



**B.Tech – COMPUTER SCIENCE AND
ENGINEERING
CBCS CURRICULUM
Honors / Specialization / Minor
(With effect from 2021-2022)**

**Department of Computer Science and
Engineering**



School of Computing

**Vel Tech Rangarajan Dr.Sagunthala R&D Institute of
Science and Technology**

Vision

The Vision of Computer Science and Engineering Department is

To produce intellectual graduates who could contribute significantly in the analysis, design, development, operation and maintenance of complex software systems for meeting the ever changing requirements and to compete globally towards professional excellence.

Mission

The mission of Computer Science and Engineering Department is to

M1: Design curricula for imparting training in adapting newer computing methods and technologies for providing effective and efficient solutions to the existing / new problems.

M2: Inculcate in-depth knowledge of various courses by employing Information and Communication Technology (ICT) based pedagogy methods.

M3: Create a conducive research environment for making technological innovations by the faculty and students.

M4: Provide leadership skills and professional ethics thereby making a prolific career in academics and industry.

PROGRAMME EDUCATIONAL OBJECTIVES

The Programme Educational Objectives (PEOs) are as follows:

After a few years of professional accomplishment in the program, the graduates will be able to

PEO1: Formulate, solve and analyze Computer Science and Engineering problems using necessary mathematical, Scientific and engineering fundamentals.

PEO2: Demonstrate the impact of cutting-edge technologies to accomplish social and professional responsibilities.

PEO3: Demonstrate critical thinking, communication, teamwork, leadership skills and ethical behavior necessary to function productively and professionally.

PEO4: Pursue higher education at reputed institution in India and abroad, work in product development companies and engage in lifelong learning.

Knowledge and Attitude Profile (WK)

WK1: A systematic, theory-based understanding of the natural sciences applicable to the discipline and awareness of relevant social sciences.

WK2: Conceptually-based mathematics, numerical analysis, data analysis, statistics and formal aspects of computer and information science to support detailed analysis and modelling applicable

to the discipline.

WK3: A systematic, theory-based formulation of engineering fundamentals required in the engineering discipline.

WK4: Engineering specialist knowledge that provides theoretical frameworks and bodies of knowledge for the accepted practice are as in the engineering discipline; much is at the forefront of the discipline.

WK5: Knowledge, including efficient resource use, environmental impacts, whole-life cost, re-use of resources, net zero carbon, and similar concepts, that supports engineering design and operations in a practice area.

WK6: Knowledge of engineering practice(technology) in the practice area as in the engineering discipline.

WK7: Knowledge of the role of engineering in society and identified issues in engineering practice in the discipline, such as the professional responsibility of an engineer to public safety and sustainable development.

WK8: Engagement with selected knowledge in the current research literature of the discipline, awareness of the power of critical thinking and creative approaches to evaluate emerging issues.

WK9: Ethics, inclusive behavior and conduct. Knowledge of professional ethics, responsibilities, and norms of engineering practice. Awareness of the need for diversity by reason of ethnicity, gender, age, physical ability etc. with mutual understanding and respect, and of inclusive attitudes.

PROGRAM OUTCOMES (POs)

Engineering Graduates will be able to:

PO1: Engineering Knowledge: Apply knowledge of mathematics, natural science, computing, engineering fundamentals and an engineering specialization as specified in WK1 to WK4 respectively to develop to the solution of complex engineering problems.

PO2: Problem Analysis: Identify, formulate, review research literature and analyze complex engineering problems reaching substantiated conclusions with consideration for sustainable development. (WK1 to WK4)

PO3: Design/Development of Solutions: Design creative solutions for complex engineering problems and design/develop systems/components/processes to meet identified needs with consideration for the public health and safety, whole-life cost, net zero carbon, culture, society and environment as required. (WK5)

PO4: Conduct Investigations of Complex Problems: Conduct investigations of complex engineering problems using research-based knowledge including design of experiments, modelling, analysis & Interpretation of data to provide valid conclusions. (WK8).

PO5: Engineering Tool Usage: Create, select and apply appropriate techniques, resources and modern engineering & IT tools, including prediction and modelling recognizing their limitations to solve complex engineering problems. (WK2 and WK6)

PO6: The Engineer and The World: Analyze and evaluate societal and environmental aspects while solving complex engineering problems for its impact on sustainability with reference to economy, health, safety, legal framework, culture and environment. (WK1, WK5, and WK7).

PO7: Ethics: Apply ethical principles and commit to professional ethics, human values, diversity and inclusion; adhere to national & international laws. (WK9)

PO8: Individual and Collaborative Teamwork: Function effectively as an individual, and as a member or leader in diverse/multi-disciplinary teams.

PO9: Communication: Communicate effectively and inclusively within the engineering community and society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations considering cultural, language, and learning differences

PO10: Project Management and Finance: Apply knowledge and understanding of engineering management principles and economic decision-making and apply these to one's own work, as a member and leader in a team, and to manage projects and in multidisciplinary environments.

PO11: Life-Long Learning: Recognize the need for, and have the preparation and ability for

i) Independent and life-long learning ii) Adaptability to new and emerging technologies and iii) Critical thinking in the broadest context of technological change. (WK8)

PROGRAM SPECIFIC OUTCOMES (PSOs)

On successful completion of the program, the graduates will be able to,

PSO1: Mathematical Concepts: Equipped with the knowledge to infer the mathematical models for problem solving using data structures, design and analysis of algorithms.

PSO2: Software Development: Exhibit proficiency to analyze, design and develop applications in various domains to provide solutions using innovative ideas.

PSO3: Transferring Skills: Demonstrate the ability to provide solutions for real world problems through acquaintance and hands-on training

COURSE OUTCOME

Abilities of the student defined in terms of Course Outcomes (COs) as per the Bloom's Taxonomy at the end of every course in the programme.

**B. Tech - COMPUTER SCIENCE AND ENGINEERING
CURRICULUM
(CBCS)**

**Honors / Specialization / Minor
2021-2022**

**Credits required for regular students in various course categories for
B. Tech(CSE)**

The students shall earn 164 credits in various course categories given below for the award of degree of B.Tech (CSE).

Course Category	Minimum Credits Required
Foundation Courses (FC)	56
Program Core (PC)	58
Program Elective (PE)	18
Open Elective (OE)	12
Independent Learning(IL)	14
Industry / Higher Institute Learning Interaction(IHL)	2
Professional Proficiency Courses (PPC)	4
TOTAL	164

Minimum credits required for regular students in various course categories for B.Tech Computer Science and Engineering with minor

The students shall earn 164 credits in various course categories and additional 18 to 20 credits in the specialized tracks / areas from other branches/Schools by satisfying the prerequisite courses for the award of degree of B.Tech Computer Science and Engineering with minor subject to the regulations.

Minimum credits required for regular students in various course categories for B.Tech Computer Science and Engineering with Honors

The students shall earn 164 credits in various course categories and additional 18 to 20 credits in the specialized tracks / areas courses by satisfying the prerequisite courses for the award of degree of B.Tech Computer Science and Engineering with Honors subject to the regulations.

Foundation Core (56 Credits) Batch- 2021-2022 Only

Foundation courses enhance the knowledge, skills and attitude of UG engineering graduates of all programmes to the expected level. The foundation courses shall have the courses related to basic sciences and mathematics, basic engineering sciences and humanities and social sciences.

L-Lecture, T-Tutorial, P-Practical, C-credit

S.No	Course Code	Subject Title	Category	L	T	P	SL	C
Lecture Courses								
1	10210MA101	Linear Algebra for Computing	BSC	3	1	0	4	4
2	10210MA102	Calculus & Ordinary differential Equations	BSC	3	1	0	4	4
3	10210MA103	Probability, Statistics and Queuing theory	BSC	3	1	0	4	4
4	10210PH101	Semiconductor Physics	BSC	3	0	0	3	3
5	10210CH101	Engineering Chemistry	BSC	3	0	0	3	3
6	10210CH103	Environmental Science	BSC	2	0	0	2	2
7	10210CS101	Problem Solving using C	ESC	3	0	0	3	3
8	10210CS102	Computational Thinking for Problem Solving	ESC	3	0	0	3	3
9	10210ME101	Design thinking	ESC	2	0	0	2	2
10	10210BM101	Biology for Engineers	ESC	2	0	0	2	2
11	10210ME103	Innovation & Entrepreneurship	ESC	2	0	0	2	2
12	10210ME102	Universal Human Values	HSC	3	0	0	3	3
13	10210ME104	Project Management & Finance	HSC	2	0	0	2	2
14	10210ME105	Engineers and Society	HSC	1	0	0	1	M
15	10210BL101	Constitution of India	HSC	1	0	0	1	M
Integrated Courses								
16	10210EN201	Professional Communication – I	HSC	1	0	2	1	2
17	10210EN202	Professional Communication – II	HSC	1	0	2	1	2
18	10210EE201	Basic Electrical, Electronics & Measurement Engineering	ESC	2	0	2	2	3
19	10210EE204	Introduction to Engineering	ESC	1	0	4	1	3
20	10210ME201	Engineering Graphics	ESC	1	0	4	1	3
Laboratory Courses								
21	10210PH301	Modern Physics Laboratory	BSC	0	0	2	-	1
22	10210CH302	Engineering Chemistry Laboratory	BSC	0	0	2	-	1
23	10210EE301	Engineering Products Lab	ESC	0	0	2	-	1
24	10210CS301	Problem Solving using C Lab	ESC	0	0	2	-	1

25	10210CS302	Computational thinking laboratory	ESC	0	0	2	-	1
26	10210CS303	IT Workshop	ESC	0	0	2	-	1
Total Credits								56

- BSC – Basic Science Courses, ESC – Engineering Science Courses, HSC – Humanities & Social Science Courses, M – Mandatory course

Foundation Core (56 Credits) (2022-2023) admitted batch Onwards

Foundation courses enhance the knowledge, skills and attitude of UG engineering graduates of all programmes to the expected level. The foundation courses shall have the courses related to basic sciences and mathematics, basic engineering sciences and humanities and social sciences.

L-Lecture, T-Tutorial, P-Practical, C-credit								
S.No	Course Code	Subject Title	Category	L	T	P	SL	C
Lecture Courses								
1	10210MA101	Linear Algebra for Computing	BSC	3	1	0	4	4
2	10210MA102	Calculus & Ordinary differential Equations	BSC	3	1	0	4	4
3	10210MA103	Probability, Statistics and Queuing theory	BSC	3	1	0	4	4
4	10210PH101	Semiconductor Physics	BSC	3	0	0	3	3
5	10210CH101	Engineering Chemistry	BSC	3	0	0	3	3
6	10210CH104	Environmental Science and Sustainability	BSC	3	0	0	2	3
7	10210CS101	Problem Solving using C	ESC	3	0	0	3	3
8	10210CS102	Computational Thinking for Problem Solving	ESC	3	0	0	3	3
9	10210ME101	Design thinking	ESC	2	0	0	2	2
10	10210BM101	Biology for Engineers	ESC	2	0	0	2	2
11	10210ME103	Innovation & Entrepreneurship	ESC	2	0	0	2	2
12	10210ME102	Universal Human Values	HSC	3	0	0	3	3
13	10210ME104	Project Management & Finance	HSC	2	0	0	2	2
14	10210ME105	Engineers and Society	HSC	1	0	0	1	M
15	10210BL101	Constitution of India	HSC	1	0	0	1	M
Integrated Courses								
16	10210EN201	Professional Communication – I	HSC	1	0	2	1	2
17	10210EN202	Professional Communication – II	HSC	1	0	2	1	2
18	10210EE201	Basic Electrical, Electronics & Measurement Engineering	ESC	2	0	2	2	3
19	10210EE204	Introduction to Engineering	ESC	1	0	4	1	3
20	10210ME201	Engineering Graphics	ESC	1	0	4	1	3

Laboratory Courses								
21	10210PH301	Modern Physics Laboratory	BSC	0	0	2	-	1
22	10210CH302	Engineering Chemistry Laboratory	BSC	0	0	2	-	1
23	10210EE301	Engineering Products Lab	ESC	0	0	2	-	1
24	10210CS301	Problem Solving using C Lab	ESC	0	0	2	-	1
25	10210CS302	Computational thinking laboratory	ESC	0	0	2	-	1
Total Credits								56

- BSC – Basic Science Courses, ESC – Engineering Science Courses, HSC – Humanities & Social Science Courses, M – Mandatory course

Program Core (58 Credits) Batch- 2021-2022 Only

L-Lecture, T-Tutorial, P-Practical, C-credit								
S.No	Course Code	Course Name	L	T	P	C	SL	Pg.No
		Theory Courses						
1	10211CS101	Discrete Mathematics	3	0	0	3	3	32
2	10211CS102	Data Structures	3	0	0	3	3	34
3	10211CS103	Operating Systems	3	0	0	3	3	37
4	10211CS129	Modern Computer Architecture	3	0	0	3	3	39
5	10211CS105	Computer Networks	3	0	0	3	3	42
6	10211CS130	Fundamentals of Computer Networks	3	0	0	3	3	45
7	10211CS106	Formal Languages and Automata Theory	3	0	0	3	3	48
8	10211CS107	Compiler Design	3	0	0	3	3	51
9	10211CS109	Microprocessor and Microcontroller	2	0	0	2	2	54
10	10211CS111	Image Processing	2	0	0	2	2	57
12	10211CS118	Image Processing Techniques	3	0	0	3	3	60
13	10211CS120	Machine Learning	3	0	0	3	3	63
		Integrated Courses						
14	10211CS201	Digital Electronics	2	0	2	2	3	66
15	10211CS202	Design and Analysis of Algorithms	3	0	2	3	4	70
16	10211CS204	Object Oriented Programming using Java	2	0	2	2	3	76
17	10211CS207	Database Management Systems	3	0	2	3	4	80
18	10211CS208	Software Engineering	2	0	2	2	3	85
19	10211CS212	Web and Mobile Application Development	3	0	2	3	4	89
20	10211CS213	Python Programming	1	0	2	1	2	94
21	10211CS223	Machine learning techniques	2	0	2	2	3	97
		Laboratory Courses						

22	10211CS302	Data Structures Laboratory	0	0	2	-	1	143
23	10211CS303	Computer Networks Laboratory	0	0	2	-	1	148
24	10211CS312	Fundamentals of Computer Networks Laboratory	0	0	2	-	1	152
25	10211CS313	Machine Learning Laboratory	0	0	2	-	1	156
26	10211CS304	Operating Systems Laboratory	0	0	2	-	1	159
27	10211CS305	Microprocessor and Microcontroller Laboratory	0	0	2	-	1	165
28	10211CS306	Competitive Coding –I	0	0	2	-	1	168
29	10211CS307	Competitive Coding –II	0	0	2	-	1	173
Total Credits						58		

Program Core (58 Credits) (2022-2023)

L-Lecture, T-Tutorial, P-Practical, C-credit

S.No	Course Code	Course Name	L	T	P	SL	C	Pg.No
		Theory Courses						
1	10211CS101	Discrete Mathematics	3	0	0	3	3	32
2	10211CS102	Data Structures	3	0	0	3	3	34
3	10211CS103	Operating Systems	3	0	0	3	3	37
4	10211CS129	Modern Computer Architecture	3	0	0	3	3	39
5	10211CS130	Fundamentals of Computer Networks	3	0	0	3	3	45
6	10211CS106	Formal Languages and Automata Theory	3	0	0	3	3	48
7	10211CS107	Compiler Design	3	0	0	3	3	51
8	10211CS109	Microprocessor and Microcontroller	2	0	0	2	2	54
9	10211CS118	Image Processing Techniques	3	0	0	3	3	60
		Integrated Courses						
10	10211CS201	Digital Electronics	2	0	2	2	3	66
11	10211CS202	Design and Analysis of Algorithms	3	0	2	3	4	70
12	10211CS204	Object Oriented Programming using Java	2	0	2	2	3	76
13	10211CS207	Database Management Systems	3	0	2	3	4	80
14	10211CS208	Software Engineering	2	0	2	2	3	85
15	10211CS212	Web and Mobile Application Development	3	0	2	3	4	89
16	10211CS213	Python Programming	1	0	2	1	2	94
17	10211CS223	Machine Learning Techniques	2	0	2	2	3	97
18	10211CS225	Problem Solving and Testing using Java	1	0	4	2	3	111

19	10211CS224	Full Stack Application Development	1	0	4	2	3	102
20	10211CS226	Java Programming	2	0	2	2	3	131
		Laboratory Courses						
18	10211CS302	Data Structures Laboratory	0	0	2	-	1	143
19	10211CS312	Fundamentals of Computer Networks Laboratory	0	0	2	-	1	152
20	10211CS304	Operating Systems Laboratory	0	0	2	-	1	159
21	10211CS305	Microprocessor and Microcontroller Laboratory	0	0	2	-	1	165
22	10211CS306	Competitive Coding –I	0	0	2	-	1	168
23	10211CS307	Competitive Coding –II	0	0	2	-	1	173
Total Credits							58	

B.Tech - COMPUTER SCIENCE AND ENGINEERING**CBCS CURRICULUM****Honors / Specialization / Minor****Minimum credits required for Lateral Entry students in various course categories for
B.Tech(CSE) – VTR UGE 2021**

The students shall earn 120 credits in various course categories given below for the award of degree of B.Tech (CSE).

Course Category	Minimum Credits Required
Foundation Courses (FC)	22
Program Core (PC)	48
Program Elective (PE)	18
Open Elective (OE)	12
Independent Learning(IL)	14
Industry / Higher Institute Learning Interaction(IHL)	2
Professional Proficiency Courses (PPC)	4
TOTAL	120

**B.Tech - COMPUTER SCIENCE AND ENGINEERING
CBCS CURRICULUM
Honors / Specialization / Minor**

**VTR UGE 2021
Lateral Entry 2022-2023 Admitted Batch (2021-2022)**

Foundation Core (22 Credits)

S.No	Course Code	Subject Title	Category	L	T	P	SL	C
Lecture Courses								
1	10210MA103	Probability, Statistics and Queuing theory	BSC	3	1	0	4	4
2	10210PH101	Semiconductor Physics	BSC	3	0	0	3	3
3	10210CS101	Problem Solving using C	ESC	3	0	0	3	3
4	10210ME101	Design thinking	ESC	2	0	0	2	2
7	10210ME103	Innovation & Entrepreneurship	ESC	2	0	0	2	2
9	10210ME102	Universal Human Values	HSC	3	0	0	3	3
11	10210ME104	Project Management & Finance	HSC	2	0	0	2	2
12	10210ME105	Engineers and Society	HSC	1	0	0	1	M
13	10210BL101	Constitution of India	HSC	1	0	0	1	M
Integrated Courses								
14	10210EN201	Professional Communication – I	HSC	1	0	2	1	2
Laboratory Courses								
15	10210CS301	Problem Solving using C Lab	ESC	0	0	2	-	1
Total Credits								22

B.Tech - COMPUTER SCIENCE AND ENGINEERING
CBCS CURRICULUM
Honors / Specialization / Minor
VTR UGE 2021
Lateral Entry Admitted 2023-2024 Batch (2022-2023)onwards

Foundation Core (22 Credits)

S.No	Course Code	Subject Title	Category	L	T	P	SL	C
Lecture Courses								
1	10210CH104	Environmental Science and Sustainability	BSC	3	0	0		3
2	10210MA103	Probability, Statistics and Queuing theory	BSC	3	1	0	4	4
3	10210CS101	Problem Solving using C	ESC	3	0	0	3	3
4	10210ME101	Design thinking	ESC	2	0	0	3	2
5	10210BM101	Biology for Engineers	ESC	2	0	0	2	2
6	10210ME103	Innovation & Entrepreneurship	ESC	2	0	0	2	2
7	10210ME102	Universal Human Values	HSC	3	0	0	3	3
8	10210ME104	Project Management & Finance	HSC	2	0	0	2	2
9	10210ME105	Engineers and Society	HSC	1	0	0	1	M
10	10210BL101	Constitution of India	HSC	1	0	0	1	M
Integrated Courses								
11	10210EN201	Professional Communication – I	HSC	1	0	2	¹	2
Total Credits								23

Lateral Entry 2022-2023 Admitted Batch (2021-2022)**Program Core (48 Credits)**

(Common for both the batches)

S.No	Course Code	Course Name	L	T	P	SL	C
		Theory Courses					
1	10211CS101	Discrete Mathematics	3	0	0	3	3
2	10211CS102	Data Structures	3	0	0	3	3
4	10211CS129	Modern Computer Architecture	3	0	0	3	3
5	10211CS105	Computer Networks	3	0	0	3	3
6	10211CS106	Formal Languages and Automata Theory	3	0	0	3	3
7	10211CS107	Compiler Design	3	0	0	3	3
8	10211CS109	Microprocessor and Microcontroller	2	0	0	2	2
9	10211CS120	Machine Learning	3	0	0	3	3
		Integrated Courses					
10	10211CS201	Digital Electronics	2	0	2	2	3
11	10211CS202	Design and Analysis of Algorithms	3	0	2	3	4
12	10211CS204	Object Oriented Programming using Java	2	0	2	2	3
13	10211CS207	Database Management Systems	3	0	2	3	4
14	10211CS212	Web and Mobile Application Development	3	0	2	3	4
15	10211CS213	Python Programming	1	0	2	1	2
		Laboratory Courses					
16	10211CS302	Data Structures Laboratory	0	0	2	-	1
17	10211CS303	Computer Networks Laboratory	0	0	2	-	1
18	10211CS313	Machine Learning Laboratory	0	0	2	-	1
20	10211CS305	Microprocessor and Microcontroller Laboratory	0	0	2	-	1
21	10211CS306	Competitive Coding –I	0	0	2	-	1
22	10211CS307	Competitive Coding –II	0	0	2	-	1
Total Credits							48/49

Lateral Entry Admitted 2023-2024 Batch (2022-2023) onwards**Program Core (48 Credits)**

(Common for both the batches)

S.No	Course Code	Course Name	L	T	P	SL	C
		Theory Courses					
1	10211CS101	Discrete Mathematics	3	0	0	3	3
2	10211CS102	Data Structures	3	0	0	3	3
4	10211CS129	Modern Computer Architecture	3	0	0	3	3
5	10211CS130	Fundamentals of Computer Networks	3	0	0	3	3
6	10211CS106	Formal Languages and Automata Theory	3	0	0	3	3
7	10211CS107	Compiler Design	3	0	0	3	3
8	10211CS109	Microprocessor and Microcontroller	2	0	0	2	2
		Integrated Courses					
10	10211CS201	Digital Electronics	2	0	2	2	3
11	10211CS202	Design and Analysis of Algorithms	3	0	2	3	4
12	10211CS204	Object Oriented Programming using Java	2	0	2	2	3
13	10211CS207	Database Management Systems	3	0	2	3	4
15	10211CS212	Web and Mobile Application Development	3	0	2	3	4
16	10211CS213	Python Programming	1	0	2	1	2
17	10211CS223	Machine Learning Techniques	2	0	2	2	3
		Laboratory Courses					
18	10211CS302	Data Structures Laboratory	0	0	2	-	1
19	10211CS312	Fundamentals of Computer Networks Laboratory	0	0	2	-	1
20	10211CS305	Microprocessor and Microcontroller Laboratory	0	0	2	-	1
21	10211CS306	Competitive Coding –I	0	0	2	-	1
22	10211CS307	Competitive Coding –II	0	0	2	-	1
Total Credits							48

Program Electives (18 Credits)

Program electives are the courses offered in the programme which covers depth and breadth. The students may register for appropriate electives offered in the programme based on their area of interest. One course under this category shall be taken from the list of approved MOOCs.

L-Lecture, T-Tutorial, P-Practical, C-credit

S.No	Course Code	Course Name	L	T	P	SL	C	PG No
		Theory Courses						
1	10212CS110	Optimization Techniques	3	0	0	3	3	181
2	10212CS112	Cyber Security Policy, Law & Ethics*	3	0	0	3	3	183
3	10212CS113	Reinforcement Learning	3	0	0	3	3	186
4	10212CS119	Cryptography and Network Security	3	0	0	3	3	188
5	10212CS121	High Performance Computing	3	0	0	3	3	191
6	10212CS122	Cognitive Computing*	3	0	0	3	3	
7	10212CS123	Data Security and Privacy*	2	0	0	2	2	
8	10212CS126	Information Security*	3	0	0	3	3	
9	10212CS127	Introduction to FinTech*	3	0	0	3	3	
10	10212CS128	Financial Accounting	3	0	0	3	3	196
11	10212CS125	Principles of Compiler Design	3	0	0	3	3	193
		Integrated Courses						
11	10212CS210	Big Data Analytics	3	0	2	3	4	198
12	10212CS211	Artificial Intelligence Techniques	3	0	2	3	4	202
13	10212CS214	Data Visualization	3	0	2	3	4	207
14	10212CS215	Deep Learning	3	0	2	3	4	211
15	10212CS216	Natural Language Processing	3	0	2	3	4	214
16	10212CS217	Data Science	2	0	2	2	3	218
17	10212CS218	Modeling for Data Science	3	0	2	3	4	222
18	10212CS219	Cyber Security	3	0	2	3	4	226
19	10212CS220	Animation Engineering	3	0	2	3	4	229
20	10212CS221	Game Design	3	0	2	3	4	233
21	10212CS222	Virtual and Augmented Reality	2	0	2	2	3	236
22	10212CS224	Computer Vision	3	0	2	3	4	239
23	10212CS225	Ethical Hacking	3	0	2	3	4	242
24	10212CS226	Secure Coding*	3	0	2	3	4	
25	10212CS228	Blockchain Technology	2	0	2	2	3	245
26	10212CS229	IoT and Cloud Computing	3	0	2	3	4	249
27	10212CS230	Forensics in Cyber Security	3	0	2	3	4	252
28	10212CS232	Vulnerability Analysis and Penetration Testing*	3	0	2	3	4	
29	10212CS239	3D Character Modeling	3	0	2	3	4	256
30	10212CS243	Mobile Game Development Tools*	3	0	2	3	4	
31	10212CS244	Computer Graphics	3	0	2	3	4	260
32	10212CS245	Story Telling*	2	0	2	2	3	
33	10212CS248	IoT with Blockchain Technology*	3	0	2	3	4	

34	10212CS249	Bitcoin and Cryptocurrencies*	3	0	2	3	4	
35	10212CS276	Generative AI	1	0	2	1	2	351
36	10212CS277	Deep Learning Techniques	3	0	2	3	4	354
37	10212CS278	Quantum Machine Learning	3	0	2	3	4	357
38	10212CS279	Application Development and Maintenance	3	0	2	3	4	361
39	10212CS280	Low code Development	1	0	2	1	2	366
40	10212CS282	Robotics & Automation	3	0	2	3	4	374
41	10212CS281	Industry Internet of Things	3	0	2	3	4	370
42	10212CS283	Industrial Automation	3	0	0	3	3	378
43	10212CS284	Business Intelligence	3	0	2		4	381
44	10212CS285	Drone Technology	3	0	2		4	383
45	10212CS286	Linux System Drivers	2	0	2		3	389
46	10212CS287	Linux System Programming	2	0	2		3	392
47	10212CS288	Salesforce Ecosystem and Application Development	3	0	2		4	395
48	10212CS292	Applied Programming Skills	3	0	2		4	404
49	10212CS293	Microprocessors and ARM Microcontrollers	2	0	2		3	419
50	10212CS291	Foundations of Mainframe Computing	3	0	2		4	398
* The proposed course subject to the approval/ratification in the upcoming BoS meetings								

Open Electives (12 Credits)

Open electives are the courses offered across the schools to enhance the knowledge breadth and professional competency of the students. The students shall register for appropriate electives offered in other schools based on their area of interest. The courses offered under this category cover the interdisciplinary knowledge.

L-Lecture, T-Tutorial, P-Practical, C-credit

S.No	Course Code	Course Name	L	T	P	SL	C
1	XXX1	Course Name- 1	3	0	0	3	3
2	XXX2	Course Name- 2	3	0	0	3	3
3	XXX3	Course Name- 3	3	0	0	3	3
4	XXX4	Course Name- 4	3	0	0	3	3

*One of the courses may be completed through MOOCs Platform like NPTEL as described by the department.

The following courses are offered to the other departments/schools by School of Computing under Open Elective category.

S.No	Course Code	Course Name	L	T	P	SL	C	Pg No
1	10213CS101	Object Oriented Programming using Java	3	0	0	3	3	494
2	10213CS102	Data Structures	3	0	0	3	3	496
3	10213CS103	Operating Systems	3	0	0	3	3	499
4	10213CS104	Database Management Systems	3	0	0	3	3	501
5	10213CS105	Computer Networks	3	0	0	3	3	504
6	10213CS106	Data warehousing and Data mining	3	0	0	3	3	507
7	10213CS107	Design and Analysis of Algorithms	3	0	0	3	3	510
8	10213CS108	Big Data Analytics	3	0	0	3	3	513
9	10213GE103	Software Development and Maintenance	3	0	0	3	3	516
10	10213GE106	Salesforce Platform Essentials	3	0	0	3	3	
11	10213GE107	Linux on embedded systems	3	0	0	3	3	

The following courses are offered to the other departments/schools by School of Computing under Open Elective category. The students will solve the problems posted by Leet Code Platform, the grades will be offered based on the scores secured by the students by solving the problems posted in Leet Code Platform.

S.No	Course Code	Course Name	L	T	P	SL	C	Pg No
1	10213GE302	Programming Challenges	0	0	4	1	2	520
2	10213GE204	Application Development and Maintenance	3	0	2	3	4	522

Independent Learning (14 Credits)

The students are expected to learn the courses offered under this category on their own. The courses offered under this category include:

L-Lecture, T-Tutorial, P-Practical, C-credit

S.No	Course Code	Course Name	L	T	P	SL	C
1	10214CS501	Community Service Project	0	0	2	-	1
2	10214CS601	Minor Project-I	0	0	4	-	2
3	10214CS602	Minor Project-II	0	0	4	-	2
4	10214CS701	Major Project	0	0	18	-	9

Industry / Higher Institute Learning Interaction (2 Credits)

The students shall earn a minimum of two credits by undergoing internship and/or specialized courses.

1. Internship:

The students shall undergo Internship in the industry/higher learning institute approved by Industry-Institute Interaction Cell (IIIC) during any time after the second academic year.

2. Specialized Courses:

The students shall undergo the courses offered either by the industrial experts whose minimum academic qualification is Bachelor of Engineering or equivalent or faculty expert from higher learning institutions approved by IIIC. The students shall choose either one two credits course or one one credit course or two one credit courses.

L-Lecture, T-Tutorial, P-Practical, C-credit

S.No	Course Code	Course Name	L	T	P	SL	C
1	10215CS901	Internship	-	-	-	-	2
2	10215CS902	Industry Expert Lecture-1	-	-	-	-	1
3	10215CS903	Industry Expert Lecture-2	-	-	-	-	1
4	10215CS951	Higher Institute Learning Interaction-1	-	-	-	-	1
5	10215CS952	Higher Institute Learning Interaction-2	-	-	-	-	1

Professional Proficiency Courses (4 Credits)

The Professional Proficiency Courses which carry four credits, to be offered in four different semesters, starting from third semester. These courses offered in this category are relevant to professional proficiency.

L-Lecture, T-Tutorial, P-Practical, C-credit

S.No	Course Code	Course Name	L	T	P	SL	C
1	10216GE901	Soft Skill-I	-	-	-	2	1
2	10216GE902	Soft Skill-I	-	-	-	2	1
3	10216GE903	Aptitude Skill-I	-	-	-	2	1
4	10216GE904	Aptitude Skill-II	-	-	-	2	1

Specialization / Honors (Specialization: 18 credits Honors: Additional 20 credits)

Track: Artificial Intelligence

S.No	Course Code	Course Name	L	T	P	SL	C	Pg No
		Artificial Intelligence – Core						
1	10212CS211	Artificial Intelligence Techniques	3	0	2	3	4	202
2	10212CS210	Big Data Analytics	3	0	2	3	4	198
3	10212CS110	Optimization Techniques	3	0	0	3	3	181
4	10212CS214	Data Visualization	3	0	2	3	4	207
5	10212CS215	Deep Learning	3	0	2	3	4	211
6	10212CS224	Computer Vision	3	0	2	3	4	239
7	10212CS216	Natural Language Processing	3	0	2	3	4	214
8	10212CS113	Reinforcement Learning	3	0	0	3	3	186
9	10212CS229	IoT and Cloud Computing	3	0	2	3	4	249
		Honors						
13	10212CS122	Cognitive Computing*	3	0	0	3	3	
	*The proposed course subject to the approval/ratification in the upcoming BoS meetings							

Track: Data Science

S.No	Course Code	Course Name	L	T	P		C	Pg. No
		Data Science – Core						
1	10212CS217	Data Science	2	0	2	2	3	218
2	10212CS218	Modeling for Data Science	3	0	2	3	4	222
3	10212CS210	Big Data Analytics	3	0	2	3	4	198
4	10212CS214	Data Visualization	3	0	2	3	4	207
5	10212CS211	Artificial Intelligence Techniques	3	0	2	3	4	202
6	10212CS110	Optimization Techniques	3	0	0	3	3	181
7	10212CS215	Deep Learning	3	0	2	3	4	211
8	10212CS229	IoT and Cloud Computing	3	0	2	3	4	249
9	10212CS276	Generative AI	1	0	2		2	351
10	10212CS277	Deep Learning Techniques	3	0	2		4	354
11	10212CS278	Quantum Machine Learning	3	0	2		4	357
		Honors						
12	10212CS234	Time Series and Forecasting*	3	0	2	3	4	424
13	10212CS233	Social Media Analytics*	3	0	2	3	4	428
14	10212CS122	Cognitive Computing*	3	0	0	3	3	
15	10212CS224	Computer Vision	3	0	2	3	4	239
16	10212CS124	Artificial Intelligence for Cyber Security*	3	0	0	3	3	
	*The proposed course subject to the approval/ratification in the upcoming BoS meetings							

Track : Artificial Intelligence and Machine Learning

S. No	Course Code	Course Name	L	T	P	SL	C	Pg. No
		Artificial Intelligence and Machine Learning Core Electives						
1	10212CS210	Big Data Analytics	3	0	2	3	4	198
2	10212CS214	Data Visualization	3	0	2	3	4	207
3	10212CS211	Artificial Intelligence Techniques	3	0	2	3	4	202
4	10212CS110	Optimization Techniques	3	0	0	3	3	181
5	10212CS215	Deep Learning	3	0	2	3	4	211
6	10212CS216	Natural Language Processing	3	0	2	3	4	214
7	10212CS122	Cognitive Computing *	3	0	0	3	3	
8	10212CS113	Reinforcement Learning	3	0	0	3	3	186
9	10212CS121	High Performance Computing	3	0	0	3	3	191
10	10212CS229	IoT and Cloud Computing	3	0	2	3	4	249
11	10212CS276	Generative AI	1	0	2	1	2	351
12	10212CS277	Deep Learning Techniques	3	0	2	3	4	354
13	10212CS278	Quantum Machine Learning	3	0	2	3	4	357
		Honors						
		Artificial Intelligence						
14		Speech Recognition*	3	0	0	3	3	
15	10212CS282	Robotics & Automation	3	0	2	3	4	374

16		Intelligent Agents*	2	0	2	2	3	
17	10212CS206	Human Computer Interaction*	3	0	2	3	4	
18	10212CS124	Artificial Intelligence for Cyber Security*	3	0	0	3	3	
19		Artificial Intelligence for Health Care*	2	0	4	2	4	
		Machine Learning						
20	10212CS224	Computer vision	3	0	2	3	4	239
21	10212CS234	Time series and Forecasting*	3	0	2	3	4	424
22	10212CS233	Social Media Analytics*	3	0	2	3	4	428

Track: Artificial Intelligence and Data Science

S.No	Course Code	Course Name	L	T	P	SL	C	Pg. No
		Artificial Intelligence and Data Science - Core						
1	10212CS110	Optimization Techniques	3	0	0	3	3	181
2	10212CS113	Reinforcement Learning	3	0	0	3	3	186
3	10212CS123	Data Security and Privacy*	2	0	0	2	2	
4	10212CS210	Big Data Analytics	3	0	2	3	4	198
5	10212CS211	Artificial Intelligence Techniques	3	0	2	3	4	202
6	10212CS214	Data Visualization	3	0	2	3	4	207
7	10212CS215	Deep Learning	3	0	2	3	4	211
8	10212CS216	Natural Language Processing	3	0	2	3	4	214
9	10212CS217	Data Science	2	0	2	2	3	218
10	10212CS218	Modeling for Data Science	3	0	2	3	4	222
11	10212CS229	IoT and Cloud Computing	3	0	2	3	4	249
12	10212CS276	Generative AI	1	0	2	1	2	351
13	10212CS277	Deep Learning Techniques	3	0	2	3	4	354
14	10212CS278	Quantum Machine Learning	3	0	2	3	4	357
		Honors						
		Artificial Intelligence						
15	10212CS122	Cognitive Computing*	3	0	0	3	3	
16	10212CS282	Robotics & Automation	3	0	2	3	4	374
17	10212CS231	Artificial Intelligence for Health Care*	3	0	2	3	4	
		Data Science						
18	10212CS234	Time Series and Forecasting*	2	0	2	2	3	424
19	10212CS233	Social Media Analytics*	2	0	2	2	3	428
20	10212CS224	Computer Vision	3	0	2	3	4	239
		*The proposed course subject to the approval/ratification in the upcoming BoS meetings						

Track : Artificial Intelligence and Robotics

S.No	Course Code	Course Name	L	T	P	SL	C	Pg. No
		Artificial Intelligence and Robotics - Core						
1	10212CS211	Artificial Intelligence Techniques	3	0	2	3	4	202
2	10212CS210	Big Data Analytics	3	0	2	3	4	198
3	10212CS214	Data Visualization	3	0	2	3	4	207
4	10212CS216	Natural Language Processing	3	0	2	3	4	214
5	10212CS281	Industry Internet of Things	3	0	2	3	4	370
6	10212CS282	Robotics & Automation	3	0	2	3	4	374
7	10212CS283	Industrial Automation	3	0	0	3	3	378
	*The proposed course subject to the approval/ratification in the upcoming BoS meetings							

Track: Cyber Security L-Lecture T-Tutorial P-Practical C-Credits

S.No	Course Code	Course Name	L	T	P	SL	C
		Cyber Security – Core Electives					
1	10212CS219	Cyber Security	3	0	2	3	4
2	10212CS225	Ethical Hacking	3	0	2	3	4
3	10212CS230	Forensics in Cyber Security	3	0	2	3	4
4	10212CS232	Vulnerability Assessment and Penetration Testing*	3	0	2	3	4
5	10212CS250	Identity Access Management	3	0	2	3	4
6	10212CS112	Cyber Security Policy, Law & Ethics	3	0	0	3	3
7	10212CS119	Cryptography and Network Security	3	0	0	3	3
8	10212CS228	Blockchain Technology	2	0	2	2	3
9	10212CS229	IoT and Cloud Computing	3	0	2	3	4
10	10212CS226	Secure Coding*	3	0	2	3	4
		Honors					
11		Disaster Recovery and Business Continuity Planning*	3	0	0	3	3
12		Computer Security Audit and Assurance*	3	0	2	3	4
13		Incident Response Management*	3	0	0	3	3
14		Reverse Engineering & Malware Analysis*	3	0	2	3	4
15		Social Network Security *	3	0	2	3	4
16		IoT and Cloud Security	3	0	2	3	4
17		Artificial Intelligence in Cyber Security*	3	0	2	3	4

Track: Financial Technology

S.No	Course Code	Course Name	L	T	P	SL	C	Pg. No
Financial Technology- Core								
1	10212CS127	Introduction to FinTech*	3	0	0	3	3	
2	10212CS210	Big Data Analytics	3	0	2	3	4	198
3	10212CS214	Data Visualization	3	0	2	3	4	207
4	10212CS211	Artificial Intelligence Techniques	3	0	2	3	4	202
5	10212CS223	Machine Learning Techniques	3	0	2	3	4	97
6	10212CS128	Financial Accounting	3	0	0	3	3	196
7	10212CS228	Block Chain Technology	2	0	2	2	3	245
8	10212CS229	IoT and Cloud Computing	3	0	2	3	4	249
Honors								
9	10212CS215	Deep Learning	3	0	2	3	4	211
10	10212CS249	Bit coin and Crypto currencies*	3	0	2	3	4	
*The proposed course subject to the approval/ratification in the upcoming BoS meetings								

Track: Gaming and Animation

S.No	Course Code	Course Name	L	T	P	SL	C	Pg. No
Gaming and Animation- Core								
1	10212CS244	Computer Graphics	3	0	2	3	4	260
2	10212CS220	Animation Engineering	3	0	2	3	4	229
3	10212CS221	Game Design	3	0	2	3	4	233
4	10212CS222	Virtual and Augmented Reality	2	0	2	2	3	236
5	10212CS239	3D Character Modeling	3	0	2	3	4	256
6	10212CS243	Mobile Game Development Tools *	3	0	2	3	4	
7	10212CS229	IoT and Cloud Computing	3	0	2	3	4	249
Honors								
8		Mobile Game Programming *	3	0	2	3	4	
9		3D Architecture and Art Design *	2	0	4	2	4	
10		Computer Animation Algorithm and Techniques*	3	0	2	3	4	
11		Visual Effects*	3	0	2	3	4	
12	10212CS245	Story Telling*	2	0	2	2	3	
*The proposed course subject to the approval/ratification in the upcoming BoS meetings								

Track: IoT and Cyber Security including Block Chain Technology

S.No	Course Code	Course Name	L	T	P	SL	C	Pg. No
Cyber Security - Core								

S.No	Course Code	Course Name	L	T	P	SL	C	Pg. No
1	10212CS119	Cryptography and Network Security	3	0	0	3	3	188
2	10212CS219	Cyber Security	3	0	2	3	4	226
3	10212CS225	Ethical Hacking	3	0	2	3	4	242
4	10212CS230	Forensics in Cyber Security	3	0	2	3	4	252
5	10212CS228	Blockchain Technology	2	0	2	2	3	245
6	10212CS248	IoT with Blockchain Technology*	3	0	2	3	4	
7	10212CS249	Bitcoin and Cryptocurrencies*	3	0	2	3	4	
8	10212CS229	IoT and Cloud Computing	3	0	2	3	4	249

Track: Cloud Infrastructure Management

S.No	Course Code	Course Name	L	T	P	SL	C	Pg. No
1	10212CS251	Virtualization Techniques	2	0	2	2	3	268
2	10212CS252	Windows Client Administration Operating System	2	0	2	2	3	272
3	10212CS253	Windows Server Administration Operating System	2	0	2	2	3	275
4	10212CS254	Cloud Fundamentals using Azure	2	0	2	2	3	278
5	10212CS255	Administering Microsoft Exchange Server	2	0	2	2	3	283
6	10212CS256	Office 365 Administration	2	0	2	2	3	288

Track: Network and Server Management

S.No	Course Code	Course Name	L	T	P	SL	C	Pg. No
1	10212CS257	Data Center Hosting – Windows operating System	2	0	2	2	3	298
2	10212CS258	Data Center Hosting – Unix operating System*	2	0	2	2	3	
3	10212CS259	Advanced Switching and Networking Techniques	2	0	2	2	3	301
4	10212CS260	Advanced Routing Techniques and Security	2	0	2	2	3	305
5	10212CS261	Juniper Networking Techniques*	2	0	2	2	3	
6	10212CS262	Storage Management	2	0	2	2	3	308

Track: Application Modernization Service

S.No	Course Code	Course Name	L	T	P	SL	C	Pg. No
1	10212CS263	IIS Web Server Technologies	2	0	2	2	3	312
2	10212CS264	Application Server Technologies	2	0	2	2	3	316
3	10212CS265	Web Sphere Application Server	2	0	2	2	3	322
4	10212CS301	WebSphere MQ	0	0	2	2	1	328
5	10212CS267	Middleware Technologies	1	0	2	2	2	332
6	10212CS268	Database Administration Using MSSQL Server	2	0	2	2	3	338
7	10212CS269	Oracle Database Administration	2	0	2	2	3	341

Track: Mainframe

S.No	Course Code	Course Name	L	T	P	SL	C	Pg. No
1	10212CS270	Introduction to System Programming z/OS commands and Panel on IBMZ	2	0	2	2	3	347
2	10212CS271	Fundamentals of Networking, CICS and IMS*	2	0	2	2	3	
3	10212CS272	z/OS Administration*	2	0	2	2	3	
4	10212CS273	z/OS System Service Structure*	2	0	2	2	3	
5	10212CS274	IBM System z Parallel Sysplex Operations*	2	0	2	2	3	
6	10212CS275	IBM MQ System Administration for z/OS*	2	0	2	2	3	

Minors tracks for B.Tech offered to the other Schools by Department of CSE (18 Credits)**Track :Artificial Intelligence**

Other branch students who want to complete minor in Artificial Intelligence, have to complete the prerequisite course Data Structures, Database Management System, Data Warehousing and Data mining and Digital Image Processing in Open Electives Category

S.No	Course Code	Course Name	L	T	P	SL	C	Pg No
		Artificial Intelligence – Core						
1	10212CS211	Artificial Intelligence Techniques	3	0	2	3	4	202
2	10212CS210	Big Data Analytics	3	0	2	3	4	198
3	10212CS110	Optimization Techniques	3	0	0	3	3	181
4	10212CS214	Data Visualization	3	0	2	3	4	207
5	10212CS215	Deep Learning	3	0	2	3	4	211
6	10212CS224	Computer Vision	3	0	2	3	4	239

7	10212CS216	Natural Language Processing	3	0	2	3	4	214
8	10212CS113	Reinforcement Learning	3	0	0	3	3	186
9	10212CS229	IoT and Cloud Computing	3	0	2	3	4	249
10	10212CS218	Modeling for Data Science	3	0	2	3	4	222
11	10212CS276	Generative AI	1	0	2	1	2	351

Track :Data Science

Other branch students who want to complete minor in Data Science, have to complete the prerequisite course Data Structures, Database management System and Data warehousing and Data mining in Open Electives Category

S.No	Course Code	Course Name	L	T	P	SL	C	Pg. No
		Data Science – Core						
1	10212CS217	Data Science	2	0	2	2	3	218
2	10212CS218	Modeling for Data Science	3	0	2	3	4	222
3	10212CS210	Big Data Analytics	3	0	2	3	4	198
4	10212CS214	Data Visualization	3	0	2	3	4	207
5	10212CS211	Artificial Intelligence Techniques	3	0	2	3	4	202
6	10212CS110	Optimization Techniques	3	0	0	3	3	181
7	10212CS215	Deep Learning	3	0	2	3	4	211
8	10212CS229	IoT and Cloud Computing	3	0	2	3	4	249
9	10212CS224	Computer Vision	3	0	2	3	4	239
10	10212CS276	Generative AI	1	0	2	1	2	351

Track :Cyber Security

Other branch students who want to complete minor in Cyber Security, have to complete the prerequisite course Data Structures, Operating Systems and Computer Networks in Open Electives Category

S.No	Course Code	Course Name	L	T	P	SL	C	Pg. No
		Cyber Security - Core						
1	10212CS219	Cyber Security	3	0	2	3	4	226
2	10212CS225	Ethical Hacking	3	0	2	3	4	242
3	10212CC230	Digital Forensics*	3	0	2	3	4	
4	10212CC232	Vulnerability Assessment and Penetration Testing*	3	0	2	3	4	
5	10212CC250	Identity Access Management*	3	0	2	3	4	
6	10212CC112	Cyber Security Policy, Law & Ethics*	3	0	0	3	3	
7	10212CS119	Cryptography and Network Security	3	0	0	3	3	188

S.No	Course Code	Course Name	L	T	P	SL	C	Pg. No
8	10212CC228	Blockchain Technology*	2	0	2	2	3	245
9	10212CS229	IoT and Cloud Computing	3	0	2	3	4	249
10	10212CC226	Secure Coding*	3	0	2	3	4	
Independent Learning Courses (MOOC/Swayam)								
1	10212CS431	Computer Networks and Internet Protocol	-	-	-		3	
2	10212CS432	Blockchain and its Applications	-	-	-		3	
3	10212CS433	Cryptography and Network Security	-	-	-		3	
4	10212CS434	Foundations of Cryptography	-	-	-		3	
5	10212CS413	Quantum Algorithms and Cryptography	-	-	-		3	447
6	10212CS435	Secure Computation: Part I	-	-	-		3	483
7	10212CS436	Privacy and Security in online Social Media	-	-	-		3	485
8	10212CS411	Deep Learning for Natural Language Processing	-	-	-		3	443
9	10212CS403	Introduction To Industry 4.0 and Industrial Internet of Things	-	-	-		3	435
10	10212CS401	Social Networks	-	-	-		3	433
11	10212CS402	Design & Implementation of Human-Computer Interfaces	-	-	-		3	434
12	10212CS410	Introduction to Internet of Things	-	-	-		3	545
13	10212CS414	Foundation of Cloud IoT Edge ML	-	-	-		3	449

Track :Gaming and Animation

Other branch students who want to complete minor in Gaming and Animation, prerequisite course is NIL

S.No	Course Code	Course Name	L	T	P	SL	C	Pg. No
		Gaming and Animation- Core						
1	10212CS244	Computer Graphics	3	0	2	3	4	260
2	10212CS220	Animation Engineering	3	0	2	3	4	229
3	10212CS221	Game Design	3	0	2	3	4	233
4	10212CS222	Virtual and Augmented Reality	2	0	2	2	3	236
5	10212CS245	Story Telling*FF	2	0	2	2	3	
6	10212CS239	3D Character Modeling	3	0	2	3	4	256
7	10212CS243	Mobile Game Development Tools *	3	0	2	3	4	
8	10212CS229	IoT and Cloud Computing	2	0	2	2	3	249
Independent Learning Courses (MOOC/Swayam)								
1	10212CS437	Games and Information	-	-	-		3	486
2	10212CS438	Human Computer Interaction	-	-	-		3	488
3	10212CS439	Introduction to Graphic Design	-	-	-		3	490

S.No	Course Code	Course Name	L	T	P	SL	C	Pg. No
4	10212CS440	Augmenting Design Thinking with Human-Computer Interaction	-	-	-		3	491

Track: Financial Technology

Other branch students who want to complete minor in Financial Technology, have to complete the prerequisite course Data Structures, Database Management Systems, Data warehousing and Data mining and Computer Networks in Open Electives Category

S.No	Course Code	Course Name	L	T	P	SL	C	Pg. No
		Financial Technology- Core						
1	10212CS127	Introduction to FinTech*	3	0	0	3	3	
2	10212CS210	Big Data Analytics	3	0	2	3	4	198
3	10212CS214	Data Visualization	3	0	2	3	4	207
4	10212CS211	Artificial Intelligence Techniques	3	0	2	3	4	202
5	10212CS128	Financial Accounting	3	0	0	3	3	196
6	10212CS228	Block Chain Technology	2	0	2	2	3	245
7	10212CS229	IoT and Cloud Computing	3	0	2	3	4	249

Track : Artificial Intelligence and Machine Learning

Other branch students who want to complete minor in Artificial Intelligence/Machine Learning, have to complete the prerequisite course Data Structures, Database Management System and Data warehousing and Data mining in Open Electives Category

S. No	Course Code	Course Name	L	T	P	SL	C	Pg. No
		Core Electives						
1	10212CS210	Big Data Analytics	3	0	2	3	4	198
2	10212CS214	Data Visualization	3	0	2	3	4	207
3	10212CS211	Artificial Intelligence Techniques	3	0	2	3	4	202
4	10212CS110	Optimization Techniques	3	0	0	3	3	181
5	10212CS215	Deep Learning	3	0	2	3	4	211
6	10212CS216	Natural Language Processing	3	0	2	3	4	214
7	10212CS122	Cognitive Computing *	3	0	0	3	3	
8	10212CS113	Reinforcement Learning	3	0	0	3	3	186
9	10212CS121	High Performance Computing	3	0	0	3	3	191
10	10212CS229	IoT and Cloud Computing	3	0	2	3	4	249
11	10212CS218	Modeling for Data Science	3	0	2	3	4	222
12	10212CS224	Computer Vision	3	0	2	3	4	239
13	10212CS276	Generative AI	1	0	2	1	2	351

Independent Learning Courses (MOOC/Swayam)								
1	10212CS415	Applied Linear Algebra in AI and ML	-	-	-		3	451
2	10212CS416	Artificial Intelligence: Knowledge Representation And Reasoning	-	-	-		3	453
3	10212CS417	Introduction to Machine Learning	-	-	-		3	454
4	10212CS418	Natural Language Processing	-	-	-		3	456
5	10212CS419	Computer Vision And Image Processing - Fundamentals And Applications	-	-	-		3	458
6	10212CS404	Reinforcement Learning	-	-	-		3	437
7	10212CS420	Deep Learning	-	-	-		3	460
8	10212CS421	Machine Learning for Soil and Crop Management	-	-	-		3	462
9	10212CS422	Algorithms in Computational Biology and Sequence Analysis	-	-	-		3	464
10	10212CS423	Introduction to Large Language Models (LLMs)	-	-	-		3	466
11	10212CS405	Responsible & Safe AI Systems	-	-	-		3	438
12	10212CS406	Deep Learning for Computer Vision	-	-	-		3	440
13	10212CS408	Software Engineering	-	-	-		3	442

Track : Artificial Intelligence and Data Science

Other branch students who want to complete minor in Artificial Intelligence/Data Science, have to complete the prerequisite course Data Structures, Database Management System and Data warehousing and Data mining in Open Electives Category

S.No	Course Code	Course Name	L	T	P	SL	C	Pg. No
Artificial Intelligence and Data Science - Core								
1	10212CS110	Optimization Techniques	3	0	0	3	3	181
2	10212CS113	Reinforcement Learning	3	0	0	3	3	186
3	10212CS123	Data Security and Privacy *	2	0	0	2	2	
4	10212CS210	Big Data Analytics	3	0	2	3	4	198
5	10212CS211	Artificial Intelligence Techniques	3	0	2	3	4	202
6	10212CS214	Data Visualization	3	0	2	3	4	207
7	10212CS215	Deep Learning	3	0	2	3	4	211
8	10212CS216	Natural Language Processing	3	0	2	3	4	214
9	10212CS217	Data Science	3	0	2	3	4	218
10	10212CS218	Modeling for Data Science	3	0	2	3	4	222
11	10212CS229	IoT and Cloud Computing	3	0	2	3	4	249
12	10212CS224	Computer Vision	3	0	2	3	4	239
13	10212CS276	Generative AI	1	0	2	1	2	351
Independent Learning Courses (MOOC/Swayam)								
1	10212CS416	Artificial Intelligence: Knowledge Representation and Reasoning	-	-	-		3	453

S.No	Course Code	Course Name	L	T	P	SL	C	Pg. No
2	10212CS417	Introduction to Machine Learning	-	-	-		3	454
3	10212CS411	Deep Learning for Natural Language Processing	-	-	-		3	443
4	10212CS424	Deep Learning for Visual Computing	-	-	-		3	
5	10212CS425	Essentials of Data Science with R Software-1: Probability and Statistical Inference	-	-	-		3	469
6	10212CS426	Essentials of Data Science with R Software-2: Sampling Theory and Linear Regression Analysis	-	-	-		3	471
7	10212CS427	Data Science for Engineers	-	-	-		2	473
8	10212CS428	Data Analytics with Python	-	-	-		3	475
9	10212CS429	Business Intelligence & Analytics	-	-	-		3	477
10	10212CS423	Introduction to Large Language Models (LLMs)	-	-	-		3	466
11	10212CS430	Predictive Analytics - Regression and Classification	-	-	-		3	479
12	10212CS412	Machine Learning for Engineering and science applications	-	-	-		3	445

Track : Artificial Intelligence and Robotics

Other branch students who want to complete minor in Artificial Intelligence/Robotics, prerequisite course is Digital Image Processing

S.No	Course Code	Course Name	L	T	P	SL	C	Pg. No
		Artificial Intelligence and Robotics - Core						
1	10212CS211	Artificial Intelligence Techniques	3	0	2	3	4	202
2	10212CS215	Deep Learning	2	0	2	2	3	211
3	10212CS224	Computer Vision	3	0	2	3	4	239
4	10212CS281	Industry Internet of Things	3	0	2	3	4	370
5	10212CS282	Robotics & Automation	3	0	2	3	4	374
6	10212CS283	Industrial Automation	3	0	0	3	3	378
*The proposed course subject to the approval/ratification in the upcoming BoS meetings								

PROGRAM CORE

COURSE CODE	COURSE TITLE	L	T	P	C
10211CS101	DISCRETE MATHEMATICS	3	0	0	3

A. Preamble

This course aims to develop the skills on network mathematics, statements in logic, and decision-making problems. The course gives a hands-on exploration of the value of mathematics in the real world problems and it develops the knowledge to rectify the analytical problems and challenges in their field.

B. Prerequisite Course

10210MA102- Calculus and Ordinary differential equations

C. Course Objectives

Learners are exposed to provide the fundamental concepts of logical statements sets, functions, permutations, recurrence relations, graphs, and algebraic structures.

D. Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Apply the logical statements and validity of arguments	K3
CO2	Implement the concepts of sets and relations in computer science engineering problems	K3
CO3	Solve the practical problems in computer science courses using functions and recurrence	K3
CO4	Demonstrate the concepts of algebraic structures and its properties	K3
CO5	Use the basic knowledge of graph theory in developing the applications	K3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of COs with Program outcomes and Programme Specific Outcomes:

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3			2								3		
CO2	3		2									3		
CO3	3			2								3		
CO4	3	2		2								3		
CO5	3		2	3								3		

3- High; 2-Medium; 1-Low

F. Course Contents

Unit 1 Propositional and Predicate Logic

L-9 Hours

Propositional Logic: Logical connectives – Truth tables – Tautologies and contradictions – Contrapositive – Logical equivalences and implications. Rules of inference: Validity of arguments. Predicate Logic: Predicates – Variables – Free and bound variables – Universal and Existential Quantifiers – Universe of discourse – Mathematical induction.

Unit 2 Sets and Relations

L-9 Hours

Finite Sets, Power Set, Cardinality of finite sets, Infinite sets, Countable and Uncountable sets, Cartesian product, Properties of Sets. Relations: Relations on sets – Partitions – Equivalence relations – Partial ordering – Posets – well-ordering – Meet and Join – Hasse diagrams – Lattices – Boolean Algebra.

Unit 3 Functions and Recurrence

L-9 Hours

Functions: Injective, Surjective and Bijective functions – Inverse of a function – Composition. Counting: Permutations and combinations – Sum and product rule, Principle of Inclusion Exclusion. Pigeon Hole Principle. Recurrence Relations: Linear recurrence relations with constant coefficients – methods of solutions – Generating Functions.

Unit 4 Systems and Groups

L-9 Hours

Algebraic systems – Semigroups, Monoids, Submonoids, Groups, Abelian groups, Subgroups Cosets, permutation group, transpositions, cycle representation symmetries as a group. Homomorphism and Isomorphism, Lagrange's Theorem – Generators – Cyclic Groups.

Unit 5 Graph Theory

L-9 Hours

Definitions – degree of a vertex, walk – trail – path – cycle and connected graphs, Matrix representation of Graph, Isomorphism, Hamiltonian and Eulerian graphs, Trees, cut edges, cut vertices, spanning tree, shortest path problem – Warshall's Algorithm.

Total: 45 Hours

G. Learning Resources

i. Text Books:

1. Trembly J.P and Manohar R, "Discrete Mathematical Structures with Applications to Computer Science", Tata McGraw-Hill Pub.Co.Ltd, New Delhi, 2008.[Unit 1-5].

ii. Reference Books:

1. Ralph. P. Grimaldi, "Discrete and Combinatorial Mathematics: An Applied Introduction", 4/e, Pearson Education Asia, Delhi, 2002.
2. Liu C. L., "Elements of Discrete Mathematics", 2/e, McGraw-Hill Int. editions, 1988.
3. Bernard Kolman, Robert C. Busby, Sharan Cutler Ross, "Discrete Mathematical Structures", Pearson Education Pvt Ltd., New Delhi, 2003.
4. Kenneth H.Rosen, "Discrete Mathematics and its Applications", 5/e, Tata McGraw – Hill Pub. Co. Ltd., New Delhi, 2003.

iii. Online References:

1. "Discrete Mathematics Shanghai". Accessed on April 22, 2021 [Online]. Available: <https://www.coursera.org/learn/discrete-mathematics>.
2. "Discrete Mathematics", 2013, Accessed on April 22, 2021 [Online]. Available: <http://discrete.openmathbooks.org/dmoi3.html>.

COURSE CODE	COURSE TITLE	L	T	P	C
10211CS102	DATA STRUCTURES	3	0	0	3

A. Preamble

This course aims at molding the learner to understand the various data structures, their organization and operations. The course helps the learners to assess the applicability of different data structures and associated algorithms for solving real world problem.

B. Prerequisite Course

10210CS101 - Problem Solving using C.

C. Course Objectives

Learners are exposed to:

- Understand basic elementary data structures.
- Illustrate the basic properties of Trees and Graphs.
- Implement various techniques of sorting and searching algorithms.
- Choose efficient data structure and apply them to solve problems.

D. Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Design modular algorithms to find solution for computational problem with time and space complexities using suitable linear data structure.	K3
CO2	Demonstrate the familiarity with tree data structure, rule to manipulate those, and their canonical applications.	K3
CO3	Solve unstructured problem using balanced search tree algorithm, hashing function.	K3
CO4	Implement an appropriate algorithm using graph ADT for an application.	K3
CO5	Apply sorting and searching techniques for real world problems.	K3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of Cos with Program Outcomes and Programme Specific Outcomes:

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	3	2				1	1		3	3		
CO2	3	3	3	2				1	1		3	3		
CO3	3	3	3	3				1	1		2	3		
CO4	3	3	3	3				1	1		3	3	2	
CO5	3	3	2	2				1	1		2	3	2	

3- High; 2-Medium; 1-Low

F. Course Contents

Unit 1 Introduction to Data Structure

L-9 Hours

Introduction: Dynamic aspects of operations on data, Characteristics of data, Creation, Manipulation and Operations on data, Data Structure and its types, Abstract Data Types (ADTs), Analysis of algorithms and its Types, Asymptotic Notation. Arrays: Allocation, Operations and Storage with one-dimensional arrays and multidimensional arrays. Stacks: Operations on Stacks, Applications of Stacks, Queues: Operations and its types. Linked lists: Operations and Types of linked list (singly, doubly and circularly linked list), Implementation of Stack and Queue using linked list, Applications of Linked List.

Case Study: Tower of Hanoi, Sparse matrix.

Unit 2 Trees

L-9 Hours

Introduction to Trees – Definitions and concepts, Representation of Binary Tree, Types of Binary tree, properties, structure and applications of binary tree, Binary Tree Traversal with Recursive and Non-Recursive - Expression Tree, Binary Search Trees – Definition, Properties and Operations of binary search tree. Priority Queue: Operations, Implementations- Heaps: properties and types of Heaps, Binary Heap: Representation, Declaration and Creation of Heap, Heapifying, Inserting and Deleting an element, Destroying Heap, Heap Sort. Case Study: Priority Queue in bandwidth management

Unit 3 Special Trees & Hashing

L-9 Hours

AVL Tree: Properties, Declarations, Rotations, Insertion, Deletion. B-Tree: Properties, Insertion, Deletion –Trie: Insertion, Deletion, Searching. Hashing - Separate Chaining – Open Addressing – Linear Probing – Quadratic Probing – Double Hashing –Rehashing. Case Study: DNA sequence matching

Unit 4 Graphs

L-9 Hours

Introduction to Graphs – Operations on graphs – Representation of Graph – Topological Sort - Graph Traversal - Depth first search – Breadth First Search – Minimum Spanning Tree – Prim's Algorithm- Kruskal's Algorithm – Disjoint subsets and Union-Find algorithms – Shortest Path Algorithms – Unweighted Shortest Paths –Dijkstra's Algorithm. Applications of Graphs. Case Study: Representing Google maps, Recommendations on e-commerce websites.

Unit 5 Sorting, Searching

L-9 Hours

Searching Techniques: Linear Search, Binary Search.Sorting techniques: Insertion Sort,Selection sort, Merge sort, Quick Sort, Shell sort, Bubble sort, Counting sort, Bucket sort, Radix sort. Case Study: Tim Sort.

Total: 45 Hours

G. Learning Resources

i. Text Books:

1. M. A. Weiss, "Data Structures and Algorithm Analysis in C", Second Edition, Pearson Education, Reprinted edition 2013.[Unit 1,2,3]
2. Ellis Horowitz, Sartaj Sahni, Susan Anderson Freed, "Fundamentals of Data Structure in C", Universities Press,2017. [Unit 4,5]

ii. Reference Books:

1. V. Aho, J. E. Hopcroft, and J. D. Ullman, “Data Structures and Algorithms”, Pearson Education, First Edition, Reprint 2017.
2. R. F. Gilberg, B. A. Forouzan, “Data Structures”, Second Edition, Thomson India Edition, 2007.
3. Seymour Lipschutz, “Data Structures with C”, Schaum’s outlines, Tata McGraw Hill, 2017.

iii. Online References:

1. “Introduction to Data Structures and Algorithms” 31.12.2009. Accessed on Apr. 20, [Online]. Available: <https://nptel.ac.in/courses/106/102/106102064/>.
2. “Data Structures and algorithms”. Accessed on: Apr. 15, 2021 [Online]. Available: <https://www.coursera.org/specializations/data-structures-algorithms>.
3. “Data structures Visualization” 2011. Accessed on: Apr. 15, 2021 [Online]. Available: <https://www.cs.usfca.edu/~galles/visualization/Algorithms.html>
4. “Data Structures Fundamentals” 2021, Accessed on: Apr. 20,2021 [Online]. Available: <https://www.edx.org/course/data-structures-fundamentals>

COURSE CODE	COURSE TITLE	L	T	P	C
10211CS103	OPERATING SYSTEMS	3	0	0	3

A. Preamble

This course provides discussion about Operating systems kernel architecture, address spaces, system call interface, processes & threads, inter process communication, CPU scheduling deadlock, scheduling, main memory, virtual memory and file systems. This course aims to understand how Operating system makes computer system convenient to use in an efficient manner.

B. Prerequisite Course

10211CS102 - Data Structures

C. Course Objective

Learners are exposed to:

- Recognize operating Systems basic concepts, structures and System Calls.
- Apply process management concepts and Synchronization techniques.
- Learn different CPU scheduling algorithms and Deadlock Handling methods.
- Understand Various Memory and File management techniques.

D. Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Describe the operating system structures, operations and system calls.	K2
CO2	Demonstrate process management concepts and process synchronization methods for real time problems.	K3
CO3	Illustrate CPU scheduling algorithms and deadlock handling methods for the given situation.	K2
CO4	Explain the concepts of various memory management techniques.	K2
CO5	Discuss the concepts of disk management and file system interface.	K2
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of COs with Program outcomes and Programme Specific Outcomes:

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3													
CO2	2	3	2	1								2		
CO3	1	3	3	1								2		
CO4	2	3	3									3		
CO5	3	3	3									2		

3- High; 2-Medium; 1-Low

F. Course Contents

Unit 1 Structure and Overview of Operating Systems

L-8 Hours

Operating system overview: Objectives – functions - Computer System Organization - Operating System Operations- System Calls, System Programs-Operating- System Structure: Traditional UNIX system structure-The Mac OS X structure - Architecture of Google's Android.

Unit 2 Process Management

L-10 Hours

Processes: Process Concept – Threads - Process Scheduling - Operations on Processes – Inter process Communication - Communication in Client–Server Systems-Pipes (RPC, Pipes) - Process Synchronization: The Critical-Section Problem - Semaphores – Mutex Locks- Classic Problems of Synchronization – Monitors. Case Study: Windows Threads and Linux Threads

Unit 3 CPU Scheduling and Deadlock Management

L-9 Hours

CPU Scheduling: Scheduling Criteria - Scheduling Algorithms. Deadlocks: Deadlock Characterization - Methods for Handling Deadlocks - Deadlock Prevention - Deadlock Avoidance - Deadlock Detection - Recovery from Deadlock. Case Study: Real Time CPU scheduling

Unit 4 Memory Management

L-9 Hours

Main Memory: Swapping - Contiguous Memory Allocation, Segmentation, Paging - Structure of the Page Table - Virtual Memory: Demand Paging - Page Replacement - Allocation of Frames – Thrashing. Case study: Virtual machine

Unit 5 Storage Structure & File Systems

L-9 Hours

Mass Storage Structure: Disk Structure - Disk Scheduling - Disk Management-Structure - File-System Interface: File Concepts -Directory Structure - File Sharing – Protection- File Allocation Methods-NFS. Case study: Recovery in Windows.

Total: 45 Hours

G. Learning Resources

i. Text Book:

1. Abraham Silberschatz, Peter Baer Galvin and Greg Gagne, “Operating System Concepts”, 9th Edition, John Wiley and Sons Inc., 2016.[Unit 1-5]

ii. Reference Books:

1. Andrew S. Tanenbaum, “Modern Operating Systems”, 4th Edition, Prentice Hall, Wesley, 2014.
2. Operating Systems: Principles and Practice (2nd Edition), by Thomas Anderson and Michael Dahlin.

iii. Online Resources:

1. “Learn Operating System” Accessed on: April 21, 2021 [Online]. Available:https://www.tutorialspoint.com/operating_system/index.htm
2. Prof.Santanu Chattopadhyay, “Operating System Fundamentals” July 25,2019. Accessed on: April 21, 2021 [Online]. Available:<https://nptel.ac.in/courses/106/105/106105214/>

COURSE CODE	COURSE TITLE	L	T	P	C
10211CS129	MODERN COMPUTER ARCHITECTURE	3	0	0	3

A. Preamble

This course specially focuses on the basic structure and functional units of a digital computer. It will explore the interaction of hardware and software and consider the efficient use of hardware to achieve high performance. Moreover, this course describes the analysis of different computer architectures and their applications. The topics of distributed computing architecture, High Performance Computing and CUDA Programming model are presented.

B. Prerequisite Course

10211CS201-Digital Electronics

C. Course Objectives

Learners are exposed to:

- Understand basic principles of Computer Systems.
- Explain various logic design techniques and their applications.
- Use high performance computing architecture.
- Implement CUDA Programming model for Parallel computing

D. Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Understand the organization of the Control unit, Arithmetic and Logical unit, Memory unit and the I/O unit.	K2
CO2	Analyse different computer architectures and applications.	K3
CO3	Explain modern design structures of Pipelined and Multiprocessors systems.	K2
CO4	Know distributed computing architecture and high-performance computing.	K2
CO5	Implement the CUDA Programming model for Parallel computing.	K3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO 1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	3			1						1		
CO2	3	3	2			1						1		

CO3	3	3				1								
CO4	3	3				1								
CO5	3	3	3	3	1	1								3

3- High; 2-Medium; 1-Low

F. Course Contents

Unit 1 Basic Structure of a Computer System

9 Hours

Functional Units, Basic Operational Concepts, Instructions, Instruction's cycle, Memory reference instruction, I/O interruption, Adder and Subtractor circuits, Booth Multiplication Algorithm, Pipelining Review, control hazards and the motivation for caches, cache characteristics and superscalar architecture basics.

Unit 2 Multi Core Architecture

9 Hours

Memory technologies, hierarchical memory systems, the locality principle and caching, direct- mapped caches, block size, cache conflicts, associative caches, write strategies, advanced optimisations, performance improvement techniques, DRAM – organization, access techniques, scheduling algorithms and signal systems. Tiled Chip Multicore Processors (TCMP), Network on Chips (NoC), NoC router – architecture, design, routing algorithms and flow control techniques, Advanced topics in NoC and storage – compression, prefetching, QoS.

Unit 3 Distributed Computing Systems and Concurrency

9 Hours

Relation to Parallel Multiprocessors/multicomputer Systems, Distributed and Concurrent Programs, Message Passing vs. Shared Memory Systems, Synchronous vs. Asynchronous Executions, Design Issues and Challenges, Distributed Computing Technologies, Clocks and Synchronization, Coordination and Agreement Algorithms, Global State and Distributed Transactions.

Unit 4 High Performance Computing (HPC)

9Hours

HPC Architecture, Parallel Processing, Parallel Memory Models, Data vs. Task Parallelism, High Throughput Computing, Vectorization, Multithreading.

Unit 5 High Performance Computing with CUDA

9 Hours

CUDA programming model, Basic principles of CUDA programming, Concepts of threads and blocks, GPU and CPU data exchange.

Total: 45 Hours

G. Learning Resources

i. Text Books:

1. Hamacher, V.C., Vranesic, Z.G., and Zaky, S.G., “Computer Organization”, 5/e. McGraw-Hill. 2014. [Unit- 1,2]
2. Patterson, D.A., and Hennessy, J.L. , “Computer Architecture : A Quantitative Approach ”, Morgan Kaufmann Publishers, , Inc.2017.[Unit -3, 4,5]

ii. Reference Books:

1. Patterson, D.A., and Hennessy, J.L. , “Computer Organization and Design: The Hardware/Software Interface”, Morgan Kaufman Publishers, 5th Edition, Inc.2013
2. Stalling W, “Computer Organization and Architecture ”, Pearson Education India. 10th Edition, 2016
3. Parallel Computer Architecture: A Hardware/Software Approach David Culler and J.P. Singh with Anoop Gupta, Morgan Kaufmann, 1998.

iii. Online References:

1. Computer Architecture and Organisation” June, 8, 2017. Accessed on April, 20, 2021 [Online]. Available: <https://nptel.ac.in/courses/106/105/106105163/>
2. “Introduction to Computer Architecture “ January, 1, 2017. Accessed on June, 30, 2021 [Online]. Available: <https://nptel.ac.in/courses/106/102/106102157/>
https://onlinecourses.nptel.ac.in/noc20_cs41/preview
3. High-performance-computing[Online] Available: <https://www.coursera.org/learn/introduction-high-performance-computing#syllabus>
4. “GPU Architectures and Programming“ April, 17, 2020. Accessed on December, 14, 2022 Available: https://onlinecourses.nptel.ac.in/noc20_cs41/preview

COURSE CODE	COURSE TITLE	L	T	P	C
10211CS105	COMPUTER NETWORKS	3	0	0	3

A. Preamble

This course is to provide students with an overview of the concepts and fundamentals of computer networks. Topics to be covered include: data communication concepts and techniques in a layered network architecture, communications switching and routing, types of communication, network congestion, network topologies, network configuration and management, network model components, layered network models (OSI reference model, TCP/IP networking architecture) and their protocols, various types of networks and their protocols.

B. Prerequisite Course

10210CS101 -Problem Solving Using C

C. Course Objectives

Learners are exposed to:

- Build an understanding of the fundamental concepts of computer networking, protocols, architectures, and applications.
- Understand the division of network functionalities into layers.
- Learn the flow control and error control mechanisms.
- Obtain the skills of subnetting and routing mechanisms.
- Explain the transport layer and application layer protocols.

D. Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Discuss the basic fundamentals of networks for data communication and protocol Layering.	K2
CO2	Know the various techniques and standards of analog and digital data communication and its standards.	K2
CO3	Apply various error detection, correction and flow control techniques for efficient transmission of data.	K3
CO4	Interpret the functions and protocols of network and transport layer.	K2
CO5	Outline the various protocols of application layer.	K2
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of COs with Program outcomes and Programme Specific Outcomes:

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2		2								3		
CO2	3	2		2								2		
CO3	3	3		2								2		
CO4	2	3		2								2		
CO5	3	2										3		

3- High; 2-Medium; 1-Low

F.Course Contents

Unit 1 Introduction

L- 9 Hours

Data Communication: system components (Passive and Active)-Topology-Network Models: -Physical and Logical Addressing- Data and Signals - Transmission Impairment - Data rate and Channel capacity- OSI-TCP /IP- Quality of Service.

Unit 2 Physical Layer

L- 9 Hours

Transmission Media: Guided Media - Unguided Media -Digital Transmission - Digital-To -Digital Conversion - Analog Transmission - Digital-To-Analog Conversion - Wireless - Wired LANs: Ethernet - Token ring - Connecting Devices – Switching techniques – Bandwidth Metrics.

Unit 3 Data Link Layer

L-9 Hours

Link Layer: Types of errors –Error detection- VRC, LRC, CRC andChecksum techniques - Data Forward and backward error correction - Hamming code. Flow control: stop and wait- sliding window protocol, Error control: Stop and wait ARQ- Go-Back-N ARQ- Selective Repeat ARQ Protocols- Asynchronous and Synchronous Protocol- Logical Link Control protocols: HDLC, ALOHA and FDDI.

Unit 4Network layer, Transport Layer

L-9 Hours

Logical Addressing: Addressing Methods - IPV4 - IPV6 - Frame Format - IPV6 implementation scenarios and challenges – ARP– RARP - DHCP – ICMP - Unicast Routing Protocols – Intra and Interdomain Routing - Distance Vector Routing - Link State Routing-Transport Layer: Process-to-Process Delivery: UDP – TCP - Congestion Control Techniques - Techniques to Improve QoS.

Unit 5 Application Layer and Network Security

L-9 Hours

Session Layer: Functions-Protocols: Winsock, Socket-Remote Procedure Call-Presentation Layer: Functions- Compression Techniques: Lossy and Lossless - Application layer protocols: REMOTE LOGGING - TELNET -EMAIL– DNS – FTP -HTTP – SDN-Introduction-Security Fundamentals.

Total: 45 Hours

G.Learning Resources

i. Text Books:

1. Behrouz Forouzan, “Data Communications and Networking”, Tata McGraw Hill, 5th Edition,2021. (Reprinted) [Unit 1- 5]

ii. Reference Books:

1. Kurose and K.W. Ross, “Computer Networking: A Top-Down Approach Featuring the Internet”, Addison-Wesley,2017.

2. Andrew S. Tanenbaum, David J Wetherall, Computer Networks, 5th Edition, Pearson Edu, 2013.
3. Foundations of Modern Networking: SDN, NFV, QoE, IoT, and Cloud, 5th Edition, 2017

iii. Online References:

1. “Computer Communication Specialization” Course Era [online] 2019. Available: <https://www.coursera.org/specializations/computer-communications>
2. “Introduction to Networking” edx [online] 2020, Available: <https://www.edx.org/course/introduction-to-networking>
3. “Computer Networking” Udacity [online] 2020, Available: <https://www.udacity.com/course/computer-networking--ud436>
4. “Computer Networks and Internet Protocol” NPTEL [online] 2018, Available: <https://nptel.ac.in/courses/106/105/106105183/>

COURSE CODE	COURSE TITLE	L	T	P	C
10211CS130	FUNDAMENTALS OF COMPUTER NETWORKS	3	0	0	3

A. Preamble

This course is to provide students with an overview of the concepts and fundamentals of computer networks. Topics to be covered include: data communication concepts and techniques in a layered network architecture, communications switching and routing, types of communication, network congestion, network topologies, network configuration and management, network model components, layered network models (OSI reference model, TCP/IP networking architecture) and their protocols, various types of networks and their protocols.

B. Prerequisite Courses

10210CS101-Problem Solving Using C

C. Course Objectives

Learners are exposed to

- Build an understanding of the fundamental concepts of computer networking, protocols, architectures, and applications.
- Understand the division of network functionalities into layers.
- Learn the flow control and error control mechanisms.
- Obtain the skills of subnetting and routing mechanisms.
- Explain the transport layer and application layer protocols.

D.Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Explain the basic fundamentals of networks and discuss the various techniques and standards of analog and digital data communication.	K2
CO2	Apply various error detection, correction and flow control techniques for efficient transmission of data.	K3
CO3	Interpret the functions and protocols of network layer.	K3
CO4	Explain the functions and protocols of transport layer.	K3
CO5	Understand the various protocols of application layer.	K2
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2		2				2				2		
CO2	3	2		2				2				2		
CO3	3	3		2				2				3		
CO4	2	3		2				2				2		
CO5	3	2						2						

3- High; 2-Medium; 1-Low

F. Course Contents**Unit 1 Introduction and Physical Layer****L-9 Hours**

ISO/OSI - TCP/IP Protocol suite, Data Communication-system components – Topology-Transmission Impairment
Transmission Media: Guided Media - Unguided Media - Wireless - Wired LANs: Ethernet - Token ring - Connecting Devices – Switching techniques – Bandwidth Metrics.

Unit 2 Data Link Layer**L-9 Hours**

Types of errors –Error detectionand correction techniques, Flow control mechanism, Multiple access protocols, Link layer addressing. Media Access Control – frame format, types, cloning. IEEE 802.3 – frame format, types.

Unit 3 Network layer**L-9 Hours**

Logical Addressing: Addressing Methods –Classfull addressing - IPV4 – Public and private addressing.Classless addressing - IPV6 - Frame format –Subnetting -Supernetting, CIDR.Inter/Intra networks,ARP– DHCP – ICMP – IGMP - SNMP. Unicast Routing Protocols – Intra and Interdomain Routing - Distance Vector Routing - Link State Routing. RIP, OSPF.

Unit 4 Transport Layer**L-9 Hours**

Transport Layer: Process-to-Process Delivery: UDP – TCP - Congestion Control and Flow control Techniques - Quality of Service - Techniques to Improve QoS, merits and de-merits.

Unit 5 Application Layer**L-9 Hours**

Session Layer: Functions-Protocols: Winsock, Socket-Remote Procedure Call-Presentation Layer: Functions- Compression Techniques: Lossy and Lossless - Application layer protocols: REMOTE LOGGING – TELNET/SSH -EMAIL– DNS – FTP/TFTP– IPSec - HTTP/HTTPS – SDN.

Total: 45 Hours**G. Learning Resources****i.Text Books:**

1. Behrouz Forouzan, “Data Communications and Networking”, Tata McGraw Hill, 5th Edition,2021. (Reprinted) (Unit 1 to 5)

ii.Reference Books:

- 1.Kurose and K.W. Ross, “Computer Networking: A Top-Down Approach Featuring the Internet”, Addison-Wesley,2017
2. Andrew S. Tanenbaum, David J Wetherall, Computer Networks, 5th Edition, Pearson Edu, 2013.
3. Foundations of Modern Networking: SDN, NFV, QoE, IoT, and Cloud,5th Edition,2017

iii.Online References:

- 1.“Computer Communication Specialization” Course Era [online] 2019.
Available:<https://www.coursera.org/specializations/computer-communications>
- 2.”Introduction to Networking” edx [online]2020,Available:
<https://www.edx.org/course/introduction-to-networking>
- 3.“Computer Networking”Udacity [online] 2020, Available:
<https://www.udacity.com/course/computer-networking--ud436>
- 4.“Computer Networks and Internet Protocol” NPTEL [online] 2018, Available:
<https://nptel.ac.in/courses/106/105/106105183/>

COURSE CODE	COURSE TITLE	L	T	P	C
10211CS106	FORMAL LANGUAGES AND AUTOMATA THEORY	3	0	0	3

A.Preamble

This course deals with the concepts of automata theory, formal languages, grammar, computability and decidability. Students learn concrete implementations, manipulations of discrete structures and their use in design and analysis of non-trivial problems for a given computational task. This course can be applied in Compilers, Artificial Intelligence, Robotics and Natural Language Processing.

B.Prerequisite Course : 10211CS101- Discrete Mathematics

C.Course Objectives

Learners are exposed to:

- Understand overview of theoretical foundations of computer science from the perspective of formal languages.
- Illustrate various automata to solve problems in computing.
- Familiarize Regular grammars, context free grammar, recursive and recursively enumerable languages.

D.Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Design the finite automata to recognize the regular languages.	K3
CO2	Construct regular expression for regular grammar and its equivalence with finite automata.	K3
CO3	Develop push down automata and context-free grammar representations for context-free languages.	K3
CO4	Model Turing Machines for accepting recursive languages and its capabilities.	K3
CO5	Apply the notions of decidability and undecidability to examine complex problems.	K3

Knowledge Level (Based on revised Bloom's Taxonomy)

K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create

E.Correlation of COs with Program outcomes and Programme Specific Outcomes:

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	3		3							3		
CO2	3	3	3		3							3		
CO3	3	3	2	1	3							3		
CO4	3	3	3	3	3							3		
CO5	3	3	2	2								3		

3- High; 2-Medium; 1-Low

F.Course Contents

Unit 1 Formal Languages and Finite Automata

L-10Hours

Introduction to automata and automata theory, Basic concepts of formal Languages, Finite Automata – DFA, NFA, Epsilon NFA, Equivalence of DFA, NFA and Epsilon NFA, Minimization of Automata, FA with output.

Case Studies: Implementation of Spell checkers, Text search, Text editors, Simulation of FA using JFLAP.

Unit 2 Regular Languages and Expressions

L-8 Hours

Regular Grammar, Regular Expressions, Converting Regular Expression to Epsilon NFA, Equivalence of regular expressions and NFA with Epsilon moves, Converting DFA to Regular Expressions, Pumping Lemma for Regular Languages, Applications of Pumping Lemma, Closure Properties of Regular sets.

Case Studies: Pattern for Mobile numbers with country code, email address.

Unit 3 Context Free Languages and Push Down Automata

L-10 Hours

Context-Free Languages and Grammar, Derivation trees, Ambiguity, Simplification of CFG, Chomsky Normal Form, Greibach Normal Forms, Deterministic Push Down Automata, Non-Deterministic Pushdown Automata (NPDA), Equivalence of acceptance by final state and empty stack in PDA, Equivalence between NPDA and CFG, Closure properties of CFLs, Pumping Lemma for CFLs.

Case Studies: Tower of Hanoi, Evaluating Arithmetic expression, Transaction Process System.

Unit 4 Turing Machine

L-9 Hours

Context-sensitive Grammar, Turing Machine (TM) – Basics and formal definition, Instantaneous Description, TMs as language acceptors, TMs as Transducers, Designing Turing Machines, Variants of TMs, Checking of Symbols, Encoding a Turing Machine and Universal Turing Machine.

Case Study: Lambda Calculus

Unit 5 Recursive and Recursively Enumerable Languages

L-8 Hours

Recursive Functions, Recursive languages and Recursively Enumerable Languages, Properties of Recursively Enumerable Languages and Recursive Languages, Decidability and Halting Problem, Reduction, P and NP, NP-completeness, Post Correspondence Problem, Rice Theorem and Chomsky Hierarchy.

Case Studies: Knapsack Problem, SAT Problem using Turing machine.

Total:45 Hours

G.Learning Resources

i. Text Books:

1. John E Hopcroft, Rajeev Motwani and Jeffrey D. Ullman, Introduction to Automata Theory, Languages, and Computation, 3/e, Pearson Education, 2013. [Unit 1– 3]
2. John C. Martin, Introduction to Languages and the theory of computation, 4/e, TMH, 2011 [Unit 4,5]

ii. Reference Books:

1. Michael Sipser, Introduction to the Theory of Computation, 3rd edition, Cengage Publishers, 2013
2. Peter Linz, “An Introduction o Formal Language and Automata”, 5th edition, Cathleen Sether Publishers, 2012.
3. S.P.Eugene Xavier, Theory of Automata, Formal Languages and Computation, New Age International Publishers, 2005.

iii. Online References:

1. “Theory Of Computation” Sep. 09, 2016. Accessed on: Feb. 16, 2021, [Online]. Available: <https://nptel.ac.in/courses/106/104/106104148/>
2. “Automata Theory”, Accessed on: Apr. 21, 2021, [Online]. Available: <https://www.edx.org/course/automata-theory>
3. “Lambda Expressions” [Online] Accessed on: Apr. 28, 2021, <https://www.udemy.com/course/java-8-functional-programming-with-lambda-expression/>

COURSE CODE	COURSE TITLE	L	T	P	C
10211CS107	COMPILER DESIGN	3	0	0	3

A.Preamble

This course introduces the basic concepts and mechanisms that are employed in language translators, with emphasis on compilers. It aims to provide the theory and practice of compilation, in particular, the lexical analysis, parsing, intermediate code generation, techniques of code optimization and approaches towards code generation.

B.Prerequisite Course

10211CS106 - Formal Languages and Automata Theory

C.Course Objectives

Learners are exposed to:

- Understand the structure of a compiler with the sequence of translation of input for each phase.
- Analyze the regular languages to describe the lexical elements of a programming language.
- Develop the syntactic structure of a programming language using the context free language and design the top-down and bottom-up parsers.
- Break down the statements and expressions to form and implement the intermediate code.
- Optimize the intermediate code of a program to machine-specific instructions to generate target code.

D.Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Interpret the different phases of compilation with compile time error handling and design the lexical analyzer for a language.	K3
CO2	Develop appropriate top down and bottomup parser to produce parse tree representation of the input.	K3
CO3	Generate intermediate code for decision control and looping control statements in high level language.	K3
CO4	Apply various optimization techniques with basic blocks of the intermediate code through global data flow analysis.	K3
CO5	Design DAG representation for basic blocks and target low level language for high level language	K3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of COs with Program outcomes and Programme Specific Outcomes:

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	1		2							3		
CO2	3	2	1		1							3		
CO3	3	3										3		
CO4	3	3										3		
CO5	3	3	1									3		

3- High; 2-Medium; 1-Low

F.Course Contents

Unit 1 Compilers and Lexical Analysis

L-9 Hours

Compilers – Interpreters – Analysis of the source program –Cousins of the Compiler – Grouping of Phases – Phases of a compiler – Compiler construction tools – Lexical Analysis – Role of Lexical Analyzer – Input Buffering – Recognition of Tokens – Specification of Tokens. Design of a Lexical Analyzer Generator.

Case Study: Lexical Analyzer for separating tokens, Counting whitespace, special characters and newline character, Pattern Matching of strings – use relevant tool.

Unit 2 Syntax Analysis and Parser

L-9 Hours

Need and Role of the parser –Writing Grammars – Context-Free Grammars – Top Down parsing – Recursive Descent Parsing – Predictive Parsing – Bottom-up parsing – Shift Reduce Parsing – Operator Precedence Parsing – LR Parsers – SLR Parser – Canonical LR Parser – LALR Parser.

Case Study: Specification for desktop calculator, Error recovery of ambiguous arithmetic grammar, Syntax Analyzer for looping construct - use relevant tool.

Unit 3 Intermediate Code Generation

L-9 Hours

Intermediate languages – Implementation of Three Address Code – Types and Declarations – Type Checking – Assignment Statements – Boolean Expressions – Switch Case Statements – Back patching – Procedure calls.

Case Study: Intermediate code for Array references with subscripts, Intermediate code for decision making and looping constructs of C, Intermediate code for Expression Parsing.

Unit 4 Code Optimization

L-9 Hours

Principal Sources of Optimization – Introduction to Data Flow Analysis – Constant propagation – Loops in Flow Graphs – Runtime Environments – Source Language issues – Storage Organization – Storage Allocation – Access to Nonlocal data - Local optimization- Global optimization.

Case Study: Implement code optimization methods using relevant data structures.

Unit 5 Code Generation

L-9 Hours

Issues in the design of code generator – The target machine – Register Allocation and Assignment- Basic Blocks and Flow Graphs – Optimization of basic Blocks – A Simple Code Generator – DAG representation of Basic Blocks – Code Generation from DAG - Heuristic Reordering - Labeling - Next-use Information – Peephole Optimization.

Case Study: Graph coloring application using Register Allocation, Generate Target code for Binary tree operations.

Total: 45 Hours

G.Learning Resources

i. Text Books:

1. Alfred V. Aho, Monica S. Lam, Ravi Sethi, and Jeffrey D. Ullman, Compilers –Principles Techniques and Tools, Pearson, Second Edition, 2013. [Unit 1-5].

ii. Reference Books:

1. Kenneth C. Loudon, Compiler Construction –Principles and Practice, Cengage Learning Indian Edition, 2006.
2. Grune D, Bal H, Jacobs C, Langendoen K, Modern Compiler Design, Wiley, 2000
3. Cooper K, Torczon L, Engineering a Compiler, Morgan Kaufmann, Second Edition, 2011

iii. Online References:

1. “COMPILER DESIGN” Sep. 25, 2014. Accessed on: Feb. 16, 2021 [Online]. Available: <https://nptel.ac.in/courses/106/105/106105190/>
2. “LEX & YACC TUTORIAL by Tom Niemann” Oct. 24, 2013. Accessed on: Apr. 20, 2021 [Online]. Available: <https://cse.iitkgp.ac.in/~bivasm/notes/LexAndYaccTutorial.pdf>
3. “Compiler Design”, Nov. 2019. Accessed on: Apr. 20, 2021 [Online]. Available: <https://www.udemy.com/course/compiler-design-n/>

COURSE CODE	COURSE TITLE	L	T	P	C
10211CS109	MICROPROCESSOR AND MICROCONTROLLER	2	0	0	2

A.Preamble

Microprocessors are a core course in the stream of Computer Science & Engineering. The objective of this course is to explore the concepts of hardware and software part of a microprocessor. It includes the architecture of 8086 Microprocessor and latest ARM processor architecture. This course covers ARM processor programming with interfacing techniques and equivalent assembly language code for high level language instructions like 'C'.

B.Prerequisite Course

10211CS201–Digital Electronics

C.Course Objective

Learners are exposed to:

- Explain the architecture and operational features of the 8086 microprocessor including signals, system configuration, and interrupts.
- Develop assembly language programs for the 8086 microprocessor using addressing modes and instruction sets.
- Describe the architecture and functional components of ARM Cortex-M processors including memory organization and interrupt handling.
- Implement ARM processor programs using addressing modes, instruction sets, and control instructions.
- Design ARM-based interfacing applications using GPIO, timers, interrupts, and serial communication interfaces.

D.Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Know the architecture of 8086 microprocessor and its basic configuration.	K2
CO2	Develop assembly language programs for the given problems using 8086 processor.	K3
CO3	Understand ARM processor architecture and the principle of its debug system.	K2
CO4	Develop programs for the problems specified using ARM processor instructions.	K3
CO5	Establish interfacing of I / O devices with ARM processor to develop real time applications.	K3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of COs with Program outcomes and Programme Specific Outcomes:

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3												
CO2	3	2			3							2		
CO3	3	2												
CO4	3	2			3							3		
CO5	3		3		3							3		

3- High; 2-Medium; 1-Low

F. Course Contents**Unit 1 Microprocessor Architecture – 8086****L - 6 Hours**

Introduction – Types of computers – Evolution of Microprocessor – 8086 Microprocessor architecture – 8086 Programming method — program development tools – Interrupts and interrupt service procedures - 8086 signals – Basic configurations – System connections –Timing – Troubleshooting – Assembler directives - Introduction to OpenRISC Architecture.

Unit 2 Programming In 8086**L - 6 Hours**

Assembly language programming – Addressing modes – Instruction set: Data transfer instructions – arithmetic and logical instructions – Branch instructions — Loop instructions – While – Do programs – Strings – Procedures – Macros.

Unit 3 Arm Processor Architecture**L - 6 Hours**

Cortex – M architecture – Block diagram –pipelined architecture – memory address map – bus system and bus matrix – memory and peripherals: memory endianness – bit banding – system stack architecture – debug system: AHB access port, FPB, DWT, ITM, ETM, TPIU – conditional execution – interrupts: exception and interrupt handling.

Unit 4 Arm Processor Programming**L - 6 Hours**

Addressing modes – ARM instruction sets: thumb instruction set – ADD and MOV data processing instructions – LDR and STR memory access instructions – data processing instructions – bitfield instructions – branch and control instructions – recursive functions If-Then conditional instructions – reset sequence – instruction encoding.

Unit 5 Arm Processor Interfacing**L - 6 Hours**

TM4C123 microcontroller block diagram – GPIO features – peripherals on the memory map – GPIO pin configuration – LED and switch interfacing – keypad interfacing – I/O synchronization – polling – interrupt driven – processor interrupt configuration – interrupt based interfacing – Timer: timer as I/O device – external event counter – input capture device – general purpose timer modules – serial communication interface – I²C interface – SPI interface – CAN interface.

Total: 30Hours (L)

G.Learning Resources

i. Text Books:

1. Douglas V. Hall, “Microprocessors and Interfacing”, TMH, Revised second edition, 2005. [Unit 1-2]
2. Muhammad Tahir, Kashif Javed, “ARM Microprocessor Systems: Cortex-M Architecture, Programming, and Interfacing”, CRC Press, Taylor & Francis group, 2017. [Unit 3-5]

ii. Reference Books:

1. Andrew N. Sloss, “ARM system developer’s guide – Designing and optimizing system software”, ELSEVIER, 2004.
2. Charles M. Gilmore, “Microprocessors: Principles and applications”, TMH, Second edition, 1995.
3. Jonathan W. Valvano, “Embedded microcomputer systems: Real time interfacing”, CENGAGE Learning, Third edition, 2012.
4. Steve Furber, “ARM system-on-chip architecture”, Pearson education, Second edition, 2015.

iii. Online References:

1. “Microprocessor interfacing”, Accessed on Sep 20, 2022 [Online], Available: https://onlinecourses.nptel.ac.in/noc20_ee11/preview.
2. “Microprocessors and Digital systems”, Accessed on Sep 21, 2022 [Online], Available: <https://archive.org/details/microprocessorsd0000hall/page/n3/mode/2up>.
3. “ARM processor architecture”, Accessed on Sep 21, 2022 [Online], Available: https://www.cs.ccu.edu.tw/~pahsiung/courses/esc/notes/ESD_03_ARM_Architecture.pdf.

COURSE CODE	COURSE TITLE	L	T	P	C
10211CS111	IMAGE PROCESSING	2	0	0	2

A.Preamble

Image processing is a core course in the stream of Computer Science & Engineering. This course provides an introduction to basic concepts, methodologies and algorithms of digital image processing focusing on the following two major problems concerned with digital images: (1) image enhancement and restoration for easier interpretation of images, and (2) image analysis and object recognition. Some advanced image processing techniques (e.g., wavelet and multiresolution processing) will also be studied in this course.

B.Pre-requisite course

10210MA101- Linear Algebra for Computing

C.Course Objectives

Learners are exposed to:

- Explain the fundamentals of digital image processing, including image acquisition, sampling, quantization, and pixel relationships.
- Apply image transforms and enhancement techniques in spatial and frequency domains for improving image quality.
- Analyze and implement image restoration and reconstruction techniques for recovering degraded images.
- Apply image compression techniques including lossless and lossy methods for efficient image storage and transmission.
- Implement image segmentation and representation techniques for object detection and feature extraction in images.

D.Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Explain the theoretical foundations for image processing techniques.	K2
CO2	Apply image transformations and enhancement techniques to improve the quality of real time images.	K3
CO3	Compare various filters for image restoration and construction techniques.	K2
CO4	Interpret the various image compression techniques.	K3
CO5	Use the image segmentation and representation techniques to detect object.	K3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E.Correlation of COs with Program outcomes and Programme Specific Outcomes:

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3				1									1
CO2	3	3		3	2							3		1
CO3	3	3			1									1
CO4	2	3		3	1							3		1
CO5	3	2		2								2		

3- High; 2-Medium; 1-Low

F.Course Contents

Unit 1 Digital Image Fundamentals

L-6 Hours

Need for Digital Image processing - Fundamentals steps in Digital Image Processing, Components of Image Processing Systems, Elements of visual perception - Light and Electromagnetic spectrum, Image sensing and acquisition, Simple image formation model, Image file formats, Image sampling and quantization, Basic relationship between pixels.

Case study: Basic operation on image

Unit 2 Image Transforms and Enhancement

L- 6 Hours

Properties and Examples of 2D Orthogonal and Unitary Transforms, Hadamard Transform, Haar Transform, Fourier Transform, KL Transform, Discrete wavelet Transform- and its application in Compression, Image Enhancement in spatial and frequency domain: Histogram processing, Smoothing, Sharpening. Case study: Medical image diagnosis

Unit 3 Image Restoration and Reconstruction

L-6 Hours

Image Restoration: Image Degradation and Observation Model, Unconstrained and constrained restorations, Detector and Recorder model noise model, Inverse filtering, Pseudo inverse filtering, Wiener Filtering, SVD, Blind Deconvolution, Image Reconstruction from Projections, Back projection operator- Projection Theorem, Principles of Computed Tomography, Projections and the Radon transform. Case study: Biometric analysis

Unit 4 Image Compression

L-6 Hours

Image Compression: Need for data compression - Fundamentals, Compression Models, Loss Less and Lossy. Loss Less: Variable Length, LZW coding, Huffman, Lossy: Predictive Coding, Transform coding, Wavelet coding, JPEG Standard. Case study: Climatic data analysis

Unit5 Image Segmentation And Representation

L-6 Hours

Fundamentals, Detection of Discontinuities, Point, link and line detection. Edge Linking and Boundary detection, Local and Global processing. Thresholding, Region based segmentation; Region growing, splitting and merging. Representation - Boundary descriptors, Simple Descriptors, Shape numbers, Regional descriptors, Simple and Topological Descriptors, Texture.

Part-1

Use Cases:

- To analyze the impact of image preprocessing techniques on real time data
- To analyze the effect of segmentation algorithms
- Analyze face image data through image processing techniques
- Plant crop detection using image segmentation
- Medical image analysis through morphological operations

Total: 30 Hours

G.Learning Resources

i. Text Books:

1. Rafael C Gonzalez, Richard E Woods, Digital Image Processing, Pearson Education, 4th edition, 2018. [Unit 1-5].

ii. Reference Books:

1. Rafael C. Gonzalez, Richard E. Woods, Steven Eddins,' Digital Image Processing using MATLAB', Pearson Education, Inc., 2004.
2. Rick Szelisk, “Computer Vision: Algorithms and Applications”, Springer 2011.
3. Malay K. Pakhira, “Digital Image Processing and Pattern Recognition”, First Edition, PHI Learning Pvt. Ltd., 2011.

iii. Online References:

1. “Fundamentals of Digital Image and Video Processing”, 2016, Accessed on June 20, 2021. [Online]. Available: <https://www.coursera.org/learn/digital>.
2. “Digital Image Processing”, 2017, Accessed on June 20, 2021, <https://nptel.ac.in/noc/courses/noc20/SEM2/noc20-ee75/>.

COURSE CODE	COURSE TITLE	L	T	P	C
10211CS118	IMAGE PROCESSING TECHNIQUES	3	0	0	3

A.Preamble

Image processing is a core course in the stream of Computer Science & Engineering. This course provides an introduction to basic concepts, methodologies and algorithms of digital image processing focusing on the following two major problems concerned with digital images: (1) image enhancement and restoration for easier interpretation of images, and (2) image analysis and object recognition. Some advanced image processing techniques (e.g., wavelet and multiresolution processing) will also be studied in this course.

B.Pre-requisite course

10210MA101- Linear Algebra for Computing

C.Course Objectives

Learners are exposed to:

- Explain the fundamentals of digital image processing, including image acquisition, sampling, quantization, and visual perception concepts.
- Apply image transform and enhancement techniques in spatial and frequency domains for improving image quality.
- Analyze and implement image restoration and reconstruction methods to recover degraded images.
- Apply lossless and lossy image compression techniques for efficient storage and transmission of images.
- Implement image segmentation and representation techniques for object detection and feature analysis.

D.Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Explain the theoretical foundations for image processing techniques.	K2
CO2	Apply image transformations and enhancement techniques to improve the quality of real time images.	K3
CO3	Compare various filters for image restoration and construction techniques.	K2
CO4	Interpret the various image compression techniques.	K3
CO5	Use the image segmentation and representation techniques to detect object.	K3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of COs with Program outcomes and Programme Specific Outcomes:

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3				1									1
CO2	3	3		3	2							3		1
CO3	3	3			1									1
CO4	2	3		3	1							3		1
CO5	3	2		2								2		

3- High; 2-Medium; 1-Low

F. Course Contents**Unit 1 Digital Image Fundamentals****L-9 Hours**

Need for Digital Image processing - Fundamentals steps in Digital Image Processing, Components of Image Processing Systems, Elements of visual perception - Light and Electromagnetic spectrum, Image sensing and acquisition, Simple image formation model, Image file formats, Image sampling and quantization, Basic relationship between pixels.

Case study: Basic operation on image

Unit 2 Image Transforms and Enhancement**L- 9 Hours**

Properties and Examples of 2D Orthogonal and Unitary Transforms, Hadamard Transform, Haar Transform, Fourier Transform, KL Transform, Discrete wavelet Transform- and its application in Compression, Image Enhancement in spatial and frequency domain: Histogram processing, Smoothing, Sharpening. Case study: Medical image diagnosis

Unit 3 Image Restoration and Reconstruction**L-9 Hours**

Image Restoration: Image Degradation and Observation Model, Unconstrained and constrained restorations, Detector and Recorder model noise model, Inverse filtering, Pseudo inverse filtering, Wiener Filtering, SVD, Blind Deconvolution, Image Reconstruction from Projections, Back projection operator- Projection Theorem, Principles of Computed Tomography, Projections and the Radon transform. Case study: Biometric analysis

Unit 4 Image Compression**L-9 Hours**

Image Compression: Need for data compression - Fundamentals, Compression Models, Loss Less and Lossy. Loss Less: Variable Length, LZW coding, Huffman, Lossy: Predictive Coding, Transform coding, Wavelet coding, JPEG Standard. Case study: Climatic data analysis

Unit 5 Image Segmentation And Representation**L-9 Hours**

Fundamentals, Detection of Discontinuities, Point, link and line detection. Edge Linking and Boundary detection, Local and Global processing. Thresholding, Region based segmentation; Region growing, splitting and merging. Representation - Boundary descriptors, Simple Descriptors, Shape numbers, Regional descriptors, Simple and Topological Descriptors, Texture.

Part-1**Use Cases:**

- To analyze the impact of image preprocessing techniques on real time data
- To analyze the effect of segmentation algorithms
- Analyze face image data through image processing techniques
- Plant crop detection using image segmentation
- Medical image analysis through morphological operations

Total: 45 Hours

G.Learning Resources

i. Text Books:

1. Rafael C Gonzalez, Richard E Woods, Digital Image Processing, Pearson Education, 4th edition, 2018. [Unit 1-5].

ii. Reference Books:

1. Rafael C. Gonzalez, Richard E. Woods, Steven Eddins,' Digital Image Processing using MATLAB', Pearson Education, Inc., 2004.
2. Rick Szelisk, “Computer Vision: Algorithms and Applications”, Springer 2011.
3. Malay K. Pakhira, “Digital Image Processing and Pattern Recognition”, First Edition, PHI Learning Pvt. Ltd., 2011.

iii. Online References:

1. “Fundamentals of Digital Image and Video Processing”, 2016, Accessed on June 20, 2021. [Online]. Available: <https://www.coursera.org/learn/digital>.
2. “Digital Image Processing”, 2017, Accessed on June 20, 2021, <https://nptel.ac.in/noc/courses/noc20/SEM2/noc20-ee75/>.

COURSE CODE	COURSE TITLE	L	T	P	C
10211CS120	MACHINE LEARNING	3	0	0	3

A. Preamble

This course introduces students to new and actively evolving interdisciplinary field of modern data analysis. The interplay between known data and unknown ones give rise to complex pattern structures and machine learning methods that are the focus of the study. In the course we will consider methods of Machine Learning and Data Mining Learning. It also includes dimensionality reduction techniques which help to deal with huge datasets.

B. Prerequisite Course

10210MA103- Probability, Statistics and Queuing theory

C. Course Objectives

Learners are exposed to:

- Apply the concepts of supervised and unsupervised learning algorithms for real time applications
- Experiment classification and association models to overcome the problem of over fitting
- Analyse and suggest appropriate machine learning approaches for various types of problems
- Demonstrate the aspects of simple perceptron and multi layer perceptron.

D. Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Examine the basic concepts of data mining and machine learning concepts	K2
CO2	Apply pre-processing techniques and explore the using visualization techniques	K3
CO3	Implement various algorithms of Regression and Classification.	K3
CO4	Use the concept of neural networks for learning linear and non-linear activation functions	K3
CO5	Experiment various clustering algorithms to solve real time problems	K3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of Cos with Program Outcomes and Programme Specific Outcomes:

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	2	2						1	1					
CO2	3	3	3	2	2			1	1			2	2	
CO3	3	3	3	2	2			1	1			2	2	
CO4	3	3	3	2	2			1	1			2	2	
CO5	3	2	2	2	2	1		1	1			2	2	

3- High; 2-Medium; 1-Low

F. Course Contents

Unit 1 Introduction

L-9 Hours

Introduction to Data mining and Knowledge Discovery – Data – Data Mining Functionalities – Architecture of Typical Data Mining systems - Classification of Data Mining Systems – Association Rule Mining: - Efficient and Scalable Frequent Item set Mining Methods – Mining Various Kinds of Association Rules – Association Mining to Correlation Analysis – Constraint-Based Association Mining. Mining Machine Learning -Types of machine learning– Model selection and generalization- Evaluation Metrics -Perspectives and Issues in Machine Learning – Concept Learning Task- FINDS- Candidate Algorithm – Applications of Data Mining and Machine Learning.

Unit 2 Data Preparation

L-9 Hours

Data Pre-processing- Needs of Pre-processing - Data Cleaning, Data Integration and Transformation, Data Reduction – Exploratory Data Analysis - Dimensionality Reduction – Feature Extraction- Variable Selection- Variable ranking- Linear Discriminant Analysis – Principal Component Analysis – Factor Analysis –Cross Validation –Resampling methods – Visualization of Data.

Unit 3 Association and Classification

L-9 Hours

Linear Regression – Multiple Regression - Bayesian Classification –Support Vector Machines - Logistic Regression - Decision Trees – Univariate Tree – Multivariate Tree –ID3- Regression Trees - Random forest- Probability and Learning – Gaussian Mixture Models – Nearest Neighbour Methods.

Unit 4 Neural Networks

L-9 Hours

ANN- Simple Perceptron- Gate implementation- Multi-Layer Perceptron – Going Forwards – Going Backwards: Back Propagation Error – Multi-layer Perceptron in Practice – Application of MLP – Overview – Deriving Back-Propagation- Optimization- Hyper parameter tuning.

Unit 5 Clustering Techniques

L-9 Hours

Cluster Analysis - Types of Clustering methods – Partitioning Clustering- K-means, K- Medoid, K Mode- Density-Based Clustering-DBSCAN Algorithm - Distribution Model Based Clustering - Maximum likelihood Estimation- Expectation-Maximization Algorithm- Hierarchical Clustering- Agglomerative and Divisive Clustering -BIRCH - ROCK- Chameleon - Fuzzy Clustering- Fuzzy C-means algorithm.

Total:45 Hours

G. Learning Resources

i. Text Books:

1. Ethem Alpaydin,, “Introduction to Machine Learning”, 4th edition, MIT Press, 2020 (Unit 1 to 4)
2. Christopher M Bishob, “Pattern Recognition and Machine Learning”, Springer-Verlag New York Inc, 2018
3. Tom Mitchell, “Machine Learning: A Guide to Current Research”, Kluwer academic publisher, 2019.

ii. Reference Books:

1. Richard S. Sutton and Andrew G. Barto, "Reinforcement learning: An Introduction", Second Edition, MIT Press, 2019.(Unit 5)
2. Oliver Theobald,“ Machine Learning for Absolute Beginners", Scatterplot Press, third edition, 2020

3. Beattie, Guy, “Machine Learning: Start with Machine Learning with all new tips tricks2020.Understand how machine learning can help”, 2020

iii. Online References:

1. Machine Learning, accessed online April 20, 2021 [online].Available: <https://www.coursera.org/learn/machine-learning>
2. Machine Learning by Georgia Tech, accessed online April 20, 2021[online].Available: <https://www.udacity.com/course/machine-learning--ud2>

COURSE CODE	COURSE TITLE	L	T	P	C
10211CS201	DIGITAL ELECTRONICS	2	0	2	3

A.Preamble

The primary aim of this course is to understand the fundamentals behind the digital logic design. From that student can gain the experience, to design any digital circuits and systems. The course includes fundamentals of Boolean algebra, combinational, sequential circuits and applications of digital electronics. Students can learn the basic programming concepts to implement digital circuits using hardware description language. The course outcome will be helpful to learn the concepts of computer architecture and organization.

B.Prerequisite Course

10210EE201 - Basic Electrical, Electronics & Measurement Engineering

C.Course Objectives

Learners are exposed to:

- Learn the basic Logic Gates
- Design of Combinational Circuits
- Apply Sequential Circuits Concepts
- Implement the Applications using sequential and combinational circuits

D.Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Apply the simplification of Boolean expressions using various mapping and mathematical methods.	K3/S3
CO2	Solve binary arithmetic operations using Combinational Circuits along with VHDL code using software tools.	K3/S3
CO3	Design synchronous counters and develop sequential circuit applications using flip flop and registers.	K3/S3
CO4	Demonstrate the variety of asynchronous sequential circuit design styles with respect to circuit properties.	K3/S3
CO5	Discuss the various applications of digital electronics.	K2/S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E.Correlation of Cos with Program Outcomes and Programme Specific Outcomes

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3			3							3		
CO2	3	3	3		3							3	1	
CO3	3	2	3		3									
CO4	3	3	3		3									
CO5	3	3			3								3	

High- 3; Medium-2; Low-1

F.Course Contents

Unit 1 Digital Fundamentals

L-6 Hours

Introduction to Boolean algebra and Switching Functions; Basic Logic Gates, Binary and Digital number system - Boolean Minimization using K Map and Tabulation method.

Unit 2 Combinational Circuits and HDL

L-7 Hours

Half adder – Full Adder – Half subtractor – Full subtractor – Multiplexer/ Demultiplexer – decoder - encoder – parity checker – parity generators – code converters - Magnitude Comparator-Introduction to Verilog / VHDL- Structural, Dataflow and Behavioral modeling. Structural, Dataflow and Behavioral modeling of combinational logic circuits (Multiplexer, Demultiplexer, decoder and encoder).

Unit 3 Memory Devices and Synchronous Sequential Circuits

L-7 Hours

Flip Flops and Memory devices: RAM – Static and Dynamic, ROM, PROM, EPROM, EEPROM; Counters: Binary, BCD and programmable modulo counters - Shift register- Sequential circuit design: using Mealy and Moore model.

Programmable Logic Devices – Programmable Logic Array (PLA) – Programmable Array Logic (PAL) – Field Programmable Gate Arrays (FPGA) – Implementation of combinational logic circuits using ROM, PLA, PAL.

Unit 4 Asynchronous Sequential Circuits

L-6 Hours

Analysis Procedure - Design Procedure-Reduction of state and flow table-Race free state assignment-Hazards-ASM chart.

Unit 5 Applications of Digital Electronics

L-4 Hours

Multiplexing displays - Frequency counters - Time measurements - ADC0804- microprocessor compatible A/D converters.

Total : 30 Hours

G. Laboratory Experiments

Total : 30 Hours

Part - 1

Task 1	Realization of Boolean Logic using Logic Gates Tools: Digital trainer kit, Integrated Circuits
Task 2	Design of Combinational Logic Circuits Half-Adder, Full-Adder, Half- Subtractor and Full-Subtractor. Tools: Digital trainer kit, Integrated Circuits
Task 3	Verification of 4-Bit Magnitude Comparator Operations Tools: Digital trainer kit, Integrated Circuits
Task4	Design of Decoders and encoders like BCD-Decimal decoder Tools: Digital trainer kit, Integrated Circuits
Task5	Design of Code Converters (Binary to Gray) Tools: Digital trainer kit, Integrated Circuits
Task6	Design of Multiplexers/De Multiplexers Tools: Digital trainer kit, Integrated Circuits

Task7	Verification of Flip-Flop operation Tools: Digital trainer kit, Integrated Circuits
Task8	Design of Shift register (To verify Serial to Parallel, Parallel to Serial, Serial to Serial and Parallel to Parallel Converters) using Flip-Flop Tools: Digital trainer kit, Integrated Circuits
Task9	Conversion of Flip-Flops (JK-T, JK-D) Tools: Digital trainer kit, Integrated Circuits
Task10	Design Synchronous and Asynchronous Counters Tools: Digital trainer kit, Integrated Circuits
Task11	Design a LED indicator system Tools: Digital trainer kit, Integrated Circuits
Task12	Design a counter using Programmable Array Logic Tools: Digital trainer kit, Integrated Circuits

Part-2

Use Cases:

Use case 1:

Design an Asynchronous sequential circuit using SR latch with two inputs A and B and one output y. B is the control input which, when equal to 1, transfers the input A to output y. when B is 0, the output does not change, for any change in input.

Use case 2:

The count has a repeated sequence of six states, with flip flops B and C repeating the binary count 00, 01, 10 while flip flop A alternates between 0 and 1 every three counts. Design with JK flipflop.

Use case 3:

Design a token number display system to drive the large LED display. This is used to display the three-digit token number on the display. These can be used in public places where people have to wait in lines like Banks, Hospitals, Airports, and Restaurants.

Use case 4:

This digital watch is used to display a 60sec time interval that counts from 0 – 59. This simple task is to design with an IC 555 & two counter ICs where 555 IC is used to generate the CLK signals and counter ICs carry the counting operation. Design a digital stopwatch.

Use case 5:

Use DTMF to control the appliances of home. This will overcome the limits of usual wall switches because these switches operate manually. The operation of this task can be done by dialing the selected number of specific loads on DTMF to control the home appliances. The operation of this system is very easy by using a mobile phone. The DTMF technology used in this system will get the commands from the mobile phone so that digital output can be generated. This output can make the relay driver ON to control the loads. The designing of this can be done using FFs, DTMF, and De-Multiplexer.

Total: Theory + Lab= 60 Hours

H.Learning Resources

i. Text Books:

1. M. Morris Mano, Digital Design, 6th Edition, Prentice Hall of India Pvt. Ltd., Pearson Education (Paperback) Pvt. Ltd., New Delhi, 2018. [Unit 1-3]
2. Thomas L. Floyd, Digital Fundamentals, 8th Edition, Pearson Education Inc, New Delhi, 2013
Donald D.Givone, Digital Principles and Design, TMH. [Unit 4-5]

ii. Reference Books:

1. John F.Wakerly, Digital Design, Fourth Edition, Pearson/PHI, 2016.
2. Donald. P.Leach, Digital principles and applications, 7th Edition, McGraw-Hill ,2012.
3. James Palmer and David Perlman, Introduction to Digital Systems, McGraw Hill, 1st Edition, 2020.

iii. Online References:

1. “Digital Electronics & Logic Design” Udemey[online] Available:
<https://www.udemy.com/course/digital-electronics-logic-design/>
2. “Digital System Design “ Nptel [online] 2021, Available:
https://onlinecourses.nptel.ac.in/noc21_ee39/preview.

COURSE CODE	COURSE TITLE	L	T	P	C
10211CS202	DESIGN AND ANALYSIS OF ALGORITHMS	3	0	2	4

A.Preamble

The course introduces the basics of computational complexity analysis and various algorithm design paradigms. The goal is to provide students with solid foundations to deal with a wide variety of computational problems, and to provide a thorough knowledge of the most common algorithms and data structures.

B.Prerequisite Course

10211CS102 – Data Structures

C.Course Objectives

Learners are exposed to:

- Develop the problem-solving ability and algorithmic analytical skills
- Produce optimized code
- Understand the different techniques involved in solving a problem
- Analyze the logic in terms of Efficiency, Time & Space Complexity
- Break the large size algorithm into smaller size components

D.Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Analyze the time complexity of algorithms using various asymptotic notations.	K3/S3
CO2	Implement solution for the given problems using Brute force and Divide and Conquer techniques.	K3/S3
CO3	Apply Dynamic programming and Greedy techniques to solve the problems.	K3/S3
CO4	Solve the problems using Backtracking and Branch & Bound techniques.	K3/S3
CO5	Interpret the computational efficiency using iterative approaches	K3/S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E.Correlation of COs with Program outcomes and Programme Specific Outcomes:

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	3	3				1	1		1	3	1	
CO2	3	3	3	3				1	1		1	3		
CO3	3	3	3	3				1	1		1	3		
CO4	3	3	3	3				1	1		1	3		
CO5	3	3	3	3				1	1		1	3		

3- High; 2-Medium; 1-Low

F.Course Contents

Unit 1 Introduction

L-9 Hours

Notion of an Algorithm – Fundamentals of Algorithmic Problem Solving – Important Problem Types – Analysis of Algorithm Efficiency - Introduction to Asymptotic Notations– Solving Recurrence relations – Master’s theorem - Mathematical analysis for Recursive and Non-recursive algorithms.

(Case Study: Design a gaming application like Chess, Candy crush etc.)

Unit 2 Brute Force and Divide & Conquer

L-9 Hours

Brute Force: Closest-Pair and convex-Hull Problems- Exhaustive Search - Traveling Salesman Problem - Knapsack Problem - Assignment problem. Divide and conquer methodology: Merge sort – Quick sort – Binary search- Heap sort- Closest-Pair and convex-Hull Problems

(Case Study: Brute Force: Solve the hide and seek game with a search time limit as 5 minutes and check its efficiency.)

(Case Study: Divide & Conquer: A king with a huge kingdom struggles to oversee from a single central city. Suggest how the king will handle the local problems of the smaller parts of the kingdom.)

Unit 3 Dynamic Programming and Greedy Technique

L-9 Hours

Dynamic Programming: Principle of optimality – Rod cutting problem - Computing Binomial Coefficient - Warshall’s and Floyd’ algorithm – Knapsack Problem. Greedy Technique: Minimum spanning tree-Prim’s algorithm- Kruskal’s Algorithm- Huffman Trees and Codes- Graph algorithms: Topological sort- Dijkstra’s Algorithm.

(Case Study: Dynamic Programming: Hill climbing, Comparison of knapsack problem with other design algorithms, A Taxi Operator firm needs to design a growth strategy plan that should focus on the revenue for the next 25 years considering various constraints like car turns old, reduction in mileages etc.,)

(Case Study: Greedy Technique: Assume the list of arrival and departure time; find the minimum number of platforms required for the railway as no train waits.)

Unit 4 Backtracking and Branch & Bound

L-9 Hours

Backtracking: n-Queens problem- Hamiltonian Circuit Problem- Subset sum problem-Branch & Bound: Assignment problem-Knapsack Problem- Traveling Salesman Problem

(Case Study: Backtracking: Solve a constraint network to find an assignment of values for each variable so that all constraints are satisfied.)

(Case Study: Branch & Bound: Optimize the tour to deliver the packages at randomly chosen pizza centres in the city of Chennai.)

Unit 5 Iterative Improvements and Limitations of Algorithm Power

L-9 Hours

The simplex method– The maximum-flow problem– The Maximum matching in bipartite graph- Introduction to unsolvable Problem-Halting problem. Limitations of Algorithm Power--Decision Trees- Lower bound arguments- P, NP and NP-Complete Problems. The Classes NP and P.

(Case Study: Identify solution to the telecommunication networks to handle the crucial problems such as flow control and routing)

Total: 45Hours (L)

G.Laboratory Experiments**Total: 30 Hours****Part – 1****Task 1: Algorithmic problem solving 1**

Calculating the time complexities for the given algorithms and estimating the orders of growth

Ex 1:

```
int i, j, k = 0;
for (i = n / 2; i <= n; i++) {
    for (j = 2; j <= n; j = j * 2) {
        k = k + n / 2;
    }
}
```

Ex 2:

```
function(int n) {
    if (n==1)
        return;
    for (int i=1; i<=n; i++)
    {
        for (int j=1; j<=n; j++)
        {
            printf("*");
            break;
        }
    }
}
```

Task 2: Algorithmic problem solving 2

Ex 2.1: Sort all the functions below in increasing order of asymptotic (Big O) growth. If some have the same asymptotic growth, then be sure to indicate that. Note: log means base 2.

i) $5n$, ii) $4 \log n$, iii) $4 \log \log n$, iv) n^4 , v) $n^{1/2} \log^4 n$, vi) $(\log n)^{\log n}$
vii) $n^{\log n}$, viii) 5^n , ix) 4^{n^4} , x) 4^{4n} xi) 5^{5n} , xii) 5^{5n} , xiii) $n^{1/5}$, xiv) $n^{n/4}$, xv) $(n/4)^{n/4}$

Solution: $4 \log \log n < 4 \log n < n^{1/2} \log^4 n < 5n < n^4 < (\log n)^{\log n} < n^{\log n} < n^{1/5}$
 $< 5^n < 5^{5n} < (n/4)^{n/4} < n^{n/4} < 4^{n^4} < 4^{4n} < 5^{5n}$

Ex 2.2 Order the functions according to their growth from slowest to fastest growing

Ex2.3: $6n^3$, $n \log_2 n$, $4n$, $8n^2$, $\log_2 n$, $n \log_2 n$, $\log_8 n$, 64 , 8^{2n}

Solution: 64 , $\log_8 n$, $\log_2 n$, $4n$, $n \log_2 n$, $n \log_2 n$, $8n^2$, $6n^3$, 8^{2n}

Task 3: D&C – Quick Sort

Sort a given set of elements using the Quicksort method and determine the time required to sort the elements. Repeat the experiment for different values of n , the number of elements in the list to be sorted and plot a graph of the time taken versus n . The elements can be read from a file or can be generated using the random number generator.

Test Case 1: Choose random element as Pivot and analyze the time taken to execute the algorithm

Test Case 2: Assume that the list of elements is in ascending order and apply quick sort and find the time complexity

Task 4: D&C – Merge Sort

Implement a parallelized Merge Sort algorithm to

sort a given set of elements and determine the time required to sort the elements. Repeat the experiment for different values of n , the

number of elements in the list to be sorted and plot a graph of the

time taken versus n . The elements can be read from a file or can be generated using the random number generator.

Test Case 1: Assume that the list of elements is in ascending order and apply merge sort and find the time complexity

Test Case 2: Assume the unsorted order of elements and apply merge sort and find its time complexity. Modify the algorithm to arrange the elements in descending order.

Task 5: Dynamic Programming – Warshall's & Floyd's algorithm

Compute the transitive closure of a given directed graph using Warshall's algorithm and for the same graph implement the all pairs shortest path problem using Floyd's algorithm.

Test Case 1: Assume the graph or a digraph with negative weights and check whether the Floyd's algorithm yields the correct result or not.

Test Case 2: Analyze that the time efficiency of Warshall's algorithm is cubic.

Task 6: Dynamic Programming – Knapsack algorithm

Implement 0/1 Knapsack problem using Dynamic Programming and find its complexity

Test Case 1: Show that a sequence of values in a row of the dynamic programming table for the knapsack problem is always non decreasing

Test Case 2: Show that a sequence of values in a column of the dynamic programming table for the knapsack problem is always non decreasing

Task 7: Greedy Technique – Prim's algorithm

Find Minimum Cost Spanning Tree of a undirected graph using Prim's algorithm

Test Case 1: Assume an undirected graph and find whether the graph is connected or not and then apply Prim's algorithm to find MST.

Test Case 2: Verify whether Prim's algorithm always work correctly on graphs with negative edge weights.

Task 8: Greedy Technique – Kruskal's algorithm

Find Minimum Cost Spanning Tree of a given undirected graph using Kruskal's algorithm

Test Case 1: Assume an input graph with more than one edge having a same weight, in such case which edge should be selected as the next visiting vertex by applying Kruskal's algorithm.

Test Case 2: Construct a Maximum Spanning tree for the input graph by applying Kruskal's algorithm.

Task 9: Greedy Technique – Dijkstra's algorithm

From a given vertex in a weighted connected graph find shortest paths to other vertices using Dijkstra's algorithm

Test Case 1: Show that the Dijkstra's Algorithm cannot work for a weighted connected graph with negative weights.

Test Case 2: Analyze the correctness of Dijkstra's algorithm for graphs with positive weights.

Task 10: Greedy Technique – Topological Sort

Obtain the Topological ordering of vertices in a digraph.

Test Case 1: Assume a Directed Acyclic Graph (DAG) as an input and check whether Topological sorting can be applied or not?

Test Case 2: Implement topological sorting using Breadth First Search (BFS)

Task 11: Backtracking - n-Queen's problem

Implement N Queen's problem using Backtracking

Test Case 1: Implement Rat in a Maze problem by applying the concept of backtracking.

Test Case 2: Modify the n-queens problem to handle the value of n in the range from 1 to 3.

Task 12: Backtracking - Subset problem

Find a subset of a given set $S = \{s_1, s_2, \dots, s_n\}$ of n positive integers whose sum is equal to a given positive integer d . For example, if $S = \{1, 2, 5, 6, 8\}$ and $d = 9$ there are two solutions $\{1, 2, 6\}$ and $\{1, 8\}$. A suitable message is to be displayed if the given problem instance doesn't have a solution.

Test Case 1: Assume an instance of the subset sum problem and verify whether the backtracking algorithm work correctly if we use just one of the two inequalities to terminate a node as non-promising.

Test Case 2: Given a set of distinct integers and return all the possible subsets or power set

Task 13: Branch & Bound - Travelling Salesman problem

Implement any scheme to find the optimal solution for the Travelling Salesperson problem

Test Case 1: Derive the solution of TSP using B&B and prove that it is optimal when compared with Brute Force technique.

Test Case 2: Analyze that the Travelling Salesman Problem (TSP) is an NP hard problem.

Task 14: Branch & Bound – Knapsack algorithm

Implement 0/1 Knapsack problem using Branch & Bound techniques and find its complexity

Test Case 1: Derive the solution of Knapsack problem using B&B and prove that it is optimal when compared with Brute Force technique.

Test Case 2: Show that a sequence of values in a column of the dynamic programming table for the knapsack problem is always non decreasing

Task 15: Iterative Improvements

Apply Ford-Fulkerson algorithm to solve the computational problems of finding maximum flow in a network to determine how fast people can be evacuated from the given city.

Test Case 1: Apply Ford-Fulkerson algorithm and find whether the augmenting path exists or not in the graph.

Test Case 2: Apply Ford-Fulkerson algorithm and find the updated residual capacity of a graph.

H.Learning Resources

i. Text Books:

1. Anany Levitin, “Introduction to the Design and Analysis of Algorithms”, Third Edition, Pearson Education, 2012. [Unit 1, 2 & 4]
2. Horowitz, S. Sahni, and S. Rajasekaran, “Computer Algorithms,” 2nd Edition, Galgotia Publications, 2007. [Unit 3 & 5]

ii. Reference Books:

1. Alfred V. Aho, John E. Hopcroft and Jeffrey D. Ullman, —Data Structures and Algorithms, Pearson Education, Reprint 2017.

2. Thomas H.Cormen, Charles E.Leiserson, Ronald L. Rivest and Clifford Stein, “Introduction to Algorithms”, Third Edition, PHI Learning Private Limited, 2012.
3. Donald E. Knuth, “The Art of Computer Programming”, Volumes 1&3 Pearson Education, 2009.

iii. Online References:

1. “Design and analysis of algorithms”, Jan 21 2015, Accessed on: Jan 18 2021, [Online]. Available: https://onlinecourses.nptel.ac.in/noc21_cs22/preview
2. "Algorithms Specialization", Accessed on: April 21 2021, [Online], Available: <https://www.coursera.org/specializations/algorithms>

COURSE CODE	COURSE TITLE	L	T	P	C
10211CS204	OBJECT ORIENTED PROGRAMMING USING JAVA	2	0	2	3

A.Preamble

This course provides basic concepts about Object Oriented Programming, Java Packages, Exceptions, Database connectivity, Java Streams, JavaFX and Java Servlets, JSP, Spring and Hibernate. After successful completion of this course learners can able to develop software modules for real world problem.

B.Pre-requisite course

10210CS101 - Problem Solving using C

C.Course Objectives:

Learners are exposed to:

- Implement Basic OOP principles to solve problems
- Practice Java basic constructs
- Build efficient Programming practice in Java
- Web application development with DB servers
- Work on Frameworks such as Spring and Hibernate

D.Course Outcomes:

Upon the successful completion of the course, students will be able to:

CO Nos	Course Outcomes	K - Level
CO1	Explain the fundamentals of Java and Object Oriented Programming (OOP) concepts to solve problems.	K2/S3
CO2	Apply the concepts of java inheritance and exception handling mechanisms for real world problem.	K3/S3
CO3	Solve the problems using Java Streams, Collections, UI Design and Threads	K3/S3
CO4	Develop simple web applications using JSP, Servlet with JDBC and Hibernate Framework	K3/S3
CO5	Build the web applications using Spring Framework.	K3/S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E.Correlation of COs with Program outcomes and Programme Specific Outcomes:

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	1											
CO2	1	3	3	1	1							2	1	
CO3	1	3	2	2	2							2	2	
CO4	1	3	3		2								3	
CO5	1	3	3		3								3	

3- High; 2-Medium; 1-Low

F.Course contents

Unit 1 Basics of OOP and JAVA

L-6 Hours

Introduction to object-oriented programming - Features of Java - JVM- Keywords- Variables - Data types – Operators - Control statements -Command line arguments - Classes and Methods - Objects – Java Packages – Build-in packages - user-defined package -Access specifiers: private, protected, default, public - Constructors - Method Overloading - Type Casting - this keyword- Static – Arrays, Multi-dimensional Arrays.

Unit 2 Inheritance, Exception Handling and string

L-6 Hours

Basics of Inheritance - Forms of Inheritance - Super keyword – Final - Method Overriding –Abstraction: Abstract Classes - Interfaces. Exception Handling: Java Exception Hierarchy - Exception Types - Throwing and Catching exceptions -Declaring New Exception Types – custom exceptions. String class – Immutable String - String handling methods.

Unit 3 Java Streams, Collections, UI Design, Thread

L-6 Hours

Java Streams: Features, methods – Generic Classes and Methods - Collections Frameworks: List, Queue, Set and Map. JavaFX: Architecture, Effects, Transformation, Animation, Layouts, UI Controls, Event Handling. Threads: Life cycle - Thread states - Thread priority– Thread Synchronization.

Unit 4 Server Programming, Hibernate Framework

L-6 Hours

Introduction to Servlet and JSP, JDBC, Example Web applications with DB - Basics of Java Networking. Introduction to Java Frameworks - Basics of Hibernate – Hibernate Web application - Hibernate Mapping and types – HQL – Annotations – Caching. Integrating Hibernate and Spring.

Unit 5 Java Spring Framework

L-6 Hours

Basics of Spring – Spring in IDE– Dependency Injection – Constructor Injection – Spring ORM: Spring with Hibernate – SPEL -Spring MVC – Spring Web – Spring JDBC – Spring with Angular – Spring Boot.

Total: 30 Hours (L)

G.Laboratory Experiments

Total: 30 Hours

Part – 1

Task 1: Package creation and String.

- Simple Java application with class (Object), Methods
- Developing user defined packages in Java
- Apply String handling methods

Task 2: Inheritance and Interfaces

- To Implement Method overloading, constructor and Method Overriding.
- Use of this, super, static and final keywords
- Developing user-defined interfaces and implementation
- Use of predefined interfaces

Task 3: Exception Handling Mechanism in Java

- Handling pre-defined exceptions
- Handling user-defined exceptions

Task 4: Java Collections

- Handle list of elements using Collection classes (List, Queue, Set, Map)

Task 5: JavaFX

- To Create Different Layouts with JavaFX animation.

- To Design a web application with JavaFX event handling.

Task 6: Threading

- Creation of thread in Java applications
- Multithreading

Task 7: Java-JDBC

- To connect Oracle/ MySQL for Table creation and Data Manipulation.

Task 8: Servlet

- Write a Servlet program to to connect Oracle/MySQL for Table creation and Data Manipulation.

Task 9: JSP

- Write a JSP program to to connect Oracle/ MySQL for Table creation and Data Manipulation.

Task 10: Spring Standalone and web Application

- Standalone application with data manipulation
- Web application with data manipulation

Task 11: Hibernate Standalone and web Application

- Standalone application with data manipulation
- Web application with data manipulation

Part– 2**Use cases:****Use case1: Validated Login for Employee management JSP:**

Company ABC planned to create a website for maintaining employee details. Login page need to be designed with username, password and captcha. Username is their mail-id and password should contain at-least 1-Uppercase alphabet, 1-special character, 1- lower alphabet. Design the page using front end and server scripting with client / server validation.

Use the technologies: HTML5, CSS5, Javascript, JSP.

Use case2: Event Registration for a University Spring:

In a University, they have planned to conduct national level symposium. Students are expected to come from various places from india. University has decided to maintain all participant details such as Participant_name, college name, branch, year, age, and phone.

And also generate unique participant ID using ID generator logic. Implement a java application for above scenario with rich user interface and database server.

Use the technologies: HTML5, CSS5, Spring framework, MySQL/Oracle.

Use case3: E – Applicant System for Student using Spring MVC:

E-Applicant system has been planned to implement by an institute. In that, student will be able to request any kind of Certificate like Transfer and Bonafiedetc by applying through this System.The student can check the status through his/her account. There are many users like Internal Staff, HOD, Dean, Registrar and Vice chancellor to verify and approve the request. Above application should be implemented using spring framework with MVC technology with suitable front end techniques and Database servers.

Use the technologies: HTML5, CSS5, Spring framework, MySQL/Oracle.

Use case4: Stock management with CRUD operations using hibernate:

A Pharma Agency decided to manage their business using software. Monitoring Stock entry, sales, shortage, offers, profit and expired medicines are the requirements from the client using the targeted software. Since inserting details, update the offer details and removing expired stock entries, CRUD

supported implementation expected to build. Instead of using JDBC API, hibernate suggested to be used for making efficient CRUD operations with database servers.

Use the technologies: HTML5, CSS5, Hibernate framework, MySQL/Oracle.

H. Learning Resources

i. Text Books:

1. Herbert Schildt, “Java: the Complete Reference”, Eleventh edition, Tata Mc-Graw Hill, 2019.[Unit 1, 2, 3]
2. Jim Keogh, “J2EE: The complete Reference”, First Edition, Tata Mc-Graw Hill, 2017. [Unit 4]
3. K. Santosh Kumar, “Spring and Hibernate”, McGraw Hill Education, 2017. [Unit 4,5]

ii. Reference Books:

1. H.M. Deitel and P.J. Deitel,” Java How to Program”, Pearson Prentice Hall Seventh edition, 2018.
2. Balaguruswamy, “Programming in java”, Sixth Edition, Tata McGraw Hill, 2019.
3. Jason Hunter, William Crawford, “Java Servlet Programming”, Second Edition, O'Reilly-2018.

iii. Digital Resources

1. “Programming in Java”, Accessed on: Apr. 20, 2021 [Online], Available: <https://nptel.ac.in/courses/106/105/106105191/>
2. “Java Tutorial”, Accessed on: Apr. 20, 2021 [Online], Available: <https://www.javatpoint.com/java-tutorial>.

COURSE CODE	COURSE TITLE	L	T	P	C
10211CS207	DATABASE MANAGEMENT SYSTEMS	3	0	2	4

A.Preamble

Databases form the backbone of all major applications today – tightly or loosely coupled, intranet or internet based, financial, social, administrative, and so on. Structured Database Management Systems (DBMS) based on relational and other models have long formed the basis for such databases. While DBMS differ in the details, they share a common set of models, design paradigms via a Structured Query Language (SQL) or NoSQL. More specifically, this course introduces relational data models; entity-relationship modeling, SQL, data normalization, database design and modern databases as well as lightweight databases.

B.Prerequisite Course

10210CS101 – Problem Solving using C

C.Course Objectives

Learners are exposed to:

- Design a database at basic level like Entity-Relationship modeling, Relational Modeling.
- Devise a database using Armstrong axioms, functional dependencies and views.
- Acquire knowledge of Query processing and concurrency transaction control.
- Learn an implementation issues with nontraditional databases.

D.Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Apply basic database designing methodologies	K3/S3
CO2	Design a database schema for a given problem	K3/S3
CO3	Apply normalization for the given database.	K3/S3
CO4	Use the properties of transaction and recovery management.	K3/S3
CO5	Implement CRUD operations in modern database.	K3/S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E.Correlation of Cos with Program Outcomes and Programme Specific Outcomes

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	2	2	2									2		2
CO2		3	3	2	3							3	3	3
CO3		3	3		2							2	2	2
CO4		2	2											2
CO5		2		2	3								2	2

High- 3; Medium-2; Low-1

F.Course Contents**Unit 1 Introduction to DBMS****L–9 Hours**

Purpose of Database System – Database Schema and Instances- Views of data – Database Languages - Database System Architecture – Database users and Administrator –Distributed Databases-DBB System Architecture-Database models-Entity–Relationship model – E-R Diagrams - Introduction to relational databases –Structure of relational databases- Relational model-Basics- From E/R diagrams to Database Design- Hierarchical model-Tree-structure diagrams- data retrieval-update facility-virtual records-mapping of hierarchies to files-Network Model- Data-structure diagrams-DBTG model- data retrieval-Find-Get-Set processing-update facility-Mapping network to files. Case study: Conference Management system.

Unit 2 Database Design and Query Optimization**L–9 Hours**

Keys and Integrity Constraints - Relational Algebra – Relational Calculus-Tuple –Structured Query language(SQL) Basic and additional Operations –Database Languages Revisited –create-alter-update-insert-rename-truncate-drop-select query commands-grant-revoke-rollback-commit control commands-Operators-aggregate functions-string-indent-date functions-set-union-union all-aliasing-Nested Queries & Join Queries–Join-Natural-inner-equi-cross/cartesian-outer-left-right-full outer join-Common Table Expressions-limit function in statistics-summarizing-Embedded SQL- Indexes and Triggers –compound- Views Definitions -Materialized view-Assertions-Query Processing and Optimization-Heuristics and cost estimates in Query Optimization-Tuning. Case study: Restaurant chain for network model.

Unit 3 Functional Dependencies and Normalization**L–9 Hours**

Introduction and problem of data redundancy-Features of good Relational database design-Functional Dependencies - Normalization – First Normal Form-minimal cover-canonical cover-Decomposition- Second Normal Form- Third Normal Form –controversies with SQL in normalization-Advanced Normalization -Boyce/Codd Normal Form, Fourth Normal Form and Fifth Normal Form- Dependencies preservation-Denormalization-Case Studies of database system like Air-cargo services.

Unit 4 Transaction and Concurrency**L–9 Hours**

Transaction Concepts –Schedules- Transaction States - Concurrent Execution- Serializability- -Two Phase locking Protocol –Timestamp ordering Protocol-Validation Based Protocol-Multiple Granularity-Intention lock modes-Multiple granularity locking scheme-Replication in Distributed databases-Deadlock-Handling-Prevention-Wait-diescheme- wound-wait scheme- Detection-Wait-for graph-Phantom phenomenon-Distributed deadlock detection-Index locking-weak levels of concurrency- Concurrency in index structures- B+ Tree crabbing- -Types of Failure-Recoverability -Recovery and Prevention- Log based recovery-undo-redo-checkpoint-Recovery Algorithm-Transaction rollback with Logical undo- Recovery Algorithm with Logical Undo-ARIES Recovery Algorithm. Case study: Instances and Log recovery process in Oracle and Microsoft Access.

Unit 5 Storage and Modern Databases**L–9 Hours**

RAID storage -Modern databases-XML Databases- CAP Theorem-Datalog-Basic structure-Syntax-semantics-Relational operation in Datalog-Recursion in Datalog-Power of Recursion-Monotonicity-Non-monotonicity-NoSQL-Data models-Key value store, column families-Document Databases-MongoDB-Introduction to MongoDB- CRUD operations-Create-Insert-Update-Query-Removing documents-Sharding- Introduction to Mongoose-core concepts-extending models-CRUD operations with mongoose. Case study: Designing local library in Amazon using mongoose-SQLite-Building and Installing SQLite-DDL-DML-TCL-Select-compound select- alternate Join- JOIN..ON, JOIN..USING-Database design-SQL functions and SQLite extensions-Virtual Tables and Modules.

Total: 45 Hours (L)

G.Laboratory Experiments**Total: 30 Hours****Part-1****Task 1: Conceptual Design after FTR**

Using basic database design methodology and ER modeler, design Entity Relationship Diagram by satisfying the following sub tasks:

1. a Identifying the entities.
1. b Identifying the attributes.
1. c Identification of relationships, cardinality, type of relationship.
1. d Reframing the relations with keys and constraints.

Task 2: Generating design of other traditional database model

Creating Hierarchical /Network model of the database by enhancing the sound abstract data by performing following tasks using forms of inheritance:

2. a Identify the specificity of each relationship, find and form surplus relations.
- 2.b Check is-a hierarchy/ has-a hierarchy and performs generalization and/or specialization relationship.
2. c Find the domain of the attribute and perform a check constraint to the applicable.
2. d Rename the relations.
2. e Perform SQL Relations using DDL, DCL commands.

Task 3: Using Clauses, Operators and Functions in queries

Perform the query processing on databases for different retrieval results of queries using DML, DRL operations using aggregate, date, string, indent functions, set clauses and operators.

Task 4: Using Functions in queries and writing subqueries:

Perform the advanced query processing and test its heuristics using the designing of optimal correlated and nested subqueries such as finding summary statistics.

Task 5: Writing Join Queries, equivalent, and/or recursive queries

Perform the advanced query processing and test its heuristics using the designing of optimizing complex queries and their equivalence queries

TASK 6: Procedures, Function and Loops:

Programming using PL/SQL Procedures, Functions and loops on Number theory and business scenarios like.

TASK 7: Triggers, Views and Exceptions

Conduct events, views, exceptions on CRUD operations for restricting phenomenon.

TASK 8: CRUD operations in Document databases

Perform Mongoose using NPM design on MongoDB designing document database and performing CRUD operations like creating, inserting, querying, finding, removing operations.

TASK 9: CRUD operations in Graph databases

Perform GraphQL/Neo4jgraph space design for recommendation engines. Also perform CRUD operations like creating, inserting, querying, finding, deleting operations on graph spaces.

Task 10: Normalizing databases using functional dependencies up to BCNF

Upon relational tables created in task-2, perform normalization up to BCNF based on given Dependencies as following for the assumed relations specified below.

- 10. a** Apply the functional dependency, normalize to 1NF
- 10. b** Normalize the relations using FD+ and α^+
- 10. c** Find the minimal cover, canonical cover.
- 10. d** Normalize to 2NF, add/alter constraints if necessary.
- 10. e** Normalize to BCNF, add/alter constraints if necessary.
- 10. f** Normalize to 3NF, add/alter constraints if necessary.

Task 11: Menus, Forms and Reports:

For an application, creating and debugging Menus, Forms and reports using Oracle Forms and Report Builder or SQL server management studio or SQLite or NetBeans

Task 12: Micro Project:

Develop micro projects based on business scenarios and use cases.

Part-II

Use cases:

Use Case-1: Building a Cart analysis for Myph

Myph has just launched their brand new phone range to the eager reception of the consumer market cart analysis. The product's data model has a unique menu that identifies the product, title, description, a stock quantity, and pricing information about the item. All products have categories. To be able to provide a list of all the products in a category, amend the data model with a collection of documents for each category and contain the path for that category in the category tree. Use cart analysis in developing different consumer selection options. Would this answer outlier selection in cart i.e., surplus selections? Is Relational database application can answer these transactions? How recovery is made through carting and commerce?

Use Case-2: Indexing various devices in IoT platform

A generic IoT platform required support for data from a wide range of devices, some of which could not be envisaged while developing the platform. The proficient work necessitated a data storage mechanism that could handle data from different types of devices. Indexing support makes it easy to pull data using a single index or multiple indexes such as device id with location id. Records for a particular device in different locations are easily accessed. Common parameters like temperature from different types of devices and their records are retrieved fast through these indexes. How the application could lead to the choice of JSON-based document database, MongoDB? Assume or create JSON script in support of this.

Use Case-3: A gift coupon application handles offers payment-related Information.

Choose the database system, a relational database, for its capability to scale horizontally while keeping the ACID property. The last was particularly important because transactional data was being handled. Eventual consistency as offered by other databases was not suitable. Going with DB and cluster gives the opportunity to scale horizontally for a large number of writes and reads without compromising the ACID and transactional capability required by the application. Will it transact and lead to no

deadlock, keeping all relational tables normalized? If so till what normal form do they sustain to offer a gift coupon application?

Use case- 4: Army Supply chain, Bill of Materials and Maintenance Cost Management

The nation's armed forces, the support for more than one million soldiers and about 200,000 civilian staff. Each of these staff members relies on multiple pieces of equipment, from helicopters and armoured vehicles to small arms and radios, to complete their missions. With maintenance, operation and support costs of equipment representing as much as 80% of total lifecycle costs, it's imperative that the Defence ministry track and analyse equipment maintenance costs including changing historical data sources Dimension with more flexibility like graph databases and to be given richer analysis like forecast replacement parts with the location and climate, mean time to failure rates, logistics and requirement processes. Is answerable the vital "what-if" questions such as the cost of deploying certain forces and supporting equipment to a new war zone? Could the model perform multi-dimensional cost comparison and trend analysis? Will the solution promises how the data management in unpredictable maintenance costs?

H.Learning Resources

i. Text Books:

1. Abraham Silberschatz, Henry F. Korth and S. Sudharshan, "Database System Concepts", Seventh Edition, Tata McGraw Hill, 2019.[Unit1-5]
2. Shannon Bradshaw, E Brazil, Kristina Chodorow, "MongoDB: The Definitive Guide - Powerful and Scalable Data Storage", Third Edition, Shroff/O'Reilly Inc., January 2020.[Unit-5]
3. AgusKurniawin, "Python and SQLite Development", First Edition, PE Press, January 2021. [Unit-5]

ii. Reference Books:

1. Raghu Ramakrishnan et al, "Database Management Systems", Third Edition, McGraw Hill, 2014.
2. Elmasri Ramez, Navathe S, "Fundamentals of Database System", Seventh Edition, Pearson, 2017.
3. J.D.Ullmann et al, "Database Systems: The Complete Book", Second Edition, Pearson Ed, Inc, 2009.

iii. Online References:

1. "Database Management System Part-1&11" Mar.25, 2019. Accessed on Feb.20, 2021[Online].Available:https://infytq.onwingspan.com/en/viewer/web_module/lex_auth_0130542736573808641006_shared?collectionId=lex_auth_01275806667282022456_shared&collectionType=Course.
2. "Database Design", Dec 31, 2009. Accessed on May05,2021[Online]. Available: <https://nptel.ac.in/courses/106/106/106106093/>.
3. "Designing local library models", Accessed on: May 05, 2021[online]. Available: https://developer.mozilla.org/en-US/docs/Learn/Server-side/Express_Nodejs/mongoose.

COURSE CODE	COURSE TITLE	L	T	P	C
10211CS208	SOFTWARE ENGINEERING	2	0	2	3

A.Preamble

Software Engineering is an engineering branch that relates to the evolution of software product using well-defined scientific principles, techniques, and procedures. This course is being offered in order to train the students to presents the concepts, methods and techniques necessary to efficiently capture software requirements, to plan the project with appropriate estimations, and to bring it to completion.

B.Prerequisite Courses

10211CS204– Object Oriented Programming using Java/ 10210CS101- Problem solving using C

C.Course Objectives

Learners are exposed to:

- Understand the best practices in software engineering.
- Develop the necessary skills to handle software projects in a principled way.
- Provide an in depth knowledge on the different phases of the software life cycle.
- Perform requirements analysis, estimation design and testing for applications.

D.Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Identify the suitable process model for software development.	K3/S3
CO2	Prepare the SRS document with project estimation and plan.	K3/S3
CO3	Design the UML Diagrams to develop a software application.	K3/S3
CO4	Apply the software design heuristics for quality improvement	K3/S3
CO5	Build Test cases using various testing strategies.	K3/S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E.Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	2	3										1	3	
CO2	2	3	3		1							2	3	
CO3	1	3	3		3								3	3

CO4	1	3	3		3	2							3	
CO5	3	3	3			1						1	3	

3- High; 2-Medium; 1-Low

F.Course Content

Unit 1 Software Development Process Models

L–6 Hours

Introduction to Software Engineering - Software Development process models – Waterfall Model, Incremental Model, Spiral Model, V Model – Agile Development- Agile Methodologies- Agile Principles- SCRUM and XP - Project management- Stages of Project management- 4 P's of project management - Process & Project metrics. Case Study: The Railways tracking - Arrival Time Prediction System

Unit 2 System Requirements Specification, Planning & Scheduling

L–6 Hours

Software Requirements Specification - Software project planning - Software Estimation - Empirical Estimation Models – Software Project Scheduling- Risk Management - Software risks – Risk identification – Risk projection – Risk refinement - Defect Prevention. Case Study: Preparation of SRS for the camera motion sensor system for home automation.

Unit 3 Analysis Using UML

L–6 Hours

Analysis Modeling - Data Modeling - Functional Modeling & Information Flow - Behavioral Modeling- Structured Analysis - Object Oriented Analysis - Domain Analysis- Object Relationship Model - Object Behavior Model, Design modelling with UML- Design modelling with UML. Case Study: UML diagrams for Stock Trading Application using the online Creatly platform.

Unit 4 Design Principles and Heuristics

L–6 Hours

Design Principles - Design Process - Design Elements- Design Concepts - Modular Design-top-down and bottom-up, Abstraction, step-wise refinement, coupling and cohesion, Refactoring, Architecture- Introduction to Software Architecture - Data Design - Transform Mapping - Transaction Mapping - Design Patterns. Case Study: Architecture diagram and design for the Safe Home security system.

Unit 5 Testing, Maintenance and Reengineering

L–6 Hours

Testing approaches -Top-Down, Bottom-Up, Big Bang and Sandwich, object oriented product Implementation & Integration- Software Testing methods- White Box- Black Box - Unit Testing - Integration testing - Validation & System testing – Software Maintenance & Reengineering. Case study: Login module of Ecommerce application - Test cases and Best Practices.

Total: 30 Hours (L)

G.Laboratory Experiments

Total: 30 Hours

Part - 1

Task 1 Design of problem statement, perform a detailed feasibility study and finalize the process model to be used.

Tools: Microsoft Word

- Task 2** To Prepare the Software Requirement Specification Document in the given template to include the functional and Non-functional requirements.
Tools: Perforce Helix RM and Microsoft Word.
- Task 3** To Prepare a detailed estimate using the FP and COCOMO Model. Also prepare a detailed schedule of the project.
Tools: GanttPRO or Lucid Chart and Microsoft Word.
- Task4** Identify Use Cases and develop the Use Case model for the given scenario Draw the relevant Entity Relationship Diagram and class diagram.
Tools: Creately, Rational Ross or any CASE Tools
- Task5** Using the identified scenarios, find the interaction between objects and represent them using UML Sequence & Communication diagrams.
Tools: Creately, Rational Ross or any CASE Tools
- Task6** To implement behavioural modelling using User Interface diagrams, state transition diagram and activity diagrams for the given scenario.
Tools: Creately, Rational Ross or any CASE Tools
- Task7** To implement partial layered, Object Oriented and logical software architecture diagram to visualize the high level design.
Tools: Rational Ross or any CASE Tools
- Task8** To Implement the data Flow Diagrams to perform the data modelling with Level 0, Level 1 and Level 2 analysis.
Tools: Creately, Rational Ross or any CASE Tools
- Task9** To Develop test cases for various Testing methodologies. Write the test cases for both Black box Testing and White Box Test Testing.
Tools: TestopiaTools and Microsoft Word.
- Task10** To Develop a prototype of the project with database management system and deploy the project.
Tools: Rational Ross or any CASE Tools

Part-2

Use Cases:

Use Case 1: Analyze and implement the Employee management and Payroll Processing application.

Use Case 2: Build the Library Management system application with project plans, schedule, estimation and design.

Use Case 3: Design,Analyze and implement the Passport Automation System application.

Use Case 4: Identify, plan, analyze and implement the E Voting System application.

H.Learning Resources

i. Text Books:

1. Roger. S. Pressman and Bruce R. Maxim, “Software Engineering – A Practitioner’s Approach”, 9th Edition, McGraw Hill, 2020. [Unit 1,2,4,5]
2. Ian Sommerville, “Software Engineering”, 10th Edition, Pearson Education, New Delhi, 2017. [Unit 3]

ii. Reference Books:

1. Fairley R, “Software Engineering Concepts”, Indian edition, Tata McGraw Hill, New Delhi, 2017.

iii. Online References:

1. Software Development Lifecycle, Accessed on: Apr. 20, 2021 [Online], Available: <https://www.coursera.org/specializations/software-development-lifecycle>
2. Writing Test Cases, Accessed on: Apr. 20, 2021 [Online], Available: <https://www.softwaretestinggenius.com/a-simplistic-approach-to-writing-test-cases-the-black-box-way/>
3. Introduction to Software Engineering, Accessed on: Apr. 20, 2021 [Online], Available: <https://www.coursera.org/learn/introduction-software-testing>

COURSE CODE	COURSE TITLE	L	T	P	C
10211CS212	WEB AND MOBILE APPLICATION DEVELOPMENT	3	0	2	4

A.Preamble

This course provides basic concepts about HTML5, CSS3, Bootstrap Framework and java script technologies to create the interactive client-side design of web applications. This course also teaches the server-side programming using Node JS, PHP and MVC Front End design for web applications using Angular. The conversion of web application to mobile apps can be performed using Angular. Finally, the creation and deployment of micro services using Seneca and Dockers will be discussed.

B.Pre-requisite course

10211CS204- Object Oriented Programming using Java

C.Course Objectives:

Learners are exposed to:

- Design the interactive and responsive web application.
- Build the efficient server-side applications.
- Develop the single page MVC application.
- Construct and deploy the Microservices.
- Conversion of web apps to Mobile apps using Angular.

D.Course Outcomes:

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Design web applications using Hypertext Mark-up Language and Cascading style sheets through bootstrap.	K3/S3
CO2	Build the interactive and dynamic web page using Java Script Technology.	K3/S3
CO3	Implement the server-side business logic to handle client request using NodeJS and PHP.	K3/S3
CO4	Make use of MVC framework for integrating the window controls and its corresponding actions through event handlers.	K3/S3
CO5	Transform the function as Microservice components to enhance the Reusability concept.	K3/S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E.Correlation of COs with Program outcomes and Programme Specific Outcomes:

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1		2	3		3	1							3	
CO2		3	3	1	2	1							3	
CO3		3	2		3	2							2	2
CO4		3	3		3	2							3	3
CO5		3	3		3	2							3	3

High- 3; Medium-2; Low-1

F.Course contents

Unit 1 Front end design using HTML5, CSS3 and BOOTSTRAP Framework L-9 Hours

Introduction to HTML5: Basic Elements, Form Elements, Media Elements, HTML5 Graphics (Canvas, SVG), **CCS3:** Selector String, Box Model, Text Properties, CSS 3D Transformation, CSS Animation, **Bootstrap Framework:** BS Grid, Tables, Images, Alerts, Form Elements. Case study: Online Blog Creation.

Unit 2 Dynamic web page design using Java Script and jQuery L-9 Hours

Java Script: Data Types and Variables - Operators - Control Statements - Functions -Objects - Build in Objects - DOM - Java Script Event Handling - Form Handling and validations - **AJAX & JQuery:** Introduction to AJAX and JQuery: Introduction - JQuery Selectors - JQuery Animations - Effects - Event Handling - JQuery DOM Traversing-JSON - JQuery AJAX. Case Study: Web Scraping and Automation

Unit 3 Server-side programming using PHP and Node JS L-9 Hours

PHP: Introduction - Variables - Program control - Builtin Functions - Connecting to Database using MySQLi - Cookies - Sessions - Regular Expression;

Node JS: Introduction - Architecture - Features of Node JS - Installation and setup - Function - Module - Creating Web Servers with HTTP (Request & Response) - Event Handling - Express JS - Get and Post Implementations - Database connectivity. Case Study: online Book Store.

Unit 4 Single Web Page Design using Angular L-9 Hours

TypeScript: Introduction to Type Script - Variables - Datatypes - Enum - Array - Tuples - Functions - OOP concepts - Interfaces - Generics - Modules - Namespaces - Decorators. **Angular:** components and Modules - Data Binding - Event Binding - Templates - Directives - Services - Dependency Injection - Routing and Navigation pages - Template based Form - Reactive Forms - Validating - Pipes - Sorting and Filtering -Decorator - HTTP Client - Data storage - Observables & RxJS Case study: Todo list

Unit 5 Creating Forms and Mobile app Conversions L-9 Hours

Microservices: Introduction - Microservices Architecture - Microservices in Node.js: Installing Node.js, NPM, Seneca and PM2-SOLID Design Principles - Seneca Toolkit: Inversion of control(IOC) - Pattern Matching - Reusing Patterns - Writing Plugins - Web Server Integration - Data Storage - PM2 Task Runner for Node.js - Writing Microservices - Integrating with Express - Testing and Documenting Microservices - Monitoring Microservices using PM2 and Keymetrics - Deploying Microservices. **Converting Web App to Mobile App Development** - Install Capacitor Package - Configuration Setup - Native IOS and Android packages - Build and Deploy the application. Case study:Chat Bot

Total: 45 Hours

G.Laboratory Experiments

Total: 30 Hours

Part – 1

Task 1	Develop a Simple College Website including all the Department Information using HTML5 and CSS3.
Task 2	Create Home Page, Sign up and Login Page for Clinic Management Service using Bootstrap Framework.
Task 3	Validate the Registration, user login, user profile and payment by credit card pages using JavaScript.
Task 4	Parse the web page to get the required information using JQuery and DOM Traversing.
Task 5	Create a simple HTTP web server using Node.js to generate a dynamic response
Task 6	Create a three-tier application using Node.js and MySQL data base.

- Task 7** Create an Reactive form for User Registration using Angular for Online Exam portal.
- Task 8** Develop web application to implement routing and navigation in Angular.
- Task 9** Develop a micro service for finding what people think by asking 500 people's opinion for any consumer product in Node.js using Seneca Toolkit.
- Task 10** Build the Web application and convert it into Mobile app.

Part-2

Use Cases:

Use Case 1: Bike Rental System

This system is named as **Bike on Rent Management System**. This system is designed to help the customers to take bikes or two-wheelers on rent. When we go on any trip outside the town or country we want to be free of time so instead of going through metros and taxis we prefer to have our own vehicle for rent. Using this system vehicle owner can register as sellers and customers who want to take bikes on rent can register them as renters and can take any bike on rent. Address of the both are required as the customer can only take bike by going to the address provided and the vehicle owners can know the address that a customer is verified or not. The customer also has to upload some proofs to take the bike on rent. Proofs like license, pan card and identity card are compulsory so that no one could run taking the bike. Any customer whose proofs are not uploaded and are not valid will not be allowed to take any bike on rent. This has one admin account who verifies the registering user and two types of the user account. One for bike sellers and one for customers who take the bike on rent. This system has only one admin account and cannot have more than one admin account. There will be simple chat room needs to be added to make instant interactions between customer and admin. Admin can verify and register the user who is registering. If the admin does not verify, the user cannot register.

Tools: HTML5, CSS3, BOOTSTRAP, Angular, NodeJS, JQuery

Use Case 2: Clinic Management Service

This system is named as Clinic management service. This system is made to keep the records about the patients, doctors and other staff members working at a clinic or hospital receptionist. One can login into the clinic management service using the email id or the user id and password. After signing in into this system there are the options to add new patients, new doctors and other new staff members like nurses and ward boys etc. This system is designed to easily maintain the data of the patients specifically. Daily many new patients visit the clinic so adding the new patient's details and keeping the records using it is very easy. There is also an option to add and delete doctors and other staff member's details. New and unique ids are given to everyone who gets registered over this system. There are also the options to check the patient's disease and course the patient is going through. Fees paid by any customer or patient can be saved on it and it is easy to calculate daily that the money is collected. Doctors and another staff member like nurses, ward boys, janitor and maid leaves can be deducted from their salary and rest of the salary can be paid easily using this clinic management service. It is easy to calculate money and handle account on monthly basis also. This system also has an option to use the backup that means if we are backing up the database it can never be lost. So, overall, this clinic management service is a solution to all the problems that we face in a clinic or hospital.

Tools: HTML5, CSS3, BOOTSTRAP, Angular, NodeJS, Java Script

Use Case 3: E-commerce service

Online shopping is a process in which people (specifically customers) are being provided with the option of purchasing goods and services directly from the seller, all in real-time environment. Online shopping is an application of the internet as electronic commerce (Like Amazon). From the business perspective, customers usually find the products more attractive, on websites, as they get all the details available there. People in large number are doing online shopping today, and it is not only because it is convenient as one can shop from home,

but also because there are ample amount of varieties available, with high competition of prices, and also it is easy to navigate for searching regarding any particular item. For sellers, their products have access to World Wide market, which also increases the number of customers and enhances the customer relationships. Also the web stores are a means for the small scale companies to launch their products at global level. The main objective behind this project is to develop a web oriented application which can provide an online shopping feature to the users. In other words, the project aimed at creating a virtual shop environment for users, in some handy form, which will be available to them through internet. This system has been designed keeping in mind all the aspects such as loading the data, complexity and maintaining the security of user credentials.

Here in this system, complexity refers to the total number of features being provided to users, and their smooth arrangement and functioning required. Following are some of the key features of our system, which distinguishes it from others:

- Display of all the available categories for shopping on the home page.
- Display all the sub-categories on the home page; those are associated with any particular item.
- Admin has the authority to add new particulars to the items list whenever needed.
- Permission to administrator to remove items, anytime.
- Allows the admin to modify the price of each item, whenever required or felt like.
- Admin has the authority to update the description of each item.
- Permission to the admin to view information about each customer who checkouts the items list.

Tools:HTML5, CSS3, BOOTSTRAP, Angular, NodeJS, Java Script

Use Case 4: College management service

This application college management system based on Internet that aims to all the levels of management providing information within an organization. This system can be used as an information management system for the college. For a given student/staff (Technical / Non-technical) the Administrator creates login id & password, using these student/ staff (Technical / Non-technical) can access the system to either upload or download some information from the database. Not only will this added user also get to know about the events and extra curriculum activities which will hold into the college campus.

The main menu will contain six parts which are as follow:

- Student Login
- Teacher login
- Deposit Fee
- Ask Queries Forum
- About Us
- Contact Us

Now, we can see that nearly everything is very much possible to perform with a single click, so this application will help you to computerize the system of college management so that student and other staff members can access the system online. These are some key features of the system which is as follow:

- To reduce the headache of maintaining the record of students and teachers related documents.
- To reduce the cumbersome job of maintaining several documents like
- It will eliminate the delays in the generation of results and free updating of the students; this system will help in maintaining the records of absent students.
- Searching will become more efficient and fast in comparison of manual searching.
- It will also provide assurance that each employee of the college marked their attendance timely.
- Overall it will reduce the cost and time of the college head in taking care of the college.

Tools: HTML5, CSS3, BOOTSTRAP, Angular, NodeJS, Java Script

Use case 5: E-Payment

In all E-commerce and other online services, payment activity is the most important section and this section mostly having the common procedures. If we make the payment service as the micro service component any web application full stack developer can easily integrate with his project instead of redesigning which is extra burden to developer. So, create the component for the Payment activity with the following operations:

- Getting card details
- Pin verification
- Encryption of card details
- Crediting and debiting the money on respective accounts.
- Payment Acknowledgement.
- Invoice Generation.

Convert this application to Mobile app

Tools:Node.js, Angular

H. Learning Resources**i. Text Books:**

1. Ben Frain “Responsive Web Design with HTML5 and CSS: Develop future-proof responsive websites using the latest HTML5 and CSS techniques”, Packt Publishing Limited, 3rd Edition, 2020. [Unit 1]
2. Mary Delamater , “Murach's JavaScript and jQuery (4th Edition) Paperback – Illustrated”, Mike Murach& Associates Inc, August 26, 2020.[Unit 2]
3. David Herron “Node.js Web Development: Server-side web development made easy with Node 14 using practical examples”, Packt Publishing Limited,5th Edition, 2020. [Unit 3]
4. Yakov Fain “Angular Development with TypeScript”, Manning Publications, 2nd Edition, December 2018. [Unit 4].

ii. Reference Books:

1. AzatMardan, “Practical Node.js: Building Real-World Scalable Web Apps”, 1st Edition.2018.

iii. Online Resources:

1. “Angular the complete guide(2021 Edition)”, Accessed on: May. 6, 2021 [Online], Available: <https://www.udemy.com/course/the-complete-guide-toangular-2/>
2. “Server-side Development with NodeJS, Express and Mongo DB”, Accessed on: May. 06, 2021 [Online], Available: <https://www.coursera.org/learn/server-side-nodejs>
3. “Node.js Micro services for beginners”, Accessed on: May. 06, 2021 [Online], Available: <https://www.udemy.com/course/nodejs-microservices-for-beginners/>

COURSE CODE	COURSE TITLE	L	T	P	C
10211CS213	PYTHON PROGRAMMING	1	0	2	2

A.Preamble

The objective of this course is to provide learners an insight into Python programming, and develop programming skills to manage the development of software systems. It covers programming structures, important instructions, and data representations, file handling, Exception handling, String handling, regular expressions and file data processing. This course also lays the foundation to data processing, UI design and gaming.

B.Prerequisite Course

Nil

C.Course Objectives

Learners are exposed to:

- Understand the basic program structure and elements of Python Language.
- Know the concept of file manipulation, string manipulation and regular expression.
- Learn the exception handling and multithreading concepts using python.
- Use the python modules for data analysis.
- Apply the python modules for gaming and UI design.

D.Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K – Level
CO1	Experiment with data types, operators, statements and functions in python.	K3/S3
CO2	Construct the file handling, string handling and regular expression functions in python	K3/S3
CO3	Make use of the Exception handling to handle errors and multithreading mechanism for parallel execution.	K3/S3
CO4	Demonstrate the use of python libraries for data analysis.	K3/S3
CO5	Apply python modules for Graphical User Interface and game design	K3/S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create S3- Apply		

E.Correlation of COs with Program outcomes and Programme Specific Outcomes:

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	1	3									3	1	
CO2	3	1	3	3	1							1	3	1
CO3	3	3	3	2									2	
CO4	3	3	3		3				1		1	3	3	3
CO5			3		3	3		3			3	3	3	3

3- High; 2-Medium; 1-Low

F.Course Contents

Unit 1 Introduction

L-3Hours

Python Basics:variables-data types-operators and expressions-control statements-comments in the program-python collections (List,Tuple,Set,Dictionary)-modules-packages and composition. Python functions-built-in functions-Lambda functions-python iterator and generator.

Case Study: Shuffling a Deck of Card

Unit 2 File Manipulation, String Handling and Regular Express

L-3Hours

Manipulating files and directories, os and sys modules; text files: reading/writing text and binary files; creating and reading a formatted file (csv or tab-separated); String manipulations: indexing, slicing a string-string operations-number system-Regular expressions- match and search functions; modifiers and patterns-python Decorators.

Case Studies: Creating a Hash File (or a message digest of a file) and pattern recognition

Unit 3 Exception Handling and Multi-Threading in python

L-3 Hours

Exception handling: try, except and finally block, handling multiple exceptions, raise an exception, User defined exception- python multithreading- thread and threading module-Synchronizing Threads in Python.

Case Study: Development of student performance evaluation report

Unit 4 Data Analysis using Python libraries

L-3Hours

NumPy: Introduction, NdArray object, Data Types, Array Attributes, Indexing and Slicing, Array manipulation, mathematical functions, Matplotlib; Pandas: Introduction to pandas data structures-series-Data Frame-Panel-basic functions-descriptive statistics function-iterating data frames-statistical functions-aggregations-visualization- plotting graphs using plotly Library.

Case Study: Sales Forecasting

Unit 5 UI and Game design using Python libraries

L-3 Hours

Tkinter module: introduction, widgets, standard attributes, Geometry management, Tkinter Event Handling; Database connectivity with MySQL. PyGame Module - PyGame Concepts- Basic Game Design-Sprites-Sprite Groups-Custom Events-Collision Detection-Sprite Images-Game Speed-Sound Effects.

Case Study: Angry bird game and UI design

Total: 30 Hours (L)

G.Laboratory Experiments

Total: 30 Hours

Part 1

Task 1	Running Python Script and various expressions in an interactive interpreter Tools:IDLE,pycharm
Task 2	Implement conditional, control and looping statements. Tools:Python Tutor (or) pycharm (or) IDLE
Task 3	Importing Python modules and packages in python programming. Tools:Jupyter,pandas, numpy,scipy
Task4	Use various data types, List, Tuples and Dictionary in python programming Tools: Python Tutor (or) pycharm (or) IDLE

- Task5** Implement various Searching and Sorting Operations in python programming
Tools: Python Tutor (or) pycharm (or) IDLE
- Task6** Implement various text file operations.
Tools: Python Tutor (or) pycharm (or) IDLE
- Task7** Utilizing ‘Functions’ concepts in Python Programming.
Tools: Python Tutor (or) pycharm (or) IDLE
- Task8** Implement python generator and decorators.
Tools: Python Tutor (or) pycharm (or) IDLE
- Task9** Implement Exceptions and Exceptional handling in Python.
Tools: Python Tutor (or) pycharm (or) IDLE
- Task10** Use the Matplotlib module for plotting in python.
Tools:matplotlib
- Task11** Use the Tkinter module for UI design.
Tools:Tkinter, jupyter
- Task12** Simulate Gaming concepts using Pygame
Tools:Pygame

Part 2

Use Cases:

Use Case 1: Finding the Winning Strategy in a Card Game using python

Use Case 2: Develop an application to perform Plotting Probabilities Using Matplotlib

Use Case 3: Design an application for Simulating Random Coin-Flips and Dice-Rolls Using NumPy

H.Learning Resources

i. Text Books:

1. Guido van Rossum, Learning Python: Crash Course Tutorial, The Python development team publishers, 2020. [Unit-1,2,3]
2. Bharti Motwani, “Data Analytics using Python”, paperback edition, Wiley Publishing Ltd, 2020. [Unit-4]
3. James R. Parker, “Game Development Using Python”, 2nd edition, Mercury Learning and Information publishers, 2021. [Unit-5]

ii. Reference Books:

1. David M.Baezly , “Python Essential Reference”, Addison-Wesley Professional; 5th edition, 2021.
2. John Shovic, Alan Simpson, “Python for dummies”, John Wiley & Sons, 2020.

iii. Online References:

1. Programming for Everybody (Getting Started with Python), Accessed on 02, April 2021 [Online] Available: <https://www.coursera.org/learn/python>
2. Learn Python, Accessed on 02, April 2021 [Online] Available: <https://www.udemy.com/course/python-the-complete-python-developer-course>
3. Introduction to Python Programming, Accessed on 02, April 2021 [Online] Available: <https://www.edx.org/professional-certificate/introduction-to-python-programming>
4. The Python Standard Library, Accessed on 02, April 2021 [Online] Available: <https://docs.python.org/3/library/>

COURSE CODE	COURSE TITLE	L	T	P	C
10211CS223	MACHINE LEARNING TECHNIQUES	2	0	2	3

A. Preamble

This course introduces students to new and actively evolving interdisciplinary field of modern data analysis. The interplay between known data and unknown ones give rise to complex pattern structures and machine learning methods that are the focus of the study. In the course we will consider methods of Machine Learning and Data Mining Learning. It also includes dimensionality reduction techniques which help to deal with huge datasets.

B. Prerequisite Course

10210MA103- Probability, Statistics and Queuing theory

C. Course Objectives

Learners are exposed to:

- Apply the concepts of supervised and unsupervised learning algorithms for real time applications
- Experiment classification and association models to overcome the problem of over fitting
- Analyse and suggest appropriate machine learning approaches for various types of problems
- Demonstrate the aspects of simple perceptron and multi-layer perceptron.

D. Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Examine the basic concepts of data mining and machine learning concepts	K2/S3
CO2	Design and evaluate the dimensionality reduction algorithms using real world datasets.	K3/S3
CO3	Apply various algorithms of Classification and Association.	K3/S3
CO4	Demonstrate experiments to evaluate and compare different unsupervised learning algorithms	K3/S3
CO5	Use the concept of neural networks for learning linear and non-linear activation functions	K3/S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of Cos with Program Outcomes and Programme Specific Outcomes:

Cos	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO9	PO1 0	PO1 1	PSO 1	PSO 2	PSO 3
CO1	2	3	3									3	3	
CO2	3	3	3	3	2	1		1	1			3	3	
CO3	3	3	3	3	2	1		1	1			3	3	
CO4	3	3	3	3	3	1		1	1			3	3	
CO5	3	3	3	3	2	1		1	1			3	3	

3- High; 2-Medium; 1-Low

F. Course Contents

Unit 1 Introduction to Data mining and Machine Learning

L-6 Hours

Introduction to Data mining and Knowledge Discovery – Data – Data Mining Functionalities – Steps in Data Mining Process, Architecture of Typical Data Mining systems - Interesting Pattern - Classification of Data Mining Systems – Machine Learning -Types of machine learning– Model selection and generalization- Evaluation Metric-Perspectives and Issues in Machine Learning – Concept Learning Task – Applications of Data Mining and Machine Learning

Unit 2 Data Preprocessing

L-6 Hours

Data Pre-processing- Needs Pre-processing the Data- Data Cleaning, Data Integration and Transformation, Data Reduction, Discretization and Concept Hierarchy Generation- Dimensionality Reduction – Feature Extraction- Variable Selection- Variable ranking- Linear Discriminant Analysis – Principal Component Analysis – Factor Analysis –Cross Validation –Resampling methods

Unit 3 Association and Classification

L-6 Hours

Association Rule Mining- Mining Multilevel Association Rules- Mining Multidimensional Association Rule-Constraint Based Association Mining –Decision Tree Induction - Bayesian Classification –Support Vector Machines – Linear Regression - Logistic Regression- Decision Trees – Univariate Tree – Multivariate Tree – Regression Trees – ID3- Random forest- Probability and Learning – Gaussian Mixture Models – Nearest Neighbour Methods

Unit 4 Clustering Techniques

L-6 Hours

Cluster Analysis - Types of Clustering methods – Partitioning Clustering- K-means, K-Medoid, K-Mode- Density-Based Clustering-DBSCAN Algorithm - Distribution Model Based Clustering - Maximum likelihood Estimation- Expectation-Maximization Algorithm- Hierarchical Clustering- Agglomerative Hierarchical algorithm- Fuzzy Clustering- Fuzzy C-means algorithm, Agglomerative and Divisive Clustering –BRICH - ROCK- Chameleon

Unit 5 Neural Networks

L-6 Hours

ANN- Simple Perceptron- Gate implementation- Multi-layer Perceptron – Going Forwards – Going Backwards: Back Propagation Error – Multi-layer Perceptron in Practice – Examples of using the MLP – Overview – Deriving Back-Propagation- Optimization- Hyper parameter tuning

Total: 30Hours

Total: 30Hours

G. Laboratory Experiments

		<u>Part-1</u>	
Task No		Task	CO/Skill Level
Task 1	i.	Implement the Find-S algorithm to learn the most specific hypothesis from a dataset.	CO1/S3
	ii.	Implement the Candidate Elimination algorithm to compute the version space for a concept.	
	iii.	Compare the outputs of Find-S and Candidate Elimination on the same dataset.	
	iv.	Understand how these algorithms behave with noisy or incomplete data.	
Platform: Orange, Google co-lab, Anaconda navigator, Language: Python			
Task 2	i.	Perform Data Pre-processing, Data Analysis and Visualization for a given dataset. Platform: Orange, Language: Python.	CO2/S3

Task 3	i. Implement Principal Component Analysis (PCA) on a high-dimensional dataset. ii. Compare results of PCA with Linear Discriminant Analysis (LDA). Platform: Orange, Google co-lab, Anaconda navigator, Language: Python	CO2/S3
Task 4	i. Use Apriori and FP-growth algorithm to find all frequent item sets for the chosen dataset and also generate Association Rules. ii. Analyze and interpret generated rules. Platform: Rapidminer, Language: Python	CO3/S3
Task 5	Build a model using any of the classification algorithms that can effectively analyze and classify a new sample. Calculate the classification rate, accuracy, precision, recall, F1-score and confusion matrix for your data set. Platform: Orange, Google co-lab, Anaconda navigator, Language: Python	CO3/S3
Task 6	i. Build a simple linear regression model. Evaluate the model's performance using appropriate metrics like R-squared and Mean Squared Error. Use the model to predict the target variable for a given value of the predictor. ii. Perform exploratory data analysis (EDA) to identify correlations between variables. Fit a multiple regression model using all the independent variables. Check for multicollinearity among predictors and address it if necessary. iii. Build a logistic regression model. What is the equation for the log-odds (logit function)? Interpret the coefficients of the predictors. How do they influence the probability of the binary outcome? Evaluate the model using metrics such as accuracy, precision, recall, F1-score, and ROC-AUC. Create a confusion matrix. Platform: Google co-lab, Language: Python	CO3/S3
Task 7	Apply partitioning and hierarchical clustering algorithms for a given dataset. Compare the results and comment on the quality of clustering using evaluation metrics Platform: Google co-lab, Language: Python	CO4/S3
Task 8	Apply back propagation neural network on image data. The idea is to build a Artificial Neural Network model that can effectively analyze and extract features from an image. Platform: Google co-lab Language: Python	CO5/S3
Task 9	Perceptron Implement a perceptron to solve a linearly separable problem. i. Demonstrate logical gate implementation using perceptron ii. Implement an MLP from scratch using Python/NumPy for a classification task. iii. Train the MLP using the backpropagation algorithm and evaluate its performance. Platform: Google co-lab, Language: Python	CO5/S3
Task 10	Hyperparameter Tuning: i. Use grid search or random search to optimize hyperparameters for an MLP. ii. Compare results before and after tuning. Platform: Google co-lab, Language: Python	CO5/S3

Part-2

Use Cases:

Use Case No	Use Case	CO/Skill Level
1	How can a machine learning system be developed to monitor student performance in an online education platform by incorporating data preprocessing (e.g., handling missing grades, normalizing time spent on activities), classification (e.g., predicting if a student will pass or fail a course), clustering (e.g., grouping students based on learning patterns or engagement levels), perceptron (e.g., using a simple neural network to classify students into high-risk or low-risk categories for dropout), hyperparameter tuning (e.g., optimizing the classification and clustering models), with inference (e.g., identifying underperforming students early) and visualization (e.g., generating dashboards showing clusters, classification probabilities, and risk levels)?	CO1-CO5/S3
2	How can a machine learning solution be developed for a retail store to analyze customer purchasing behavior, involving data preprocessing (e.g., cleaning and normalizing transaction data), classification (e.g., predicting whether a customer will buy a particular product), clustering (e.g., segmenting customers into groups based on their purchase patterns), perceptron (e.g., using it to build a simple binary model for predicting high-value customers), and hyperparameter tuning (e.g., optimizing the classification and clustering algorithms for better performance)?	CO1-CO5/S3
3	How can a machine learning system be designed to optimize vehicle maintenance schedules for a logistics company by using data preprocessing (e.g., cleaning sensor data, imputing missing values, and scaling usage metrics), classification (e.g., predicting whether a vehicle is likely to require maintenance soon), clustering (e.g., grouping vehicles based on usage patterns and wear-and-tear factors), perceptron (e.g., using a simple neural network to classify vehicles into "maintenance required" or "maintenance not required"), hyperparameter tuning (e.g., optimizing the classification and clustering algorithms), with inference (e.g., identifying high-risk vehicles in real time) and visualization (e.g., generating heatmaps and timelines of maintenance predictions and vehicle clusters)?	CO1-CO5/S3
4	How can a machine learning system be implemented to detect fraud in online financial transactions by incorporating data preprocessing (e.g., removing duplicates, handling missing values, and normalizing transaction amounts), classification (e.g., predicting whether a transaction is fraudulent or legitimate), clustering (e.g., grouping transactions to identify anomalous behavior patterns), perceptron (e.g., applying a basic neural network for binary fraud detection), hyperparameter tuning (e.g., fine-tuning the model to improve precision and recall), with inference (e.g., flagging suspicious transactions in real time) and visualization (e.g., creating dashboards to display fraud patterns and cluster distributions)?	CO1-CO5/S3

G. Learning Resources

i. Text Books:

1. Ethem Alpaydin,, “Introduction to Machine Learning”, 4th edition, MIT Press, 2020 (Unit 1 to 4)
2. Christopher M Bishop, “Pattern Recognition and Machine Learning”, Springer-Verlag New York Inc, 2018
3. Tom Mitchell, “Machine Learning: A Guide to Current Research”, Kluwer academic publisher, 2019.

ii. Reference Books:

1. Han, J., Pei, J., &Kamber, M. *Data mining: Concepts and techniques* (3rd ed.). Morgan Kaufmann., reprint, 2019
2. Oliver Theobald, “Machine Learning for Absolute Beginners", Scatterplot Press, third edition, 2020
3. Beattie, Guy, “Machine Learning: Start with Machine Learning with all new tipstricks2020.Understand how machine learning can help”, 2020

iii. Online References:

1. Machine Learning, accessed online April 20, 2021 [online] <https://www.coursera.org/learn/machine-learning>
2. Machine Learning by Georgia Tech, accessed online April 20, 2021[online] Available: <https://www.udacity.com/course/machine-learning--ud2>
3. Data mining, Stanford University, Accessed on 19th June 2021, [Online] http://statweb.stanford.edu/~lpekelis/13_datafest_cart/WekaManual-3-7-8.pdf
4. Data Warehouse: Edgewood Solutions, Accessed on 28th June 2021, [Online] <https://www.mssqltips.com/sqlservertip/5690/create-a-star-schema-data-model-in-sql-server-using-the-microsoft-toolset/>
5. Data mining, University of Waikato, Accessed on 19th June 2021, [Online] Available :https://waikato.github.io/weka-wiki/downloading_weka/

COURSE CODE	COURSE TITLE	L	T	P	C
10211CS224	Full Stack Application Development	1	0	4	3

A. Preamble

This course provides practical knowledge of database systems, web development, backend programming, and cloud technologies. Students gain hands-on experience in building dynamic web applications using modern frameworks, version control tools, micro services, DevOps practices, and cloud platforms. The course focuses on applying concepts to develop secure, scalable, and deployable software solutions.

B. Prerequisite Course: Nil

C. Course Objectives

Learners are able to:

- Understand and use database queries and web technologies to build interactive applications.
- Apply Spring Boot concepts to develop backend and RESTful services.
- Utilize micro services architecture for building scalable applications.
- Practice version control, build tools, and DevOps workflows for software deployment.
- Explore cloud platforms and AI-assisted coding techniques for modern application development.

D. Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K Level
CO1	Use database queries and web technologies to create dynamic web pages.	K3
CO2	Implement backend services using Spring Boot and RESTful APIs.	K3
CO3	Demonstrate micro service-based applications using Spring frameworks.	K3
CO4	Apply version control, build tools, and CI/CD practices for application delivery.	K3
CO5	Build cloud-enabled applications using cloud services and AI-based coding tools.	K3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of COs with Program Outcomes and Programme Specific Outcomes

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	2	2	3	1	2	-	-	-	-	-	-	2	3	2
CO2	3	2	3	2	3	-	-	-	-	-	-	2	3	2
CO3	3	3	3	2	3	-	-	-	-	-	-	2	3	3
CO4	2	2	2	2	3	-	-	-	-	-	-	1	3	3
CO5	2	2	3	2	3	-	-	-	-	-	-	1	3	3

3- High; 2-Medium; 1-Low

F. Course Contents

Unit I: Database Management and Web Development basics

L - 3 Hours

SELECT statement – Aggregate functions - Sorting and Filtering Data - Joins - Subquery. Working with Date- Triggers, Indexes, and Views - Transactions, Commit, and Rollback – SQL Built-in and User-Defined Functions-GraphQL HTML5 – Tags in HTML – Design HTML Forms, Handling user inputs, - use of required. CSS3. JavaScript for Dynamic Web - ES6 features - DOM manipulation – GetElementById - Form validation. Event handling - Mouse click and double click events - Keyboard events – Developer tools.

Unit II: Spring Boot Core and Backend Development

L - 3 Hours

Spring MVC – Annotation - Spring Core: Inversion of Control and Dependency Injection – Auto wired - BeanFactory- Spring Boot Web Stack (Servlet-based Architecture) - Object-Relational Mapping Related Annotations - Data Access Layer using Spring Data JPA. RESTful API Design and Development - Exception Handling and Validation – Spring Security Implementation.

Unit III: Advanced Spring, Microservices Programming

L – 3 Hours

Microservices Design Principles - Service Registry and Discovery - API Gateway and Load Balancing - Spring Messaging (JMS) – React JS - TypeScript- Web application into mobile application – Capacitor package.

Unit IV: Version Control, Build Tools and DevOps Foundations

L - 3 Hours

Git & GitHub Fundamentals – Repository Management - Versioning - Branching, Merging. Build Automation using Maven - DevOps - Continuous Integration and Continuous Deployment (CI/CD) using Jenkins, GitHub Actions, and GitLab CI/CD.

Unit V: Cloud Computing and Vibe Coding

L - 3 Hours

Cloud Services: API Gateway, File Storage, Compute, Relational Database Service– Pay as you Use model - Introduction to AWS - Google Cloud - Azure - Platform as a Service (PaaS) – Infrastructure as a Service (IaaS) - Software as a Service (SaaS) – Cloud applications for business. Vibe Coding - Prompt Engineering – Fine tuning the prompt - Generative AI.

TOTAL: 15 Hours

G. Laboratory Experiments

Part A

Task 1: Student Registration & Data Storage

Design a Student Registration Form using HTML5 and CSS3 that collects:

Name, Email, DOB, Department, Phone

Store the data in a database table and retrieve it using SELECT.

Task 2: Data Retrieval & Sorting Dashboard

Description: Create a page that displays student or employee records with:

- Sorting by name or date

- Filtering by department

- Count of students per department

Task 3: Login System with Validation

Description:

- Develop a Login Page:

- Validate inputs using JavaScript

- Check credentials from database

- Show error messages dynamically

Real-Time Usage: Authentication systems for any web application.

Task 4: Order Management using Joins

Concepts Used: Joins, Subqueries, ORDER BY, CSS layout

Create tables for:

- Customers

- Orders

- Products

Display a customer order history using JOIN queries and a subquery to find:

- Highest value order

- Most active customer

Real-Time Usage: E-commerce and retail systems.

Task 5: Transaction-Based Payment Simulation

Description: Simulate an online payment process:

- Deduct balance from user account

- Add amount to merchant account

- Use COMMIT on success, ROLLBACK on failure

- Real-Time Usage: Banking and digital payment applications.

Task 6: Automated Logging using Triggers & Views

A trigger that logs every INSERT or UPDATE

A view that shows daily activity reports

Real-Time Usage: Audit logging in enterprise databases.

Task 7: Interactive Web Form with Events & Functions

Description: Build an interactive feedback form:

- Validate inputs on keypress

- Highlight fields on mouse hover

- Show confirmation on double-click submit

Use JS functions for reusable validation logic
Real-Time Usage: Customer feedback and survey systems

Task 8: Create a simple Employee Management module using Spring Core. Demonstrate Inversion of Control and Dependency Injection using annotations such as `Component` and `Autowired`. Use `BeanFactory` to manage beans and store employee data in memory.

Task 9: Develop a basic Spring MVC application using annotation-based configuration. Create a controller to accept user requests and display employee details using the MVC flow without XML configuration.

Task 10: Build a Student CRUD application using Spring Boot. Map the database table using JPA annotations such as `@Entity`, `@Id`, `@Table`, and `@Column`. Perform Create, Read, Update, and Delete operations using a relational database.

Task 11: Implement a Data Access Layer using Spring Data JPA. Use `JpaRepository` and custom query methods to retrieve student records based on conditions such as department or age, and include sorting and pagination.

Task 12: Design and develop a RESTful API for Product Management using Spring Boot. Implement GET, POST, PUT, and DELETE methods and return JSON responses. Test the APIs using a REST client.

Task 13: Add exception handling and validation to the RESTful application. Implement global exception handling and apply validation annotations to ensure correct input data and meaningful error responses.

Task 14: Design a simple microservices-based system by splitting a monolithic application into at least two independent services. Apply microservices design principles such as loose coupling, single responsibility, and independent deployment.

Task 15: Implement Service Registry and Discovery using a tool such as Eureka. Register multiple microservices and enable dynamic service lookup instead of hard-coded service URLs.

Task 16: Configure an API Gateway to route client requests to appropriate microservices. Implement basic load balancing so that requests are distributed across multiple instances of a service.

Task 17: Enable inter-service communication using REST-based communication. Create one microservice that consumes REST APIs exposed by another microservice and handle responses and failures gracefully.

Task 18: Write unit test cases for microservices using a testing framework. Test service logic, REST controllers, and data handling independently to ensure reliability.

Task 19: Create a Git repository for a sample project and demonstrate basic Git operations such as initializing a repository, staging files, committing changes, and

maintaining version history. Push the repository to GitHub and manage it using remote repositories.

Task 20: Implement branching strategies in Git by creating feature branches. Perform merging and rebasing operations, and intentionally create merge conflicts to understand and resolve them effectively.

Task 21: Implement a CI/CD pipeline using Jenkins, GitHub Actions, or GitLab CI/CD. Automate code checkout, build, test, and deployment steps, and demonstrate continuous integration and continuous deployment for the application

Task 22: Explore major cloud service providers such as AWS, Google Cloud, and Microsoft Azure. Identify and compare their core services related to compute, storage, databases, and networking, along with the pay-as-you-use pricing model.

Task 23: Apply Vibe Coding and Prompt Engineering techniques using a Generative AI tool. Design effective prompts to generate code snippets, cloud configuration templates, or application logic, and evaluate the quality and efficiency of the generated outputs for real-world business applications.

Task 24: Use Vibe Coding to build a complete cloud-based feature using Generative AI. Write iterative prompts to generate a REST API, cloud deployment steps, and security configurations. Refine the prompts based on the output quality, document prompt versions, and evaluate how prompt engineering improves code accuracy, scalability, and real-world usability.

PART – B

Project -1: Design and develop a React JS based web application for Ticket Booking of an Internal Department Event (such as technical fest, or seminar).

The application should allow users (students/faculty) to view event details and book tickets online.

The system must include the following requirements:

Event Details Module

- Display event name
- Department name
- Event date and time
- Venue
- Ticket price
- Available number of tickets
- Ticket Booking Module

Allow user to enter:

- Name
- Email ID
- Department

- Number of tickets required
- Prevent booking more tickets than available
- Display booking confirmation message after successful booking

Validation Requirements

- All input fields must be mandatory
- Email ID should be in valid format
- Number of tickets should be a positive number
- Display appropriate error messages for invalid inputs

React Concepts to be Used

- Functional components
- JSX
- useState hook for state management
- Event handling
- Conditional rendering
- Output Requirements
- Display updated available ticket count after booking

Show booking summary including:

- User name
- Event name
- Number of tickets booked
- Total amount
- Exception / Error Handling
- Handle invalid inputs gracefully
- Display user-friendly error messages

Additional Features (Optional)

- Reset booking form
- Simple CSS styling for better UI
- Separate components for Event Details and Booking Form

The application should be developed using React JS/Any Other JS and run in a browser. Proper component structure and code readability must be maintained.

Full Stack Project-2: Smart Campus Event Management System

A web application for a college/university to manage campus events, workshops, and seminars. Students can view upcoming events, register, and provide feedback, while admins can create, update, or delete events.

Key Features:

Student Account:

Browse upcoming events (REST APIs + MVC pages)

Register for events with form validation

View registered events

Admin Account:

Add / Edit / Delete events (CRUD using Spring Boot + JPA)

Search events using filters (date, department, type)

View event registration statistics (aggregate functions)

Technical Highlights:

- Spring Core: Dependency Injection (@Autowired), Bean management
- Spring MVC: Annotation-based controllers (@Controller, @RequestMapping)
- Spring Boot: Embedded Tomcat server, REST API development (@RestController)
- Spring Data JPA: Entity mapping (@Entity, @Id, @Table), CRUD operations
- Validation: @NotNull, @Size for form inputs
- Security: Basic authentication for admin login
- Exception Handling: Global exception handling using @ControllerAdvice
- HTML5/CSS3 + Thymeleaf for frontend

Project-3: Job Portal Management System

A web application that connects job seekers and employers. Students can create profiles, upload resumes, browse job listings, and apply online. Employers can post jobs, view applicants, and shortlist candidates.

Key Features:

- User Side (Job Seeker / Student):
- Register/login and maintain profile
- Upload resumes (file storage in database or local folder)
- Search and filter jobs by category, location, or experience
- Apply for jobs and track application status
- Receive notifications for shortlisted jobs

Admin/Employer Side:

Post new jobs (title, description, skills required, salary)

Edit or delete jobs

View applicants for posted jobs

Shortlist or reject applications

Technical Highlights:

Backend (Spring Boot + Spring MVC):

REST APIs for CRUD operations

Controllers using @RestController and @Controller

Validation using @NotNull, @Size

Exception handling with @ControllerAdvice

Database (Spring Data JPA):

Entities: User, Job, Application

Relationships: One-to-Many (Employer → Jobs), Many-to-One (Application → Job/User)

Queries: Custom finder methods, sorting, and filtering

Security:

Spring Security for login and role-based access (Student, Employer, Admin)

Password hashing for security

Frontend:

HTML5/CSS3 and Thymeleaf

Responsive forms and dashboards

Dynamic table filtering (JavaScript/ES6)

Optional Add-ons (Industry-Relevant):

File uploads for resumes (using MultipartFile)

Email notifications for job status

Simple dashboard analytics (number of applications, shortlisted candidates)

Resources Needed:

- MySQL / H2 Database
- Spring Boot starter projects (can use Spring Initializr)
- Any code editor: VS Code, IntelliJ Community
- Localhost server for testing (no paid cloud required)

H. Learning Resources

i. Text Books:

1. Ashok Shah & K. K. Saxena. Web Technologies. University Press. 2019.
2. Douglas Crockford. JavaScript: The Good Parts. O'Reilly Media. 2008.
3. Greg L. Turnquist, Learning Spring Boot 3.0, Simplify the development of production-grade applications using Java and Spring, Third Edition, Packt Publishing Ltd., 2022, ISBN-10 1803233303 and ISBN-13 978-1803233307.
4. Jaya V. & Ramesh P. Vibe Coding and Generative AI for Developers. TechPress Publications. 2024.

ii. Reference Books:

1. Craig Walls. Spring in Action, Fifth Edition, Manning Publications. 2018.
2. Abraham Silberschatz, Henry F. Korth, & S. Sudarshan. Database System Concepts, Seventh Edition, McGraw-Hill Education. 2019.

iii. Online Resources:

1. W3Schools– Free tutorials for HTML, CSS, JavaScript, and more with simple examples and runnable code. Assessed on: Dec. 20, 2025 [Online]

<https://www.w3schools.com>

2. Spring Documentation, Assessed on: Dec. 20, 2025 [Online]
https://spring.io/guides/gs/spring-boot?utm_source=chatgpt.com
3. Engineers Coding Hub –Spring Boot, HTML/CSS/JS tutorials. Assessed on: Dec. 20, 2025 [Online] <https://engineerscodinghub.com/full-stack-projects-with-github-code>.
- 3 Awesome Spring Boot – Curated GitHub repository with blogs, tutorials, and learning resources for Spring Boot. Assessed on: Dec. 20, 2025 [Online]
<https://github.com/RameshMF/awesome-spring-boot>
4. Cloud Computing Platforms (AWS, Google Cloud, Azure) – Accessed on: Dec. 20, 2025 [Online] Available <https://aws.amazon.com>, <https://cloud.google.com>, <https://azure.microsoft.com>
5. Instant Development Environments, Stackblitz.com, Assessed on: Dec. 20, 2025 [Online]
 Available [https://www. Stackblitz.com](https://www.Stackblitz.com)
6. Vercel.com Assessed on: Dec. 20, 2025 [Online] Available
<https://www.vercel.com>
7. Github.com Assessed on: Dec. 20, 2025 [Online] Available
<https://www.Github.com>

COURSE CODE	COURSE TITLE	L	T	P	C
10211CS225	Problem Solving and Testing using Java	1	0	4	3

A. Preamble

This course focuses on applying functional programming, algorithmic problem-solving patterns, object-oriented design, and efficient data handling techniques. It equips students with practical skills for competitive programming and software development through structured coding practices and systematic testing methods.

B. Prerequisite Course: Nil

C. Course Objectives

Learners are able to:

- Practice functional programming features for writing clean and efficient code.
- Apply algorithmic patterns to solve time and space-constrained problems.
- Design reusable object-oriented solutions for programming challenges.
- Select suitable collections for efficient data handling and performance.
- Assess program correctness using testing, debugging, and logging tools.
- **D. Course Outcomes**

Upon the successful completion of the course, students will be able to:

Cos No	Course Outcomes	K Level
CO1	Use functional programming constructs to process data effectively.	K3
CO2	Apply algorithmic thinking to develop optimized solutions.	K3
CO3	Design object-oriented structures for problem solving.	K3
CO4	Utilize Java collections to manage data with performance awareness.	K3
CO5	Implement testing and debugging techniques to ensure program reliability	K3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of COs with Program Outcomes and Programme Specific Outcomes

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	2	1	2	1	3	-	-	-	-	-	-	2	3	2
CO2	3	3	2	2	2	-	-	-	-	-	-	3	3	2
CO3	2	2	3	1	2	-	-	-	-	-	-	2	3	2
CO4	2	2	2	1	3	-	-	-	-	-	-	2	3	2
CO5	1	2	2	2	3	-	-	-	-	-	-	1	3	3

3- High; 2-Medium; 1-Low

F. Course Contents

UNIT I – Functional Programming

L - 3 Hours

Introduction to functional programming concepts – Lambda expressions – Functional interfaces – Method references – Streams API – Date and Time API (java.time package) – Collectors API - Custom comparators and sorting logic

UNIT II – Algorithmic Thinking & Competitive Problem Patterns

L - 3 Hours

Constraint-driven solution design – Competitive problem patterns - Writing Efficient Code: Strassen's matrix multiplication - Kadane's algorithm for finding maximum subarray sum- String Handling – Knuth Morris Pratt pattern matching – Boyer Moore Algorithm for substring search – Manacher Algorithm for finding longest palindromic substring.

UNIT III – Object-Oriented Problem Solving

L - 3 Hours

Object modelling for algorithmic problems - Class to class interaction, Class to method interaction - Inheritance- Designing reusable classes - Polymorphic behavior in algorithm design - Design Pattern - Interface-based design for extensibility - Immutable objects for safe computation - Exception-driven flow control in competitive environments.

UNIT IV – Advanced Data Handling using Collections

L - 3 Hours

Advanced problem solving using Vector, LinkedList, Arrays, Collections, ArrayList, HashSet, HashMap – Frequency counting using HashMap - Nested Collections.

UNIT V – Advanced Software Testing

L - 3 Hours

Black box vs White Box – Stress testing - JUNIT – Test Life cycle methods - assertEquals - TimeOut error - Advanced features — Parameterized Testing - Logging: Log4j framework utilization. Debugging: How to Debug in Integrated Development Environment, Launch/Attach Breakpoints/Conditional Breakpoints - Step In/Out/Over Variables. Edge case identification and Stress testing.

TOTAL: 15 Hours

G. Laboratory Experiments

PART –A

Task 1. Given a list of integers, find the second highest number in the list using stream API. If there is no second highest number (all elements same or only one element), print -1.

Input Format:

First line: An integer N - the number of integers in the list.

Second line: N space-separated integers - the elements of the list.

Output Format:

Print a single integer - the second highest number. Print -1 if it does not exist.

Sample Input 1:

5

4 2 8 1 5

Sample Output 1:

5

Sample Input 2:

3

5 5 5

Sample Output 2:

-1

Task 2. Write the following methods that return a lambda expression performing a specified action:

- PerformOperation isOdd(): The lambda expression must return true if a number is odd or false if it is even.
- PerformOperation isPrime(): The lambda expression must return true if a number is prime or false if it is composite.
- PerformOperation isPalindrome(): The lambda expression must return true if a number is a palindrome or false if it is not.

Input Format

The first line contains an integer, (the number of test cases).

The subsequent lines each describe a test case in the form of space-separated integers:

The first integer specifies the condition to check for (for Odd/Even, for Prime, or for Palindrome). The second integer denotes the number to be checked.

Sample Input:

5

1 4

2 5

3 898

1 3

2 12

Sample Output

EVEN

PRIME

PALINDROME

ODD

COMPOSITE

Task 3. Given a list of persons, each with a name (String) and age (integer). Sort the persons alphabetically by name using method references. Filter the persons older than a given age limit using a static method reference. Convert all names to uppercase using an instance method reference.

Input Format

The first line contains an integer *n* — the number of persons.

The next *n* lines each contain a string *name* and an integer *age* separated by a space.

The last line contains an integer *ageLimit* — the age threshold for filtering.

Output Format

First line: sorted names (alphabetically), space-separated.

Second line: names of persons older than *ageLimit*, space-separated, in the order they appear in the input.

Third line: all names in uppercase, space-separated, in the original order.

Name contains only uppercase and lowercase English letters

Sample Input-1:

```
3
Alice 23
Bob 30
Charlie 25
24
```

Sample Output-1:

```
Alice Bob Charlie
Bob Charlie
ALICE BOB CHARLIE
```

Task 4. A list of events is given, each with a unique name and a date in the format yyyy-MM-dd. The goal is to process this list using Java's Date and Time API (*java.time*) to perform the following operations: first, sort all events chronologically by date; second, determine the earliest and latest events in the list; and third, given a month number, identify all events that occur in that specific month. The

solution should handle parsing dates, sorting them, and filtering based on the month efficiently using the Date and Time API.

Sample Input-1:

```
5
EventA 2025-12-17
EventB 2024-01-10
EventC 2025-06-05
EventD 2024-12-31
EventE 2025-12-01
12
```

Sample Output-1:

```
EventB EventD EventC EventE EventA
EventB
EventA
EventE EventA
```

Task 5. A list of students is given, where each student has a name and total marks. The task is to sort students based on marks in descending order. If two students have the same marks, they must be sorted alphabetically by name. Finally, collect the top K students using Java Streams and Collectors API.

Input Format

The first line contains an integer N, the number of students.

The next N lines contain student details in the format: name marks

The last line contains an integer K, representing the number of top students to display.

Output Format:

Print the names of the top K students in sorted order, separated by space.

Sample Input-1:

```
6
Arun 85
Bala 92
Charan 85
```

Divya 95

Esha 92

Farhan 88

3

Sample Output-1:

Divya Bala Esha

Task 6. A list of sales records is given, where each record contains a category name, item name, and sales amount. The task is to calculate the total sales for each category and also identify the item that contributed the maximum sales amount within that category.

Input Format

The first line contains an integer N, representing the number of sales records.

The next N lines contain sales details in the format:

category itemName amount

Output Format

For each category, print the category name, total sales amount, and the item name with the highest sales in that category. Categories should be printed in the order of first appearance.

Sample Input-1:

7

Electronics TV 50000

Electronics Mobile 40000

Furniture Sofa 30000

Electronics Laptop 70000

Furniture Chair 15000

Furniture Table 20000

Electronics Headphones 10000

Sample Output-1:

Electronics 170000 Laptop

Furniture 65000 Sofa

Task 7. Given an array `arr[]`, Find the prefix sum of the array. A prefix sum array is another array `prefixSum[]` of the same size, such that `prefixSum[i]` is `arr[0] + arr[1] + arr[2] . . . arr[i]`.

Sample Input-1:

5

10 20 10 5 15

Sample Output-1:

10 30 40 45 60

Explanation: For each index `i`, add all the elements from 0 to `i`

Task 8. You are given an array of integers `nums`, there is a sliding window of size `k` which is moving from the very left of the array to the very right. You can only see the `k` numbers in the window. Each time the sliding window moves right by one position. Return the max sliding window.

Sample Input 1: `nums = [1,3,-1,-3,5,3,6,7]`, `k = 3`

Sample Output 1: `[3,3,5,5,6,7]`

Explanation:

Window position	Max
-----	-----
[1 3 -1] -3 5 3 6 7	3
1 [3 -1 -3] 5 3 6 7	3
1 3 [-1 -3 5] 3 6 7	5
1 3 -1 [-3 5 3] 6 7	5
1 3 -1 -3 [5 3 6] 7	6
1 3 -1 -3 5 [3 6 7]	7

Sample Input-2:

`nums = [1]`, `k = 1`

Sample Output-2: `[1]`

Task 9. Given an integer array `nums` sorted in non-decreasing order, remove the duplicates in-place such that each unique element appears only once. The relative order of the elements should be kept the same.

Consider the number of unique elements in `nums` to be `k`. After removing duplicates, return the number of unique elements `k`.

The first k elements of `nums` should contain the unique numbers in sorted order. The remaining elements beyond index $k - 1$ can be ignored.

Sample input 1:

`nums = [1,1,2]`

Sample Output: 2, `nums = [1,2]`

Explanation: Your function should return $k = 2$, with the first two elements of `nums` being 1 and 2 respectively.

Sample input 2:

`nums = [0,0,1,1,1,2,2,3,3,4]`

Sample Output -2: 5, `nums = [0,1,2,3,4]`

Explanation: Your function should return $k = 5$, with the first five elements of `nums` being 0, 1, 2, 3, and 4 respectively.

Task 10. Given a word pattern and a string text consisting of lowercase characters, the task is to return the count of substrings in text which are anagrams of the pattern.

Sample Input-1: `text = "forxxorfxdofr"`, `pattern = "for"`

Sample Output-1: 3

Explanation: Anagrams present are for, orf and ofr. Each appears in the text once and hence the count is 3.

Sample Input-2: `text = "aabaabaa"`, `pattern = "aaba"`

Sample Output-2: 4

Explanation : Anagrams present are aaba and abaa. Each appears twice in the text and hence the count is 4.

Task 11. You are given N integers. Find the count of numbers divisible by K .

Input Format

The input begins with two positive integers N, K . The next N lines contain one positive integer each denoted by A_i . **Output a single number denoting how many integers are divisible by K .**

Sample Input 1:

7 3

1

51

966369

7

9

999996

11

Sample Output-1:

4

Task 12. Given a lowercase English string *s* of length *n*, encrypt it as follows:

Replace each contiguous group of the same character with the character followed by the lowercase hexadecimal representation of its length.

Reverse the entire modified string.

Return the encrypted string.

Note: All Hexadecimal letters should be converted to Lowercase letters.

Sample Input - 1: *s* = "aaaaaaaaaa"

Sample Output - 1: ba

Explanation: Count consecutive characters: "aaaaaaaaaa" → "a11".

Convert frequency to hexadecimal: "a11" → "ab".

Reverse the string: "ab" → "ba".

Sample Input 2: *s* = "abc"

Sample Output 2: 1c1b1a

Explanation: Convert to character-frequency format: "abc" → "a1b1c1".

Convert counts to hexadecimal (no change as 1 remains 1).

Reverse the string: "a1b1c1" → "1c1b1a".

Task 13. Design a simple EntityRecord system to store and display information about multiple entities. Each EntityRecord has the following attributes:

Enrollment_ID (integer, unique identifier)

Student (string, name of the entity or person)

Course (string, type or classification of the entity)

Instructor (string, person responsible or associated with the entity)

startDate (string in YYYY-MM-DD format, representing creation or start date)

duration (integer, representing duration in appropriate units, e.g., weeks or months)

The task is to implement a program that reads multiple records, stores them, and displays the details of all records in the order they were added.

Sample Input-1:

3

101 Alice JavaBasics John 2025-12-20 6

102 Bob PythonAdvanced Mary 2025-12-22 8

103 Charlie DataScience Mike 2026-01-05 10

Sample Output-1:

Enrollment_ID: 101, Student: Alice, Course: JavaBasics, Instructor: John, Start Date: 2025-12-20, Duration: 6 weeks

Enrollment_ID: 102, Student: Bob, Course: PythonAdvanced, Instructor: Mary, Start Date: 2025-12-22, Duration: 8 weeks

Enrollment_ID: 103, Student: Charlie, Course: DataScience, Instructor: Mike, Start Date: 2026-01-05, Duration: 10 weeks

Task 14. Design a simple vehicle rental system using inheritance. There is a base class **Vehicle** with the following attributes:

vehicleId (integer, unique ID)

modelName (string)

baseRent (double)

There are two subclasses that inherit from Vehicle:

Car: has an additional attribute seats (integer).

The total rent for a car is calculated as $\text{baseRent} + (\text{seats} * 100)$.

Bike: has an additional attribute engineCapacity (integer, in cc).

The total rent for a bike is calculated as $\text{baseRent} + (\text{engineCapacity} * 2)$.

The task is to implement these classes and write a program that reads multiple vehicle records (Car or Bike), stores them, calculates the total rent for each vehicle, and then displays the vehicle details along with the total rent.

Sample Input-1:

3

C 101 HondaCivic 2000 5

B 102 YamahaR15 500 150

C 103 ToyotaCorolla 2500 4

Sample Output-1:

Vehicle ID: 101, Model: HondaCivic, Total Rent: 2500.0

Vehicle ID: 102, Model: YamahaR15, Total Rent: 800.0

Vehicle ID: 103, Model: ToyotaCorolla, Total Rent: 2900.0

Task 15. Design a payment processing system for an e-commerce platform using polymorphism.

There is a base class Payment with a method **processPayment**(double amount) which returns a confirmation message.

There are multiple subclasses that inherit from Payment:

CreditCardPayment:

Charges a 2% processing fee on the amount.

PayPalPayment: Charges a flat fee of \$1.50 per transaction.

UPIPayment: No extra fee.

The task is to implement these classes and a program that reads multiple payment instructions, creates the appropriate payment objects, and prints the result of processPayment() for each payment. Use polymorphism so that the main program can call processPayment() on a Payment reference without knowing the specific payment type.

Sample Input-1:

4

C 100

P 50

U 200

C 250

Sample Output-1:

Processed CreditCard payment: Total Amount = 102.00

Processed PayPal payment: Total Amount = 51.50

Processed UPI payment: Total Amount = 200.00

Processed CreditCard payment: Total Amount = 255.00

Task 16. Design a notification system for a web application using interface-based design to ensure extensibility. There is an interface Notification with a method:

```
void sendNotification(String message);
```

There are multiple implementations of this interface:

EmailNotification – sends the message via email.

SMSNotification – sends the message via SMS.

PushNotification – sends the message as a push notification.

The main program reads multiple notification requests, creates the appropriate Notification objects, and calls sendNotification() on them.

Goal: Use an interface so that new types of notifications (like WhatsAppNotification, SlackNotification) can be added in the future without modifying existing code.

Sample Input-1:

4

E Welcome_to_our_platform

S Your_OTP_is_1234

P New_offer_available

E Password_changed_successfully

Sample Output-1:

Sent Email notification: Welcome_to_our_platform

Sent SMS notification: Your_OTP_is_1234

Sent Push notification: New_offer_available

Sent Email notification: Password_changed_successfully

Task 17. Design an online order processing system where certain operations may fail and should be handled using exception-driven flow control.

Each **Order** has the following attributes:

orderId (integer, unique ID)

productName (string)

quantity (integer)

availableStock (integer)

The main program reads multiple orders and attempts to process each order.

If the quantity requested is greater than availableStock, an exception should be thrown and caught to display:

Order <orderId> failed: Insufficient stock

If the order is valid, print:

Order <orderId> processed successfully

The program must continue processing the remaining orders even if some orders fail.

Sample Input-1:

5

101 Laptop 2 5

102 Mouse 10 8

103 Keyboard 3 3

104 Monitor 1 0

105 Speaker 5 10

Sample Output-1:

Order 101 processed successfully

Order 102 failed: Insufficient stock

Order 103 processed successfully

Order 104 failed: Insufficient stock

Order 105 processed successfully

Task 18.

Design a simple banking system where each account has a unique account number, an account holder name, and a balance.

Implement a class Account to store this information.

Create another class Bank that manages multiple accounts and provides the following operations:

depositing money into an account,

withdrawing money from an account,
and transferring money between accounts.

The **Bank** class should interact with **Account** objects to perform these operations safely, ensuring that withdrawals or transfers do not exceed the available balance and that accounts exist before performing any operation.

Your task is to implement these classes and write a program that can process a sequence of banking operations and produce the appropriate output for each operation.

Sample Input-1:

```
2
101 Alice 1000
102 Bob 500
5
DEPOSIT 101 200
WITHDRAW 102 100
TRANSFER 101 102 300
WITHDRAW 102 1000
TRANSFER 101 103 50
```

Sample Output-1:

```
Deposited 200 to Alice
Withdrawn 100 from Bob
Transferred 300 from Alice to Bob
Insufficient balance
Account not found
```

Task 19. Given an integer array `nums` and an integer `k`, return the `k` most frequent elements. You may return the answer in any order.

Sample Input 1:

```
nums = [1,1,1,2,2,3], k = 2
```

Sample Output-1: [1,2]

Sample Input 2:

nums = [1], k = 1

Sample Output-2: [1]

Sample Input -3:

nums = [1,2,1,2,1,2,3,1,3,2], k = 2

Sample Output- 3: [1,2]

PART – B

Task 20. Design a data structure that internally uses **HashMap** that follows the constraints of a Least Recently Used (LRU) cache.

Implement the LRUCache class:

LRUCache(int capacity)

Initialize the LRU cache with positive size capacity.

int get(int key)

Return the value of the key if the key exists, otherwise return -1.

void put(int key, int value)

Update the value of the key if the key exists. Otherwise, add the key-value pair to the cache. If the number of keys exceeds the capacity from this operation, evict the least recently used key.

The functions get and put must each run in $O(1)$ average time complexity.

Sample Input-1:

["LRUCache", "put", "put", "get", "put", "get", "put", "get", "get", "get"]

[[2], [1, 1], [2, 2], [1], [3, 3], [2], [4, 4], [1], [3], [4]]

Sample Output-1:

[null, null, null, 1, null, -1, null, -1, 3, 4]

Explanation

LRUCache lRUCache = new LRUCache(2); // Object creation

lRUCache.put(1, 1); // cache is {1=1}

lRUCache.put(2, 2); // cache is {1=1, 2=2}

lRUCache.get(1); // return 1

```

LRUCache.put(3, 3); // LRU key was 2, evicts key 2, cache is {1=1, 3=3}
LRUCache.get(2); // returns -1 (not found)
LRUCache.put(4, 4); // LRU key was 1, evicts key 1, cache is {4=4, 3=3}
LRUCache.get(1); // return -1 (not found)
LRUCache.get(3); // return 3
LRUCache.get(4); // return 4

```

Task 21. For Given Number A, find if it's a COLORFUL number or not. [Hint: Use HashSet]

Definition of COLORFUL number:

A number can be broken into different contiguous sub-subsequence parts.

Suppose, a number 3245 can be broken into parts like 3 2 4 5 32 24 45 324 245.

And this number is a COLORFUL number, since product of every digit of a contiguous subsequence is different. **Return 1 if A is a COLORFUL number, else return 0.**

Problem Constraints

$0 \leq A \leq 109$

Input Format

The first argument is an integer A.

Output Format

Return 1 if A is a COLORFUL number, else return 0

Sample Input-1:

A = 23

Sample Output-1:

1

Explanation

A = 23

2 3 23

2 -> 2

3 -> 3

23 -> 6. This number 23 is a COLORFUL number since the product of every digit of a sub-sequence is different.

Task 22. Google needs to schedule meetings in different rooms. Each meeting has a startTime and endTime. You are given a list of meetings. Determine the minimum number of meeting rooms required so that all meetings can take place without overlap. Use TreeMap to track ongoing meetings efficiently

Sample Input-1:

3

30 75

0 50

60 150

Sample Output-1:

2

Task 23. Given a list of items, each with a name and a rating, the task is to:

Filter items with rating greater than or equal to a specified threshold (e.g., 8.0).Sort the filtered items by rating in descending order.

If two items have the same rating, sort them lexicographically by name. The items should be stored and manipulated using ArrayList to allow dynamic operations.

Sample Input-1:

5

ItemA 8.8

ItemB 7.5

ItemC 8.6

ItemD 7.9

ItemE 8.4

Sample Output-1:

ItemA 8.8

ItemC 8.6

ItemE 8.4

Task 24.A streaming platform has a RecommendationEngine class that generates movie recommendations based on user preferences.

Write JUnit tests to cover:

Users with no watch history

Users with only one watched movie

Users with identical ratings across multiple genres

Users with extremely large watch history (stress test)

Input Format:

```
User = new User("user1", new ArrayList<>());
```

Output Format:

Ensure recommendations always return at least 5 movies if possible.

JUnit Assertion Example:

```
List<Movie> recommendations = engine.getRecommendations(user);
assertNotNull(recommendations);
assertTrue(recommendations.size() >= 5);
```

Task 25. A company develops a MathUtils class containing methods for: Finding the maximum subarray sum, Computing prefix sums of an array, Performing frequency counting of elements. Write JUnit 5 unit tests for these methods covering:

Empty input arrays

Arrays with negative/positive numbers

Very large arrays (stress testing)

Input Format-1:

Arrays passed programmatically in test cases

Output Format-1:

Test report indicating pass/fail for all scenarios

Task 26. A UserAnalytics class processes a list of users with attributes like age, location, and activity score. The class uses Streams API and Lambda expressions to filter active users and compute average scores. A bug is causing incorrect averages for certain edge cases (empty list or all inactive users). Use IDE debugging tools (breakpoints, conditional breakpoints, step in/out) to identify and fix the bug.

Input Format:

Programmatically created list of user objects

Output Format:

Correct average score and fixed filtering

Task 27. A streaming platform logs events (e.g., play, pause, skip) for millions of users. Implement a `LoggingService` using `Log4j`: INFO for normal events, WARN for repeated fast events, ERROR for system failures (e.g., null event). Write JUnit tests to validate that correct log levels are generated for different scenarios.

Input Format:

Simulated event objects in test cases

Output Format:

Log output verification in console or file

A banking application allows multiple users to perform transactions (deposit and withdraw) on the same account simultaneously. Design an `AccountService` class with the following methods:

Boolean deposit(int amount)

Boolean withdraw(int amount)

Float getBalance()

The system must ensure thread safety, so that the final balance is always correct even when multiple threads access the account concurrently. Implement the `AccountService` class using proper synchronization techniques.

Write JUnit 5 test cases to verify:

- Concurrent deposits do not cause incorrect balance updates.
- Concurrent withdrawals do not allow balance to go negative.
- Mixed deposit and withdrawal operations maintain data consistency.
- Perform a stress test using multiple threads (e.g., 100+ threads).
- Use IDE debugging tools and breakpoints to observe thread execution behavior.

Input Format

Initial balance is set programmatically.

Transactions are simulated using multiple threads in test cases.

Output Format

Final account balance after all operations.

JUnit test report showing pass/fail for concurrency scenarios.

Task 28. A recommendation engine uses `TreeMap` to track top-N items for users based on frequency and ratings. Implement stress tests to handle: Very large input size (10^6 items), Frequent updates and queries, Log execution time using `Log4j` for performance verification.

Input Format:

Simulated item additions and queries

Output Format:

Performance logs

Verification that top-N logic is correct

TOTAL:60 Hours

H. Learning Resources**i. Text Books:**

1. Raoul-Gabriel Urma, Mario Fusco, & Alan Mycroft. (2019). Modern Java in Action, Second Edition, Manning Publications.
2. Joshua Bloch. (2018). Effective Java, Third Edition, Addison-Wesley Professional.
3. Kent Beck & Erich Gamma. (2019). JUnit in Action, Third Edition, Manning Publications.
4. M. A. Weiss, Data Structures and Algorithm Analysis in Java, 3rd ed., Pearson Education, New Delhi, India, ISBN 978-8131760635.
5. M. T. Goodrich, R. Tamassia, M. H. Goldwasser, and S. Banerjee, Data Structures and Algorithms in Java, 6th ed., Indian Adaptation, Wiley India Pvt. Ltd., 2022, ISBN 978-9354247934.

ii. Reference Books:

1. Kathy Sierra & Bert Bates. (2017). Head First Java (2nd ed.). O'Reilly Media.
2. Steven S. Skiena. (2020). The Algorithm Design Manual, Third Edition, Springer.

iii. Online Resources:

1. Competitive Programming - A Complete Guide, [Online] Last Updated: Dec. 22, 2025. Available: <https://www.geeksforgeeks.org/dsa/competitive-programming-a-complete-guide/>
2. Junit5 - [Online] Last Updated: Dec. 22, 2025. <https://www.geeksforgeeks.org/software-testing/introduction-to-junit-5/>
3. Take U Forward - Strivers SDE Sheet - [Online] Last Updated: Dec. 22, 2025. www.takeuforward.org
4. Leetcode.com Assessed on: Dec. 22, 2025 [Online], Available: <https://www.leetcode.com>
5. Codechef.com Assessed on: Dec. 22, 2025 [Online], Available: <https://www.leetcode.com>

COURSE CODE	COURSE TITLE	L	T	P	C
10211CS226	Java Programming	2	0	2	3

A. Preamble

This course focuses on basics of java programming, Object oriented programming, java collection and effective data handling techniques. It equips students with practical skills for competitive programming and software development through structured coding practices and systematic testing methods.

B. Prerequisite Course:

Nil

C. Course Objectives

Learners are able to:

- Apply basic java programming for writing clean code.
- Implement object oriented programming skills and handle exception to solve problems.
- Demonstrate solutions using suitable collections and multi-threaded programming.
- Utilize java data base connectivity to manage data.
- Validate program correctness using testing, debugging, and explore java version 8 features.

D. Course Outcomes

Upon the successful completion of the course, students will be able to:

COs	Course Outcomes	K Level
CO1	Explain the fundamentals of Java and Object Oriented Programming Concepts to solve problems.	K3
CO2	Apply the concepts of java inheritance and exception handling mechanisms for real world problems.	K3
CO3	Solve the problems using Java Collection and Threads	K3
CO4	Utilize JDBC to handle data with database.	K3
CO5	Implement testing techniques to ensure program reliability and learn java version 8 features.	K3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of COs with Program Outcomes and Programme Specific Outcomes

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	1	-	-	-	-	-	-	-	-	-	-	-

CO2	1	3	3	1	1	-	-	-	-	-	-	1	1	-
CO3	1	3	2	2	2	-	-	-	-	-	-	1	1	-
CO4	1	3	3	-	2	-	-	-	-	-	-	-	2	-
CO5	1	3	3	-	3	-	-	-	-	-	-	-	2	-

3- High; 2-Medium; 1-Low

F. Course Contents

UNIT I – Basics of OOP and JAVA

L-6 Hours

Introduction to object-oriented programming - Features of Java - JVM- Keywords- Variables – Data types – Operators - Control statements -Command line arguments - Compile Time Error - Classes and Methods - Objects – Number based problems - Java Packages – Build-in packages - user-defined package - Access specifiers: private, protected, default, public - Constructors - Method Overloading - Type Casting - this keyword- Static – Arrays, Multidimensional Arrays - Array based problems.

UNIT II – Inheritance, Exception Handling and string

L- 6 Hours

Basics of Inheritance - Forms of Inheritance - Super keyword – Final - Method Overriding –Abstraction: Abstract Classes - Interfaces. Exception Handling: Java Exception Hierarchy - Exception Types - Throwing and Catching exceptions - Declaring New Exception Types – Custom exceptions. String class – Immutable String - String handling methods- String based problems - Runtime Time Errors.

UNIT III – Collection and Problem Solving

L - 6 Hours

Java.lang.Object – Array of Objects – java.lang.Arrays – Collection: ArrayList, LinkedList, Vector, HashMap, TreeMap - Debugging and Bug Fixing – Logic Mistake – Infinite Loops – Timeout error. Thread - Life Cycle - Multi-Threaded Programming — Thread Synchronization.

UNIT IV – Data Base Connectivity

L – 6 Hours

Establish Connection to Data Base– Performing Create, Read, Update, Delete (CRUD) operations on Database – Insert, Delete – Update – Select - Join – Commit and Rollback – Debugging the JDBC Code.

UNIT V –Testing and Java 8 features

L - 6 Hours

Software Testing Life Cycle - JUnit - Introduction to Junit Annotations — Configuring unit tests in Integrated Development Environments - Writing and executing unit tests- Java 8 features – Lambda expressions – Functional interfaces – Method references – Streams API – Date and Time API (java.time package) – Collectors API.

TOTAL: 30 Hours

G. Laboratory Experiments

Part –A

Task 1: Basic Conditional Statements and Looping Programs

- A. Given 5 numbers, how many are even or odd?
<https://tests.mettl.com/authenticatekey/607636d7>
- B. Sum of last digit of two given numbers
<https://tests.mettl.com/authenticateKey/783a1fcf>
- C. To check whether a given number is Prime
<https://tests.mettl.com/authenticateKey/b1efaa3d>
- D. Factorial of a number
<https://tests.mettl.com/authenticateKey/8c1f2ae>
- E. Nth Fibonacci
<https://tests.mettl.com/authenticateKey/f390cadf>

Task 2: Array Manipulation

- . Given an array arr [], Find the prefix sum of the array. A prefix sum array is another array prefixSum [] of the same size, such that prefixSum[i] is arr[0] + arr[1] + arr[2] . . . arr[i].

Sample Input-1:

5
10 20 10 5 15

Sample Output-1:

10 30 40 45 60

Explanation: For each index i, add all the elements from 0 to i

Task 3: Array Manipulation and Analysis Programs

A. RETURN THE MAX SLIDING WINDOW.

You are given an array of integers nums, there is a sliding window of size k which is moving from the very left of the array to the very right. You can only see the k numbers in the window. Each time the sliding window moves right by one position.

Sample Input-1: nums = [1,3,-1,-3,5,3,6,7], k = 3

Sample Output-1: [3,3,5,5,6,7]

Explanation:

Window position	Max
-----	-----
[1 3 -1] -3 5 3 6 7	3
1 [3 -1 -3] 5 3 6 7	3

1 3 [-1 -3 5] 3 6 7 5

1 3 -1 [-3 5 3] 6 7 5

1 3 -1 -3 [5 3 6] 7 6

1 3 -1 -3 5 [3 6 7] 7

Sample Input -2: nums = [1], k = 1

Sample Output -2: [1]

B. ADJUSTED SUM

You are given an integer array of size N. If an element is even, add it to the sum. If an element is odd, subtract it from the sum. Find the final adjusted sum.

Sample Input-1:

N = 5

Array = 10 15 20 25 30

Sample Output-1:

20

Explanation:

Even sum = 10 + 20 + 30 = 60

Odd sum = 15 + 25 = 40

Adjusted sum = 60 - 40 = 20

C. ROTATE ARRAY AND FIND MAX DIFFERENCE

Rotate the given array to the right by K positions. After rotation, find the maximum absolute difference between any two adjacent elements.

Sample Input-1:

Array = 3 8 9 7 6

K = 1

Rotated Array:

6 3 8 9 7

Sample Output-1:

5

Explanation:

Absolute differences:

$$|6 - 3| = 3$$

$$|3 - 8| = 5$$

$$|8 - 9| = 1$$

$$|9 - 7| = 2$$

Maximum difference = 5

Task 4: Inheritance Based Application

Design a simple vehicle rental system using inheritance. There is a base class Vehicle with the following attributes:

- vehicleId (integer, unique ID)
- modelName (string)
- baseRent (double)

Create derived classes to calculate rental cost for different vehicle types.

Sample Input:

Car Details: Vehicle ID : 101

Model Name : Honda City

Base Rent : 2000

Bike Details: Vehicle ID : 201

Model Name : Yamaha R15

Base Rent : 800

Sample Output:

---- Car Details ----

Vehicle ID : 101

Model Name : Honda City

Base Rent : 2000.0

Total Rent : 2500.0

---- Bike Details ----

Vehicle ID : 201

Model Name : Yamaha R15

Base Rent : 800.0

Total Rent : 1000.0

Task 5: String Handling

Given a word pattern and a string text consisting of lowercase characters, the task is to return the count of substrings in text which are anagrams of the pattern.

Example 1:

Input : text = "forxxorfxdofr", pattern = "for"

Output : 3

Explanation : Anagrams present are for, orf and ofr. Each appears in the text once and hence the count is 3.

Example 2:

Input : text = "aabaabaa", pattern = "aaba"

Output : 4

Explanation: Anagrams present are aaba and abaa. Each appears twice in the text and hence the count is 4.

Task 6: Exception Handling

A. To demonstrate the following Exceptions Using Separate try–catch blocks:

1. `ArrayIndexOutOfBoundsException`,

Input:

Array size = 3
Array elements = 10 20 30
Access index = 5

Output: `ArrayIndexOutOfBoundsException: Index 5 out of bounds for length 3`

2. `StringIndexOutOfBoundsException`,

Input:

String = "JAVA"
Access character index = 10

Output: `StringIndexOutOfBoundsException: String index out of range: 10`

3. NullPointerException,**Input:**

String reference = null

Operation = length()

Output:

NullPointerException: Cannot invoke "String.length()" because the object is null

4. NumberFormatException**Input:**

String value = "ABC123"

Conversion = Integer.parseInt()

Output: NumberFormatException: For input string: "ABC123"

B. User-defined exception

Define a custom exception class and throw the exception when a user enters an invalid input (for example, age less than 18 or negative balance). Handle the exception using a try-catch block and display an appropriate error message.

Input:

Enter age: 16

Output:

InvalidAgeException: Age must be 18 or above

Task 7: Collection Based Problems**A. HashSet based problems:**

For Given Number A, find if it's a COLORFUL number or not.

Definition for COLORFUL number:

A number can be broken into different contiguous sub-subsequence parts. Suppose, a number 3245 can be broken into parts like 3 2 4 5 32 24 45 324 245. And this number is a COLORFUL number, since product of every digit of a contiguous subsequence is different. Return 1 if A is a COLORFUL number, else return 0.

Problem Constraints: $0 \leq A \leq 10^9$

Input Format: The first argument is an integer A.

Output Format: Return 1 if A is a COLORFUL number, else return 0

Sample Input-1: A = 23

Sample Output-1: 1

Explanation

A = 23

2 3 23

2 -> 2

3 -> 3

23 -> 6

This number is a COLORFUL number since the product of every digit of a sub-sequence is different.

B. ArrayList based problems:

Given a list of items, each with a name and a rating, the task is to: Filter items with rating greater than or equal to a specified threshold (e.g., 8.0). Sort the filtered items by rating in descending order. If two items have the same rating, sort them lexicographically by name. The items should be stored and manipulated using ArrayList to allow dynamic operations.

Sample Input: 5

ItemA 8.8

ItemB 7.5

ItemC 8.6

ItemD 7.9

ItemE 8.4

Sample Output:

ItemA 8.8

ItemC 8.6

ItemE 8.4

Task 8: Thread Based Problems

A. Creating Threads

Write a Java program to create two threads:

One thread should be created by extending the Thread class.

The other thread should be created by implementing the Runnable interface.

Each thread should print its name and run for 5 iterations.

B. Thread Lifecycle Demonstration

Create a Java program that demonstrates different thread states:

New ; Runnable; Running ;

Waiting / Timed Waiting

Terminated

Print the thread state at different points during execution.

C. Thread Sleep and Join

Write a Java program with two threads:

The first thread prints numbers from 1 to 5 with a delay of 1 second.

The second thread should start only after the first thread finishes execution.

Use appropriate thread methods to control execution.

Task 9: JDBC – Database Connectivity

To establish a database connection and create a Student table.

Sample Input:

Database Name: college

User Name : root

Password : password

Sample Output:

Student table created successfully.

A. JDBC – Insert and Display Student Records

Sample Input

Student Records:

101 Anitha 85

102 Ramesh 78

Sample Output

101 Anitha 85

102 Ramesh 78

B. JDBC – Update Student Marks

Sample Input

Student ID : 101

Updated Marks : 90

Sample Output

101 Anitha 90

Task 10: Java 8 Features and Testing

A. Lambda Expression Sorting.

Sample Input: List of integers: 25, 10, 5, 30, 15

Sample Output: Sorted List: [5, 10, 15, 25, 30]

B. Streams Filter Even Numbers.

Sample Input: List of integers: 12, 15, 20, 33, 40

Sample Output:

12

20

40

C. Date and Time API

Write a Java program to display the current date and time, Add 10 days to the current date and Display the updated date using **Java 8 Date and Time API**

Sample Input: system date

Sample Output:

Current Date: 2026-01-26

Date after 10 days: 2026-02-05

D. Write a JUnit program to demonstrate the use of the following annotations:

@Before

@After

@Test

Sample Output:

Before Test

Test Passed

After Test

E. JUnit Basic Unit Test

Write a Java program to perform addition of two numbers. Create a JUnit test case to verify whether the addition method returns the correct result.

Sample Input: 10 20

Sample Output: Test Passed

USE CASE / MINI PROJECT – 1

Title: Student Result Management System

Question:

Design and implement a Java console-based **Student Result Management System**.

The system should:

- Accept student details such as Registration Number, Name, and Marks in multiple subjects
- Calculate total marks, average, and grade
- Store student records using appropriate **Collection classes**
- Display student results in a structured format
- Handle invalid inputs using **exception handling**

Input:

Student details and marks entered through console

Output:

Student result report with total, average, grade, and status

USE CASE / MINI PROJECT – 2

Title: Library Management System using JDBC

Question:

Design and implement a simple **Library Management System** using Java and JDBC.

The system should:

- Connect to a database using JDBC
- Create a table to store book details (Book ID, Book Name, Author, Availability)
- Insert new book records
- Display all available books
- Update book availability when a book is issued or returned

Input:

Book details and user operations

Output:

Display book records and update status messages

USE CASE / MINI PROJECT – 3

Title: Employee Skill Evaluation System using Java 8

Question:

Design and implement an **Employee Skill Evaluation System** using Java 8 features.

The system should:

- Store employee names and skill scores in a collection
- Use **Lambda Expressions** and **Streams** to:
 - Filter employees with skill score above a given threshold
 - Sort employees based on skill score
- Validate inputs and display appropriate messages for invalid data

Input:

Employee details and skill scores

Output:

Filtered and sorted employee skill evaluation report

TOTAL: 60 Hou

H. Learning Resources

i. Text Books:

1. Balaguruswamy E, *Programming in java*, Sixth Edition, Tata McGraw Hill, 2019.
2. Mark Allen Weiss, *Data Structures and Algorithm Analysis in Java*, 3rd ed., Pearson Education, New Delhi, India, ISBN 978-8131760635.
3. Raoul-Gabriel Urma, Mario Fusco, & Alan Mycroft. (2019). *Modern Java in Action*, Second Edition, Manning Publications.
4. Joshua Bloch. (2018). *Effective Java*, Third Edition, Addison-Wesley Professional.
5. Kent Beck & Erich Gamma. (2019). *JUnit in Action*, Third Edition, Manning Publications.
6. M. T. Goodrich, R. Tamassia, M. H. Goldwasser, and S. Banerjee, *Data Structures and Algorithms in Java*, 6th ed., Indian Adaptation, Wiley India Pvt. Ltd., 2022, ISBN 978-9354247934.

ii. Reference Books:

1. Kathy Sierra & Bert Bates. (2017). *Head First Java* (2nd ed.). O'Reilly Media.
2. Steven S. Skiena. (2020). *The Algorithm Design Manual*, Third Edition, Springer.

iii. Online Resources:

1. Competitive Programming - A Complete Guide, Assessed on: Dec. 20, 2025, [Online]. Available: <https://www.geeksforgeeks.org/dsa/competitive-programming-a-complete-guide/>
2. W3schools.com Assessed on: Dec. 20, 2025 [Online], Available: <https://www.w3schools.com/java/default.asp>
3. Codingbat.com Assessed on: Dec. 20, 2025 [Online], Available: <https://www.codingbat.com>
4. Leetcode.com Assessed on: Dec. 20, 2025 [Online], Available: <https://www.leetcode.com>
5. Codechef.com Assessed on: Dec. 20, 2025 [Online], Available: <https://www.codechef.com>
6. TakeUforward.com Assessed on: Dec. 20, 2025 [Online], Available: <https://www.takeuforward.com>

COURSE CODE	COURSE TITLE	L	T	P	C
10211CS302	DATA STRUCTURES LABORATORY	0	0	2	1

A. Preamble

This course is designed to develop skills to design and analyze linear and nonlinear data structures. This laboratory gives a way for students to get hands-on experience in identifying and applying the suitable data structure for the given real-world problem. It enables them to gain knowledge in practical applications of data structures.

B. Prerequisite Course

10210CS101 - Problem Solving using C

C. Corequisite Course

10211CS102- Data Structures

D. Course Objectives

Learners are exposed to:

- Introduce various techniques for representation of the data in the real world.
- Design and implement various data structure algorithms through ADT including List, Stack, Queues.
- Get familiarize and implement nonlinear data structures including Trees and Graphs.
- Implement various techniques of sorting and searching algorithms.

E. Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Design and implement basic array and various linear data structure operations.	S3
CO2	Develop an appropriate algorithm using tree data structure to specified problem definition.	S3
CO3	Utilize appropriate hashing techniques and develop solutions for a range of problems.	S3
CO4	Apply Graph data structure and implement appropriate algorithm for given problem.	S3
CO5	Implement appropriate sorting and searching techniques for solving the problem.	S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

F. Correlation of Cos with Program Outcomes and Programme Specific Outcomes

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	3	2							3	3	3	
CO2	3	3	3	3							3	3	1	
CO3	3	3	3	3							3	3	1	

CO4	3	3	3	3							3	3	1	
CO5	3	3	3	3							3	3	3	

3- High; 2-Medium; 1-Low

G. Course Contents

Part -1

Task 1: Array ADT

Using Array ADT perform the following:

- i) Basic Operations in ADT: Creating an array, displaying of array elements, Inserting an element
- ii) Problems related to: Finding duplicates, missing element, minimum difference and rotations.

Task 2: Linked List

Design, Develop and Implement the following tasks using LinkedList ADT.

- i) Singly Linked List: Operations in Ascending Order
- ii) Doubly Linked List: Get reversed in from the given order
- iii) Circular Linked List: Choose the place in the end of the circle so that you are the last one.

Task 3: Stack ADT

Develop and implement the following tasks using Stack ADT:

- i) Stack using Array
- ii) Find all elements greater than their Right element
- iii) Infix To Postfix Conversion
- iv) Balanced Parenthesis
- v) Tower of Hanoi

Task 4: Queue ADT

Using Queue ADT perform the following:

- i) Basic Operations of Queue ADT Using Array
- ii) Queue Implementation using Stack
- iii) Circular Queue of Characters

Task 5: Binary search Tree

Design, Develop and Implement the following tasks using Binary search Tree.

- i) Basic Operations of BST
- ii) Tree Traversal with Recursion
- iii) Tree Traversal without Recursion
- iv) Find the Depth and Height of the tree
- v) Find the MIN and MAX element of the tree.

Task 6: Hashing

Using Hashing Technique perform the following:

- i) Basic Hashing Technique
- ii) Linear Probing
- iii) Separate Chaining
- iv) Character Frequency Counting
- v) Number Frequency Counting

Task 7: Graph

Design, Develop and Implement the following tasks using Graph Concept.

- i) Read the adjacent Vertices from given Matrices
- ii) Find the Shortest Path using Dijkstra's Algorithm
- iii) Find Minimum Spanning Tree using Prim's algorithm (based on PriorityQueue)
- iv) Find Minimum Spanning Tree using Kruskal's algorithm - using Disjointsubsets and Union find algorithm
- v) Topological Sorting
- vi) Graph Traversal using Depth First Search
- vii) Graph Traversal using Breadth First Search

Task 8: Sorting

Develop and implement the following sorting algorithms:

- i) Insertion Sort
- ii) Merge sort
- iii) Quick Sort
- iv) Bubble sort
- v) Radix sort.

Task 9: Searching

- i) Implementation of Linear Search
- ii) Implementation of Binary Search

Part II

Use Cases

Use Case 1: Railroad Car

Imagine a railroad switching circuit such as the one below. Railroad cars are given unique numbers, from 1 onwards. Cars come in from the right in a random order, and the goal is to assemble the cars in numeric order on the left. For example, to assemble the cars in the sequence shown (2, 1, 3) the car numbered 2 would be switched on to the siding. Then car numbered 1 would be moved on to the siding and then to the left. Next car numbered 2 would be moved up from the siding and assembled behind car 1. Finally, car numbered 3 would be moved from the right to the left, via the siding. Notice that the cars in the siding work as a stack, since if a car is moved on to a nonempty siding the existing cars cannot be accessed until the new car is removed. Develop and implement an algorithm that will rearrange the cars in sequential

order regardless of the order in which they appear on the left. Use the Linear data structure ADT for implementing the use case.

Use Case 2: Card Shuffling

A new casino has recently opened in ChefTown - the capital of Chef Land. Amongst a lot of games, there are also a few games that are played with a pile of cards. In total, there are N cards in the pile. Let us enumerate them by positive integers $1, 2, \dots, N$ in order they appear in the pile from top to bottom. Obviously, the cards must be shuffled before the game. Our protagonist Alex, being a mathematician wants to analyze this process. According to Alex, the card shuffling process in the casino is quite similar to how we do in the real life. Namely, it consists of the following: The croupier takes a consecutive "sub-pile" of cards, starting from the $L1$ -th card from the top and ending at the $R1$ -th card from the top (both inclusive) and moves this sub-pile from the middle of the pile to the very top, not changing the order of the cards in the taken sub-pile. Then she takes a sub-pile, starting at the $L2$ -th card and ending this one at the $R2$ -th one from the top of the current pile and moves it to the top, and so on. At last, she takes a sub-pile, starting at the LM -th card and ending this one at the RM -th one from the top of the current pile and moves it to the top. The process, described above, repeats Q times. Alex wants to collect some data in order to be able to predict the order in which the cards will be lying in the pile after the shuffling, but he is not really good at working with big piles and large repetitions. Design, Develop and implement an algorithm to do the following:

- i) For a given set of segments $([L1, R1], [L2, R2], \dots, [LM, RM])$ and a given permutation, Find the minimum Q such that after Q repetitions of the first step in the algorithm above, the permutation will look like this. It is given that this number won't ever be bigger than 1012.
- ii) For a given set of segments $([L1, R1], [L2, R2], \dots, [LM, RM])$ and a given Q , Determine the order of the cards after the shuffling algorithm above.

Use Case 3: Multiuser Environment

Linux is designing a new OS to operate in a multiuser environment. All the users would be connected to a central machine. This central machine can allocate or de-allocate resources to the connected users in order to maximize overall performance. The OS has a limitation. If a particular program is running on more than 2 computers the central machine terminates this program on all the machines running it. To do this, the central machine assigns a unique identification number Z to each of the programs. The central machine runs a periodic check in which it obtains a list from each of users stating identification number of all the programs running on their systems. If the central machine finds that a particular program is running on more than 2 machines it terminates it on all machines.

This task of periodic checks and obtaining the lists has been implemented in the OS. In order to terminate the programs, the OS need to know in advance the exact number of programs it is going to terminate. So given these lists, design a program which can tell the OS the number of programs it is going to terminate.

Note: You need to specify only the number of ‘programs’ it is going to terminate and not the number of machines on which that program is running. For e.g. If a program is running on 4 computers, for the OS this counts as a single program which it needs to terminate.

Use Case 4: Delivery Vehicles

A computer company in the Silicon Valley area (see Figure) needs to route delivery vehicles between cities on the shortest route. Recognize the appropriate algorithm and demonstrate the following tasks:

- a. Modify the graph ADT to store weights in the arc nodes.
- b. Implement a program, when given the start and destination and display the shortest route between them.

Total: 30 Hours

H. Learning Resources

i. Text Books:

1. Narasimha Karumanchi, "Data Structures and Algorithms made Easy" 5th Edition, Career Monk Publication, Print Replica, 2020.
2. Seymour Lipschutz, "Data Structures with C", Schaum's outlines, Tata McGraw Hill, 2017.

ii. Reference Books:

1. V. Aho, J. E. Hopcroft, and J. D. Ullman, "Data Structures and Algorithms", Pearson Education, First Edition, Reprint 2017.
2. Ellis Horowitz, Sartaj Sahni, Susan Anderson Freed, "Fundamentals of Data Structure in C", Universities Press, 2017.

iii. Online References:

1. "Data Structures using C" 2003- 2021. Accessed on: Apr. 15, 2021 [Online]. Available: <http://www.academictutorials.com/data-structure/>
2. "Linked List" 2015, Accessed on: Apr. 15, 2021 [Online]. Available: <http://www.c4learn.com/mcq/data-structure-mcq/linked-list-mcq-general-questions/>
3. "Advanced Data Structures" 1999 – 2021. Accessed on: Apr. 15, 2021 [Online]. Available: <http://randu.org/tutorials/c/ads.php>
4. "C Data Structure Fundamentals" 2021. Accessed on Apr. 15, 2021 [Online]. Available: <http://www.zentut.com/c-tutorial/>

COURSE CODE	COURSE TITLE	L	T	P	C
10211CS303	COMPUTER NETWORKS LABORATORY	0	0	2	1

A.Preamble

This course provides students with hands on training regarding the design, troubleshooting, modeling and evaluation of computer networks. In this course, students are going to experiment in a passive cable connectivity, and learn about network design and troubleshooting topics such as: frame analysis, routing and routing table updating, route discovery, Switch Trunk Port Issues and many others.

B.Prerequisite Course

10210CS101 -Problem Solving using C

C. Corequisite Course

10211CS105 - Computer Networks

D.Course Objectives

Learners are exposed to:

- Understand the working difference between straight cable and cross over cable.
- Gain core knowledge of Network layer routing protocols and IP addressing
- Learn about basic security configuration and troubleshooting

E. Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Choose the different types of transmission media for real time applications.	S3
CO2	Identify data flow between peer to peer in an IP network.	S3
CO3	Demonstrate the configuration of switches and routers.	S3
CO4	Implement the routing protocols to route the packet in a given network.	S3
CO5	Implement file transfer mechanisms using Socket programming with security configurations.	S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

F.Correlation of Cos with Program Outcomes and Programme Specific Outcomes

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	2				3									2
CO2	2	2	3		3									2
CO3	2	2	3		3									2
CO4	2	2	3		3									2
CO5	2	2	3		3									2

3- High; 2-Medium; 1-Low

G.Course Contents

Part-1

Physical Layer:

Task 1: Passive cable connectivity:

Using raw Un Twisted Pair Cable (UTP) cables and connectors create Cross-over /straight-through cables as per the network requirements to fulfil the hybrid topology creation.

Data Link Layer:

Task 2:Frame Analysis:

- A) Using PCAP tool capture the real time frame and analyses the all-header fields of the same.
- B) Configure a layer 2 switch to send a packet from one LAN (Local Area Network) to another LAN by Link Layer Address.

Task 3: Layer 2 Route Discovery:

Create and implement the following

Execute the ping command from 'PC2' to 'PC5' successfully

Network Layer:

Task 4: Layer 3 Routing:

A) Consider networks A, B and C consist of 4 computers in each network are directly connected with Router 'R'.

Condition: I) network A and C can only communicate each other

II) Portion of Network B can communicate with Network A

Configure the above scenario using Cisco packet tracer or equivalent opensource tool

B) Using a wireless router configure the following

I) Device ID / name II) Device Password III) SSID IV) Wi-Fi Password

V) DHCP (Dynamic Host Configuration Protocol) Address range VI) MAC Binding

Task 5:Routing Table Update:

Implement to update routing table IPV4 and IPV6 addressing table before and after route 2(R2) has been removed from network communication for the given network.

Task 6: Subnetting:

Calculate the range of network and host addresses and subnet mask for the given IP addresses (ex. 172.17.1.5/24) and construct a network for the same using packet tracer/NS2/NS3.

Task 7: Protocol Performance:

Implement Distance Vector and Link State Routing Protocol using packet tracer with at least 4 routers. Justify the performance.

Transport Layer:**Task 8: Socket Programming:**

Using TCP/IP sockets, create a client – server program to make the client send the file name and to make the server send back the contents of the requested file if present.

Task 9: Basic Security Configurations:

Configure basic security consideration in a computer, Switch, Router and Firewall devices.

Task 10: Troubleshooting:

Study and Devise Solutions with appropriate commands to trouble shoot

i) Missing VLANs ii) Switch Trunk Port Issues iii) Switch Access Port Issues iv) Router Configuration Issues v) Remote management issue vi) restoring a managed switch with earlier configuration

Application Layer:**Task 11: Remote Login:**

Access the server's FTP service from the PC, capture the packet and analyze it.

Part-2**Use Cases****Use Case1: Designing LAN for an Operating Centre**

Design an operational LAN network-1 with departments A, B are having 2 computers each, accessing the switch-1 located in Building 'B1' and the same part of departments which is operational at Building 'B2' are accessing the switch-2. Both switches are connected with Router 'R1' and the router 'R1' is serially connected with router 'R2'. R2 is directly connected with a PC ("stu VTU number") using a Switch-3.

A Laptop 'Admin-PC' with Wireless connectivity to manage all network devices remotely and an application server 'Server-S1' is located in a network operating Centre connected with Router 'R1' via Switch-4 and is accessible to all departments.

Connectivity:

- All Network devices are connected to each other through FOC
- In-case of any additional devices required can be added
- Department C shall not communicate with Department A& B but can communicate with Server
- All not used Ports shall be Disabled and All switch requires login Password facility for remote users
- From laptop all switches need to be configured/Access for remote management
-

Use Case 2:

Create a Lan Network with 5 Wi-Fi device and 5 CCTV camera under a switch. Both traffic channels shall be separated and Only CCTV cameras need to be accessed from remote network beyond router. For that student may configure Port forwarding method.

Total: 30 Hours

G.Learning Resources**i. Text Books:**

1. Behrouz Forouzan, “Data Communications and Networking”, Tata McGraw Hill, 5th Edition, 2021.
2. Andrew S. Tanenbaum, David J Wetherall, Computer Networks, 5th Edition, Pearson Edu, 2016.

ii. Reference Books:

1. Kurose and K.W. Ross, “Computer Networking: A Top-Down Approach Featuring the Internet”, Addison-Wesley, 2013
2. W. Stallings, “Data and Computer Communication”, PHI, 10th Edition, 2017.
3. Foundations of Modern Networking: SDN, NFV, QoE, IoT, and Cloud, 5th Edition, 2015

iii. Online References:

1. “Networking Fundamentals” CISCO [online] 2019, https://www.cisco.com/c/dam/global/fi-fi/assets/docs/SMB_University_120307_Networking_Fundamentals.pdf
2. “Software Testing Help” [Online] 2020, <https://www.softwaretestinghelp.com/router-setup-configuration/>
3. “Cisco ICND1 Foundation Learning Guide: LANs and Ethernet” CISCO [online] 2013, <https://www.udacity.com/course/computer-networking--ud436>.
4. “Fundamentals of Network Communication” Course Era [Online] 2021, <https://www.coursera.org/learn/fundamentals-network-communications>.
5. “Introduction to packet tracer” CISCO [online] 2019, <https://www.netacad.com/courses/packet-tracer/introduction-packet-tracer>

COURSE CODE	COURSE TITLE	L	T	P	C
10211CS312	FUNDAMENTALS OF COMPUTER NETWORKS LABORATORY	0	0	2	1

A. Preamble

This course provides students with hands on training regarding the design, troubleshooting, modelling and evaluation of computer networks. In this course, students are going to experiment in a passive cable connectivity, and learn about network design and troubleshooting topics such as: frame analysis, routing and routing table updating, route discovery, Switch Trunk Port Issues and many others.

B.Prerequisite Courses

10210CS101 - Problem Solving using C

C.Course Objectives

Learners are exposed to

- Understand the working difference between straight cable and cross over cable.
- Gain core knowledge of Network layer routing protocols and IP addressing
- Learn about basic security configuration and troubleshooting

D.Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Implement the network using network components	S3
CO2	Examine the frame analysis and routing in layer2 network.	S3
CO3	Apply the routing technique and create the subnetting in layer3 network.	S3
CO4	Demonstrate the Socket programming and datagram approaches.	S3
CO5	Explain the application layer protocols for a given real-time scenario.	S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create S3 – Skill Level, Precision		

E.Correlation of COs with POs:

CO s	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO1 0	PO1 1	PSO 1	PSO 2	PSO 3
CO 1	2				3									2
CO 2	2	2	3		3									2
CO 3	2	2	3		3									2

CO 4	2	2	3		3									2
CO 5	2	2	3		3									2

3- High; 2-Medium; 1-Low

F.Course Contents

Part 1

Physical Layer:

Task 1: Implement Peer-to-Peer network using network components:

CO1, K3

Using raw Un Twisted Pair Cable (UTP) cables and connectors create Cross-over /straight-through cables as per the network requirements to fulfil the hybrid topology creation.

Task 2: Implement a client-server architecture using wireless medium.

CO1, K3

Data Link Layer:

Task 3:Frame Analysis:

CO2, K3

A) Using PCAP tool capture the real time frame and analyses the all-header fields of the same.

B) Configure a layer 2 switch to send a packet from one LAN (Local Area Network) to another LAN by Link Layer Address.

Task 4: Examine the MAC table using ARP

CO2, K3

Task 5: Layer 2 Route Discovery:

CO2, K3

Configure a layer 2 switch to send a packet from one LAN (Local Area Network) to another LAN by Link Layer Address.

Execute the ping command from 'PC2' to 'PC5' successfully.

Network Layer:

Task 6: Subnetting

CO3, K3

Calculate the range of network and host addresses and subnet mask for the given IP addresses (ex. 172.17.1.5/24) and construct a network for the same using packet tracer/NS2/NS3.

Task 7: Layer 3 Routing:

CO3, K3

A) Consider networks A, B and C consist of 4 computers in each network are directly connected with Router 'R'.

Condition: I) network A and C can only communicate each other

II) Portion of Network B can communicate with Network A

Configure the above scenario using Cisco packet tracer or equivalent opensource tool

B) Using a wireless router configure the following

I) Device ID / name II) Device Password III) SSID IV) Wi-Fi Password

V) DHCP (Dynamic Host Configuration Protocol) Address range VI) MAC Binding.

Task 8: Configuring RIP:

CO3, K3

Implement to update routing information protocol table for the given network.

Task 9:Routing Table Update:

CO3, K3

Implement to update routing table IPV4 and IPV6 addressing table before and after route 2(R2) has been removed from network communication for the given network.

Task 10: Protocol Performance:**CO3, K3**

Implement Distance Vector and Link State Routing Protocol using packet tracer with at least 4 routers. Justify the performance.

Transport Layer:**Task 11: Socket Programming:****CO4, K3**

Using TCP/IP sockets, create a client – server program to make the client send the file name and to make the server send back the contents of the requested file if present.

Task 12: Implement the datagram approaches.**CO4, K3****Task 13: Basic Security Configurations:****CO4, K3**

Configure basic security consideration in a computer, Switch, Router and Firewall devices.

Task 14: Troubleshooting:**CO3, CO4, K3**

Study and Devise Solutions with appropriate commands to trouble shoot

i) Missing VLANs ii) Switch Trunk Port Issues iii) Switch Access Port Issues iv) Router Configuration Issues v) Remote management issue vi) restoring a managed switch with earlier configuration.

Application Layer:**Task 15: Remote Login:****CO5, K3**

Access the server's FTP service from the PC, capture the packet and analyze it.

Task 16: Implement the DNS services.**CO5, K3****Task 17: Implement the web services using HTTP/HTTPS.****CO5, K3****Task 18: Implement the mail service using SMTP, POP3, and IMAP.****CO5, K3****Part 2****Use Case1: Designing LAN for an Operating Centre**

Design an operational LAN network-1 with departments A, B are having 2 computers each, accessing the switch-1 located in Building 'B1' and the same part of departments which is operational at Building 'B2' are accessing the switch-2. Both switches are connected with Router 'R1' and the router 'R1' is serially connected with router 'R2'. R2 is directly connected with a PC ("stu VTU number") using a Switch-3.

A Laptop 'Admin-PC' with Wireless connectivity to manage all network devices remotely and an application server 'Server-S1' is located in a network operating Centre connected with Router 'R1' via Switch-4 and is accessible to all departments.

Connectivity:

- # All Network devices are connected to each other through FOC
- # In-case of any additional devices required can be added
- # Department C shall not communicate with Department A& B but can communicate with Server
- # All not used Ports shall be Disabled and All switch requires login Password facility for remote users
- # from laptop all switches need to be configured/Access for remote management

Use Case 2:

Create a Lan Network with 5 Wi-Fi device and 5 CCTV camera under a switch. Both traffic channels shall be separated and Only CCTV cameras need to be accessed from remote network beyond router. For that student may configure Port forwarding method.

Total: 30 Hours**F.Learning Resources**

i.Text Books:

1. Behrouz Forouzan, “Data Communications and Networking”, Tata McGraw Hill, 5th Edition, 2021.
2. Andrew S. Tanenbaum, David J Wetherall, Computer Networks, 5th Edition, Pearson Edu, 2016.

ii.Reference Books:

1. Kurose and K.W. Ross, “Computer Networking: A Top-Down Approach Featuring the Internet”, Addison-Wesley, 2013
2. W. Stallings, “Data and Computer Communication”, PHI, 10th Edition, 2017.
3. Foundations of Modern Networking: SDN, NFV, QoE, IoT, and Cloud, 5th Edition, 2015

iii.Online References:

1. “Networking Fundamentals” CISCO [online] 2019,
https://www.cisco.com/c/dam/global/fi-fi/assets/docs/SMB_University_120307_Networking_Fundamentals.pdf

COURSE CODE	COURSE TITLE	L	T	P	C
10211CS313	MACHINE LEARNING LABORATORY	0	0	2	1

A. Preamble

This course introduces students to new and actively evolving interdisciplinary field of modern data analysis. The interplay between known data and unknown ones give rise to complex pattern structures and machine learning methods that are the focus of the study. In the course we will consider methods of Machine Learning and Data Mining Learning. It also includes dimensionality reduction techniques which help to deal with huge datasets

B. Prerequisite Course

10210MA103- Probability, Statistics and Queuing theory

C. Corequisite Course

10211CS120–Machine Learning

D. Course Objectives

Learners are exposed to:

- Implement the Machine Learning concepts for real time applications
- Experiment classification and association models to overcome the problem of over fitting
- Analyse and suggest appropriate machine learning approaches for various types of problems
- Demonstrate the aspects of neural networks for real scenario.

E.Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Apply data mining algorithms and concept learning algorithms	S3
CO2	Experiment pre-processing techniques and dimensionality reduction techniques in real time problems	S3
CO3	Use the concept of various regression and classification algorithms	S3
CO4	Implement artificial neural networks concepts to solve complex problems	S3
CO5	Model various clustering algorithms to solve real time problems	S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

F. Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	2	2										3		
CO2	3	3	3	2	3							3	2	2
CO3	3	3	3	2	3							3	2	2
CO4	3	3	3	2	2							3	2	2
CO5	3	2	2	2	2							3	2	2

3- High; 2-Medium; 1-Low

G.Course Contents**Part-1**

- TASK 1** Implement Apriori and FP-growth algorithm to find all frequent item sets for the chosen dataset and also generate Association Rules.
Platform: Rapidminer, Language: Python
- TASK 2** Demonstrate the FIND-S algorithm to find a most specific hypothesis and Candidate-Elimination algorithm to output a description of the set of all hypotheses consistent with the training examples.
Platform: Orange, Google co-lab, Anaconda navigator, Language: Python
- TASK 3** Perform Data Pre-processing, Data Analysis and Visualization for a given dataset. **Platform: Orange, Language: Python**
- TASK 4** Apply PCA and LDA algorithm to select the appropriate features from the given data set.
Platform: Orange, Google co-lab, Anaconda navigator, Language: Python
- TASK 5** Build a model using any of the classification algorithms that can effectively analyze and classify a new sample. Calculate the classification rate, accuracy, precision, and recall for your data set.
Platform: Alteryx Google co-lab, Language: Python
- TASK 6** Implement parametric and non-parametric Regression algorithm in order to fit data points. Select appropriate dataset, visualize the output and Interpret the results.
Platform: Google co-lab, Language: Python
- TASK 7** Apply back propagation neural network on image data. The idea is to build a Artificial Neural Network model that can effectively analyze and extract features from an image.
Platform: Google co-lab Language: Python
- TASK 8** Create a perceptron with appropriate number of inputs and outputs using Medical data. Train it using fixed increment learning algorithm until no change in weights is required. Output the final weights.
Platform: Google co-lab, Language: Python
- TASK 9** Apply partitioning clustering algorithms for a given dataset. Compare the results and comment on the quality of clustering using evaluation metrics.
Platform: Google co-lab, Language: Python

TASK 10 Apply hierarchal clustering algorithms for a given dataset. Compare the results and comment on the quality of clustering using evaluation metrics.

Platform: Google co-lab, Language: Python

Part-2

Use Cases:

1. Predicting Illness
2. Recommendation System
3. Email monitoring
4. Product Classifier
5. Marketing Analysis
6. Ranking of Posts on Social Media

Total: 30 Hours

G. Learning Resources

i. Text Books:

1. Ethem Alpaydin,, “Introduction to Machine Learning”, 4th edition, MIT Press, 2020.
[Unit 1 -4]
2. Christopher M Bishop, “Pattern Recognition and Machine Learning”, Springer-Verlag New York Inc, 2018
3. Tom Mitchell, “Machine Learning: A Guide to Current Research”, Kluwer academic publisher, 2019.

ii. Reference books:

1. Richard S. Sutton and Andrew G. Barto, "Reinforcement learning: An Introduction", Second Edition, MIT Press, 2019.[Unit 5]
2. Oliver Theobald,“ Machine Learning for Absolute Beginners", Scatterplot Press, third edition, 2020
3. Beattie, Guy, “Machine Learning: Start with Machine Learning with all new tips tricks2020.Understand how machine learning can help”, 2020

iii. Online References:

Machine Learning, accessed online April 20, 2021 [online].Available:
<https://www.coursera.org/learn/machine-learning>

COURSE CODE	COURSE TITLE	L	T	P	C
10211CS304	OPERATING SYSTEMS LABORATORY	0	0	2	1

A.Preamble

The Operating System Laboratory is a course that will teach students about practicing Unix commands, Shell programming, System calls, working with process management, different CPU scheduling algorithms, Deadlock handling methods and apply Memory management techniques.

B.Prerequisite Course

10211CS102 - Data Structures

C. Corequisite Course

10211CS103 - Operating Systems

D.Course Objectives

Learners are exposed to:

- Practice on Unix commands and Shell programming.
- Illustrate System calls implementation and process management techniques.
- Demonstrate different CPU scheduling algorithms and Deadlock Handling methods
- Apply Various Memory and Disk scheduling techniques.

E.Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Develop Shell scripts& system calls	S3
CO2	Apply the process management techniques and scheduling algorithms for the given problem.	S3
CO3	Make use of Banker's algorithm to detect and avoid dead lock	S3
CO4	Implement various memory management strategies and page replacement algorithm.	S3
CO5	Implement the various disk scheduling algorithms	S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

F.Correlation of COs with Program outcomes and Programme Specific Outcomes:

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	1												
CO2	3	3	3	1								3		1
CO3	3	3	3	3								3		
CO4	3	3	3									3		
CO5	3	3	3									3		3

3- High; 2-Medium; 1-Low

G.Laboratory Experiments:**Total:30Hours****Part-1****Task 1. Practice on UNIX Commands**

- For the given task find out the correct command and produce the output
- Ex: create 100 directories named from 1 to 100

Task 2. Interpret Shell programming

- For the given problem statement, write shell program.
- Find out the appropriate shell script to get the output.
- Implement the shell program to perform I/O operations on file.

Task 3. Illustrate System calls of UNIX operating system: fork, exec, getpid, exit, wait, close.

- Implement the appropriate program to implement the given system calls.
- Files created automatically by using system call implementation.
- Use relevant system calls in the implementation.

Task 4. Interpret I/O System calls of UNIX operating system (open, read, write, etc).

- Create the file by using appropriate system call in C program
- Perform read operation in the mentioned file and print it on screen
- Read input from user and write the same to the file.
- Copy content from one file, write into another file and display the content of destination file in display.

Task 5. Demonstrate Inter-Process Communication (using shared memory, pipes or message queues).

- Create number of processes and pass information between them.
- Pass information between parent and child process using pipes.
- Use shared memory concept, perform read and write operation by the process to communicate between them.

Task 6. Use of semaphores (using UNIX system calls)

Write program to ,

- Implement producer-consumer environment using common buffer.
- Implement readers-writers problem.
- Use appropriate semaphore variable for the given problem statement to avoid synchronization problems.

Task 7. Practice on CPU Scheduling algorithms: FCFS, SJF, PRIORITY, ROUND ROBIN

For the given scenario write C program to,

- Find out the waiting time for each process.
- Find out the Turn around time of each process.

- Display the table which contains process names, burst time, arrival time, waiting time and turnaround time.
- Display average waiting time and Average turnaround time.

Task 8. Apply **Bankers Algorithm** to avoid Deadlock.

For the given scenario write C program to,

- Calculate need matrix.
- Find out sequence of execution of process.
- Print the Sequence of execution and conclude the system in safe state or not.

Task 9. Interpret Memory allocation schemes like **First fit, Best fit and Worst fit.**

Write a C program to,

- Simulate contiguous memory allocation techniques.
- Each allocation Techniques, find out the mapping of process with corresponding frames.
- Display mapping details .

Task 10. Illustrate **page Replacement algorithms: FIFO, LRU, OPTIMAL**

Write a C program to,

- Implement different Page replacement algorithms, for each Algorithm,
- Show the contents of frames for each reference of a page.
- If page hit, display “No page Fault”
- Display the number of page faults at the end.

Task 11. Apply Disk scheduling algorithms: **FCFS, SSTF, SCAN, C-SCAN**

Write a C program to simulate Disk Scheduling techniques and for each algorithm

- Display the order of servicing the request.
- Calculate the number of read / write head movements.
- Display the number of movements.

Part-2

Use Cases

Use Case 1:

Illustrate the given scenario by a program to implement

a) create a file “input.txt”

[one
Two
Three
Four
Five]

b) Copy “input.txt” to “output.txt”

c) Append to “output.txt”

[Six
Seven
Eight]

d) Delete “input.txt” file

Use Case 2:

- a) Demonstrate a program in C that creates a child process, waits for the termination of the child and lists its PID
- b) Create a file named my file.txt that contains the following four lines :

Child 1 reads this line

Child 2 reads this line

Child 3 reads this line

Child 4 reads this line

Write a C program that forks four other processes. After forking the parent process goes into wait state and waits for the children to finish their execution. Each child process reads a line from the file my file.txt (Child 1 reads line 1, child 2 reads line 2, child 3 reads line 3 and child 4 reads line 4) and each prints the respective line. The lines can be printed in any order.

Use Case 3:

Illustrate a program to coordinate the barber and the customers..A barbershop consists of a waiting room with n chairs, and the barber room containing the barber chair. If there are no customers to be served, the barber goes to sleep. If a customer enters the barbershop and all chairs are occupied, then the customer leaves the shop. If the barber is busy, but chairs are available, then the customer sits in one of the free chairs. If the barber is asleep, the customer wakes up the barber. Write a program to coordinate the barber and the customers.

Usecase 4.

A computer processor is given N tasks to perform ($1 \leq N \leq 50,000$). The i -th task requires T_i seconds of processing time ($1 \leq T_i \leq 1,000,000,000$). The processor runs the tasks as follows: each task is run in order, from 1 to N , for 1 second, and then the processor repeats this again starting from task 1. Once a task has been completed, it will not be run in later iterations. Determine, for each task, the total running time elapsed once the task has been completed.

Input

The first line of the input contains the integer N , and the next N lines contain the integers T_1 through T_N .

Output

Output N lines, the i -th of which contains an integer representing the time elapsed when task i has been processed.

Example**Input:**

5

8

1

3

3

8

Output:

22

2

11

12

23

The second task is completed during the first iteration, finishing 2 seconds in. On the third iteration, the third and fourth tasks complete at 11 seconds and 12 seconds, respectively. Finally, on the eighth iteration, the first and last tasks complete at 22 seconds and 23 seconds, respectively.

Usecase 5.

A single processor system has three resource types HDD, Printer and USB , which are shared by three processes. There are 5 units of each resource type. Consider the following scenario, where the column alloc denotes the number of units of each resource type allocated to each process, and the column request denotes the number of units of each resource type requested by a process in order to complete execution. Which of these processes will finish LAST? Develop a program to solve this.

0. P0

1. P1

2. P2

None of the above since the system is in a deadlock

	Alloc			Request		
	HDD	Printer	USB	HDD	Printer	USB
P0	1	2	1	1	0	3
P1	2	0	1	0	1	2
P2	2	2	1	1	2	0

Usecase 6.

Interpret a C program to simulate page replacement algorithms

a) FIFO b) LRU c) LFU

A system uses 3 page frames for storing process pages in main memory. Assume that all the page frames are initially empty. What is the total number of page faults that will occur while processing the page reference string given below-Apply all replacement algorithms

1,2,3,4,2,1,5,6,2,1,2,3,7,6,3,2,1,2,3,6

Also calculate the hit ratio and miss ratio.

Usecase 7:

Demonstrate a C program to simulate the following contiguous memory allocation techniques. Consider six memory partitions of size 200 KB, 400 KB, 600 KB, 500 KB, 300 KB and 250 KB. These partitions need to be allocated to four processes of sizes 357 KB, 210 KB, 468 KB and 491 KB in that order.

Perform the allocation of processes using-

1. First Fit Algorithm
2. Best Fit Algorithm
3. Worst Fit Algorithm

Usecase 8:

Implement a C program to simulate the following Disk Scheduling techniques

Input:

Suppose the order of request is- (82,170,43,140,24,16,190) print the order of visit and

Seek time and current position of Read/Write head is : 50

a) FCFS b) SSTF

H.Learning Resources

i. Text Books:

1. Abraham Silberschatz, Peter Baer Galvin and Greg Gagne, “Operating System Concepts”, 9th Edition, John Wiley and Sons Inc., 2016.

ii. Reference Books:

1. Abraham Silberschatz, Peter Baer Galvin and Greg Gagne, “Operating System Concepts”, 9th Edition, John Wiley and Sons Inc., 2016.
2. Richard Petersen, “Linux: The Complete Reference”, 6th Edition McGrawHill Education 2017.

iii. Online Resources:

1. “Learn Unix” Accessed on: May 25, 2021 [Online]. Available: <https://www.tutorialspoint.com/unix/index.htm>
2. “Linux programming and scripting” June 18,2015. Accessed on: May 25, 2021 [Online]. Available: <https://nptel.ac.in/courses/117/106/117106113/>

COURSE CODE	COURSE TITLE	L	T	P	C
10211CS305	MICROPROCESSOR AND MICROCONTROLLER LABORATORY	0	0	2	1

A. Preamble

This laboratory course motivates the students to understand and develop assembly language programming skill by learning simple arithmetic, conditional programs. It teaches the students to realize the difference between microprocessor and microcontroller. Students are also exposed about interfacing techniques which has been the pathway for developing real time applications.

B.Prerequisite Course

10211CS201 – Digital Electronics

C. Corequisite Course

10211CS109 - Microprocessor and Microcontroller

D.Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Develop programming skill using 8086 assembly language instructions.	S3
CO2	Identify the list of ARM processor related instructions and develop simple programs using it.	S3
CO3	Establish ARM processor interfacing with keyboard and display unit using embedded C programs.	S3
CO4	Realize UART data transmission between two ARM processor boards.	S3
CO5	Measure the temperature of an object using ARM processor.	S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E.Correlation of Cos with Program Outcomes and Programme Specific Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3											2	
CO2	3	3			3								3	
CO3	3	3	3		3								3	
CO4	3	3	3		3								3	
CO5	3	3	3		3								3	3

High - 3, Medium - 2, Low – 1

F. Course Content

Part – 1

Task 1: 16bit arithmetic and logical operations using 8086 instruction set.

Construct assembly language programs for addition, subtraction, multiplication and division.

Task 2: String operations using 8086 instruction set

Develop programs for string concatenation and conversion of uppercase letters into lowercase letters and vice versa.

Task 3: Smallest and largest number using 8086 instruction set

Realize a program that is able to identify the smallest and largest number from the given set of data.

Task 4: Ascending and descending order using 8086 instruction set

Write assembly language program to arrange the given number from smallest to largest and vice versa.

Task 5: Data transfer using ARM processor

Employ an assembly language program which is able to transfer data from ARM processor register to memory or memory to ARM processor.

Task 6: Count of negative numbers in an array using ARM processor instructions

Formulate a logic to identify the list of negative numbers from the given numbers and develop the program using ARM processor instruction set.

Task 7: Factorial of positive integer N using ARM processor instructions

Calculate the factorial of a given numbers using ARM processor instructions.

Task 8: GCD algorithm using ARM processor instructions

Identify the greatest common divisor of given numbers by executing program developed using ARM processor instructions.

Task 9: Seven segment LED display

Control the display of an LED through an ARM processor.

Task 10: Key board and display interfacing using ARM processor kit development board

Realize the display of the information typed using a keyboard and processed by an ARM processor.

Task 11: UART device driver for data transmission

Experiment the process of data transfer that can be established between two ARM processor boards.

Task 12: Temperature measurement using ARM processor kit development board

Construct a data acquisition mechanism of any physical parameter like temperature using an ADC which is in-built in an ARM processor.

Part-2**Use Cases****Use Case 1:**

Develop an application which counts the number of customers entering into a supermarket for purchasing using 8086 microprocessor. IR sensor shall be used to identify the customer entering into the supermarket. 8086 Processor recognizes the output of IR sensor as a digital input and a suitable ALP is able to initiate the count as and when it receives a logic '1'.

Use Case 2 :

Implement a scheme to measure the distance between two cars that helps to avoid collision due to over speed. GP2D120 sensor can be used to measure the distance between two objects. Output of the sensor is analog signal. A/D converter in an ARM CORTEX processor recognizes this output as an input to it. Suitable Keil 'C' coding implemented in the ARM processor converts the analog signal into digital and displays the result through LCD display.

Use Case 3 :

Interpret the given digital signal, develop a Keil 'C' program to implement in an ARM CORTEX processor for identifying the frequency of the signal and display its value in an LCD.

Use Case 4 :

Design a Keil ‘C’ program to generate a PWM signal based on the input value given by the user. The program must be implemented in an ARM CORTEX processor and the output shall be displayed by changing the ON / OFF condition of an LED. Timer in the ARM processor can change the width of the pulse depending the input applied to it. Frequency of the timer is the frequency of the ARM processor.

Total: 30 Hours

H. Learning Resources**i. Text Books:**

1. Douglas V. Hall, “Microprocessors and Interfacing”, TMH, Revised second edition, 2005.
2. Muhammad Tahir, Kashif Javed, “ARM Microprocessor Systems: Cortex-M Architecture, Programming, and Interfacing”, CRC Press, Taylor & Francis group, 2017.

ii. Reference Books:

1. Andrew N. Sloss, “ARM system developer’s guide – Designing and optimizing system software”, ELSEVIER, 2004.
2. Charles M. Gilmore, “Microprocessors: Principles and applications”, TMH, Second edition, 1995.
3. Jonathan W. Valvano, “Embedded microcomputer systems: Real time interfacing”, CENGAGE Learning, Third edition, 2012.
4. Steve Furber, “ARM system-on-chip architecture”, Pearson education, Second edition, 2015.

iii. Online References:

1. “Microprocessor interfacing”, Accessed on Sep 20, 2022 [Online], Available: https://onlinecourses.nptel.ac.in/noc20_ee11/preview.
2. “Microprocessors and Digital systems”, Accessed on Sep 21, 2022 [Online], Available: <https://archive.org/details/microprocessorsd0000hall/page/n3/mode/2up>.
3. “ARM processor architecture”, Accessed on Sep 21, 2022 [Online], Available: https://www.cs.ccu.edu.tw/~pahsiung/courses/ese/notes/ESD_03_ARM_Architecture.pdf.

COURSE CODE	COURSE TITLE	L	T	P	C
10211CS306	COMPETITIVE CODING - I	0	0	2	1

A.Preamble

This course provides the way to trace the code using pen and paper handy. Students are expected to understand the notions of syntax and semantics of programming languages. Also, the students are expected to carry out the tasks which in turn helps them to develop their logic and problem-solving skills using learning-by-doing approach

B.Prerequisite Courses

10210CS101 - Problem solving Using C

C.Course Objectives

Learners are exposed to:

- Understand in depth knowledge in tracing the code.
- Identify the syntax and semantic errors and to debug the program.
- Solve programming problems using basic number theory concepts
- Implement the concepts of linear and Nonlinear data structures for the real-world problem

D.Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Trace and debug the program / code segment which includes syntax and semantic errors.	S3
CO2	Analyze the functional, logic and programming paradigm.	S3
CO3	Apply number theory and mathematical concepts to solve the computational problem.	S3
CO4	Implement algorithm for real word problems using elementary data structures.	S3
CO5	Analyze the given partial code / Pseudocode and find correct solution.	S3

Knowledge Level (Based on revised Bloom's Taxonomy)

K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create

E.Correlation of Cos with Program Outcomes and Programme Specific Outcomes

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	3	3							3	3		
CO2	3	3	1	1							3	3		
CO3	3	3	3	3							3	3	3	
CO4	3	3	3	3							3	3	3	
CO5	3	3	3	1							3	3		

3- High; 2-Medium; 1-Low

F.Course Contents

The course describes how programs work and how data is stored- Point out the Syntax and Semantic errors - Conceptual Questions in Programming- Solve programming problems implementing loops- Basic Number Theory 1, Basic Number Theory 2- Primality Tests- Data structure algorithms and finding complexity- Code Optimization -Partial Code Completion -Techniques for readability and ease of maintenance. -Concepts behind compiling, linking, OS, Hardware-Practice Problems for Placement Preparation

The above contents are covered under the following tasks:

Task 1: Point out the Syntax and Semantic errors in the code snippet, and debug

[Ref: T1 – Pg.no: 18,32,33,141-143,159-161,191-192,248,320-321,401,402, Appendix C]

Task 2: Conceptual Questions in Programming

[Ref: T1 – Pg.no:34,35,140,141,171,172,190,191,211,212,247-248,269-271,319,320,400,401]

Task 3: Solve programming problems implementing necessary loops

[Ref: T1 – Pg.no:48-53,66-72,84-86,102-106,120-124]

Task 4: Basic Number Theory 1

To solve programming problems which includes the following topics:

1. **Modular arithmetic:** Properties with example
2. **Modular exponentiation**
 - a. Basic method and its time complexity with example
 - b. Optimized method and its time complexity with example

Task 5: Basic Number Theory 1

To solve programming problems which includes the following topics:

1. **Greatest Common Divisor (GCD)**
 - a. Naïve Approach and its time complexity with example
 - b. Euclid Algorithm and its time complexity with example
2. **Extended Euclidean algorithm:** Basic method and its time complexity
3. **Modular multiplicative inverse**
 - a. Naïve Approach and its time complexity with example
 - b. Extended Euclidean algorithm
 - c. Fermat's Little Theorem and its time complexity with example

Task 6: Basic Number Theory 2

To solve programming problems which includes the concept of Prime numbers and its related properties.

- a. **Naive Approach** and its time complexity with example
- b. **Sieve of Eratosthenes** and its time complexity with example

Task 7: Primality Tests

To solve programming problems which includes the concept of Prime numbers and its related properties.

- a. **Sieve of Eratosthenes** with example
- b. **Fermat's Primality Testing** with example
- c. **Miller-Rabin Primality Testing** with example

Task 8: Arrays -Introduction, memory allocation (with row/column major), operations (insert, delete, search) sorted and unsorted array, suffix array, subsequence and subarray.

Practice Problems:

- i) Two Sum :- <https://leetcode.com/problems/two-sum/>
- ii) Rotate Array :- <https://leetcode.com/problems/rotate-array/>
- iii) Maximum Subarray :- <https://leetcode.com/problems/maximum-subarray/>
- iv) Best Time to Buy and Sell Stock :- <https://leetcode.com/problems/best-time-to-buy-and-sell-stock/>
- v) Container with most water:- <https://leetcode.com/problems/container-with-most-water/>

Additional Problems:

- i) Contiguous array:- <https://leetcode.com/problems/contiguous-array/>
- ii) Maximum product subarray:- <https://leetcode.com/problems/maximum-product-subarray/>
- iii) Find the Duplicate Number:- <https://leetcode.com/problems/find-the-duplicate-number/>
- iv) First Missing Positive:- <https://leetcode.com/problems/first-missing-positive/>

Task 9: String Processing- Basics, String functions (based on different language), String operations, Two algorithms for string pattern matching - Naïve approach and Robin karp approach

Practice Problems:

- i) Reverse words in a string:- <https://leetcode.com/problems/reverse-words-in-a-string/>
- ii) Valid Palindrome:- <https://leetcode.com/problems/valid-palindrome-ii/>
- iii) Longest Palindromic Substring:- <https://leetcode.com/problems/longest-palindromic-substring/>
- iv) Palindromic Substrings:- <https://leetcode.com/problems/palindromic-substrings/>
- v) Subsequence Matching:- <https://practice.geeksforgeeks.org/problems/subsequence-matching/0>

Additional Problems:

- i) Reverse String:- <https://leetcode.com/problems/reverse-string/>
- ii) Reverse words in a string-III:- <https://leetcode.com/problems/reverse-words-in-a-string-iii/>
- iii) Roman to Integer:- <https://leetcode.com/problems/roman-to-integer/>
- iv) Good Substrings:- <http://codeforces.com/problemset/problem/271/D>
- v) Pattern Finding:- <https://www.spoj.com/problems/NAJPF/>

Task 10: Bit Map-Introduction, XOR, AND, OR, right shift, left shift.

Practice Problems:

- i) Reverse Bits: <https://leetcode.com/problems/reverse-bits/>
- ii) Number of 1 bits: <https://leetcode.com/problems/number-of-1-bits/>
- iii) Hamming Distance: <https://leetcode.com/problems/hamming-distance/>
- iv) Single Number: <https://leetcode.com/problems/single-number/>
- v) Single number II: <https://leetcode.com/problems/single-number-ii/>

Additional Problems:

- i) Total Hamming Distance: <https://leetcode.com/problems/total-hamming-distance/>
- ii) Counting Bits: <https://leetcode.com/problems/counting-bits/>
- iii) Single Number III: <https://leetcode.com/problems/single-number-iii/>

Task 11: Recursion and its concepts**Practice Problems:**

- i) Fibonacci Number: <https://leetcode.com/problems/fibonacci-number/>
- ii) Nth Tribonacci number: <https://leetcode.com/problems/n-th-tribonacci-number/>
- iii) K equal sum subsets: <https://leetcode.com/problems/partition-to-k-equal-sum-subsets/>
- iv) Permutations of string: <https://practice.geeksforgeeks.org/problems/permutations-of-a-given-string/0>
- v) Counting Bits: <https://leetcode.com/problems/counting-bits/>

Additional Problems:

- i) Total Hamming Distance: <https://leetcode.com/problems/total-hamming-distance/>
- ii) Single number II: <https://leetcode.com/problems/single-number-ii/>
- iii) Single Number III: <https://leetcode.com/problems/single-number-iii/>

Task 12: Elementary data structure algorithms-based Questions and finding complexity

S.No.	Data Structure Algorithms	Reference no- T2
12.1	Linked List	Problem no:2,3,11, 14,15,16,17,23,24,29
12.2	Stack	Problem no: 2, 3, 22, 23, 24, 25
12.3	Queue	Problem no: 1 to 6
12.4	Binary Tree	Problem no: 8, 10, 17, 18, 24, 27, 28
12.5	Binary Search Tree	Problem no: 49, 50, 51, 52, 56, 59
12.6	Sorting	Page no :246-255 Performance analysis and Comparison
12.7	Searching	Problem no: 1, 2, 36, 58, 59, 64, 67, 69

Task 13: Partial Code Completion [Ref: Online references]**Task 14: Practice Problems for Coding Preparation [Ref: T1 – Pgno:406-424]****Total: 30 Hours**

G.Learning Resources

i. Text Books:

1. Yeshwant P. Kanetkar,” Let us C: Authentic Guide to C Programming Language”,17th Edition, 2020, BPB Publications.
2. Narasimha Karumanchi,” Data Structures and Algorithms made Easy”5th Edition, Career Monk Publication, Print Replica, 2020.

ii. Reference Books:

1. Marty Erickson, Anthony Vazzana, David Garth,”Introduction to Number Theory”, Chapman and Hall, CRC; Copyright year 2016.
2. Thomas H.Cormen,” Introduction to Algorithms”,3rd Edition,2009, MIT Press Cambridge.

iii. Online References:

1. “Number Theory”, Accessed on Apr.20, 2021 [Online]. Available: <https://www.hackerearth.com/practice/math/number-theory/>
2. “All Problem sets from Leet code”, Accessed on Apr.20, 2021 [Online]. Available: <https://leetcode.com/problemset/all/>

COURSE CODE	COURSE TITLE	L	T	P	C
10211CS307	COMPETITIVE CODING -II	0	0	2	1

A.Preamble

This course is recommended for students hoping to learn how to solve difficult problems that appear in multi-stage programming contests. The students will learn to design time and space efficient algorithms to solve challenging contest problems, and produce bug-free code under the pressure of time in contest. Students will learn to handle corner cases, invalid inputs, recursion conditions, memory leaks and test cases effectively. Learning by doing true end goal is to produce all-rounder computer scientists/programmers who are much readier to produce better software and to face harder CS research problems in the future.

B.Prerequisite Course

10211CS306 – Competitive Coding -I

C.Course Objectives

Learners are exposed to:

- Understand the basic concepts for solving real world problems
- Impart and implement the concepts of data structures
- Identify the basic properties of Trees and Graphs.
- Hands on Experience in analysis of algorithms for various algorithmic techniques
- Implement algorithm design methods such as dynamic programming and greedy method

D.Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Apply advanced Array, String and mathematical algorithms for problem solving	S3
CO2	Demonstrate stack, queue and sorting algorithms to solve the problems.	S3
CO3	Implement classical algorithms for trees and graphs with backtracking concepts, being aware of their trade-offs in terms of complexity in time/space	S3
CO4	Analyse and implement efficient algorithms for challenging computational problems on advanced data structures, in practice, quickly and correctly.	S3
CO5	Solve complex problems using dynamic programming and greedy algorithm.	S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E.Correlation of Cos with Program Outcomes and Programme Specific Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	3	3							3	3	2	
CO2	3	3	3	3							3	3	2	
CO3	3	3	3	3							3	3	2	
CO4	3	3	3	1							3	3	2	
CO5	3	3	3	3							3	3	3	

High - 3, Medium - 2, Low - 1

F.Course Contents

This course introduces advanced algorithms and data structures concepts useful for competing effectively in programming contests. This course provides familiarity with and proficiency in solving intermediate-difficulty algorithmic programming problems using dynamic programming, graph algorithms, mathematics, computational geometry, combinatorial games, and standard library data structures. It includes connectivity-based dynamic programming, multi-dimensional computational geometry, advanced string algorithms, advanced data structures, advanced graph and tree algorithms, linear programming, and satisfiability.

Task 1: Mathematics-Why math in coding? Greatest Common Divisor, Lowest Common Multiple, Euclidean Algorithm, Extended Euclidean Algorithm, Prime Numbers, divisibility of numbers and Sieve of Eratosthenes and Segmented Sieve, Modulo arithmetic, Modulo exponentiation and Modulo inverse, Math Probability, Counting, Game Theory, Group Theory, Generating functions, Permutation Cycles.

Practice Problems:

- i) Sqrt(x): <https://leetcode.com/problems/sqrtx/>
- ii) Trapping rain water: <https://leetcode.com/problems/trapping-rain-water-ii/>
- iii) Permutation sequence: <https://leetcode.com/problems/permutation-sequence/>
- iv) Ugly Number: <https://leetcode.com/problems/ugly-number-ii/>
- v) Allocate Mailboxes: <https://leetcode.com/problems/allocate-mailboxes/>

Additional Problems:

- i) Sum of square numbers: <https://leetcode.com/problems/sum-of-square-numbers/>
- ii) Next permutation: <https://leetcode.com/problems/next-permutation/>
- iii) Product of array: <https://leetcode.com/problems/product-of-array-except-self/>
- iv) Trapping rain water: <https://leetcode.com/problems/trapping-rain-water/>
- v) Find the median problem: <https://www.hackerrank.com/challenges/find-the-median/problem>
- vi) Straight Line: <https://leetcode.com/problems/check-if-it-is-a-straight-line/>
- vii) Number of digit one: <https://leetcode.com/problems/number-of-digit-one/>
- viii) Largest component size by common factor: <https://leetcode.com/problems/largest-component-size-by-common-factor/>
- ix) Reaching Points: <https://leetcode.com/problems/reaching-points/>
- x) Beautiful Numbers: <https://codeforces.com/problemset/problem/300/C>
- xi) Number Expression: <https://practice.geeksforgeeks.org/problems/check-if-a-number-can-be-expressed-as-xy1606/1>
- xii) Pairs of Prime Numbers: <https://practice.geeksforgeeks.org/problems/pairs-of-prime-number/0>
- xiii) Fibonacci number: <https://practice.geeksforgeeks.org/problems/check-if-the-number-is-fibonacci4654/1/>
- xiv) Maximum product of three numbers: <https://leetcode.com/problems/maximum-product-of-three-numbers/>
- xv) Count all valid options: <https://leetcode.com/problems/count-all-valid-pickup-and-delivery-options/>
- xvi) Largest divisible subset: <https://leetcode.com/problems/largest-divisible-subset/>

Task 2: Advanced concept on Arrays – Two Pointer problem One maintains two pointers (indices) into either one array or two arrays, and move the pointers based on values at the pointers.

Practice Problems:

- i) Interval list intersections: <https://leetcode.com/problems/interval-list-intersections/>
- ii) Merge Intervals: <https://leetcode.com/problems/merge-intervals/>
- iii) Merge sorted array: <https://leetcode.com/problems/merge-sorted-array/>
- iv) Three sum: <https://leetcode.com/problems/3sum/>
- v) Smallest range covering element: <https://leetcode.com/problems/smallest-range-covering-elements-from-k-lists/>

Additional Problems:

- i) Longest substring without repeating characters: <https://leetcode.com/problems/longest-substring-without-repeating-characters/>
- ii) Push dominoes: <https://leetcode.com/problems/push-dominoes/>

Task 3: Advanced concept on String – String searching, Knuth–Morris–Pratt algorithm, Boyer–Moore string-search algorithm, Manacher's algorithm and z-function, suffix and prefix.

Practice Problems:

- i) Longest Prefix Suffix: <https://practice.geeksforgeeks.org/problems/longest-prefix-suffix2527/1>
- ii) Palindrome substring: <https://codeforces.com/gym/101806/problem/Q>
- iii) Password problem: <https://codeforces.com/problemset/problem/126/B>
- iv) Prefixes and suffixes: <https://codeforces.com/problemset/problem/432/D>
- v) Pattern find: <https://www.spoj.com/problems/NAJPF/>

Additional Problems:

- i) Palindrome: https://onlinejudge.org/index.php?option=com_onlinejudge&Itemid=8&category=26&page=show_problem&problem=2470
- ii) Chef and String: <https://www.codechef.com/problems/CHSTR>
- iii) Anthem of Berland: <https://codeforces.com/contest/808/problem/G>

Task 4: Stacks and Queues

Practice Problems:

- i) Sort characters by frequency: <https://leetcode.com/problems/sort-characters-by-frequency/>
- ii) Valid Anagram: <https://leetcode.com/problems/valid-anagram/>
- iii) Stack using Two queue : <https://practice.geeksforgeeks.org/problems/stack-using-two-queues/1>
- iv) Bag of numbers : <https://www.hackerearth.com/practice/data-structures/stacks/basics-of-stacks/practice-problems/algorithm/bag-of-numbers/>
- v) Disk tower : <https://www.hackerearth.com/practice/data-structures/queues/basics-of-queues/practice-problems/algorithm/disk-tower-b7cc7a50/>

Additional Problems:

- i) Queries with fixed length problem: <https://www.hackerrank.com/challenges/queries-with-fixed-length/problem>

- ii) Eerie Planet : <https://www.hackerearth.com/practice/data-structures/queues/basics-of-queues/practice-problems/algorithm/weird-planet-2000a170/>
- iii) Number of atoms: <https://leetcode.com/problems/number-of-atoms/>

Task 5: Linked List-Single linked list, Doubly linked list, operations on linked list.

Practice Problems:

- i) Add two Numbers: <https://leetcode.com/problems/add-two-numbers/>
- ii) Remove nth Node: <https://leetcode.com/problems/remove-nth-node-from-end-of-list/>
- iii) Merge two sorted list: <https://leetcode.com/problems/merge-two-sorted-lists/>
- iv) Insertion sort list: <https://leetcode.com/problems/insertion-sort-list/>
- v) Remove element: <https://leetcode.com/problems/remove-linked-list-elements/>

Additional Problems:

- i) Sort List: <https://leetcode.com/problems/sort-list/>
- ii) Reverse nodes in k groups: <https://leetcode.com/problems/reverse-nodes-in-k-group/>
- iii) Remove duplicates: <https://leetcode.com/problems/remove-duplicates-from-sorted-list-ii/>
- iv) Linked List cycle ii: <https://leetcode.com/problems/linked-list-cycle-ii/>
- v) Odd Even linked list: <https://leetcode.com/problems/odd-even-linked-list/>
- vi) Add two numbers II: <https://leetcode.com/problems/add-two-numbers-ii/>
- vii) Swap nodes in pairs: <https://leetcode.com/problems/swap-nodes-in-pairs/>

Task 6: Sorting- Counting sort, Radix sort, Heap sort, Bucket sort

Practice Problems:

- i) Firstnonrepeating character in a stream: <https://practice.geeksforgeeks.org/problems/first-non-repeating-character-in-a-stream/0/>
- ii) LRU Cache: <https://leetcode.com/problems/lru-cache/>
- iii) Finding pairs: <https://www.hackerearth.com/practice/algorithms/sorting/counting-sort/practice-problems/algorithm/finding-pairs-4/>
- iv) Cube change : <https://www.hackerearth.com/practice/algorithms/sorting/heap-sort/practice-problems/algorithm/cube-change-qualifier2/>
- v) DescendingWeights : <https://www.hackerearth.com/practice/algorithms/sorting/bucket-sort/practice-problems/algorithm/sort-the-array-5/>

Additional Problems:

- i) Binary Land: <https://www.codechef.com/MAY20A/problems/BINLAND>
- ii) Divide Apples : <https://www.hackerearth.com/practice/algorithms/sorting/heap-sort/practice-problems/algorithm/divide-apples/>
- iii) Shil and birthday present : <https://www.hackerearth.com/practice/algorithms/sorting/counting-sort/practice-problems/algorithm/shil-and-birthday-present/>

Task 7: Trees: Introduction, tree traversal (in-order, pre-order, post-order), Binary search tree

Practice Problems:

- i) Longest Uni value Path: <https://leetcode.com/problems/longest-univalue-path/>
- ii) Path Sum-III: <https://leetcode.com/problems/path-sum-iii/>

- iii) Distinct count : <https://www.hackerearth.com/practice/data-structures/trees/binary-search-tree/practice-problems/algorithm/distinct-count/>
- iv) MST revisited : <https://www.hackerearth.com/practice/data-structures/trees/binary-search-tree/practice-problems/algorithm/mst-revisited-3f9d614c/>
- v) Vertical order:<https://leetcode.com/problems/vertical-order-traversal-of-a-binary-tree/>

Additional Problems:

- i) Level order traversal:<https://leetcode.com/problems/binary-tree-zigzag-level-order-traversal/>
- ii) Level order traversal -ii:<https://leetcode.com/problems/binary-tree-level-order-traversal-ii/>
- iii) Preorder and inorder:<https://leetcode.com/problems/construct-binary-tree-from-preorder-and-inorder-traversal/>

Task 8: Graph - Introduction Adjacency Matrix and List, Depth First search, Breadth First Search, Manacher's algorithm

Practice Problems:

- i) Decode string:<https://leetcode.com/problems/decode-string/>
- ii) Cheapest Flights:<https://leetcode.com/problems/cheapest-flights-within-k-stops/>
- iii) Network delay time:<https://leetcode.com/problems/network-delay-time/>
- iv) Keys and rooms:<https://leetcode.com/problems/keys-and-rooms/>
- v) Minimum cost to connect:<https://leetcode.com/problems/minimum-cost-to-connect-two-groups-of-points/>

Additional Problems:

- i) Clone Graph:<https://leetcode.com/problems/clone-graph>
- ii) Find the town judge:<https://leetcode.com/problems/find-the-town-judge>
- iii) Bipartite:<https://leetcode.com/problems/is-graph-bipartite/>
- iv) Path with minimum effort:<https://leetcode.com/problems/path-with-minimum-effort/>

Task 9: Backtracking-Introduction, Backtrack vs Divide and Conquer

Practice Problems:

- i) Letter combination:<https://leetcode.com/problems/letter-combinations-of-a-phone-number/>
- ii) Regular expression matching:<https://leetcode.com/problems/regular-expression-matching/>
- iii) Generate parentheses:<https://leetcode.com/problems/generate-parentheses/>
- iv) Combination sum:<https://leetcode.com/problems/combination-sum/>
- v) Sequential digits:<https://leetcode.com/problems/sequential-digits/>

Additional Problems:

- i) Permutations:<https://leetcode.com/problems/permutations/>
- ii) Subsets:<https://leetcode.com/problems/subsets/>
- iii) Path with maximum gold:<https://leetcode.com/problems/path-with-maximum-gold/>

Task 10: Heap Introduction, min-max heap, Segment tree & Trie

Practice Problems:

- i) Find k pairs with smallest sums:<https://leetcode.com/problems/find-k-pairs-with-smallest-sums/>
- ii) Last stone weight:<https://leetcode.com/problems/last-stone-weight/>
- iii) Kth largest element in an array:<https://leetcode.com/problems/kth-largest-element-in-an-array/>
- iv) K-closest points to origin:<https://leetcode.com/problems/k-closest-points-to-origin/>
- v) Sliding window maximum:<https://leetcode.com/problems/sliding-window-maximum/>
- vi) Range sum Query:<https://leetcode.com/problems/range-sum-query-mutable/>
- vii) Maximum number of events:<https://leetcode.com/problems/maximum-number-of-events-that-can-be-attended/>
- viii) Minimum number of increments:<https://leetcode.com/problems/minimum-number-of-increments-on-subarrays-to-form-a-target-array/>
- ix) Cost of Data :<https://www.hackerearth.com/practice/data-structures/advanced-data-structures/trie-keyword-tree/practice-problems/algorithm/cost-of-data-11/> Random Generator : <https://www.hackerearth.com/practice/data-structures/advanced-data-structures/trie-keyword-tree/practice-problems/algorithm/dexters-random-generator/>

Additional Problems:

- i) Skyline problem:<https://leetcode.com/problems/the-skyline-problem/>
- ii) Count of smaller numbers:<https://leetcode.com/problems/count-of-smaller-numbers-after-self/>

Task 11: Dynamic programming - Introduction, Tabulation and memorization algorithms

Practice Problems:

- i) Maximum sub array:<https://leetcode.com/problems/maximum-subarray/>
- ii) Partition equal subset:<https://leetcode.com/problems/partition-equal-subset-sum/>
- iii) Best time to buy and sell stock:<https://leetcode.com/problems/best-time-to-buy-and-sell-stock/>
- iv) House robber:<https://leetcode.com/problems/house-robber/>
- v) Minimum cost for tickets:<https://leetcode.com/problems/minimum-cost-for-tickets/>

Additional Problems:

- i) Unique Paths:<https://leetcode.com/problems/unique-paths/>
- ii) Minimum path sum:<https://leetcode.com/problems/minimum-path-sum/>
- iii) Unique binary search:<https://leetcode.com/problems/unique-binary-search-trees/>
- iv) Unique paths-ii:<https://leetcode.com/problems/unique-paths-ii/>
- v) Longest increasing subsequence:<https://leetcode.com/problems/longest-increasing-subsequence/>
- vi) Integer Break:<https://leetcode.com/problems/integer-break/>
- vii) Longest palindromic substring:<https://leetcode.com/problems/longest-palindromic-substring/>

Task 12: Greedy algorithm -Introduction, algorithms based on greedy

Practice Problems:

Lemonade change:<https://leetcode.com/problems/lemonade-change/>

- i) Water bottles:<https://leetcode.com/problems/water-bottles/>
- ii) Best time to buy and sell stock:<https://leetcode.com/problems/best-time-to-buy-and-sell-stock-with-transaction-fee/>
- iii) Couples holding hands:<https://leetcode.com/problems/couples-holding-hands/>

iv) Wildcard matching:<https://leetcode.com/problems/wildcard-matching/>

Additional Problems:

- i) Partition labels:<https://leetcode.com/problems/partition-labels/>
- ii) Queue reconstruction:<https://leetcode.com/problems/queue-reconstruction-by-height/>
- iii) Reorganize string:<https://leetcode.com/problems/reorganize-string/>
- iv) Jump game II:<https://leetcode.com/problems/jump-game-ii/>
- v) Minimum number of taps:<https://leetcode.com/problems/minimum-number-of-taps-to-open-to-water-a-garden/>

Total: 30 Hours

F.Learning Resources

i. Text Books:

1. Antti Laaksonen,” Guide to Competitive Programming: Learning and Improving Algorithms Through Contests”, 2nd ed. 2020 Edition,UTiCS, Springer.
2. Antti Laaksonen,” Competitive Programmer's Handbook”, 3rd ed. 2020 Edition, , Springer.
3. Steven Skiena, “The Algorithm Design Manual” 3rd Edition,2020 Springer
4. Steven Halim and Felix Halim, “Competitive Programming, 3rd Edition”, Springer

ii. Reference Books:

1. Johan Sannemo “Principles of Algorithmic Problem Solving”, 2nd edition, 2018.
2. Thomas H.Cormen,”Introduction to Algorithms”,3rd Edition,2009, MIT Press Cambridge.
3. Gayle Laakmann McDowell “Cracking the Coding Interview”3rd Ed, career cup publisher, 2015.

iii. Online Resources:

1. “Number Theory”, Accessed on Apr.20, 2021 [Online]. Available: <https://www.hackerearth.com/practice/math/number-theory/>
2. “All Problem sets from Leet code”, Accessed on Apr.20, 2021 [Online]. Available: <https://leetcode.com/problemset/all/>

PROGRAM ELECTIVE

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS110	OPTIMIZATION TECHNIQUES	3	0	0	3

A.Preamble

This course offers a holistic and hands-on introduction to the fundamentals of mathematical optimization for machine learning and deep learning. The course also explores about the fundamental optimization algorithms for popular machine learning / deep learning models including: least squares linear and nonlinear regression, logistic regression, support vector machines, feedforward neural networks, convolutional networks, gradient boosting, K-Means, Principal Component Analysis, and Recommender Systems.

B.Pre-requisite course : 10210MA103 - Probability, Statistics and Queuing Theory

C.Course Objectives:

Learners are exposed to:

- Mathematically characterize optimal solutions for nonlinear optimization models
- Know the fundamental algorithms for unconstrained optimization
- Know how to specialize these algorithms for data science problems

D.Course Outcomes:

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Develop optimization thinking and problem solving methods for solving different types of problems.	K3
CO2	Examine the optimization principle and its application in formulation of engineering optimization problems.	K3
CO3	Optimize the mathematical problems using nature based algorithms	K3
CO4	Solve various constrained and unconstrained problems in Single variable as well as multivariable.	K3
CO5	Apply the modern optimization methods for real world problems.	K3

Knowledge Level (Based on revised Bloom's Taxonomy)

K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create

E.Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO 1	PSO 2
CO1	3	3	2	2	2		2			2		3	
CO2	3	3	3	3	3		3					3	
CO3	3	3	3	2	2		2			2		3	
CO4	3	3	3		2								
CO5	3	3	3	3	3		3			3		3	2

High- 3; Medium-2; Low-1

F.Course contents

Unit 1 Mathematical Optimization for ML and DL**L-9 hours**

Supervised learning and unconstrained optimization: deep nets, trees, and kernels; The tools in the warchest: first and second order methods; Deep learning and engineering-focused first order methods; Dimension reduction, clustering, and constrained optimization Dealing with discontinuities via heuristic methods.

Unit 2 Fundamental mathematical and computational tools**L-9 hours**

Review of Linear Algebra, probability and statistics; Calculus of multivariate functions, Taylor series and local function approximation; Optimality conditions for unconstrained / constrained optimization; Numerical methods for computing derivatives, automatic differentiation, Numerical computation of eigen values / singular values.

Unit 3 Unconstrained optimization via local search**L-9 hours**

Local search as a general optimization paradigm; Random local search and the curse of dimensionality; Step length rules and intuition; Formal principles of random local search; Gradient descent - normalized and unnormalized versions; Step size selection and convergence; Conservative theoretical guarantees and line search; Steepest descent - norm generalization; Batch / Stochastic gradient descent; Variations on normalized stochastic gradient descent.

Unit 4 Second order methods**L-9 hours**

Newton's method basics - normalized and unnormalized versions; Newton's method stepsize, and backtracking linesearch - Honest adjustments for non-convex functions; Scaling issues with Newton's method; Newton and secant method as zero-finding algorithm; Quasi-Newton Methods.

Unit 5 Modern Methods of Optimization**L-9 hours**

Genetic Algorithms - Representation of Design Variables- Representation of Objective Function and Constraints- Genetic Operators; Simulated Annealing- Procedure- Algorithm- Features of the Method; Particle Swarm Optimization- Computational Implementation of PSO- Improvement in PSO- PSO Constrained Optimization Problem; Ant Colony Optimization- Ant Searching Behavior- Path Retracing and Pheromone Updating- Pheromone Trail Evaporation.

Total: 60 Hours**G.Learning Resources****i. Text Books:**

1. Watt, Jeremy, Reza Borhani, and Aggelos K. Katsaggelos. Machine learning refined: Foundations, algorithms, and applications. Cambridge University Press, 2020.
2. Sra, Suvrit, Sebastian Nowozin, and Stephen J. Wright, eds. Optimization for machine learning. Mit Press, 2012. (ISBN: 9780262016469)

ii. ReferenceBooks

1. Battiti, Roberto, and Mauro Brunato. "The LION way." Machine Learning plus Intelligent Optimization. LIONlab, University of Trento, Italy 94 (2014).
2. S. Bubeck, Convex Optimization: Algorithms and Complexity, Foundations and Trends in MachineLearning, 8(3-4): 231-357, 2015.
3. T. Hastie, R. Tibshirani and M. J. Wainwright, Statistical Learning with Sparsity: the Lasso andGeneralizations, Chapman and Hall/CRC Press, 2015.

iii. OnlineResource

1. <http://web.stanford.edu/~hastie/StatLearnSparsity/>
2. <http://simons.berkeley.edu/talks/peter-richtarik-2013-10-23>

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS112	Cybersecurity Policy, Law and Ethics	3	0	0	3

A. Preamble

This course will enable the students to operate effectively in the cybersecurity governance arena, starting from introduction to cybersecurity by implementing frameworks and developing policy responses to issues that arise with respect to the cyber laws and regulations. The course will give deep insight about the changing nature of cybersecurity threats and the requirement for managers to maintain an up-to-date tool kit in relation to cyber law and policy. The significance of ethics in various cybersecurity applications are catered in this course.

B. Prerequisite Course

10211CS130-Fundamentals of Computer Networks

C. Course Objectives

Students are able to

- Understand the basic concepts and need for cybersecurity and security policies.
- Learn about Indian and international cyber laws and their role in handling cybercrimes.
- Identify key cybersecurity regulations, standards, and privacy laws used in the industry.
- Analyze different types of cybercrimes and explore how legal systems respond to them.
- Apply ethical principles to digital activities, focusing on responsible behavior in cyberspace.

D. Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K – Level/ S - Level
CO1	Interpret the need of Cybersecurity policy and its operation	K2
CO2	Classify the cyber laws and the restrictions in cybersecurity	K2
CO3	Understand the Cybersecurity regulations and risk management	K2
CO4	Infer the vulnerabilities and threats in cybersecurity and exploitation	K2
CO5	Summarize the concept of cyber ethics, and the accountability standards	K2
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of COs with Program outcomes and Programme Specific Outcomes:

Cos	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PSO 1	PSO 2	PSO 3
CO 1	2		2									-	-	-
CO 2			2									-	-	-
CO 3		2	3			2						2	-	-

CO 4	2	2										-	-	2
CO 5						2		3				-	-	2

3- High; 2-Medium; 1-Low

F.Course Contents

Unit 1 Introduction to Cybersecurity

L-9 Hours

Essential Terminologies: The hierarchy of standards and policies – Need of Security Policies – Firewall Policy - Host Integrity policy - Service level agreement – Risk based approach – Information Gathering - Collection limitation principles – Transfer and accountability principles - Social Engineering. Case study: Scan a network and determine whether the remote host has a firewall enabled.

Unit 2 Cyber Laws

L-9 Hours

Introduction to Cyber Law - An Overview of Cyber law in India - Cybercrime and Legal perspectives - Need of Improvement in Indian Cyber Law - International perspective on Cyber Laws – Investigations on Cyber laws – Phases of digital crime investigation - Governance of cybercrime and the internet - Corporate policy- Policy framework – Intellectual property issues – Legal aspects of Digital contracts. Case study: Reporting of network breach in sensitive domains such as banks and their impact in cyber laws

Unit 3 Cyber Security Regulations

L-9 Hours

Introduction - Roles of International Law - the state and Private Sector in Cyberspace - Cyber Security Standards - National Cyber Security Policy - Continuous improvement principles - Privacy Acts-ISO/IEC 27,000 – Systrust – Webtrust – Statement on auditing standards 70 – Sabanes oxley Act – HIPAA - Information Technology Act -Information Security Governance and Risk Management – NIST: Standard reference data – National Vulnerability database
Case Study on implementation of secure architecture for financial services adhering to the regulations

Unit 4 Cybercrimes and laws

L- 9 Hours

Introduction – Cyberattack - An Important Commonality for Exploitation and Attack - Inherent Vulnerabilities of Information Technology –Cyber Lawyers – Cyber law advantages – Two sides of cyber law – Computer crime and insecurity – IT amendment act 2008 – Role of Computer emergency response team (CERT) – Digital products and federal policy – Intellectual property rights considerations – Bayh-Dole Act – Patents and Trade secrets on digital products.

Unit 5 CyberEthics

L- 9 Hours

Introduction to dimensions of cyberethics in cybersociety – Significance of Cyberethics- Corevalues and virtues – Cyberethics by norms, laws and relations – Artificial Intelligence Ethics: AI for good – Ethics in information society: Nine P's- CyberBullying – Limiting access of technology – Accountability and Liability. Case study: Analyse and judge the cybercrimes in e-commerce and social media.

Total: 45 Hours

H.Learning Resources

i. Text Books:

1. Jeff Kosseff, “Cybersecurity Law”, ISBN: 978-1-119-23150-9, Wiley Publications, 2017.
(Unit II, III, IV)

2. David Clark, Thomas Berson and Herbert, “At the Nexus of Cybersecurity and public policy, Basic concepts and issues”, National academic press, ISBN 13: 978-0-309-30318-7, 2014. (Unit I, IV)
3. Christopher Stuckelbergerc, Pavan Duggal, “Cyber Ethics 4.0, Serving humanities with values”, Globalethics publications 4th edition, ISBN 978-2-88931-265-8 2018 (Unit V)

ii. Reference Book(s):

1. Joakim Kavrestad, “Fundamentals of digital forensics, theory, methods, and real life applications”, Second Edition, ISBN 978-3-030-38954-3, Springer, 2020

iii. Web Resource(s) :

1. “Cybersecurity”. Accessed on: June 20, 2021 [Online]. Available: <https://www.nap.edu/catalog/18749/at-the-nexus-of-cybersecurity-and-public-policy-some-basic>.
2. “ISO 27000 standards”. Accessed on: June 15, 2021 [Online]. Available: <https://www.iso.org/standards.html>.
3. “Cyber Conflicts”. Accessed on: June 20, 2021 [Online]. Available: <https://www.coursera.org/lecture/cyber-conflicts/introduction-to-cybercrime-and-fundamental-issues-xndSq>.

“Cyber Laws”. Accessed on: July 2, 2021 [Online]. Available: <https://www.cyberlawsindia.net/2sides.html>

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS113	REINFORCEMENT LEARNING	3	0	0	3

A. Preamble

This course provides the overview of reinforcement learning paradigm that aims to model the trial-and-error learning process. It is a core area of research in artificial intelligence and machine learning, and today provides one of the most powerful approaches to solving Decision problems

B. Prerequisite Course

10210MA103- Probability, Statistics and Queuing theory

C. Course Objectives

Learners are exposed to:

- Knowledge of basic and advanced reinforcement learning techniques.
- Identification of suitable learning tasks to which these learning techniques can be applied.
- Appreciation of some of the current limitations of reinforcement learning techniques.

D. Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Understand the fundamental knowledge on reinforcement learning	K2
CO2	Interpret the tabular solution methods for reinforcement learning	K2
CO3	Examine the policy gradient methods to provide the solution for real use cases	K2
CO4	Identify e the model-based learning approaches in RL environment	K2
CO5	Summarize the prototype and implement RL algorithm for real world application use cases	K2
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of COs with Program outcomes and Programme Specific Outcomes:

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	3									2	2	
CO2	3	3	3	2								2	2	
CO3	3	3	3	2								2	2	
CO4	3	3	3	2								2	2	
CO5	3	3	3	2								2	2	

3-High; 2-Medium; 1-Low

E. Course Contents

Unit 1 Introduction to Reinforcement Learning

L-9Hours

Introduction and Basics of RL- Defining RL Framework- Elements and History of RL- Markov Decision Process- Policies, Value Functions and Bellman Equations- Exploration vs. Exploitation

Unit 2 Tabular Solution Methods**L-9Hours**

Dynamic Programming- Policy Evaluation- Policy Improvement-Policy Iteration- Value Iteration- Generalized Policy Iteration- Monte Carlo- Predictions- Control- Incremental implementation- Off policy- Temporal-Difference learning methods -TD(0)- SARSA- Deep Q-networks – DQN- DDQN- Dueling DQN- Prioritized Experience Replay

Unit 3 Policy Gradient Methods**L-9 Hours**

Introduction to policy-based methods- Vanilla Policy Gradient- REINFORCE algorithm and stochastic policy search- Actor-critic methods - A2C- A3C- Advanced policy gradient-PPO-TRPO-DDPG- Function Approximation- Value prediction and control- Gradient Descent Methods-Linear methods planning- prioritized sweeping-Heuristic search

Unit 4 Model Based RL And Classical Conditioning**L-9Hours**

Model Based RL- Bayesian Approaches to Reinforcement Learning- Data-efficient Reinforcement Learning- Learning with off-line data- Learning with incompletely specified models- RL and planning- Classical Conditioning : Blocking and Higher-order Conditioning -The Rescorla - Wagner Model - TD Model -Simulations - Instrumental Conditioning – Delayed Reinforcement- Cognitive Maps.

Unit 5 Applications**L-9Hours**

Meta-learning-Multi-Agent Reinforcement Learning- TD-GammonSamuel's Checkers Player Watson's Daily-Double Wagering-Mastering the Game of Go- AlphaGo-AlphaGo Zero- Ethics in RL- Applying RL for real-world problems - Optimizing Memory Control- Human-level Video Game Play- Autonomous UAV Navigation and path planning -Drones for Field Coverage

Total: 45 Hours**G. Learning Resources****i. Text Books:**

1. Richard S. Sutton and Andrew G. Barto. Introduction to Reinforcement Learning, 2nd Edition, MIT Press. 2018.[Unit 1,2,3, and 4]
2. Csaba Szepesvári, Algorithms for Reinforcement. Morgan & Claypool. 2010. [Unit 4 and 5]

ii. Reference books:

1. D.P.Bertsekas, Dynamic Programming and Optimal Control, Vol. I, Athena Scientific, 2017
2. Sudharsan Ravichandiran, Hands-On Reinforcement Learning with Python-Master Reinforcement and Deep Reinforcement Learning Using OpenAI Gym and TensorFlow, Packt Publishing ISBN- 9781788836913, 178883691X, 2018.

iii. Online References:

1. https://web.mst.edu/~gosavia/rl_website.html
2. https://link.springer.com/referenceworkentry/10.1007%2F978-0-387-30164-8_714
3. https://onlinecourses.nptel.ac.in/noc19_cs55/preview

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS119	CRYPTOGRAPHY AND NETWORK SECURITY	3	0	0	3

A. Preamble

The purpose of this course is to understand principles of encryption algorithms, conventional and public key cryptography. It also includes knowledge of authentication, hash functions and also application level of security mechanisms.

B. Prerequisite course

10211CS105 - Computer Networks

C. Course Objectives:

Learners are exposed to:

- Understand the basics of cryptography, its techniques and principles
- Make use of mathematical concepts to apply the public key cryptography
- Utilize the authentication hash functions and digital signatures for security
- Organize various cryptographic standards to apply over security applications
- Identify the preventive security mechanisms for efficient security management

D. Course Outcomes:

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Understand the encryption and decryption techniques using block ciphers.	K2
CO2	Apply key exchange and management schemes using public key cryptography	K3
CO3	Demonstrate techniques to sign and verify messages using signature generation and verification algorithms.	K3
CO4	Implement cryptographic algorithm for various network security applications	K3
CO5	Illustrate the technologies to protect cyberspace against security threats	K3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of COs with Program outcomes and Programme Specific Outcomes:

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3										2		
CO2	3	2		3		2						3		
CO3	3	2		3		2						2		
CO4	2				1	3						3		
CO5	2	3		2	2	2								

High- 3; Medium-2; Low-1

F.Course contents

Unit 1 Introduction to Cryptography

L-9 Hours

OSI Security Architecture - Classical Encryption techniques – Cipher Principles – Data Encryption Standard – Block Cipher Design Principles and Modes of Operation - Evaluation criteria for AES – AES Cipher – Triple DES – Placement of Encryption Function – Traffic Confidentiality-Case study on Barclays Bank

Unit 2 Public Key Cryptography

L-9 Hours

Number Theory concepts: Primes and Prime Factorization – Congruent modulo n , equivalent class modulo n , Integer modulo n , Multiplicative inverse, Relatively prime, Euler's theorem, Fermat's little theorem, Extended Euclidean Algorithm, Chinese Remainder Theorem. Confidentiality using Asymmetric Encryption – Public Key Cryptography and RSA- Key Management - Diffie-Hellman key Exchange – Elliptic Curve Architecture and Cryptography – Case study on Elan Financial Services

Unit 3 Authentication and Hash Function

L-9 Hours

Authentication requirements – Authentication functions – Message Authentication Codes – Hash Functions – Security of Hash Functions and MACs – MD5 message Digest algorithm – Secure Hash Algorithm – RIPEMD – HMAC Digital Signatures – Authentication Protocols – Digital Signature Standard- Case study on Swedbank

Unit 4 Network Security Applications

L-9 Hours

Authentication Applications: Kerberos – X.509 Authentication Service – Electronic Mail Security – PGP – S/MIME - IP Security- Policy, Encapsulating Security Payload, Combining Security Associations, Internet Key Exchange, Authentication Header.

Unit 5 Security Management

L-9 Hours

Intrusion detection and Prevention System – password management – Viruses and related Threats – Virus Countermeasures- Worms Security Risks – Firewall Design Principles – Trusted Systems- Log Management. Case study on Biometric deployment for secure password management.

Total: 45 Hours

G.Learning Resources

i.Text Books:

1. William Stallings, —Cryptography And Network Security – Principles and Practices, Prentice
2. Hall of India, Eighth Edition, 2020 [Unit 1-5].
3. David Kim and Michael G.Solomon, “Fundamentals of Information Systems Security”, Jones and Bartlett Publishers, Third Edition, 2018 [Unit 1-5].

ii.References Books:

1. 1AtulKahate, —Cryptography and Network Security, Tata McGraw-Hill, 2011.
2. Bruce Schneier, —Applied Cryptography, John Wiley & Sons Inc, 2011.
3. Charles B. Pfleeger, Shari Lawrence Pfleeger, —Security in Computing, Third Edition, Pearson Education, 2010.

iii. Online References

1. “Cryptography techniques”, Accessed on: July 2022, [online]. Available: <http://Cryptographywilliamstallings.com/Extras/Security-Notes/>
2. “Authentication algorithms”, Accessed on: July 2022 [online]. Available: <http://www.cs.bilk.ent.edu.tr/~selcuk/teaching/cs519/>
3. “Network security concepts”, Accessed on: July 2022[online]. Available: <http://freevidelectures.com/Course/3027/Cryptography-andNetwork-Security>

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS121	HIGH PERFORMANCE COMPUTING	3	0	0	3

A.Preamble:

This course introduces high performance computing foundations for developing, analyzing, and implementing parallel and distributed algorithms. This course focuses on modern HPC systems, such as OpenMP on shared memory nodes, CUDA for graphics co-processors (GPUs), and MPI models for distributed memory systems.

B.Prerequisite Course

10211CS104 - Computer Organization and Architecture

C.Course Outcomes:

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Understand the concepts used in modern processors for increasing the performance	K2
CO2	Recite the knowledge on optimization techniques for serial code	K2
CO3	Illustrate the concept of Parallel Computing paradigms	K2
CO4	Infer the distributed parallel programming concepts with efficient MPI programming	K2
CO5	Interpret the shared memory parallel programming with OpenMP and CUDA	K2
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

D.Correlation of COs with Program outcomes and Programme Specific Outcomes:

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	2	3		1	1	1						2		
CO2	3	3	2	1		1						3		
CO3	3	3	3	2	2							3	1	
CO4	3	2	3	2	3							3	2	
CO5	3	2	3	2	3							3	2	

3- High; 2-Medium; 1-Low

E.Course Contents:**Unit 1 Modern Processors****9 Hours**

Modern Processors : Stored Program Computer Architecture- General purpose cache- based microprocessor-Performance based metrics and benchmarks- Moore's Law- Pipelining- Super scalarity-SIMD- Memory Hierarchies Cache- mapping- prefetch- Multicore processors- Mutithreaded processors- Vector Processors- Design Principles- Maximum performance estimates- Programming for vector architecture.

Unit 2 Optimization Techniques for Serial Code**9 Hours**

Basic optimization techniques for serial code : scalar profiling- function and line based runtime profiling- hardware performance counters- common sense optimizations- simple measures, large impact- elimination of common sub expressions- avoiding branches- using SIMD instruction sets- the role of compilers - general optimization options- inlining - aliasing- computational accuracy- register optimizations- temporaries- dynamic memory management- loop kernels and iterators data access optimization: balance analysis and light speed estimates- storage order.

case study: jacobi algorithm and dense matrix transpose.

Unit 3 Parallel Computers

9 Hours

Parallel Computers : Taxonomy of parallel computing paradigms- Shared memory computers- Cache coherence- UMA - ccNUMA- Distributed-memory computers- Hierarchical systems- Networks- Basic performance characteristics- Buses- Switched and fat- tree networks- Mesh networks- Hybrids - Basics of parallelization - Why parallelize - Data Parallelism - Function Parallelism- Parallel Scalability- Factors that limit parallel execution- Scalability metrics- Simple scalability laws- parallel efficiency - serial performance Vs Strong scalability- Refined performance models- Choosing the right scaling baseline

Case Study: Can slow processors compute faster- Load balance

Unit 4 Distributed Memory Parallel Programming with MPI

9Hours

Message passing - Introduction to MPI – example - messages and point-to-point communication - collective communication – nonblocking point-to-point communication- virtual topologies - MPI parallelization of Jacobi solver- MPI implementation - performance properties . Efficient MPI programming : MPI performance tools- communication parameters- Synchronization, serialization, contention- Reducing communication overhead- optimal domain decomposition- Aggregating messages – Nonblocking Vs Asynchronous communication- Collective communication- Understanding intra-node point-to-point communication.

Unit 5 Shared Memory Parallel Programming with OPENMP and CUDA

9Hours

Introduction to OpenMp - parallel execution - data scoping- OpenMp work sharing for loops- synchronization - reductions - loop scheduling - tasking - case study: OpenMp- parallel jacobi algorithm- advanced OpenMp wavefront parallelization- Efficient OpenMP programming: Profiling OpenMP programs - Performance pitfalls, Introduction to GPU Computing, CUDA Programming Model, CUDA API, Simple Matrix, Multiplication in CUDA , CUDA Memory Model, Shared Memory Matrix Multiplication, Additional CUDA API Features - Case study: Parallel Sparse matrix-vector multiply

Total: 45 Hours

H.Learning Resources

i.Text Book:

1. Georg Hager, Gerhard Wellein, “Introduction to High Performance Computing for Scientists and Engineers”, Chapman & Hall / CRC Computational Science series, reprint 2010 .[Unit 1-5].

ii.Reference Book(s):

1. Victor Eijkhout, Edmond Chow, Robert van de Geijn, “Introduction to High Performance Scientific Computing”, 2nd edition, revision 2016
2. Zbigniew J. Czech, “Introduction to parallel computing”, 2nd edition, Cambridge University Press, 2016

iii.Online Reference(s):

1. Introduction to High Performance Computing for Scientists and Engineers, https://cds.cern.ch/record/1338418/files/9781439811924_TOC.pdf
2. Charles Severance, Kevin Dowd, High Performance Computing <http://cnx.org/content/col11136/1.5/>

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS125	PRINCIPLES OF COMPILER DESIGN	3	0	0	3

A.Preamble

This course introduces the basic concepts and mechanisms that are employed in language translators, with emphasis on compilers. It aims to provide the theory and practice of compilation, in particular, the lexical analysis, parsing, intermediate code generation, techniques of code optimization and approaches towards code generation.

B.Prerequisite Course

10211CS106 - Formal Languages and Automata Theory

C.Course Objectives

Learners are exposed to:

- Understand the structure of a compiler with the sequence of translation of input for each phase.
- Analyze the regular languages to describe the lexical elements of a programming language.
- Develop the syntactic structure of a programming language using the context free language and design the top-down and bottom-up parsers.
- Break down the statements and expressions to form and implement the intermediate code.
- Optimize the intermediate code of a program to machine-specific instructions to generate target code.

D.Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Interpret the different phases of compilation with compile time error handling and design the lexical analyzer for a language.	K3
CO2	Develop appropriate top down and bottomup parser to produce parse tree representation of the input.	K3
CO3	Generate intermediate code for decision control and looping control statements in high level language.	K3
CO4	Apply various optimization techniques with basic blocks of the intermediate code through global data flow analysis.	K3
CO5	Design DAG representation for basic blocks and target low level language for high level language	K3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

F.Correlation of COs with Program outcomes and Programme Specific Outcomes:

Cos	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO1 0	PO1 1	PSO 1	PSO 2	PSO 3
CO 1	3	3	1		2							3		

CO 2	3	2	1		1							3		
CO 3	3	3										3		
CO 4	3	3										3		
CO 5	3	3	1									3		

3- High; 2-Medium; 1-Low

F.Course Contents

Unit 1 Compilers and Lexical Analysis

L-9 Hours

Compilers – Interpreters – Analysis of the source program –Cousins of the Compiler – Grouping of Phases – Phases of a compiler – Compiler construction tools – Lexical Analysis – Role of Lexical Analyzer – Input Buffering – Recognition of Tokens – Specification of Tokens. Design of a Lexical Analyzer Generator.

Case Study: Lexical Analyzer for separating tokens, Counting whitespace, special characters and newline character, Pattern Matching of strings – use relevant tool.

Unit 2 Syntax Analysis and Parser

L-9 Hours

Need and Role of the parser –Writing Grammars – Context-Free Grammars – Top Down parsing – Recursive Descent Parsing – Predictive Parsing – Bottom-up parsing – Shift Reduce Parsing – Operator Precedence Parsing – LR Parsers – SLR Parser – Canonical LR Parser – LALR Parser.

Case Study: Specification for desktop calculator, Error recovery of ambiguous arithmetic grammar, Syntax Analyzer for looping construct - use relevant tool.

Unit 3 Intermediate Code Generation

L-9 Hours

Intermediate languages – Implementation of Three Address Code – Types and Declarations – Type Checking – Assignment Statements – Boolean Expressions – Switch Case Statements – Back patching – Procedure calls.

Case Study: Intermediate code for Array references with subscripts, Intermediate code for decision making and looping constructs of C, Intermediate code for Expression Parsing.

Unit 4 Code Optimization

L-9 Hours

Principal Sources of Optimization – Introduction to Data Flow Analysis – Constant propagation – Loops in Flow Graphs – Runtime Environments – Source Language issues – Storage Organization – Storage Allocation – Access to Nonlocal data - Local optimization- Global optimization.

Case Study: Implement code optimization methods using relevant data structures.

Unit 5 Code Generation

L-9 Hours

Issues in the design of code generator – The target machine – Register Allocation and Assignment- Basic Blocks and Flow Graphs – Optimization of basic Blocks – A Simple Code Generator – DAG

representation of Basic Blocks – Code Generation from DAG - Heuristic Reordering - Labeling - Next-use Information – Peephole Optimization.

Case Study: Graph coloring application using Register Allocation, Generate Target code for Binary tree operations.

Total: 60 Hours (L)

G.Learning Resources

i. Text Books:

1. Alfred V. Aho, Monica S. Lam, Ravi Sethi, and Jeffrey D. Ullman, Compilers –Principles Techniques and Tools, Pearson, Second Edition, 2013. [Unit 1-5].

ii. Reference Books:

1. Kenneth C. Loudon, Compiler Construction –Principles and Practice, Cengage Learning Indian Edition, 2006.
2. Grune D, Bal H, Jacobs C, Langendoen K, Modern Compiler Design, Wiley, 2000
3. Cooper K, Torczon L, Engineering a Compiler, Morgan Kaufmann, Second Edition, 2011

iii. Online References:

1. “COMPILER DESIGN” Sep. 25, 2014. Accessed on: Feb. 16, 2021 [Online]. Available: <https://nptel.ac.in/courses/106/105/106105190/>
2. “LEX & YACC TUTORIAL by Tom Niemann” Oct. 24, 2013. Accessed on: Apr. 20, 2021 [Online]. Available: <https://cse.iitkgp.ac.in/~bivasm/notes/LexAndYaccTutorial.pdf>
3. “Compiler Design”, Nov. 2019. Accessed on: Apr. 20, 2021 [Online]. Available: <https://www.udemy.com/course/compiler-design-n/>

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS128	FINANCIAL ACCOUNTING	3	0	0	3

A.Preamble

To learn the basic Business types, impact of the Economy on Business and Firms specifically. To analyze the Business from the Financial Perspective. Student will be able to read the three most common financial statements: the income statement, balance sheet, and statement of cash flows

B.Prerequisite Course : 10210MA103 - Probability, Statistics and Queuing Theory

C.Course Objectives

Learners are exposed to

- This course will enable the students to combine practice and theoretical knowledge of financial accounting.
- The students of this course will be active learners and develop awareness of emerging trends in financial accounting,
- The course will provide decision making skills to the students in the financial analysis context.

D.Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Understand the basics of financial accounting, investments and Indian Accounting standards.	K2
CO2	Summarize the overview on accounting system, transactions and statements	K2
CO3	Outline on depreciation and various inventory evaluation methods	K2
CO4	Explain the methods of cash flow statements and accounting rules	K2
CO5	Explore the inventory evaluation methods and errors	K2
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E.Correlation of COs with Program outcomes and Program Specific Outcomes

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1										3				
CO2	2									2		2		
CO3	2									2		2		
CO4	2									2				
CO5	2									2		2		

3- High; 2-Medium; 1-Low

F.Course Contents

Unit 1 Basics of Financial Accounting

L-9 Hours

Introduction-Accounting information system- Users -Needs- Qualitative characteristics- Advantages and limitations of accounting- Branches of accounting- Financial accounting principles –Capital Receipts- Revenue and Capital Expenditure- Deferred Revenue expenditure- Indian Accounting Standards –AS-1, AS-2, AS-7, AS-9, AS-6, AS-10 and GAAP.

Unit 2 Accounting Process

L-9 Hours

Double-entry accounting system- Debits and Credits- Accounting rules and procedures- Terms: Transaction, Capital, Drawings- Liabilities- Assets- Fixed assets, Expenditure-Expense, Income, Profit, Gain, Loss, Purchase, Sales, Goods, Stock, Debtor, Creditor, Voucher, Discount -Trade discount and Cash Discount- Analyzing and journalizing transactions- Posting transactions to ledger- Accounting Rules-Preparing the trail balance- Effects of errors.

Unit 3 Depreciation

L-9 Hours

Depreciation, Provisions and Reserves- Depreciation: Concept, Features, Causes, factors - Other similar terms: Depletion and Amortization - Methods of Depreciation-Straight Line Method (SLM) -Written Down Value Method (WDV)- Inventories: Meaning. Significance of inventory valuation- Inventory Record Systems: periodic and perpetual- Methods: FIFO, LIFO and Weighted Average.

Unit 4 Cash Flows

L-9 Hours

Overview the cash flow statement- Preparing the cash flow statement- Types of cash flow activities-inflows and outflows of cash flows-Cash flow statement and balance sheet- Examples of cash flow statements- Methods of preparing cash flow statements- Accounting rules- Importance of cash flow statements.

Unit 5 Mining Data Streams

L-9 Hours

Adjustments in preparation of financial statements with respect to closing stock- outstanding expenses- Prepaid expenses- Accrued income- Income received in advance- Depreciation- Bad debts- Doubtful debts- Provision for discount on debtors- Abnormal loss- Goods taken for personal use/staff welfare- Interest on capital and managers commission-Preparation of Trading, Profit and Loss account - Balance Sheet of a profit-oriented organization-Single entry system.

Total: 45 Hours

G.Learning Resources

i.Text Books:

1. Horngren, Sundem, Elliott and Philbrick, “Introduction to Financial Accounting”, Pearson Education India Ltd., 11th edition, 2017.

ii.Reference Books:

1. Libby, Robert., Libby, Short, “Financial Accounting”, Tata McGraw-Hill., 7th edition, 2016
2. R Narayanaswamy, Financial Accounting: A managerial perspectives, 6th edition, PHI 2017

iii.Online References:

- 1.<http://www.xn--klker-kva.hu/wp-content/uploads/2013/03/30459588-Principles-of-Managerial-Finance-by-Gitman.pdf>
- 2.[http://www.mim.ac.mw/books/Principles-of-Managerial-Finance-13th Edition_Full.pdf.9Ukgy8w1RYp7U1dls6CxRLDn1i8aezIq](http://www.mim.ac.mw/books/Principles-of-Managerial-Finance-13th-Edition_Full.pdf.9Ukgy8w1RYp7U1dls6CxRLDn1i8aezIq)

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS210	BIG DATA ANALYTICS	3	0	2	4

A.Preamble

The key objective of this course is to make the students to be familiar with the concepts of data warehouses, modelling and design of data warehouse. This course includes the most significant technologies used for manipulating, storing, and analyzing big data. In addition, the basic tools of data warehouse and big data analytics will be explored. These tools make the students to understand the strategies involved in data warehouse design and Big Data Platform.

B.Prerequisite Course

10211CS207 - Database Management Systems

C.Course Objectives

Learners are exposed to:

- Understand the principles of data warehousing, multidimensional data model.
- Familiar with the data warehouse architecture and OLAP tools.
- Impart the architectural concepts of big data file management system.
- Explore PIG and HIVE tools to develop applications in Big data analytics.
- Implement best practices for Big Data Optimization.

D.Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Build and process the data warehouse model for given application	K3/S3
CO2	Design Job Execution procedures in Map Reduce and Apache Spark Paradigm	K3/S3
CO3	Design memory efficient solutions for Big Data Applications	K3/S3
CO4	Build Big Data Solutions for social media network applications	K3/S3
CO5	Apply cloud and optimization techniques to develop the solutions for real time scenarios	K3/S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E.Correlation of COs with Program outcomes and Programme Specific Outcomes:

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3		3	3									3	
CO2		2	2	2	3								3	3
CO3	2		3		3	2						2	2	2
CO4	2	3		3	3	3						3	3	1
CO5	2	2		3		3						3	3	

3- High; 2-Medium; 1-Low

F.Course Content

Unit 1 Data Warehousing And Business Analysis

L-9 Hours

Introduction to Data warehousing – Evolution of Decision Support systems – Modeling a Data Warehouse – Granularity in the Data Warehouse - Data Warehousing Components, Building a Data Warehouse, Warehouse Database- Extract, Transform and Load: ETL Overview, ETL Requirements and Steps, Data Transformation, Data Loading, ETL Tools. – Reporting and Query tools and Applications –Online Analytical Processing (OLAP) – Need – Multidimensional Data Model.

Unit 2 Big Data Processing

L-9 Hours

Introduction to Big Data, Big Data Analytics, Evolution of Big data – Best Practices for Big data Analytics – Big data characteristics- Understanding Big Data Storage – A General Overview of High-Performance Architecture – HDFS – Map Reduce Programming Model-Understanding the basics of MapReduce, Loading data into HDFS, Introduction-Apache Spark, Features, Components, Resilient Distributed Datasets, Data Sharing using Spark RDD, Spark Programming.

Unit 3 Big Data Storage System

L-9 Hours

Apache Camel- Introduction, Overview, features, camel context, architecture,message queues. Apache Ignite-Memory Architecture, Memory Pages, Lifecycle, In-Memory Data Grid, Caching Support, Streaming Support. Cassandra-Architecture, Data Model, Referenced Api, Cqlsh, Keyspace Operations, Table Operations, Batch, CURD Operations. Apache Kafka- Fundamentals, Cluster Architecture, Work Flow, Basic Operations, Producer and Consumer Example.

Unit 4Big data Visualization and Prediction

L-9 Hours

Pig : Introduction to PiG, Execution Modes of Pig, Comparison of Pig with Databases, Grunt, Pig Latin, User Defined Functions, Data Processing operators. Hive : Hive Shell, Hive Services, Hive Metastore, Comparison with Traditional Databases, HiveQL, Tables, Querying Data and User Defined Functions, NoSQL Databases : Schema-less Models: Increasing Flexibility for Data Manipulation-Key Value Stores- Document Stores – Tabular Stores – Object Data Stores – Graph Databases Hive – Sharding- Hbase – Analyzing big data with twitter – Big data for E-Commerce Big data for blogs.

Unit 5 Big Data Cloud Concepts And Optimization

L-9 Hours

Big data Cloud Computing-Features, Cloud Deployment Models, Cloud Delivery Models, Cloud for Big Data,Real time Analytics Platform(RTAP) applications – Using Graph Analytics for Big Data: Graph Analytics, Big Data Optimization- Smooth Convex Optimization-Projection-free methods, Accelerated gradient descent methods, Non smooth Convex Optimization-Smoothing techniques, Mirror-Prox method, Sparsity learning, Large-scale kernel machines.

Total : 45 Hours

G.Laboratory Experiments

Total : 30 Hours

Part - 1

Task 1

Design an multi-dimensional data model schema namely Star, Snowflake and Fact Constellations for a Categorical data using SQL Server Management Studio (SSMS).
(Perform the above for Banking, Healthcare, Manufacturing, Sales and Automobile)

Tools: SQL Server Management Studio (SSMS), Microsoft Azure SQL Pool

- Task 2** To configure, monitor, and administer a Data warehouse and perform basic Query operations on the DW.
- Tools: SQL Server Management Studio (SSMS), Microsoft Azure SQL Pool**
- Task 3** Perform Data Cube Operations (OLAP Operations) using SQL Queries
- Rollup
Rolldown
Slicing
Dicing
- Database: MySQL**
- Task4** Implement matrix multiplication with Map Reduce.
- Tools: LINUX**
- Task5** Write a Spark application to perform word count in the input file.
- Tools: APACHE SPARK**
- Task 6** Implement CURD operations on Casandra
- Tools: Casandra**
- Task 7** Implementing Producer and Consumer problem in kafka.
- Tools:kafca**
- Task8** i. Implement basis commands in HIVE.
ii. Use Hive to create, alter, and drop databases, tables, views, functions, and indexes
- Tools: HIVE , LINUX**
- Task9** Write Pig Latin scripts sort, group, join, project, and filter the data
- Tools: Pig, LINUX**
- Task10** i. Construct the Pig Latin Scripts to find character Count
ii. Construct the Pig Latin Scripts to find a max temp for each and every year.
Tools: Pig, LINUX
- Task11** Collect any Social Media Data from a Twitter to a Local File with the Topic 'covid 19'. Download and Set Up MongoDB Server and a Client Mongo shell.
- Tools: MongoDB, Python: Scipy**
- Task12** Retrieve Analytic Information given below from MongoDB created in task 9:
- For each "place_type", Find total favorite_count
 - For each "country_code", find total "retweet_count"
 - Find out top 10 most frequent topic words of the entire tweet message texts of your collection after lemmatization/stemming and removing all the Stop Words.
- Tools: MongoDB, Python: Scip**

Part – 2

UseCases:

Use case 1:

Data analytics using Apache Spark on Amazon food dataset, find all the pairs of items frequently reviewed together.

Write a single Spark application that:

- Transposes the original Amazon food dataset, obtaining a PairRDD of the type: user-id – list of the product-ids reviewed by user-id
- Counts the frequencies of all the pairs of products reviewed together;
- Writes on the output folder all the pairs of products that appear more than once and their frequencies.
- The pairs of products must be sorted by frequency.

Use case 2:

Construct MapReduce program to perform data analysis on weather dataset.

Dataset is available at: <https://www.kaggle.com/datasets/zaraavagyan/weathercsv>

Use case 3:

Apache Hive for Real-time applications

It is often the case that data has to be retrieved in real-time from the source. The processing is done as soon as the data is inputted. For example, Google maps processes traffic data in real-time. As soon as it receives information from the source, the data is output onto its application. Develop the Real-time data processing using Apache Hive for analysing the Google Map data.

Use case 4:

Build a Data Pipeline based on Messaging using PySpark and Hive:

Implement a data pipeline that ingests raw data from a source and moves this data to a destination where it can be stored or processed further for analysis. The pipeline should also be able to filter or clean the data for various purposes. Then create a Hive external on top of HDFS which will allow the cleaned, processed data to be deployed.

Total:75 Hours

H. Learning Resources

i. Textbooks:

- Jiawei Han and Micheline Kamber, “Data Mining Concepts and Techniques”, Third Edition, Elsevier, 2012. (Unit 1)
- Seema Acharya, Subhasini Chellappan, "Big Data Analytics" Wiley 2015 (Unit 2-5).

ii. Reference Books:

- G. SudhaSadasivam & R. Thirumahal, “Big Data Analytics”, First Edition, Oxford University Press, 2020
- Capriolo, E., Wampler, D., & Rutherglen, J., “Programming Hive”, O’Reilly Media, Inc., 2012

iii. Web Resources:

- “Data Warehouse Design: Modern Principles and Methodologies”, Accessed on Oct 27, 2022 [online]. Available: https://cdn.ttgtmedia.com/searchDataManagement/downloads/Data_Warehouse_Design.pdf
- “Big data Analytics”, Accessed on Apr. 5, 2022 [online]. Available: <https://www.analyticsvidhya.com/learning-paths-data-science-business-analytics-business-intelligence-big-data/learning-path-data-science-python/>
- “Apache Spark Tutorial”, Accessed on Oct 28, 2022 [online]. Available: https://www.tutorialspoint.com/apache_spark/apache_spark_deployment.htm
- “Apache Camel Tutorial”, Accessed on Oct 28, 2022 [online]. Available: https://www.tutorialspoint.com/apache_camel/apache_camel_quick_guide.html

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS211	ARTIFICIAL INTELLIGENCE TECHNIQUES	3	0	2	4

A. Preamble

The primary objective of this course is to introduce the basic principles, techniques, and applications of Artificial Intelligence. In this course students can explore the various Search techniques like basic, advanced and heuristic approach are universal problem-solving methods to solve a specific problem and provide the best result. Evolutionary algorithms are used to optimize the AI applications. Goal and constraint-based approach, Knowledge reasoning and inference helps the students to develop the decisions making strategies in intelligent agents and expert systems.

B. Prerequisite Course

10210CS101 - Problem Solving using C

C. Course Objectives

Learners are exposed to:

- Understand and apply the basic principles of artificial intelligence concepts towards problem solving, knowledge representation, reasoning, inference and learning.
- Explore the artificial intelligence techniques in intelligent agents, expert systems, artificial neural networks, Genetic algorithms, Heuristics approaches and Evolutionary algorithms models.

D. Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Identify the problem-solving techniques by using State Space search and heuristic search approach	K3/S3
CO2	Use evolutionary and advanced search techniques to provide optimized solution.	K3/S3
CO3	Solve unified planning approach and Constraint-satisfaction problems.	K3/S3
CO4	Apply Logical knowledge representation for rules and fact-based approach.	K3/S3
CO5	Build the logic of Uncertainty and Reasoning using resolution methods.	K3/S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of Cos with Program Outcomes and Programme Specific Outcomes

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3		1	1			1	1		1	2		3
CO2	3	3	3	1	1			1	1		1	2		3
CO3	3	3	3	1	1			1	1		1	2		3
CO4	3	3	2	1	2			1	1		1	1		3
CO5	3	3	3	1	2			1	1		1	2		3

High - 3, Medium - 2, Low – 1

F. Course Contents**Unit 1 Basic and Heuristic Search Techniques****L-9 Hours**

Artificial Intelligence – Introduction - Evolution of AI- State Space Search: Generate and Test- Simple Search- Depth First Search- Breadth First Search- Comparison of BFS and DFS- - Depth Bounded DFS- Depth First Iterative Deepening. Heuristic Search: Heuristic Functions- Best First Search- Hill Climbing- N-Puzzle Problem- Block-world Problem Local Maxima- Variable Neighborhood Descent- Beam Search- Tabu Search- Case Studies – Stock Market prediction, Weather Forecasting.

Unit 2 Evolutionary and Advanced Search Techniques**L-9 Hours**

Randomized Search and Emergent Systems: stochastic and evolutionary search algorithms- Simulated Annealing- - Emergent Systems- Ant Colony Optimization- Finding Optimal Paths: The Travelling Salesman Problem- Dynamic Programming- Algorithm A*- Iterative Deepening A* - Min-Max- Alpha-Beta Tree search – Monte Carlo Tree Search - Case Studies – Vehicle Routing and Map Navigation.

Unit 3 Planning and Strategies**L-9 Hours**

Planning: planning as search, partial order planning - A Unified Planning Framework-Forward and Backwards State Space Planning- Goal Stack Planning- Plan Space Planning- The STRIPS Domain- Constraint Satisfaction Problem: N-Queens-Cryptarithmic- Graph Coloring- Constraint Propagation- Scene Labeling- Higher Order and Directional Consistency- Algorithm Backtracking: Graph Coloring - Case Studies – Robot traversal- Object recognition- Game Playing.

Unit 4 Knowledge Representation**L-9 Hours**

Ontologies, foundations of knowledge representation- Logical representation -The Scheme- Frames- Semantic Net- Production rules -Scripts - Inheritance in Taxonomies- Description Logics- Formal Concept Analysis- Conceptual Graphs Knowledge reasoning - Reasoning about objects- relations – events- actions- time and space-reasoning with defaults- Reasoning about knowledge Case studies – chatbot system- recommendation system.

Unit 5 Knowledge Facets and Logic Inferences**L-9 Hours**

Knowledge Facets: Intelligent agents: reactive, deliberative, goal-driven, utility-driven, and learning agents- The Wumpus World. Logic and Inferences: Formal Logic- Resolution Method in Propositional Logic- Resolution method in First Order Logic- Deductive Retrieval – Horn Clauses – Forward Chaining- Backward Chaining- Uncertain Knowledge- Probabilistic Reasoning- connection to logic-independence- Bayes rule- Bayesian networks- probabilistic inference, Case studies – Multi- Agent Decision Making System- Human Computer Interaction.

Total: 45 Hours**G.Laboratory Experiments****Total: 30 Hours****Part - 1**

Task 1 Implementation of Graph search algorithms (Breadth first search and Depth First Search) using following constraints.

BFS: Pick any node, visit the adjacent unvisited vertex, mark it as visited, display it, and insert it in a queue. If there are no remaining adjacent vertices left, remove the first vertex from the queue. Apply recursion concept to follow the above steps until the queue is empty or the desired node is found.

DFS: Pick any node. If it is unvisited, mark it as visited and recur on all its adjacent nodes. Repeat until all the nodes are visited, or the node to be searched is found.

Tools- Prolog

Task 2 Implementation of Hill climbing algorithm for Heuristic search approach using following constraints in python.

- i. Create a function generating all neighbours of a solution
- ii. Create a function calculating the length of a route
- iii. Create a random solution generator
- iv. Create a Travelling salesman problem

Tools- Python, Online Simulator - <https://graphonline.ru/en/>

Task 3 Implementation of A * Algorithm to find the optimal path using Python by following constraints.

- The goal of the A* algorithm is to find the shortest path from the starting point to the goal point as fast as possible.
- The full path cost (f) for each node is calculated as the distance to the starting node (g) plus the distance to the goal node (h).
- Distances is calculated as the manhattan distance (taxicab geometry) between nodes.

Tools- Python, Online Simulator - <https://graphonline.ru/en/>

Task 4 Implementation of Mini-Max algorithm uses recursion to search through the game-tree using python by applying following constraints.

- In this algorithm two players play the checker's game; one is called MAX and other is called MIN.
- Both the players fight it as the opponent player gets the minimum benefit while they get the maximum benefit.
- Both Players of the game are opponent of each other, where MAX will select the maximized value and MIN will select the minimized value.
- The minimax algorithm performs a depth-first search algorithm for the exploration of the complete game tree.
- The minimax algorithm proceeds all the way down to the terminal node of the tree, then backtrack the tree as the recursion.

Tools : Python

Task 5 Implementation of Ant Colony Optimization to Optimize Ride-Sharing Trip Duration using Python by following constraints.

- To forecast travel times between every pair of pick-up and drop-off locations.
- To find the shortest route that visits a set of locations.
- To implement optimization techniques are required to intelligently search the solution space and find near-optimal solutions.

Tools: Python

Task 6 Solve a Map Coloring problem using constraint satisfaction approach by applying following constraints

- Assign each territory a color such that no two adjacent territories have the same color by considering following parameters: Domains, Variables and Constraints
- Apply Basic Greedy Coloring Algorithm: Color first vertex with first color, do following for remaining V-1 vertices.
- Consider the currently picked vertex and color it with the lowest numbered color that has not been used on any previously colored vertices adjacent to it. If all previously used colors appear on vertices adjacent to v, assign a new color to it.

Tools- Python, Online Simulator - <https://graphonline.ru/en/>

Task 7 Implementation of Monkey Banana Problem in Goal Stack planning using prolog by applying following constraints.

Imagine a room containing a monkey, chair and some bananas. That have been hanged from the centre of ceiling. If the monkey is clever enough, he can reach the bananas by placing the chair directly below the bananas and climb on the chair. The problem is to prove the monkey can reach the bananas. The monkey wants it, but cannot jump high enough from the floor. At the window of the room there is a box that the monkey can use. The monkey can perform the

Following actions: -

- 1) Walk on the floor.
- 2) Climb the box.
- 3) Push the box around (if it is beside the box).
- 4) Grasp the banana if it is standing on the box directly under the banana.

Tools: Prolog

Task 8 Implementation of N-queen problem using backtracking algorithm using prolog

In the 4 Queens problem the object is to place 4 queens on a chessboard in such a way that no queens can capture a piece. This means that no two queens may be placed on the same row, column, or diagonal.

Tools: Prolog

Task 9 To Build an Intelligent Chatbot system with Python and Dialog-flow using Interactive Text Mining Framework for Exploration of Semantic Flows in Large Corpus of Text.

- To integrate with Google Cloud Speech-to-Text and third-party services such as Google Assistant, Amazon Alexa, and Facebook Messenger.

- Configure Dialogflow to manage your data across GCP services and let you optionally integrate Google Assistant.

Tools- Python, Dialog-flow Framework

Task 10 Implement simple fact for following:

- Ram likes mango.
- Seema is a girl.
- Bill likes Cindy.
- Rose is red.
- John owns gold.

Tools-Prolog

Part-2

Use Cases:

Use Case – 1:Implementation of Map navigation using Heuristic search approach in python.

Use Case – 2:Develop a simple neural network for simulating logic gate operations.

Use Case – 3:Solve Multiple criteria production scheduling problem using Genetic algorithm

Use Case – 4: Implementation of Text Lemmatization using NLTK Python Package.

Use Case – 5: Implementation of Intelligent Chat-Bot system using Python and Dialogflow

Use Case – 6: Implementation of Block world problem using STRIPS domain precondition rules in prolog.

Total:75 Hours

G. Learning Resources

i.Text Books:

1. Stuart. J. Russell, et al. “Artificial Intelligence by Pearson: A Modern Approach” 4th Ed, 2020.

ii.Reference Books:

1. Rich and K. Knight,” Artificial Intelligence”, Mc Graw Hill Publishers Inc, 3rd Edition, 2017.
2. Deepak Khemani, “A First Course in Artificial Intelligence”, McGraw Hill Education, 2013.

iii.Online References:

1. “Artificial Intelligence :Search Methods for Problem Solving”, May. 06, 2014 [Online]. Available: <https://nptel.ac.in/courses/106/106/106106126/>
2. “Artificial Intelligence: Knowledge representation and reasoning” Jan. 19, 2016 [online] Available : <https://nptel.ac.in/courses/106/106/106106140/>

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS214	DATA VISUALIZATION	3	0	2	4

A.Preamble

Data visualization is the visual and interactive exploration and graphic representation of data of any type. This course covers data visualization concepts, practices, and tools particularly for analyzing and presenting business data. Students will evaluate, design, and develop effective visualizations and dashboards, using various development tools.

B.Prerequisite Course

10211CS207-Database Management Systems

C.Course Objectives

Learners are exposed to:

- Employ best practices in data visualization to develop charts, maps, tables, and other visual representations of data.
- Use visualization tools to conduct data analysis, especially exploration of a large dataset.

D.Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Explain the visualization process and identify the type of data	K2/S3
CO2	Apply various visualization techniques for Exploratory Data Analysis	K3/S3
CO3	Explore visualization techniques Trees, Graphs, Networks, Text and documents	K3/S3
CO4	Utilize various visualization techniques for spatial and Geospatial data	K3/S3
CO5	Make use of different visualization tools for various applications	K3/S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E.Correlation of COs with Program outcomes and Programme Specific Outcomes:

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO 1	PSO2	PSO 3
CO1	3	3	2		1			1				2		
CO2	3	3	3	3	2			1	2			3	3	
CO3	3	3	3	3	2			1	2			3	3	
CO4	3	3	3	3	2			1	2			2	3	2
CO5	3	3	3	3	3	3		1	3			3	3	2

3- High; 2-Medium; 1-Low

F.Course Contents

Unit 1 Introduction to Visualization

L-9 Hours

Vision and Color Perception : Human Perception and Information Processing - Model of perceptual processing - Perception in visualization - Metrics - Visualization Process. **Concepts and theories of data visualization:** Gestalt principles, Information overloads – Gibson's Affordance theory - Costs and Benefits of visualization – Data abstraction and Data types – Dataset – Visualization Foundation.

Unit 2 Visualization for EDA

L-9 Hours

Univariate analysis: Categorical Data - Bar chart – Pie Chart - Continuous data – Scatterplot-Line Plot-Strip Plot – Swarm Plot – Histogram – Density Plot – Rug Plot. **Bivariate analysis:** Categorical vs. Categorical: Stacked Bar Chart - Grouped Bar chart – Segmented Bar Chart - Mosaic Plots - Continuous vs. Continuous: Scatterplot Fit Lines - Categorical vs. Continuous : Bar Chart (Summary statistics) - Grouped Kernel Density Plots - Box Plots- Violin Plots - Ridgeline Plots- Beeswarm Plots. **Multivariate Analysis :** Point-Based Techniques - Line-Based Techniques - Region-Based Techniques - Combinations of Techniques.

Unit 3 Visualization for Trees, Graphs, Networks and Text

L-9 Hours

Displaying Hierarchical Structures - Displaying Arbitrary Graphs/Networks - Issues. Levels of Text Representations - The Vector Space Model - Single Document Visualizations - Document Collection Visualizations - Extended Text Visualizations - Designing Effective Visualizations - Steps in Designing Visualizations - Problems - Comparing and Evaluating Visualization Techniques.

Unit 4 Visualization for Spatial and Geospatial data

L-9 Hours

Visualization Techniques for Spatial Data - One-Dimensional Data - Two-Dimensional Data -Three-Dimensional Data - Dynamic Data. **Visualization Techniques for Geospatial Data** - Visualizing Spatial Data - Visualization of Point Data -Visualization of Line Data - Visualization of Area Data - Issues in Geospatial Data Visualization.

Unit 5 Visualization for Time Oriented Data

L-9 Hours

Characteristics of Time Oriented Data - Relating Data and Time - Visualizing Time-Oriented Data - Visualization Techniques to Analyze Outcomes of Feature Selection - Classification – Clustering – Regression - Model Selection

Total:45 Hours

G.Laboratory Experiments

Total:30 Hours

Part- 1

Task 1: Exploration of Data Visualization Tools like Tableau, Python libraries, D3.js

- Connecting Dataset
- Preparation of data

Exploratory Data Analysis

Task 2: To visualize and perform Univariate analysis using continuous and categorical data

Categorical Data - Bar chart, Pie Chart

Continuous data – Scatterplot, Line Plot, Strip Plot, Swarm Plot, Histogram, Density Plot, Rug Plot.

Tools: Tableau, Language :Python

Task 3: To visualize and perform Bivariate analysis using continuous and categorical data

Categorical vs. Categorical: Stacked Bar Chart, Grouped Bar chart, Segmented Bar Chart, Mosaic Plots - Continuous vs. Continuous: Scatterplot Fit Lines

Categorical vs. Continuous: Bar Chart (Summary statistics), Grouped Kernel Density Plots, Box Plots, Violin Plots, Ridgeline Plots, Beeswarm Plots.

Tools: Tableau, Language:Python

Task 4: To visualize and perform Multivariate analysis using Multiple variables involving Multiple measures

Scatterplot Matrix, Parallel Coordinates, Line Graph, Stacked Bar Chart,

Tools: Tableau, Language:Python

Task 5: To design and perform visualization for Trees

- TreeMap, Sun Burst Display

Tools: Tableau, Language:Python

Task 6: To design and perform visualization for Graphs and Networks

- Force based Layout

Tools: Tableau, Language:Python

Task 7: To generate insight using Text Network Analysis and Visualization

Tools: Wordle, Tag Cloud, WordTree, InfraNodus

Task 8: To analyze and visualize Spatial and Geospatial data

Geographical Map, Map Projections

Tools: Tableau, QGIS, Language:Python

Task 9:To analyze and visualize Time Oriented Data

Analysis to identify systemic patterns in the data that help to form trends, cycles or seasonal variances and to forecast the data. - Line Graph, Trend Lines, Area Chart

Tools: Tableau, Language :Python

Part- 2

1. Performance of sales representatives
2. Performance of different company departments over year
3. Company Sales Branches Comparison
4. Call Time Analysis
5. Earthquake and Geospatial Data Analysis
6. Top10 startup Investment Analysis
7. Health Care analysis for patient care and reducing costs
8. Create an interactive dashboard, to convey the data that had been collected over the financial year.
9. Creating a Dashboard using COVID-19 data

Total:75 Hours

H.Learning Resources

i.Text Books:

1. Matthew Ward, Georges Grinstein, Daniel Keim, “Interactive Data Visualization Foundations, Techniques, and Applications”, 2nd Edition, A K Peters Ltd.Natick, Massachusetts, 2015.

ii.Reference Books:

1. Donabel Santos, “Tableau 10 Business Intelligence Cookbook”, Packt Publishing, 1786465639, 9781786465634, 2016.

iii.Online References:

1. “Data Science for beginner”, Accessed on April.11.2021[Online]. Available: <https://bookdown.org/BaktiSiregar/data-science-for-beginners/Visualization.html>
2. “Exploratory data analysis and Data visualization”, Accessed on April.11.2021 [Online]. Available: <https://www.creative-wisdom.com/teaching/WBI/EDA.shtml>.
3. “Visualization of Multivariate Data”, Accessed on April.11.2021 [Online].<https://people.stat.sc.edu/hansont/stat730/MultivariateDataVisualization.pdf>

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS215	DEEP LEARNING	3	0	2	4

A.Preamble

In this course students will be gain knowledge on how to build and train neural network architectures such as Convolutional Neural Networks, Recurrent Neural Networks, LSTMs, and Autoencoders. Understand the capabilities, challenges, and consequences of deep learning compared to Machine learning and prepare students in the development of leading-edge AI technology.

B.Prerequisite Course

10212CS223 - Machine Learning Techniques

C.Course Objectives

Students are able to

- Understand basic concepts of deep learning Framework
- Understand CNN, RNN and GAN architecture models
- Perform model optimization and deployment

D.Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Explore data processing techniques using various deep learning frameworks for classification.	K3/S3
CO2	Build and train Convolutional Neural Network (CNN) model for real time applications.	K3/S3
CO3	Use various Convolutional Neural Network architectures for model building.	K3/S3
CO4	Demonstrate Recurrent Neural Networks and LSTM Networks	K3/S3
CO5	Choose the appropriate optimization technique to improve the accuracy of a model.	K3/S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E.Correlation of COs with Program outcomes and Programme Specific Outcomes:

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	3	3		2						3		
CO2	3	2	2	2	2	2						3	2	
CO3	3	3	3	3	2	2						3	2	
CO4	3	3	3	3	2	2						2	2	
CO5	3	3	3	3	3	2						2	2	

3- High; 2-Medium; 1-Low

F.Course Contents**Unit 1- Data Processing and Deep Learning Framework****L-9Hours**

Introduction to Data Handling – Data processing and Normalization using Deep Learning libraries and Data Augmentation Techniques. Deep Learning Frameworks – PyTorch, TensorFlow and Keras. Deep Learning Parameters – Epoch, Learning rate, Batch size, Open-source deep learning tools.

Unit 2 – DeepConvolutional Neural Networks (CNN)**L-9Hours**

Deep learning Layers – Convolutional Layer- Convolutional Properties and operations, Pooling Layer – Role and its types. Activation Layer – Significance and its types. Optimizers – Stochastic gradient, RMSProp, Ada Delta and Adam optimizers. Loss Function – Mean Square Error (MSE), Binary Cross Entropy, Categorical cross Entropy and Sparse Categorical cross Entropy.

Unit 3- Convnet Architecture and Transfer Learning**L-9Hours**

Introduction to Convnet - Architectures – AlexNet, VGG, Inception, ResNet - Training a Convnet: weights initialization, batch normalization, hyperparameter optimization, Addressing Overfitting and Underfitting issues. Interpretation of CNN – Localization Maps - Heat Maps - GradCAM. Transfer Learning, Advance architectures – Generative Adversarial Networks (GANs), Autoencoders, One shot learning, Siamese Networks and Capsule networks.

Unit 4 – Recurrent Neural Networks (RNN)**L-9 Hours**

Introduction to Recurrent Neural Networks (RNN)–RNN architecture, Building RNN step by step, Gated Recurrent Unit (GRU), Long-Short Term Memory (LSTM), LSTM architecture, Bi-LSTM. Language Models- Word embedding, Word-Level RNNs. Advance Networks – GPT3, BERT and Transformer models.

Unit 5- Optimization Techniques**L-9Hours**

Optimization in deep learning – Model Optimization, Non-convex optimization for deep networks- Stochastic Optimization. Generalization in neural networks- Spatial Transformer Networks.

Total: 45 Hours**G.LaboratoryExperiments:****Total:30 Hours****Part - 1**

Task 1	Implementation of a Binary classification model for grey-scale image (Framework: PyTorch).
Task 2	Implementation of a Convolutional Neural Network(CNN) classification model. (Framework: PyTorch).
Task 3	Construct a Convnet model for image classification. (Framework: PyTorch).
Task4	Application of Transfer Learning to build an image classification model (Framework: PyTorch).
Task5	Implement a simple Generative Adversarial Network (GAN) (Framework: PyTorch).

- Task6** Implementation of word embedding using RNN
(Framework: PyTorch).
- Task7** Implementation of Autoencoders
(Framework: PyTorch).
- Task8** Implementation of prediction using LSTM networks
(Framework: PyTorch).
- Task9** Implementation of object detection and localization
(Tool: OpenCV).
- Task10** Apply optimization techniques to improve model performance.
(Framework: PyTorch).

Part-2

Use Cases:

Use Case1: MNIST data classification

Use Case2: Dog vs Cat image classification

Use Case3: Synthetic image generation

Use Case4: Synthetic music generation using Autoencoder

Use Case5: Time series forecasting

Use Case6: Car detection using YOLO

Use Case7: Model optimization for Fruit data classification

[All experiments can be executed using standard dataset from Kaggle or ImageNet]

Total:75 Hours

H.Learning Resources

i. Text Books:

1. I Goodfellow, Y Bengio and A Courville, "Deep Learning", 1st Edition MIT Press, 2017. [Unit 1-5]

ii. Reference Books:

1. Venkatesan, Ragav, and Baoxin Li. "Convolutional neural networks in visual computing: a concise guide". CRC Press, 2017.[Unit-2,3]
2. Nikhil Buduma., "Fundamentals of Deep Learning", O'Reilly Media, 2017. [Unit-4,5]

iii. Online References:

1. "Deep Learning Specialization". Accessed on: April 20, 2021 [Online]. <https://www.coursera.org/specializations/deep-learning>.
2. "Deep Learning Nanodegree", Accessed on: April. 16, 2021 [Online]. Available: <https://www.udacity.com/course/deep-learning-nanodegree--nd101>

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS216	NATURAL LANGUAGE PROCESSING	3	0	2	4

A.Preamble

Natural language processing deals with written text. Students will learn how to process written text from basic of fundamental knowledge starts with Finite automata, Regular expression and probabilistic model with n-grams. This course also covers basis of semantic analysis and discourse analysis and drives it to machine translation

B.Prerequisite Courses

10211CS102-Data Structures

C.Course Objectives

Student will be able to

- Provide the student with knowledge of various levels of analysis involved in NLP
- Understand the applications of NLP
- Gain knowledge in automated Natural Language Generation and Machine Translation
- Understand Deep learning concepts for NLP.

D.Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Comprehend the fundamental concepts of Natural Language Processing.	K3/S3
CO2	Use the concept of Probabilistic model of defining language and POS tagging for NLP applications	K3/S3
CO3	Develop the Syntactic parser and Semantic Parser for Translation for word forms	K3/S3
CO4	Interpret the concept of text analysis, summarization and extractions for Information Extraction.	K3/S3
CO5	Apply the concepts of machine translation and deep learning for NLTK Modelling.	K3/S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E.Correlation of COs with Program outcomes and Programme Specific Outcomes:

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO 1	PSO 2	PSO 3
CO1	3	3	3	1	3						2	3		
CO2	3	3	3	2	3						1	3		
CO3	3	3	2	3	3	3					2	3		
CO4	3	3	2	3	3	3					2	3	2	2
CO5	3	3	3	3	3	3					3	3	3	3

3- High; 2-Medium; 1-Low

F.Course Contents

Unit 1 Introduction o Natural Language Processing

L-9 Hours

History of NLP, Components of NLP, Applications of NLP, Phases of NLP, NLP APIs, NLP Libraries, Knowledge in Speech and Language processing, Ambiguity and models and algorithm, language and understanding, brief history. Regular Expressions, Extracting Terms from Tokens, Vector Space Representation and Normalization, Similarity Computation in Text

Unit 2 Language Modelling And Part-Of-Speech (PoS) Tagging

L-9 Hours

Unigram Language Model, , Counting Words in Corpora, Simple (Unsmoothed) N-grams, Smoothing, Back off, Deleted Interpolation, N-grams for Spelling and Pronunciation, Entropy Natural Language Generation, Parts of Speech Tagging, Morphology, Named Entity Recognition, Overview of Hidden Markov Models.

Unit 3 Words And Word Forms

L-9 Hours

Context-Free Grammars for English, Lexicalized and Probabilistic Parsing, Semantic Analysis, Lexical Semantics, WordNet: A Database of Lexical Relations, Bag of words, skip-gram, Continuous Bag-Of-Words, Embedding representations for words Lexical Semantics, Word Sense Disambiguation, Knowledge Based and Supervised Word Sense Disambiguation

Unit 4 Text Analysis, Summarization And Extractions

L-9 Hours

Sentiment Mining, Entity Linking, Text Classification, LDA & Factorization - Text Summarization, Information Extraction, Named Entity Recognition, Relation Extraction, Question Answering in Multilingual Setting; NLP in Information Retrieval, Vector space model, Cross-Lingual IR

Unit 5 Machine Translation And Deep Learning

L-9 Hours

Need of MT, Problems of Machine Translation, MT Approaches, Statistical Machine Translation (SMT), Parameter learning in SMT (IBM models) using EM, Encoder-decoder architecture, Neural Machine Translation Statistical Language Models, Kernel Methods, Word-Context Matrix Factorization Models, Neural Language Models, Recurrent Neural Networks, Sentiment Analysis, Opinion Mining.

Total :45 Hours

G.Laboratory Experiments

Total : 30 Hours

Part - 1

Task 1	Introduction to Python & NLP • Get familiar with Python programming • Practice NLTK and libraries Tools: Python, NLTK, spaCy, Libraries
Task 2	To perform Word Analysis • To tokenize a text using the `transformers` package • To tokenize text with stopwords as delimiters Tools: Python, NLTK & spaCy
Task 3	To analyze algorithms for Word Generation • Single Layer LSTM Model Tools: Python, NLTK & spaCy

- Task4** Morphology is important factor for word Embedding. Develop algorithms for finding morphology of Word Documents.
Tools: Python, NLTK & spaCy
- Task5** Develop N-Grams for Word Document
 - Text generation using N-gram model**Tools: Python, NLTK & spaCy**
- Task6** To perform N-Grams Smoothing in Word Documents
Tools: , Python
- Task7** To perform the POS Tagging using Hidden Markov Model
Tools: Pandas, Python NLTK & spaCy, Gensim
- Task8** Use POS Tagging using Viterbi Decoding
To perform t log-linear model for PoS tagging problem . Compare the performance of log-linear model with HMM's performance
Tools: Python NLTK & spaCy, Gensim
- Task9**
 - To Building POS Tagger for unstructured Web Documents
 - Extracting some particular information form an unstructured text**Tools: PyTorch, Keras**
- Task10** To perform Chunking for Web Documents
 - Content Chunking
 - Designing Precise
 - Engaging Web Pages**Tools: PyTorch, Keras**
- Task11** To build Build Chunker for Web Documents
 - Content processing
 - Extracting information from text
 - Mining web pages**Tools: PyTorch, Keras**
- Task12** Python Data Visualizations Libraries
Tools: matplotlib, seaborne

Part-2

Use Cases:

- Use Case 1: Game Winning Prediction (Cricket)
 Use Case 2: Machine Translation from English-Hindi.
 Use Case 3: Query Expansion for Information Retrieval.
 Use Case 4: Emotion detection for texts.
 Use Case 5: Semantic Search Engine for Indian Languages.
 Use Case 6: Large Hindi Corpora
 Use Case 7: Crisp Query relevant Summary of each retrieved webpage
 Use Case 8: Semantic Query-Webpage Relevance

Total:75 Hours

H.Learning Resources

i.Text Books:

1. Daniel Jurafsky, James H. Martin, “Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech”, Pearson Publication, 2014.(All 5 Units)
2. Steven Bird, Ewan Klein and Edward Loper, —Natural Language Processing with Pythonl, First Edition, O‘Reilly Media, 2009.

ii.Reference Books:

1. Breck Baldwin, “Language Processing with Java and LingPipe Cookbook”, Atlantic Publisher, 2015
2. Charu C.Aggarwal, “Machine Learning for Text”- by Springer,2018 edition
3. Nitin Indurkha and Fred J. Damerau, —Handbook of Natural Language Processing, Second Edition, Chapman and Hall/CRC Press, 2010.

iii.Online References:

1. “Natural Language Processing”, Mar. 2015. Accessed on: Apr. 16, 2021 [Online]: https://onlinecourses.nptel.ac.in/noc19_cs56
2. “Natural Language Processing” Jun 2016. Accessed on: Apr. 16, 2021 [Online]. <https://www.udacity.com/course/natural-language-processingnanodegree>
3. “Natural Language Processing” Jul. 2, 2018. Accessed on: Apr. 16, 2021 [Online]. <https://www.coursera.org/learn/language-processing>

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS217	DATA SCIENCE	2	0	2	3

A.Preamble

This course provides an introduction to data science and highlights its importance in business decision making. It provides an overview of commonly used data science tools along with spreadsheets, relational databases, statistics, and programming assignments to lay the foundation for data science applications.

B.Prerequisite Course

10210MA103 - Probability, Statistics and Queuing Theory

C.Course Objectives

Learners are exposed to:

- Identify general statistical techniques for data analysis
- Determine how to summarize data to present information using descriptive statistics
- Use a correlation chart to determine the strength of a correlation.

D.Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Explain the life cycle of data science process for building real world applications	K2/S3
CO2	Understand different types of data description for data science process	K2/S3
CO3	Gain knowledge on relationships between data	K2/S3
CO4	Use the Python Libraries for Data Wrangling	K3/S3
CO5	Apply visualization Libraries in Python to interpret and explore data	K3/S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E.Correlation of COs with Program outcomes and Programme Specific Outcomes:

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1 1	PSO2 2	PSO3 3
CO1	3	3	3									2		
CO2	2	3	2	3				1	1			2	2	
CO3	2	3	2	3				1	1			2	2	
CO4	3	3	2	3				1	1			3	2	
CO5	3	3	2	3				1	1			3	2	

3- High; 2-Medium; 1-Low

F.Course Contents

Unit 1 Introduction

L-6 Hours

Introduction to Data Science – Evolution of Data Science – Data Science Roles – Life cycle of Data Science Project - Data Science: Benefits and uses – facets of data – Data Science Process: Overview – Defining research goals – Retrieving data – Data preparation – Exploratory Data analysis – Descriptive –diagnostic-Predictive-Prescriptive- feature engineering –algorithm selection- Build the model– algorithm tuning - presenting findings and building applications

Unit 2 Describing Data

L-6 Hours

Types of Data - Qualitative –Quantitative – Categorical –Nominal –ordinal –numerical -discrete – continuous –interval –ratio- Types of Variables –Univariateanalysis – Bivariateanalysis – Multivariateanalysis-Describing Data with Tables and Graphs –Measures of variability - Describing Data with Averages – Describing Variability – Normal Distributions and Standard (z) Scores

Unit 3 Describing Relationships

L-6 Hours

Covariance - Correlation –positive – negative –simple – partial – multiple - linear – non-linear –Pearson correlation - spearman ranking- Scatter plots – box plot – cross table - Histogram –correlation coefficient for quantitative data –computational formula for correlation coefficient – Regression - Simple and Multiple Regression –regression line – best fit line –least squares regression line- Polynomial Regression- – ridge – lasso regression - Regression assumptions –Standard error of estimate – Mean Square Error –root mean square error -Mean Absolute error/deviation–mean absolute percentage error - interpretation of r^2 –adjusted R square - multiple regression equations –regression towards the mean

Unit 4 Python Libraries

L-6 Hours

Basics of Numpy arrays –aggregations –computations on arrays –comparisons, masks, boolean logic – fancy indexing – Slicing of arrays –concatenate – reshape –broadcasting – transpose -structured arrays – Multidimensional –numpy universal functions - Pandas series – data frame –describe –modifying datatypes - duplicates- drop- windowing operations- date time- time series -Data manipulation with Pandas – data indexing and selection – operating on data – missing data – Hierarchical indexing – combining datasets – aggregation and grouping –merge –sorting - pivot tables

Unit 5 Model Development

L-6 Hours

Binary Classification – Multiclass Classification – Multi target – Decision Tree –Gini / information gain – root node – branch node –leaf node - interpretation of decision tree- Pruning – Model Evaluation metrics – Accuracy – confusion matrix – Precision –recall – AUC ROC curve - - Residual Plot – Distribution Plot –degree of polynomial – Bias – Variance –under fitting - overfitting low bias and low variance –Generalised model - Pipelines – Measures for In-sample Evaluation – Prediction and Decision Making – Imbalanced classification – under sampling – over sampling - Smote– Hyper parameter tuning – Best parameter – Cross validation – Grid and Randomised search cross validation- deployment – monitoring -MLOps

Total: 30 Hours

G. Laboratory Experiments**Total: 30 Hours****Part – I****Task 1:** Explore python modules for data collection and preparation**Exploratory Data Analysis****Task 2:** Write programs to perform descriptive statistics, subset of dataset.**Task 3:** Write programs to perform exploratory data analysis: variance, standard derivation, summarization, distribution and statistical inference.**Task 4:** Write programs to find the data distributions using box and scatter plot, outliers using plot on sample dataset**Task 5:** Write programs to plot the data using X-Y graph, Bar- chart, histogram, piechart and using other plotting techniques on sample dataset**Task 6:** Find the correlation matrix, covariance, plot the correlation plot on dataset and visualize giving an overview of relationships among data on sample dataset.**Model Building, Evaluation and Visualization****Task 7:** Write a program to build, evaluate the model using regression for a sample dataset. Apply multiple regressions, if data have a continuous independent variable**Task 8:** Write a program to build, evaluate the model using ridge and lasso regressions for a sample dataset.**Task 9:** Write a program to build, evaluate the model using binary classification for a sample dataset. Evaluate the performance of classifier.**Task 10:** Write a program to build, evaluate the model using Decision Tree for a sample dataset. Evaluate the performance of classifier.**Part – II****Use Case 1:** Let us use the built-in dataset air quality which has Daily air quality measurements in NewYork, May to September 1973. Create a histogram by using appropriate arguments for the following statements.

- a. Assigning names, using the air quality data set.
- b. Change colors of the Histogram
- c. Remove Axis and Add labels to Histogram
- d. Change Axis limits of a Histogram
- e. Create a Histogram with density and Add Density curve to the histogram

Use Case 2: Create a dataset or table [‘Smart Phone’] in an excel sheet that stores the mobile information [price, company name, model, SalePercent] of five different companies. Store at least 20 rows. Write the scripts and find out the output for the following information.

- a. Maximum price of the mobile of each company
- b. Minimum price of mobile of each company
- c. Average price of mobile of each company
- d. Total Price of mobile of each company

Use Case 3: The Tooth Growth data are from a study which examined the growth of teeth in guineapigs (n=10) in response to three dose levels of Vitamin C (0.5, 1, and 2 mg), which was administered using two delivery methods (orange juice or ascorbic acid). Data from the Tooth Growth Study is available as an R dataset and information about this study can be found by using R help.

- a. How many rows are there is Tooth Growth?
- b. What is the mean and standard deviation of Tooth length
- c. Which treatment is the best in terms of tooth growth? Derive the findings based on correlation between Dosage and Length for both supplements.

Use Case 4: Predict the sales of any product by performing the following

- a. Data collection from any source
- b. Data cleaning
- c. Model Building, Evaluation and Visualization

Use Case 5: Perform the Customer Churn analysis for a banking application with the following.

- a. Data collection from any source
- b. Data cleaning
- c. Model Building Evaluation and Visualization

Total:60 Hours

H.Learning Resources

i.Text Books:

1. David Cielen, Arno D. B. Meysman, and Mohamed Ali, “Introducing Data Science”, Manning Publications, 2016. (Unit I)
2. Robert S. Witte and John S. Witte, “Statistics”, Eleventh Edition, Wiley Publications, 2017. (Units II and III)
3. Jake VanderPlas, “Python Data Science Handbook”, O’Reilly, 2016. (Units IV and V)

ii.Reference Books:

1. Allen B. Downey, “Think Stats: Exploratory Data Analysis in Python”, Green Tea Press, 2014.

iii.Online References:

1. “Introduction to Data Science”, April. 11. 2021. Accessed on April. 22.2021 [Online]. Available: <https://rafalab.github.io/dsbook/>
2. "Introducing Data Science" 2016, Accessed: April 20, 2021, [Online]. Available: <http://bedford-computing.co.uk/learning/wp-content/uploads/2016/09/introducing-data-science-machine-learning-python.pdf>

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS218	MODELING FOR DATA SCIENCE	3	0	2	4

A. Preamble

Modelling is the process of applying statistical analysis to a dataset. A statistical model is a mathematical model of observed data. This course applies statistical models to the data by investigating, understanding and interpreting the information more strategically. Rather than sifting through the raw data, this practice allows the students to identify the relationships between variables, make predictions about future sets of data, and visualize the data so that non-analysts and stakeholders can consume and leverage it.

B. Prerequisite Course

10210MA103- Probability, Statistics and Queuing theory

C. Course Objectives

Learners are exposed to:

- Apply statistical methods to hypotheses testing and inference problems.
- Interpret the results in a way that addresses the question of interest.
- Use data to make evidence-based decisions that are technically sound.

D. Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Apply fundamental statistical modeling techniques to interpret linear regression model	K3/S3
CO2	Perform least square procedure to find the best fit for set of data points	K3/S3
CO3	Build interference in Linear regression to fit a linear equation to observed data	K3/S3
CO4	Analyze the linear regression and predict the data through R studio	K3/S3
CO5	Apply the Regression analysis and investigate the assumptions made about the data and the model	K3/S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of COs with Program outcomes and Programme Specific Outcomes:

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO ₁	PSO2	PSO3
CO1	3	2	1	1								3	3	
CO2	3	3	3	3	3	1	1	1	1	1		3	3	
CO3	3	3	3	3	3	1	1	1	1	1		3	3	
CO4	3	3	3	3	3	1	1	1	1	1		3	3	
CO5	3	3	3	3	3	1	1	1	1	1		3	3	

3-High; 2-Medium; 1-Low

F. Course Contents

Unit 1 Introduction To Statistical Models

L - 9 Hours

Frameworks and Goals of Statistical Modelling - The Assumption of Concept Validity - The Linear Regression Model - Matrix Representation of the Linear Regression Model - Assumptions of Linear Regression - The Appropriateness of Linear Regression - Interpreting the Linear Regression Model I - Interpreting the Linear Regression Model II

Unit 2 Linear Regression Parameter Estimation

L - 9 Hours

Introduction to Least Squares - Linear Algebra for Least Squares - Deriving the Least Squares Solution - Regression Modelling in R: a First Pass - Justifying Least Squares: The Gauss-Markov Theorem and Maximum Likelihood Estimation - Sums of Squares and Estimating the Error Variance - The Coefficient of Determination - The Problem of Non-identifiability - Regression Modelling in R: a Second Pass

Unit 3 Inference In Linear Regression

L - 9 Hours

Motivating Statistical Inference in the Linear Regression Context - The Sampling Distribution of the Least Squares Estimator - T-Tests for Individual Regression Parameters - T-Tests in R - Motivating the F-Test: Multiple Statistical Comparisons - The F-Test - The F-Test in R - Confidence Intervals in the Regression Context

Unit 4 Prediction And Explanation In Linear Regression Analysis

L - 9 Hours

Differentiating Prediction and Explanation - Point Estimates for Prediction - Interval Estimates for Prediction - Making Predictions Using Real Data in R - When Prediction Goes Wrong - Defining Causality.

Unit 5 Regression Diagnostics

L - 9 Hours

Linear Regression Diagnostic Methods - Violations of the Linearity Assumption - Violations of the Independence Assumption - Violations of the Constant Variance Assumption - Violations of the Normality Assumption - Diagnostics in R

Total: 45 Hours

G. Laboratory Experiments

Total: 30 Hours

Part - 1

Task 1	Implement the Linear Regression model to predict the salary. Consider the dataset with the following attributes i)Year of experience ii)Qualification iii)Salary. Tool: Python
Task 2	Implement the Linear Regression model to predict the age of the person. Consider the dataset with the following attributes i)Height ii)Weight iii)Gender Tool: Python

Task 3	Do a least squares regression with an estimation function defined by use of direct inverse method. Tools: RStudio, Python
Task4	Implement the linear regression model that uses a straight line to describe the relationship between the variables. i) Use only one independent variable in simple linear regression ii) Use two or more independent variable in Multiple linear regression Tools: RStudio, Python
Task5	Perform the test in statistics by use of t.test () in R, to determine the means of two groups are equal to each other in normal distribution Tools: RStudio, Python
Task6	Implement the F-Test in R Programming and calculates the ration between the larger variance and the smaller variance Tools: RStudio, Python
Task7	Build a linear regression model to predict that stopping distances of cars on the basis of the speed. Tools: RStudio, Python
Task8	Use the predict () function to make predictions from that model on new data. The new dataset must have all of the columns from the training data, but they can be in a different order with different values Tools: RStudio, Python
Task9	Implementation of regression diagnostics by using R base functionplot () or the autoplot() function. Tools: RStudio, Python
Task10	Conduct the test using R's build in function called Durbin Watson test to check the assumption of independence. Tools: RStudio, Python

Part-2

Use Cases:

Use Case 1: I wanted to see if there was a significant linear relationship between the number of fish meals consumed per week and the total mercury levels found amongst fishermen. The dataset used in this analysis is attached as an appendix item at the end

of the article. Since we have data between two variables only, I looked at applying a simple linear regression model to the dataset in question.

Let us use the Linear Regression in R to evaluate the extent of a linear relationship between two variables in a dataset using R.

Use Case 2: You're planning a road trip to Las Vegas with two of your best friends. You start off in San Francisco and you know it's going to be a ~9h drive. While your friends are in charge of the party operations, you're in charge of the all the logistics involved. You have to plan every detail: the schedule, when to stop and where, make sure you get there on time.

Prepare a dataset and use Linear Regression in R to find How much money should allocate for fuel? With the attributes of Total Payed, Total Miles.

Total:75 Hours

H.Learning Resources

i.Text Books:

1. Douglas C. Montgomery, Elizabeth A. Peck, G. Geoffrey Vining, “*Introduction to Linear Regression Analysis*”, Wiley Publication, Fourth edition, 2013
2. Alvin C. Rencher, G. Bruce Schaalje, “*Linear Models in Statistics*”, Wiley Publication, Second edition, 2007.

ii.Reference books:

1. James D.Miller, “*Statistics for Data Science*”, Packt Publishing, 2017.

iii.Online References:

1. “Statistical Modelling for Data Science Applications Specialization”, by coursera
<https://www.coursera.org/specializations/statistical-modeling-for-data-science-applications>

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS219	CYBER SECURITY	3	0	2	4

A. Preamble

Cyber Security is the body of technologies, processes, and practices designed to protect networks, computers, and data from attack, damage, and unauthorized access. This course provides the skills in cyber security in view of cybercrime, cyber offenses, frauds in mobile and wireless devices, handling techniques of cybercrime, organizational implications and cyber terrorism.

B. Prerequisite courses

10211CS130–Fundamentals of Computer Networks

C. Course Objectives:

Learners are exposed to:

- Assess threat models and their influence on an organization
- Compare the various uses and approaches to cryptography
- Prepare for and respond to security incidences
- Design effective and efficient password schemes
- Plan environments that are resistance to malware

D. Course Outcomes:

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Understand the basic concepts of cyber security and its principles.	K2/S3
CO2	Interpret the attacks, security issues, and management standards in wireless devices.	K2/S3
CO3	Apply the tools and methods to detect cyber attacks	K3/S3
CO4	Identify the phishing techniques and associated vulnerabilities.	K3/S3
CO5	Utilize the preventive measures for cyber security safeguards.	K3/S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of COs with Program outcomes and Programme Specific Outcomes:

Cos	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO1 0	PO1 1	PS O 1	PSO 2	PSO 3
CO 1	2	2				2	2	1						
CO 2	2	3		3		2	2	1						
CO 3	1	2		2	3	2	2	2						3

CO 4	1	3			3	2	2	2						3
CO 5	2		3	2	3	2	2	2						3

High- 3; Medium-2; Low-1

F.Course contents

Unit 1 Introduction

L-9Hours

Introduction: Definition and Scope- Risks- Threats- Classifications of Cybercrimes- The Legal Perspectives: Indian Perspective- Global Perspectives. Cyber offenses: Categories of Cybercrime- Social Engineering and its classification- Cyberstalking- Cybercafe and Cybercrimes- Botnets.

Unit 2 Cybercrime:Mobile and Wireless Devices

L-9 Hours

Trends in Mobility: Credit Card Frauds in Mobile and Wireless Computing Era, Security Challenges Posed by Mobile Devices, Authentication Service Security, Attacks on Mobile/Cell Phone - Identity and access management – Architecture – IAM Standards.

Unit 3 Tools and Methods used in Cybercrime

L-9 Hours

Introduction: Proxy Servers and Anonymizers, Password Cracking: Online Attacks, Offline Attacks, - Control Mechanisms - Preventive and Detective controls - Metasploit framework - Keyloggers and Spywares: Software Keyloggers, Hardware Keyloggers, Anti-keylogger, Spywares.

Unit 4 Phishing and Organizational Implications

L-9 Hours

Phishing and Identity Theft: Phishing, Types of Phishing - Phishing Countermeasures - Virus- Worms- Trojan Horse-Identity Theft (ID Theft)- Cybersecurity vulnerabilities – vulnerabilities in software and system administration- Web threats for organizations

Unit 5 Cyberterrorsim

L-9 Hours

Introduction- Access control- Audit- Authentication- Biometrics- Cryptography- Deception- Denial of Service Filters- Ethical Hacking- Firewalls- IDS- IPS- Scanning- Security policy- Threat Management.

Total: 45 Hours

G. Laboratory Experiments

Total: 30 Hours

Part – 1

- Task 1** Installations of VM work station and Kali Linux Operating System.
- Task 2** Identify the vulnerabilities and exploit attack vectors on a target device by scanning different parameters to accomplish common platform enumeration and common vulnerabilities exposure using legion in kali linux
- Task 3** Retrieving PII of a host such as a registrant name, organization, country, name servers, and date of creation, expiry, and updation.
- Task 4** Port Scanning using network mapper (nmap) in kali linux to expose the active and non-active ports and its associated vulnerabilities
- Task 5** Managing network traffic using wireshark tool.
- Task 6** Apply steganography attack using OpenStego.
- Task 7** Create and verify the virus file.
- Task 8** Terminate and Stay Resident (TSR) program analysis for identify the virus.
- Task 9** Introduction to operations on Metasploit framework and Scripting for penetration testing – Recon and Enumeration scripts
- Task 10** Configuring Windows and Linux firewall and managing user permissions

Part-2

Use Cases:

Use Case 1: A National Vehicles Insurance company contacting you for assistance regarding a cyber incident raised by the company. The company customers' data leaked to the internet and some investigations indicate that their Customer Management System which runs on IP Address was targeted by somebody and had stolen critical data. *Using your penetration testing skills & techniques, find the system weaknesses & investigate the incident.*

Use Case 2: In a Denial-of-Service (DoS) attack, threat actors flood a targeted system's network by directing lots of traffic towards it, from multiple systems under their control. It is a common tactic used by attackers via a network of compromised systems to render an online service unusable. DoS attacks can end up hurting an organization's reputation by affecting its services uptime, customer activity, and business operations. The motives behind DoS attacks can also include extortion, hacktivism, cyber warfare, etc., Explain the process to Detect and respond to break DOS attacks.

Total:75 Hours

H.Learning Resources

i.Text Books:

1. Nina Godbole, SumitBelapure, "Cyber Security", Willey India, Edition 1, 2012.
2. B. B. Gupta, D. P. Agrawal, Haoxiang Wang, "Computer and Cyber Security: Principles, Algorithm, Applications, and Perspectives", CRC Press, ISBN 9780815371335, 2018.
3. Raef Meeuwisse, "Cyber Security for Beginners", Cyber Simplicity Ltd., 2017.

ii.References Books:

1. Gary M. Jackson, "Predicting Malicious Behavior: Tools and Techniques for Ensuring Global Security", John Wiley & Sons Publisher, June 2012.
2. Roger Grimes, "Hacking the Hacker", Wiley India, 2017.
3. Donaldson, S., Siegel, S., Williams, C.K., Aslam, A., "Enterprise Cybersecurity -How to Build a Successful Cyber defense Program against Advanced Threats", A-press, 2015.

iii. Online References

1. The Complete Cyber Security Course: Hackers Exposed", 2018. [online]. Available: <https://www.udemy.com/the-complete-internet-security-privacy-course-volume-1/>, [Accessed: May 2022].
2. Cyber Security Pentesting Tools, [online]. Available: <https://www.cybrary.it /0p3n/7-cyber-security-pentesting-tools/>, [Accessed: May 2022].
3. Cyber security insights, [online], Available: <https://www.sans.org/reading-room/>, [Accessed: May 2022]

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS220	ANIMATION ENGINEERING	3	0	2	4

A. Preamble

The objective of the course is to provide the students an insight into the basics and the more advanced skills to masterfully navigate and create their own works of animation. At the end of the course, students will be able to create their own characters and animation scenes. Students able to acquire knowledge to create, develop, and execute animation sequences.

B.Prerequisite Course

10212CS244 –Computer Graphics

C.Course Objectives

Learners are exposed to:

- To familiarize the students with various approaches, methods, styles and techniques of animation technology, and how to apply.
- To enable students to manage Animation Projects from its Conceptual Stage to the final Product creation.
- To develop competencies and skills needed for becoming an effective Animator.
- Enriching the student skill set to meet professional expectations necessary for a career in the expanding industry of entertainment.

D.Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Understand the history of animation films and different types of animation techniques	K2/S3
CO2	Apply the design principles, styles and theories to animation production	K3/S3
CO3	Create a 2D animation starting from developing a script to final output	K3/S3
CO4	Develop 3D animation content by using 3D animation concepts	K3/S3
CO5	Use required software and hardware tools for Animation making	K3/S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E.Correlation of COs with Program outcomes and Program Specific Outcomes

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3		2		1			1	1				1	
CO2	3		3		2			2	1				2	
CO3	2		3		3			1	1				2	2
CO4	3		3		3			1	1				2	2
CO5	3		2		2			1	1				2	2

3- High; 2-Medium; 1-Low

F.Course Contents

Unit 1 Introduction to Animation

L-9 Hours

Introduction to graphics and animation – History of animation– Disney’s 12 Principles of Animation – Application of Graphics and Animation – Different types of Animation – Animation Techniques – Future trends of computer animation.

Unit 2 Styles in Animation

L-9 Hours

Introduction to the equipment required for animators: The animator’s drawing tools – the animation table – Perspective in animation – Styles in Animation –Animation Making Process –Pre–production phase – Production Phase – Post Production Phase.

Unit 3 2D Animation

L-9 Hours

Introduction to 2D animation–Pre visualization–Developing the idea–Writing the script – Storyboarding–2D animationTechniques–CelAnimation –Traditional 2D Animation –Digital 2D animation – Voiceover and music clips

Unit 4 3D Animation

L-9 Hours

Introduction to 3D animation – Modeling:3D Modeling basics – Types of 3D Modeling – Introduction to Modeling Techniques–Texturing: Texture Mapping –Texture making– Lighting: Types of Lights – Lights attributes – Rigging and Animation –Composition –Rendering.

Unit 5 Hardware and Software Tools and Industry Trends

L-9 Hours

Choosing a Computer – Using Monitors/Displays– Using 3D Scanners– Finding Data Storage Solutions – Choosing Software – Comprehensive 3D Animation Packages – CAD – Adobe Photoshop – Blender – 3Ds Max – Maya – Adobe Animate–Using Motion Capture–Creating Stereoscopic 3D – Integrating Point–Cloud Data – Providing Real–Time Capabilities – Working in Virtual Studios.

Total: 45 Hours

G.Laboratory Experiments

Total: 30 Hours

Part–1

Task 1 Conceptualizing and creating the storyboard:

- Prepare a storyboard for a given concept.
- Identify the environment, objects, and Characters
- Design an External and Interior Background

Tool: Adobe Photoshop

Task 2

- Create animation that showcases Weighted object lifting
- Create an Animation involving basic principle like Bouncing Ball
- Create an Animation involving basic principle like ball rolling on wooden ramp.

Tool: Blender/Pencil 2D

Task 3

- Animate a Greeting Card, Create an animated advertisement.
- E– content Creation : Use animation to explain a concept
- Design an Interactive Presentation, Animation Portfolio

Tool: Blender

Task4

For the given 3D animation conceptperform the following operation,

- Create a suitable 3D model from primitive objects.
- Apply various transformations to the 3D model.
- Apply modifiers to bring realistic 3D model.

Tool: Blender

Task5

For the developed 3D model perform the following operation,

- Create an image texture to apply on the 3D model

b) Import the texture to the 3D animation tool

c) Apply texture on the 3D model

Tool: Blender

Task 6 Animate the objects in following way:

Moving as a whole object

a) Changing their position,

b) Orientation or size in time

c) Animating their vertices or control points

Tool: Blender

Task 7 Inherited animation

- Causing the object to move based on the movement of another object

Tool: Blender

Task8 Render the Animated 3D Model/Character Using following render methods.

- Extra Easy Virtual Environment Engine (Eevee)
- Blender's physically-based path tracer for production rendering (Cycles)
- The Workbench Engine

Tool: Blender

Task 9 For the rendered 3D model/character perform the following operations.

a) Compositing – Assemble video clips together and enhance them

b) Motion Tracking & Masking – perform masking and tracking with the Movie Clip Editor

Tool: Blender

Task 10 To bring the final output perform the following operations

a) Combine multiple video channels and add effects to them

b) Add audio and synchronize the timing of the video sequence to match it–Video Sequencer

c) Display result of rendering Sequencer timeline–Sequencer preview

Tool: Blender

Part–2

Use Cases:

Case 1: Make use 2D animation to display aProduct demo

Case 2: Develop an explainer video for public awareness about a product or a service

Case1: **Building an Interactive 3D Animation System**

Develop a 3D animated video to illustrate the Basics of Computer Network

Case 2: **3D animation as an educational tool**

Use 3D animation to explain a complex subject that requires deep understanding.

H. Learning Resources

i. Text Books:

1. Richard Williams, “The Animators Survival Kit”, Barnes & Noble
2. Ande Beane, “3D Animation Essentials”, John Willey & Sons 2012
3. King R. “3D Animation for the Raw Beginner Using Autodesk Maya 2e”. CRC Press; 2019

ii. Reference Books:

1. Tinwell A. The uncanny valley in games and animation. AK Peters/CRC Press; 2014

iii. Online References:

1. “Beginning 2D Animation: The Absolute Beginner's Entry” Last updated 9/2018, [Online]. Available: <https://www.udemy.com/course/absolute-beginners-entry-into-animation/>
2. “2D Animation” [Online]. Available: <https://www.animatorisland.com/courses/2d-animation>
3. 3D Animation, [Online]. Available: <https://www.mooc-list.com/tags/3d-animation>
Getting started in 3D Animation: [Online]. Available: <https://www.udemy.com/course/learn-3d-animation-in-just-2-weeks/>
4. Introduction to 3d Animation Software, [Online]. Available: <https://www.educba.com/3d-animation-software/>

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS221	GAME DESIGN	3	0	2	4

A.Preamble

Game design is a field with a broad focus. This course is an introduction to the primary concepts of gaming, and an exploration of how these basic concepts affect the way gamers interact with games. In this course students will understand game design concepts, the mechanics, and rules behind different types of games. Students will be able to design and develop various games.

B.Prerequisite Course

10211CS102- Data Structures.

C.Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Understand the stages of the game design process	K2/S3
CO2	Develop the Game environment, Game objects, and Characters	K3/S3
CO3	Identify the various genres of games	K3/S3
CO4	Use Game Engines to develop Game User Interface	K3/S3
CO5	Build the Game from game engine based on the target platform	K3/S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

D.Correlation of COs with Program outcomes and Programme Specific Outcomes:

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3		2		2			1	1				1	
CO2	3	2	2		2			2	1				2	
CO3	3	2	3		3			2	1				3	
CO4	3	2	3		3			2	1				3	2
CO5	3	2	3		3			2	1				3	2

High-3; Medium-2; Low-1

E.Course Contents

Unit 1 Introduction to Game Design

L- 9 Hours

History of Game - Conventional Games Versus Video Games - The Key Components of Video Games - The Structure of a Video Game - The Stages of the Design Process - The Game Design Team Roles - The Anatomy of a Game Designer - Getting an Idea - From Idea to Game Concept - The Player's Role - Choosing a Genre - Defining Your Target Audience and Target Platform.

Unit 2 Game World and Character Design**L-9 Hours**

Introduction to Game World - The Purposes of a Game World - The Dimensions of a Game World - Self-Defining Play - Creative Play - Storytelling Play - Game Modifications - The Goals of Character Design - The Relationship Between Player and Avatar - Visual Appearances - Character Depth - Audio Design

Unit 3 Level Design and Genres of Game**L-9 Hours**

Game Balancing - General Principles of Level Design - Key Design Principles - Layouts - The Level Design Process - Pitfalls of Level Design - THE GENRES OF GAMES: Action Games - Strategy Games - Role-Playing Games - Sports Games - Vehicle Simulations - Adventure Games - Construction and Management Simulations - Artificial Life and Puzzle Games

Unit 4 User Interfaces design and Interaction Models**L-9 Hours**

User Interface- Player-Centric Interface Design - The Design Process - Managing Complexity - Interaction Models - Camera Models - Visual Elements- Audio Elements - Input Devices - Navigation Mechanisms Gameplay: Making Games Fun - The Hierarchy of Challenges - Skill, Stress, and Absolute Difficulty - Commonly Used Challenges

Unit 5 Game Development and Testing**L-9 Hours**

Working with audio – Saving player details and progress – 2d Vs. 3D game development – Single player Vs. Multiplayer games – Testing and refining – Game build – Target Platform – Future Trends: Cloud Gaming and Streaming - Next-Generation Consoles – VR and AR games – Mixed Reality Games.

Total: 45 Hours**G. Laboratory Experiments****Total:30 Hours****Part-1****Task 1 Conceptualizing and creating the storyboard:**

- Develop a storyboard for a given game concept.

Task 2 For the given game concept,

- Identify the Game environment, Game objects, and Characters
- Identify Target Platform, End Users, Types of game, Genre of Game, and Target Audients

Task 3 For the given game concept,

- Model the Game environment, Game objects, and Characters
- Integration of Game world with Game Engine

Tool: Blender, Unity 3D**Task 4**

- Levels design – number of Levels
- Core Mechanics Implementation

Tool: Unity 3D, Visual Studio**Task 5** Perform the following operations to implement the Scene Management

- Create multiple scenes
- Switching between the scenes – Using scene name and scene number

Tool: Unity 3D, Visual Studio**Task 6** Perform the following operations

- User Interface Design – Panels, Text, Buttons etc.,

Tool: Unity 3D, Visual Studio

- Task 7** Audio Design – Sounds and Music
- Apply Suitable Audio to all the scene
- Tool: Unity 3D, Visual Studio**
- Task 8** Saving player details and player's progress
- Save Player's information such as name, Id, Gender etc.,
 - Save Player's progress such as score, high score, level cleared etc.,
- Tool: Unity 3D, Visual Studio**
- Task 9** For the rendered 3D model/character perform the following operations.
- Testing and debugging
- Tool: Unity 3D, Visual Studio**
- Task 10** Conversion of game into Build (Android/IOS/Windows)
- Tool: Unity 3D**

Part-2

Use Cases:

Use Case 1: Design and Develop a puzzle game to improve communication skill of children

Use Case 2: Design and Develop a computer game for a given requirement

The basic idea behind the game “Bouncy Ball” is that the main screen displays a ball and the player has to click the ball which changes its position randomly. There are three lives given to the player. For every wrong click a life is reduced. As soon as the three lives are over the game gets over and the final score is displayed. There are three stages in the game that are Easy, Normal and Hard. On every stage the ball's appearing speed increases thus increases the difficulty level.

Total:75 Hours

H.Learning Resources

i.Text Books:

1. [Jesse Schell](#) - The Art of Game Design: A Book of Lenses, CRC Press, Third edition, 2020
2. [Joshua Bycer](#) - Game Design Deep Dive into Platformers, CRC Press, First edition 2020

ii.Reference books:

1. [John P. Doran](#) - Unity 2020 Mobile Game Development, Packt Publishing, Second Edition, 2020
2. Robert Zubek - Elements of Game Design, MIT Press, 2020

iii.Online References:

1. Beginner Design: Unity Game Development Course, Offered By Plural sight Company, <https://learn.unity.com/course/beginner-design-unity-game-development-course>

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS222	VIRTUAL AND AUGMENTED REALITY	2	0	2	3

A. Preamble

This course creates a foundational understanding of the differences between AR and VR. It provides the students with a broad overview of the rapidly evolving space and the tools to characterize future VR&AR developments and assess their potential and limitations. It establishes key concepts and dives into the device landscape, helping learners to differentiate current and future platforms and devices as well as critique the use of VR or AR in applications.

B.Prerequisite Course

10211CS111/10211CS118- Image Processing/ Image Processing Techniques

C.Course Objectives

Students are able to

- To introduce the relevance of this course to the existing technology through demonstrations, and applications with a futuristic vision along with socio-economic impact and issues.
- To understand virtual reality, augmented reality and using them to build applications/simulations.

D.Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K – Level
CO1	Understand the components of the Virtual Reality System	K2/S3
CO2	Construct the Geometrical Transformations and Understand Motion & Tracking of VR	K3/S3
CO3	Utilize the Augmented Reality and its components of the Augmented Reality System	K3/S3
CO4	Make use of Game Engines to develop various VR/AR Applications	K3/S3
CO5	Identify the usage of VR/AR Technology in various fields	K3/S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E.Correlation of COs with Program outcomes and Programme Specific Outcomes:

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO 1	PSO 2	PSO3
CO1	3	1	2		2			2	1					
CO2	3	2	3		3			2	1			2		
CO3	2	2	3		3			2	1			2		
CO4	2	1	3		3			2	1			2		
CO5	3	1	3		3			2	1			2		

High-3; Medium-2; Low-1

F.Course Contents

Unit 1 Introduction to Virtual Reality

L-6 Hours

Defining Virtual Reality - Human Physiology and Perception - Key Elements of Virtual Reality Experience - Virtual Reality System - Interface to the Virtual World - Input & output - Visual, Aural & Haptic Displays - Types of Virtual Reality - Components of Virtual Reality -Applications of Virtual Reality - challenges in VR.

Unit 2 The Geometry of Virtual Worlds, Motion & Tracking

L-6 Hours

3D Computer Graphics - Geometric Models - Changing Position and Orientation - Axis-Angle Representations of Rotation - Viewing Transformations - Chaining the Transformations - Motion in Real and Virtual Worlds - Physics in the Virtual World - Tracking Position and Orientation - Collision detection.

Unit 3 Introduction to Augmented Reality

L-6 Hours

Technology and features of augmented reality - difference between AR and VR - AR systems and functionality - Input & output - Visualization techniques for augmented reality - Types of augmented reality - Interaction for AR - Tracking for AR - AR challenges.

Unit 4 Development Tools And Frameworks In VR/AR

L-6 Hours

Frameworks of Software Development Tools in VR/AR- Game Engines (Unity and Unreal) - Steam VR, Oculus VR, AR Core, Vuforia - UI/UX Design - Virtual Environment Design, Choosing low polygonal 3d models for simulations - Choosing Target Platform - Testing and refining.

Unit 5 Applications and Future of VR/AR

L-6 Hours

APPLICATIONS OF VR/AR - Medical Applications, Education, Arts and Entertainment, Military, Manufacturing, Industrial Training, VR/AR application development – 3D modeling, C# Scripting Emerging Applications of VR/AR - AR in Shopping & Retail, Mobile VR/AR, VR/AR Games.

Total: 30 Hours

G.Laboratory Experiments

Part - 1

Task 1	Create a virtual environment and objects using Unity 3D Game Engine Tools: Unity 3D
Task 2	Create 3D models using Unity 3d Game Engine Tool: Unity 3D
Task 3	Develop a scene to implement transformations Tool: Unity 3D, Visual Studio
Task4	Develop a scene to implement Colliders, Object movement Tool: Unity 3D, Visual Studio
Task5	Apply Suitable Audio/Video to the created scene. Tool: Unity 3D, Visual Studio
Task6	Moving in the Virtual Environment (Teleportation), grabbing game objects Tool: Unity 3D, Visual Studio
Task7	Add multimedia content (text, image, video, animation) to the scene Tool: Unity 3D, Visual Studio
Task8	Developing different targets (Image, Model, Area) for AR application Tool: Unity 3D, Visual Studio
Task9	Testing and refining the project – Scenario-based Testing Tool: Unity 3D, Visual Studio
Task10	Conversion of VR/AR project into the application Tool: Unity 3D, Visual Studio

Part-2

Use Cases:

Use Case 1: Implement Virtual tour of campus using Unity 3D

Use Case 2: Develop an AR Application to showcase Human Anatomy

Total:30 Hours

H.Learning Resources

i.Text Books:

1. Steven M. LaValle - Virtual Reality, Cambridge University Press, 2020. [Unit 1&2]
2. Linda Daniela - New Perspectives on Virtual and Augmented Reality, Routledge, 2020[Unit 3,4,5]

ii.Reference Books:

1. Philipp A. Rauschnabel- Augmented Reality and Virtual Reality Changing Realities in a Dynamic World, Springer International Publishing, 2020
2. Erin Pangilinan, Steve Lukas, Vasanth Mohan - Creating Augmented and Virtual Realities: Theory and Practice for Next-Generation Spatial Computing, O'Reilly Media, First Edition, 2019

iii.Online References:

1. Introduction to XR: VR, AR, and MR Foundations, Offered By Unity Technologies, [Online]. Available: <https://learn.unity.com/course/introduction-to-xr-vr-ar-and-mr-foundations>

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS224	COMPUTER VISION	3	0	2	4

A.Preamble

This course offers broad area of computer vision algorithms, methods and concepts which will enable the student to implement computer vision systems with emphasis on applications and problem solving. Students will learn principles of image processing algorithms, image segmentation/classification techniques, 3D vision and 3D motion.

B.Prerequisite Courses

10211CS120/Machine Learning/ 10211CS223 –Machine Learning Techniques

Machine

C.Course Objective

Learners are exposed to

- Understand the basic concepts of computer vision and image formation.
- Learn shape detection techniques such as Hough Transform and RANSAC.
- Understand image segmentation and classification methods.
- Gain knowledge of 3D vision and object recognition techniques.
- Learn digital video processing techniques including motion detection and tracking

D.Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Apply filters and edge detection techniques to process the images	K3/S3
CO2	Use hough transform for line and object detection	K3/S3
CO3	Perform various segmentation techniques on images	K3/S3
CO4	Implement known methods to process 2D images to obtain 3D representations.	K3/S3
CO5	Explore state-of-the-art techniques for digital video processing	K3/S3

Knowledge Level (Based on revised Bloom's Taxonomy)

K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create

E.Correlation of COs with Program outcomes and Programme Specific Outcomes:

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO 1	PSO2	PSO 3
CO1	3	3	3		3							3		
CO2	3	3	3		3							3		
CO3	3	3	2		3							3		
CO4	3	3	2		3							3		
CO5	3	3	3		3							3		

3- High; 2-Medium; 1-Low

F.Course Contents

Unit 1 Introduction to Computer Vision

L-9 Hours

Introduction to Computer Vision, Fundamentals of Image Formation - Camera Imaging Geometry – Camera Calibration – Color Representations - Linear Filters and Convolution (Discrete/Continuous) – Gradient-Based Edge Detection - Vector Quantization and Textons – Synthesis by Sampling Local Models - Binocular Camera Geometry and the Epipolar Constraint

Unit 2 Hough Transform

L-9Hours

Line detection – Hough Transform (HT) for line detection – The Foot-of-Normal Method - RANSAC for straight line detection – HT based circular object detection – Unknown Circle Radius – speed problem – ellipse detection

Unit 3 Image Segmentation and Classification

L-9Hours

Shot Boundary Detection - Interactive Segmentation - Forming Image Regions - The Watershed Algorithm - Segmentation Using K-means - Encoding Layout with GIST Features - Summarizing Images with Visual Words - The Spatial Pyramid Kernel - Classifying images of single objects

Unit 4 3D Vision

L-9 Hours

Methods for 3D vision - Projection Schemes - Shape from Shading - Photometric Stereo – The Assumption of Surface Smoothness - Shape from Texture – Horaud's Junction Orientation Technique - 3D object recognition – Multiple View Vision

Unit 5 Digital Video Processing

L-9Hours

Video Sampling and Interpolation, Motion Detection and Estimation, Video Enhancement and Restoration, Video Quality Assessment, Video Segmentation, Motion Tracking, Video Surveillance

Total: 45 Hours

Laboratory Experiments

Total: 30 Hours

Part - 1

Task 1	Implement the linear and non-linear calibration algorithm. Tool: python
Task 2	Using the given spectra data set, fit a finite-dimensional linear model to a set of illuminants and surface reflectances using principal components analysis, render the resulting models, and compare your rendering with an exact rendering. Tool: python
Task 3	Detect the edges of the given image using gradient-based approach. Tool: python
Task4	Use k-means clustering algorithm to do vector quantization. Tool: python
Task5	Implement the non-parametric texture synthesis algorithm. Use your implementation to study: (a) the effect of window size on the synthesized texture; (b) the effect of window shape on the synthesized texture; and (c) the effect of the matching criterion on the synthesized texture (i.e., using a weighted sum of squares instead of a sum of squares, etc.). Tool: python

- Task6** Implement a mean shift segmenter.
Tool: python
- Task7** Implement the two version of graph-based segmenter discussed in the lecture hours. Use your graph-based segmenter to build an interactive segmentation system.
Tool: python
- Task8** Build a classifier that classifies materials using the given dataset. Compare the performance of your system using the main feature constructions described here (GIST features; visual words; spatial pyramid kernel). Investigate the effect of varying the feature construction; for example, is it helpful to use C-SIFT descriptors?
Tool: python
- Task9** Implement photometric stereo for the given set of similar images.
Tool: python
- Task10** Perform 3D object detection using ZED SDK.
Tool: python
- Task11** Process the 2D image to obtain a 3D image representation.
Tool: python
- Task12** For a moving object, detect its motion and track it using python and OpenCV.
Tool: python
- Task13** Compute the depth map of a given image programmatically
Tool: python
- Task14** Implement a Bayesian Model for optimizing the Video Quality.
Tool: python, OpenVeno

Part-2

Use Cases:

Use case 1: Human Iris Location for determining the gaze direction.

Use case 2: Face detection and recognition for security applications.

Use case 3: Surveillance in transport to monitor traffic and people.

Use case 4: Application of Computer Vision in Visual Hulls: Tracing Intersection Curves, Clipping Intersection Curves, Triangulating Cone Strips, Carved Visual Hulls.

G.Learning Resources

i. Text Books:

1. David A. Forsyth, Jean Ponce, “*Computer Vision - A Modern Approach*”, Pearson Education, Second Edition, 2003 (UNIT I, III)
2. E. R. Davies , "Computer Vision: Principles, Algorithms, Applications, Learning", Academic Press, Fifth Edition, 2018 (UNIT II, IV)
3. Alan C Bovik, “*The Essential Guide to Video Processing*”, Academic Press, Second Edition, 2009 (UNIT V)

ii. Reference Books:

1. Richard Szeliski, “*Computer Vision: Algorithms and Applications*”, Springer-Verlag London Limited, 2011
2. Simon J. D. Prince, “*Computer Vision - Models, Learning, and Inference*”, Cambridge University Press, 2012

iii. Online Resources:

1. Computer Vision, onlinecourses.nptel.ac.in/noc21_ee23/preview, 2021 [Online]

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS225	ETHICAL HACKING	3	0	2	4

A.Preamble

This course provides the skills in hacking techniques, methodologies, tools, tricks, and security measures to secure an organization's IT systems. It is a course that has become especially important in the present-day context and can help individuals and organizations to adopt safe practices and usage of their IT infrastructure. Starting from basic topics like networking, network security, and cryptography, the course will cover various attacks, vulnerabilities, and ways to secure them. There will be hands-on demonstrations that will be helpful to the students.

B.Prerequisite Courses

10211CS105–Computer Networks

C.Course Objectives

Students undergoing this course are expected to

- Understand the concepts of ethical hacking with terminologies and the laws criticizing those.
- Familiarize the tools to help competitive intelligence, DNS zone, and port scanning.
- Mitigate enumeration in the system and its vulnerabilities.
- Understand how to hack the server and applications, procure knowledge of attacks, and standards in technology.

D. Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Explain the various types of cyber-attacks and apply the security policies to protect against it.	K3/S3
CO2	Apply knowledge gained from reconnaissance and scanning to create a comprehensive network map.	K3/S3
CO3	Apply Windows and Linux operating systems specific enumeration techniques to find vulnerabilities.	K3/S3
CO4	Demonstrate various attacks on web application vulnerabilities and wireless networks.	K3/S3
CO5	Interpret the cyber laws and write the report of vulnerabilities present in assets, data, network and applications.	K3/S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of COs with Program outcomes and Programme Specific Outcomes:

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1		2		2	3		3						
CO2	2	3		3	3		3						
CO3	2	2		3	3		3						
CO4	3	3		3	3		3						

CO5	3	3		3	3		3					2	

3- High; 2-Medium; 1-Low

F. Course Contents

UNIT – 1 Ethical Hacking Overview

L-9 Hours

Introduction to Ethical Hacking– TCP/IP concepts - Essential Terminologies – Penetration Testing - Malicious Software – Protection against malware attacks – Physical Security – Security Policies and Procedures

UNIT – 2 Reconnaissance and Scanning

L-9 Hours

Reconnaissance and Footprinting – Active and Passive – Google Hacking - Social Engineering – Port Scanning – Types of Port Scanning – Port Scanning tools – Ping Sweeps – Scripting for vulnerability scan

UNIT – 3 Enumeration and System Hacking

L-9 Hours

Introduction to Enumeration – DNS Attacks -DNS Zone Transfers – NetBIOS Enumeration Tools- Windows and Linux Operating System Vulnerabilities – Metasploit framework- System Hacking – Malware Creation – Evading IDS, Firewall and Honeypot - Maintaining Access and Clearing Tracks.

UNIT – 4 Web Application and Wireless Hacking

L-9 Hours

Understanding Web Application – Web Application vulnerabilities – OWASP Top 10 Vulnerabilities – Session Hijacking - Code injection, SQL injection, cross-site scripting – Denial of service attacks - Wireless Technology – Wireless attacks – WEP/WPA vulnerabilities – Wifi Hacking.

UNIT – 5 Reporting and Cyber Law

L-9 Hours

Skills required for an ethical hacker – Incident Handling - CVE and CVSS – Report Writing – Laws of Land – Ethics Vs Law – Indian IT Policy 2000 – Compliance and Risk Assessment - Case Studies (Ransomware Attacks/Stuxnet/DataBreach/Pegasus).

Total: 45 Hours

G. Laboratory Experiments

Total: 30 Hours

Part - 1

TASK 1	Installation of VM work station and Kali Linux/Parrot Operating System.
TASK 2	OSINT Framework for gathering of information about the target system, network and human factors.
TASK 3	Gather the information about the network or webserver by using nslookup, whois lookup, Google dorks Tools: NSLookup, Whois, Google dorks
TASK 4	Scan and identify the network range, different types of ports open and vulnerabilities exist using Nmap Tool Tools: Nmap
TASK 5	Perform the vulnerability scan and identify the different types of vulnerabilities present in the network. Tool: Nessus
TASK6	Demonstrate proxy, intruder attack and Session Hijacking attack Tools: Burpsuite
TASK7	Perform Cross site scripting, and Injection attacks Tools: DVWA

- TASK8** Perform the automated SQL injection attack for the given target.
Tools: SQLMap
- TASK9** Perform Windows and Linux System Hacking
Tools: Metasploit
- TASK10** Malware Creation
Tools: Msfvenom

Part-2

Use Cases:

Use Case 1: A company wants to assess the security of its network infrastructure. As an ethical hacker, perform network reconnaissance to identify live hosts, open ports, and services using tools like Nmap. Provide a detailed report on the discovered information.

Use Case 2: A financial institution has an online banking portal. Conduct a penetration test on the web application to identify and exploit vulnerabilities such as SQL injection or cross-site scripting. Report on potential risks and recommend mitigation strategies.

Use Case 3: An organization is concerned about its employees falling victim to social engineering attacks. Conduct a simulated phishing campaign to assess the susceptibility of employees. Provide recommendations for improving awareness and training.

Use Case 4: A popular e-commerce website is vulnerable to Cross-Site Scripting attacks. Ethically hack the website to demonstrate how an attacker could inject malicious scripts into web pages, potentially stealing user credentials. Propose measures to mitigate XSS vulnerabilities.

Total: 75 Hours

H. Learning Resources

i.Text Books: Michael T. Simpson, Kent Backman, James Corley - Hands-On Ethical Hacking and Network Defense, Cengage Learning, Second Edition, 2012 (Unit I-III,V)

1. “Ethical Hacking and Countermeasures”, EC-Council, Publisher: Course Technology; 2nd edition, 2016. (UNIT-IV)

ii.Reference Books:

1. Allen Harper, Shon Harris, Jonathan Ness, Chris Eagle - Gray Hat Hacking The Ethical Hackers Handbook, McGraw Hill Education, Third Edition, 2017
2. Sean Philip Oriyano and Michael Gregg - Hacker Techniques, Tools and Incident Handling,
3. Jon Ericson, “Hacking: The Art of Exploitation”, Second Edition, No Starch Press, 2008, ISBN 978-1593271442
4. Michael Sikorski, Andrew Honig, “Practical Malware Analysis: The Hands-On Guide to Dissecting Malicious Software”, No Starch Press, 2012.

iii.Online References:

1. Grey Campus - “Ethical Hacking”. Accessed on: Apr. 20, 2021 [Online]. Available: <https://www.greycampus.com/opencampus/ethical-hacking>

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS228	BLOCKCHAIN TECHNOLOGY	2	0	2	3

A.Preamble

Blockchain technology has become popularly known because of its use in the implementation of Cryptocurrencies such as BitCoin, Ethereum. The technology itself holds much more promise in various areas such as time stamping, logging of critical events in a system, recording of transactions and trustworthy e-governance. This course covers the technical aspects of public distributed ledgers, blockchain systems, cryptocurrencies, and smart contracts. Students will learn how these systems are built, how to interact with them, how to design and build secure distributed applications.

B.Prerequisite Courses

10211CS109–Computer Networks

C.Course Objectives

Students undergoing this course are expected to

- Understand the concepts of ethical hacking with terminologies and the laws criticizing those.
- Familiarize the tools to help competitive intelligence, DNS zone, and port scanning.
- Mitigate enumeration in the system and its vulnerabilities.
- Understand how to hack the server and applications, procure knowledge of attacks, and standards in technology.

D.Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Explain Blockchain with types of consensus mechanisms and protocol	K2/S3
CO2	Illustrate the concepts of Cryptocurrency and Smart Contracts to build blockchain	K3/S3
CO3	Construct Public, Private and ConsortiumBlockchain System for real time implementation	K3/S3
CO4	Use Initial Coin Offering, securityand privacy in blockchain with Hyperledger Fabric	K3/S3
CO5	Implement Blockchain applications in real time using solidity.	K3/S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E.Correlation of COs with Program outcomes and Programme Specific Outcomes:

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO2	PSO3
CO1	2												
CO2	3	3									2		
CO3		3											
CO4					3								
CO5			3		3							3	

3- High; 2-Medium; 1-Low

F.Course Contents

Unit 1 Introduction to Blockchain, Types and Consensus Mechanism

L-6 Hours

Fundamentals of Blockchain- Introduction - Origin of Blockchain - Blockchain Solution - Components of Blockchain - Block in a Blockchain - Technology and the Future

Blockchain Types and Consensus Mechanism - Introduction - Decentralization and Distribution - Types of Blockchain - Consensus Protocol.

Unit 2 Cryptocurrency and Smart Contracts

L-6 Hours

Cryptocurrency – Bitcoin, Altcoin and Token -Bitcoin and the Cryptocurrency - Cryptocurrency Basics - Types of Cryptocurrencies - Cryptocurrency Usage

Smart Contracts - Characteristics of a Smart Contract - Types of Smart Contracts -Types of Oracles - Smart Contracts in Ethereum - Smart Contracts in Industry

Unit 3 Types of Blockchain

L-6 Hours

Public Blockchain System - Popular Public Blockchain - Bitcoin Blockchain -Ethereum Blockchain

Private Blockchain System - Key Characteristics of Private Blockchain – Need of Private Blockchain -Private Blockchain Examples- Private Blockchain and Open Source - Various Commands (Instructions) in E-commerce Blockchain -Smart Contract in Private Environment - State Machine -Different Algorithms of Permissioned Blockchain -Byzantine Fault - Multichain
Consortium Blockchain - Key Characteristics of Consortium Blockchain -Need of Consortium Blockchain - Hyperledger Platform -Overview of Ripple - Overview of Corda

Unit 4 Security and Application of Blockchain

L-6 Hours

Initial Coin Offering - Blockchain Fundraising Methods -Launching an ICO - Investing in an ICO - Pros and Cons of Initial Coin Offering - Successful Initial Coin Offerings - Evolution of ICO - ICO Platforms

Security in Blockchain - Security Aspects in Bitcoin -Security and Privacy Challenges of Blockchain in General - Performance and Scalability -Identity Management and Authentication Regulatory Compliance and Assurance -Safeguarding Blockchain Smart Contract (DApp) - Security Aspects in Hyperledger Fabric

Unit 5: Development Tools and Frameworks

L-6 Hours

Blockchain Ethereum Platform using Solidity - Remix IDE -Structure of a Smart Contract Program - Using Remix to Write and Run a Solidity Program – Modifiers – Events - Arrays in Solidity - Function Visibility - Variable Visibility - Function Modifier Keyword -How Funds Are Accepted - Fallback Function - Contract Inheritance -Contract Communicating with Another Contract - External Libraries - ERC20 Token Transfer - Error Handling in Solidity - Application Binary Interface (ABI) - Swarm (Decentralized Storage Platform) -Whisper (Decentralized Messaging Platform)

Total : 30 Hours

G.Laboratory Experiments

Part - 1

TASK 1 Creating MetaMask wallet and crypto

- Meta Mask Setup
- Deposit the Crypto
- Make a Transaction and Extracting the transaction details
- Ethereum Transaction

Tool: MetaMask

TASK 2 Implement the Smart Contract using Remix, Solidity

- Working With Variables
- Sending Money
- Starting, Pausing, Stopping and Deleting transaction
- Mapping and Struct
- Error Handling
- View/Pure, Receive Function and Fallback Function
- Inheritance, Modifier and Importing
- Events and Return Variables

Tool: Remix, Solidity

TASK 3 Installation and Configuration of Node.js and Web3.js

- Transfer Ether
- Interacting with Smart Contract
- Using Web3.js with Chrome to Interact with Smart Contracts

Tool: Node.js and Web3.js

TASK 4 Creating the Hyperledger Fabric using solidity

- Create the Fabric Test Network
- ringing up the test network
- Creating a channel
- Starting a chaincode on the channel
- Interacting with the network
- Bringing down the network

Tool: Node.js and Web3.js

TASK 5 Creating the Shared Wallet using solidity

TASK 6 Deploying MEW with Ganache

TASK 7 Establish the transaction using Remix with Ganache

Tool: Remix with Ganache

Part-2

Use Cases:

- **Use Case 1:** Retail - Establishing unconditional transparency in the food supply chain using blockchain Hyperledger fabric
- **Use Case 2:** Banking and Financial Services – Increasing transaction volume and network resilience while maintaining confidentiality requirements for real time gross settlement among different banks of a central banking consortium

- **Use Case 3:** Healthcare - Electronic health records and medical research data digitization
- **Use Case 4:** Energy and Utilities – Renewable energy trading which are locally owned deliver benefits that are environmentally relevant for the communities involved
- **Use Case 5:** Blockchain in Real-estate – The real estate market currently faces several issues related to housing affordability, rising rate and economy, many of which cannot be addressed with blockchain

Total: 60 Hours

H.Learning Resources

i.Text Books:

1. Chandramouli Subramanian, Asha A George, Abhilash K A and MeenaKarthikeyan, “Blockchain Technology”, Universities Press,2021(Unit1-Unit5)
2. Imran Bashir, “Mastering Blockchain: Distributed ledger technology, decentralization, and smart contracts explained”, Birmingham – Mumbai: Packt, 2nd Edition Kindle, 2018 (Unit 1&2)

ii.Reference Book(s):

1. Narayanan, Arvind, et al. Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction. Princeton University Press, 2016.
2. Antonopoulos, Andreas M. Mastering Bitcoin: Programming the Open Blockchain. O’Reilly Media, Inc., 2017.
3. Antonopoulos, Andreas M. and Wood, Gavin. Mastering Ethereum. O’Reilly Media, Inc., 2018.

iii.Web Resource(s):

1. Ethereum project documentation”, Jan 21 2016, Accessed on: Jan 21 2016 [Online]. Available: <https://ethdocs.org/en/latest/>
2. “Monero Research Lab papers” 01 April2014, Accessed on: 01 April 2014 [Online]. Available: <https://www.getmonero.org/resources/research-lab/>
3. “ZCash documentation”, 10 Jan 2021 Accessed on: 10 Jan 2021 [Online]. Available: <https://z.cash/>

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS229	IOT AND CLOUD COMPUTING	3	0	2	4

A. Preamble

This course provides an overview of the Internet of Things (IoT) and Cloud Computing concepts, infrastructures and capabilities. This will help students gain the necessary knowledge to construct IoT systems and use cloud services for processing and storage of the data produced by the IoT devices. Emphasis will be placed on the architecture and design of IoT systems, the different technologies governing system implementation and the migration of the data to the Cloud for processing.

B. Prerequisite Course:

10211CS129–Modern Computer Architecture

C. Course Objectives

Students are able to

- Know the IoT architecture stack.
- Understand the different IoT platforms
- Analyze the different IoT Protocols
- Learn the cloud computing basics
- Summarize role of Cloud Computing in IoT

D. Course Outcomes:

Upon the successful completion of the course, students will be able to:

CO Nos.	Course Outcomes	K-Level
CO1	Understand the various concepts of IoT and their technologies	K2/S3
CO2	Choose appropriate physical devices to develop IoT applications.	K3/S3
CO3	Use various protocols in IoT applications	K3/S3
CO4	Demonstrate the use of cloud technology in IoT applications.	K3/S3
CO5	Develop a smart application using IoT and Cloud.	K3/S3
K1-Remember; K2-Understand; K3- Apply; K4-Analyze; K5 – Evaluate; K6- Create		

E. Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	3	1	3	3		2					2	3
CO2	3	3	3	1	3	3		2					2	3
CO3	3	3	3		3	2		2					2	3
CO4	3	3	3		2	2		2					2	3
CO5	3	3	3	1	2	3		2				1	2	3

3- High; 2-Medium; 1-Low

F. Course Contents

Unit 1 IoT Introduction

L-9 Hours

Introduction to IoT – IoT – Characteristics – IoT Complete Architectural Stack – IoT enabling Technologies – IoT Challenges.

Case studies: Architectural analysis for IoT applications

Unit 2 Hardware Platforms For IoT

L-9 Hours

Sensors and Hardware for IoT – Hardware Platforms – Arduino, Raspberry Pi, Node MCU.

Case studies: Data acquisition from sensors with hardware platforms

Unit 3 Protocols For IOT

L-9 Hours

Protocols for IoT – Infrastructure protocol, Identification, Transport Layer Protocolos, Discovery, Data Protocols, Device Management Protocols.

Case Study: MQTT/CoAP usage in IoT applications

Unit 4 Cloud Computing

L-9Hours

Introduction to Cloud Computing - Service Model – Deployment Model- Cloud Platforms – Amazon AWS – Microsoft Azure – Google APIs.

Case Studies: Risk Issues, and Legal Aspects of Cloud Computing

Unit 5 IoT And Cloud

L-9 Hours

Role of Cloud Computing in IoT - Cloud Components - S3 – Lambda - IoT Core -Connecting a IoT applications to cloud Case Study: Cloud Data Security

Total: 45 Hours

G. Laboratory Experiments

Total: 30 Hours

Part-1

Task 1 : Familiarization with Arduino/Raspberry Pi/Node MCU and perform necessary software installation.

Task 2 : Familiarization with Fritzing and Thnikercad for design and fabrication of circuits.

Task 3 : To interface LED/Buzzer with Arduino/ Raspberry Pi and write a program to implement traffic light system.

Task 4 : To interface Digital sensor (IR/LDR) with Arduino/Raspberry Pi and write a program for object detection and find the distance between sensor and object.

Task 5 : To interface DHT11 sensor with Arduino/Raspberry Pi and write a program to find the weather conditions and control the air conditioner.

Task 6 : To interface motor using relay with Arduino/Raspberry Pi and write a program to control the direction and speed of the motor.

Task 7 : To interface Bluetooth/WiFi with Arduino/Raspberry Pi and write a program to send sensor data to smartphone using Bluetooth.

Task 8 : Write a program on Arduino/Raspberry Pi to push the data to cloud for detecting the fire event using temperature, light and smoke sensors.

Task 9 : Write a program on Arduino/Raspberry Pi to upload/retrieve/visualize the sensor data to thingspeak/thingsio cloud for predicting the future data.

Task 10 :Write a program on Arduino/Raspberry Pi to stream the sensor data with IoT core and use Bigquery to store data being streamed from IoT devices. Create Pub/Sub model for event driven systems.

Task 11 :Use Cloud IoT Core for IoT based environmental monitoring application.

Task 12 :Use Cloud IoT Core for IoT based Smart Light Controlling& Monitoring System.

Part-2

Task 13: Implement a smart dust bin to detect level of Garbage, Temperature, Humidity, Gases like Ammonia, Carbon Monoxide and send a message with GPS coordinates to garbage collector using GSM module.

Task 14: Implement a person centric health care management system to detect the medical events using IoT components and cloud

Task 15:Implement a Smart home with IoT components and cloud for occupant monitoring.

Total: 45 Hour

H. Learning Resources

ii. Text Books:

1. Alam, Mansaf, Kashish Ara Shakil, and Samiya Khan, eds. Internet of Things (IoT): Concepts and Applications. Springer Nature, 2020. [Unit 1-3]
2. Mangla, Monika, Integration of Cloud Computing with Internet of Things: Foundations, Analytics and Applications. John Wiley & Sons, 2021. [Unit 4-5]

iii. Reference Books:

1. Lakhwani, Kamlesh, et al. Internet of Things (IoT): Principles, Paradigms and Applications of IoT. Bpb Publications, 2020.
2. Parikshit N. Mahalle, Nancy Ambritta P., Gitanjali Rahul Shinde, Arvind Vinayak Deshpande, The Convergence of Internet of Things and Cloud for Smart Computing, CRC Press, 2021.

iii. Online Resources:

1. “IoT (Internet of Things) Wireless & Cloud Computing Emerging Technologies” Accessed on: April. 16, 2021 [Online]. Available: <https://www.coursera.org/lecture/iot-wireless-cloud-computing/2-1-iot-architecture-part-1-K6pdR>
2. “Practical IoT Concepts-Devices, IoT Protocols & Servers”, Accessed on: April. 16, 2021 [Online]. Available: <https://www.udemy.com/course/practical-iot-devices-protocols-servers/>
3. “Arduino Tutorials”, Accessed on: April. 16, 2021 [Online]. Available: <https://www.arduino.cc/en/Tutorial/HomePage>
4. “Thingspeak IoT Cloud”, Accessed on: April. 16, 2021 [Online]. Available: <https://thingspeak.com/>
5. “Tinkercad Tutorials”, Accessed on: April. 16, 2021 [Online]. Available: <https://www.tinkercad.com/>
6. Industrial IoT on Cloud platform, Accessed on April. 16, 2021 [Online]. Available: [https://www.coursera.org/learn/iiot-google-cloud-platform\(online\)](https://www.coursera.org/learn/iiot-google-cloud-platform(online)): NPTEL.

COURSECODE	COURSE TITLE	L	T	P	C
10212CS230	FORENSICS IN CYBER SECURITY	3	0	2	4

A. Preamble

This course will address methods to properly conduct a computer forensics investigation. It includes digital evidence collection and evaluation of network and host system intrusions with hands-on use of powerful forensic analysis tools.

B. Prerequisite Courses

10211CS130 – Fundamentals of Computer Networks
10211CS103-Operating systems

C. Course Objectives

Learners are exposed to

- Introduction to Digital Forensics and Evidences
- Network Forensics
- Forensic Approaches and Forensic Duplication
- Digital Forensic Tools
- Incident Response

D. Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K-Level
CO1	Outline and Explore the fundamentals of cyber forensic analysis.	K2/S3
CO2	Infer and conduct basic network forensic analysis	K3/S3
CO3	Classify the evidence acquisition and forensic duplication	K2/S3
CO4	Evaluate and validate the types of Forensic tools	K2/S3
CO5	Detect cyber security incidents and its response	K2/S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02	PS03
CO1	2	1		1								2		
CO2	1	1	1	1	1	1	1					2		
CO3	1	2				2	3				2		2	
CO4	2		2		2						1			2

CO5		3		3	3			2						

3 -High, 2 – Medium, 1-Low

F. Course Outcomes

Unit1 Forensic Science

L-9Hours

Principles and Methods: Scientific approach to Forensics, Identification and Classification of Evidence, **Location of Evidence:** Storage Media, Hard drives, Hardware Interfaces, Recovering Data, Media File Forensic Steps-**Forensic Analysis:** Planning, Case Notes and Reports, Quality Control.

Unit2 Network Forensic

L-9Hours

Network forensics overview-Securing a Network-Developing procedures for network forensics-Investigating virtual networks-Examining Honeynet projects-E-mail Investigations: Role of client and server in E-mail, Investigating E-mail crimes and violations, E-mail Servers, E-mail Forensic tools.

Unit3 Digital Forensic Approaches and Forensic Duplication

L-9Hours

Introduction-Policy and Procedure Development-Evidence Assessment Evidence Acquisition – Evidence Examination - Documenting and Reporting- **Introduction to Forensic Duplication** - Rules of Forensic Duplication (Thumb Rule) - Necessity of Forensic Duplication - Forensic Duplicates as Admissible Evidence - Important Terms in Forensic Duplicate - Forensic Duplication Tool Requirements - Creating a Forensic Duplicate of a Hard Drive-Creating a Qualified Forensic Duplicate of a Hard Drive.

Unit4 Forensics Tools

L-9Hours

Evaluating Forensics Tool Needs-Tasks performed by forensics tools-**Forensics Software Tools:** Command-line forensic tools, Linux forensic tools-**Forensics Hardware Tools:** Forensic workstation, Write-Blocker-Validating and Testing Forensics Software.

Unit5 Incident Response

L-9Hours

Incidence Response Goals of Incident Response-People Involved in Incident Response Process - Incident Response Methodology - Activities in Initial Response - Phases after Detection of an Incident –Report Writing and Presentation

Total: 45Hours

G. Laboratory Experiments**Total:30Hours****Part-1**

TASK1	Study of Computer Forensics and different tools used for forensic investigation
TASK2	To extract Exchangeable image file format (EXIF) Data from Image Files. Tool-ExifreaderSoftware
TASK3	Forensics Case Investigation using live data Tool-Autopsy
TASK4	To Recover Deleted Files using Forensics Tools Tool-FTK/File Analyzer
TASK5	To Find Last Connected USB on your system Tool-Parse
TASK6	To View Last Activities on Your PC Tool-Register Editor/Registry Explorer
TASK7	To analyze network-related incident Tool-Wireshark
TASK8	To make the forensic image of the hard drive to find the suspicious activity Tool-EnCase
TASK9	To Restore the Evidence Image to find criminal activity Tool-EnCase
TASK10	To hide and extract any text file behind an image file/Audio file using Command Prompt. Shellcomments/ShellBagsExplorer
TASK11	To capture the physical memory of a computer and analyze artifacts in memory Tool-Magnet RAM Capture
TASK12	To Collect Email Evidence in Victim PC Tool-Email Analyzer

Part-2**Use cases1:**

You are on-site, conducting a preliminary examination of a Linux system. The hardware suite includes a 56KB modem. What areas of search should be included in your examination? Prepare an examination plan that details what you will look for, and why.

Use cases2:

You receive a Windows OS X system and are asked to summarize the applications and data on the hard drive. In addition, you are asked to report any recent system usage and any signs of encryption, external storage media, or clock tampering.

Use cases3:

As a security investigator, you have been asked to determine if company confidential information (intellectual property) has been copied from enterprise computers. Investigation centers on a particular computer that has shown a high volume of network traffic at unusual times. In the course of conducting your investigation you discover that large capacity removable media has been attached to the suspect computer. A preview examination reveals that software used for secure

deletion had been downloaded to the desktop. Prepare an investigative plan listing the lines of investigation that you plan to pursue.

Usecases4:

A large accounting MNC company is going to audit certain activities by officers of a medium size, publicly traded bank. During the investigation, the appointed auditor needed to examine several computer systems used by certain Bank employees. After collecting the evidence, the digital forensic examiners were immediately dispatched and sent in to arrange for the formal investigation of those systems to search for corroborating evidence in support of the audit team's suspicions and findings.

Usecases5:

Discover you have known the existence of threatening e-mails being sent to the CEO of your company at that time predict will you do the examination of the e-mails revealed that they originated from outside of the country? Describe the steps you would take in your investigation. In particular, address the issue of jurisdiction and locating your counterparts in the target country.

H.Learning Resources

i. Text Books:

1. Chuck Eastom, "Certified Cyber Forensics Professional Certification, McGraw Hill, July 2017. (Unit 1)
2. Bill Nelson, Amelia Philips and Chris Steuart, "Guide to Computer Forensics and Investigations", Cengage Learning, 6th Edition, 2019 (Unit 2)
3. Greg Gogolin, "Digital Forensics Explained", CRC Press Taylor and Francis Group, 2013 (Unit 3, 5)
4. Nilakshi Jain and Dhananjay R Kalbande, "Digital Forensic: The Fascinating World of Digital Evidences", Wiley Press, 2017 (Unit 4)

ii. Reference Books:

1. Nhlen-AnLe-Khac, "Security, Privacy, and Digital Forensics in the Cloud" Wiley Press, 2019, ISBN-13: 978-1119053286 ISBN-10: 1119053285.
2. Nihad A. Hassan "Digital Forensics Basics A Practical Guide Using Windows OS", Alpha Press Publishers, 2019.
3. Mike Sheward "Digital Forensic Diaries" Secure owl, 2017, ISBN: 9781521514467.
4. Gerard Johansen, "Digital Forensics and Incident Response A practical guide to deploying digital forensic techniques in response to cyber security incidents".

iii. Online References:

1. <https://online.norwich.edu/academic-programs/resources/5-steps-for-conducting-computer-forensics-investigations>
2. <http://resources.infosecinstitute.com/computer-forensics-tools>
3. <http://www.cybrary.it/course/computer-hacking-forensics-analyst>

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS239	3D CHARACTER MODELLING	3	0	2	4

A. Preamble

3D Character Modelling will be able to transform a concept/idea into a three-dimensional model, with a desire to learn modern character modelling and sculpture workflows. It helps the students to blend the organic and hard surface modelling concepts into a cohesive design with a basic understanding of prop modelling.

B. Prerequisite Course

10211CS220 - Animation Engineering

C. Course Objectives

- Aims to equip students with the fundamental knowledge required for the development of 3D character models for interactive digital environments
- The course also aims to provide practical sessions, to enable students explore the theoretical knowledge and express their ideas as animated 3D Character Models

D. Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Develop a 3D Character model using modeling techniques	K3/S3
CO2	Identify the use of surface modeling concepts in 3D Character modeling	K3/S3
CO3	Use Materials to Improve the Visuals of 3D character	K3/S3
CO4	Build textures for 3D characters	K3/S3
CO5	Apply Rigging and Animation techniques on 3D Character model	K3/S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of COs with Program outcomes and Programme Specific Outcomes:

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO 1	PSO2	PSO 3
CO1	2		2		2			1	1					2
CO2	2		2		2			1	1					2
CO3	2		3		2			2	1					2
CO4	2		3		2			2	1					2

CO5	2		3		2			2	1					2

3- High; 2-Medium; 1-Low

F. Course Contents

Unit 1 Modeling

L-9 Hours

Introduction to Modeling - 3ds Max/Blender Modeling interface - Poly Modeling – Low poly Vs High Poly– Using Primitives – Modelling with Modifiers – Subdivision Surfaces

Unit 2 Surface Modeling

L-9 Hours

Patch Grid – Patch modelling, Compound Objects Modelling – 2D to 3D - 3D to 2D - Introduction to NURBS - NURBS Splines and Surfaces - Working with NURBS Curves and Surfaces - NURBS Curves Modelling - NURBS Surface Modelling

Unit 3 Material Creation

L-9 Hours

External References – Objects and Scenes - File Formats - Materials – 2D Maps - 3D Maps - Compositors and Compound Materials – Blend - Composite - Double Sided Materials - Smoothing Polygons - Subdivide Polygons

Unit 4 Texture Mapping

L-9 Hours

Working with poly sub-object - modeling a Low poly Character model for Game Design - Creating Texture for object model - UVs and Mapping - Material using UVW Mapping - Unwrap Mapping - UVW Mapping - Edit UVW modifier

Unit 5 Rigging and Animation

L-9 Hours

Rigging techniques - Setting Biped Animation for character model - Rendering Techniques – Arnold - Art - Scan line – Ray tracer, Radiosity

Total: 45 Hours

G. Laboratory Experiments

Total:30 Hours

Part – 1

Conceptualizing and creating the storyboard:

TASK 1

- Prepare a storyboard for a given concept.
- Identify the 3D Character's Properties

For the given 3D animation/Game concept perform the following operation,
Use various transformations and modifiers to bring realistic 3D Character.

TASK 2

- Model the Head
- Model the Arms/Legs
- Model the Eyes
- Model the Body

Tool: Blender

- TASK 3** For the developed 3D model perform the following operation,
 d) Create an image texture to apply on the 3D Character
 e) Import the texture to the 3D animation tool
 f) Apply texture on the 3D Character
Tool: Blender
- TASK 4** For the developed 3D character perform the following operation,
 a) Create Joints - Leg Joints, Arm Joints, Backbone Joints, Head and Neck Joints.
 b) Setting Up Controls - Leg Controls, Arm Controls, Root and Hip Controls, Shoulder Controls and Neck Control
 c) Finishing up the Controls
 d) Finishing up the rig
Tool: Blender
- TASK 5** Animate the 3D character in following way:
 Moving as a whole object
 d) Changing their position,
 e) Orientation or size in time
Tool: Blender
- TASK 6** Animate the 3D character in following way:
 ByDeforming them
 a) Animating their vertices or control points
Tool: Blender
- TASK 7** Inherited animation
 • Causing the object to move based on the movement of another object
Tool: Blender
- TASK 8** Render the Animated 3D Model/Character Using following render methods.
 • Extra Easy Virtual Environment Engine (Eevee)
 • Blender's physically-based path tracer for production rendering (Cycles)
 • The Workbench Engine
Tool: Blender
- TASK 9** For the rendered 3D Character perform the following operations.
 c) Compositing - Assemble video clips together and enhance them
 d) Motion Tracking & Masking - perform masking and tracking with the Movie Clip Editor
Tool: Blender
- TASK 10** To bring the final output perform the following operations
 d) Add audio and Lip sync to the Developed 3D character
 e) Display result of rendering Sequencer timeline - Sequencer preview
Tool: Blender

Part - 2

Use Cases:

Use Case 1: A Game Development company requested to model a 3D female character for their 3D car game. As a 3d modeler design and model the 3D character using any 3D modeling tool.

Use Case 2: A film production company needs a 10 years old boy character for a 3D animation movie. As a 3D modeler design and model the 3D character using any 3D modeling tool.

H. Learning Resources

i. Text Books:

1. Bill Culbertson. “3Ds Max Basics; for Modeling Video Game Assets Model, Rig and Animate Characters for Export to Unity or Other Game Engines”. CRC Press; 2021

ii. Reference Books:

1. Oliver Villar “Learning Blender: A Hands-On Guide to Creating 3D Animated Characters” Addison-Wesley, 2017

iii. Online References:

1. “Blender: Character Modelling – Easy Beginner’s Guide”, [Online]. Available:<https://all3dp.com/2/blender-2-8-character-modeling-simply-explained>
2. “Blender 3D Modeling Tutorials for Beginners: The Ultimate Collection”, [Online]. Available: <https://conceptartempire.com/blender-modeling-tutorials>

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS244	COMPUTER GRAPHICS	3	0	2	4

A. Preamble

Computer graphics courses enable students to create 2D and 3D designs, use multimedia techniques and study advances in the field.

B. Prerequisite Courses

1150MA101-Linear algebra for Computing

C. Course Objectives

Students are able to understand

- An understanding of the physical and geometrical principles used in computer graphics
- An understanding of rendering algorithms, and the relationship between illumination models and the algorithms used to render them
- An understanding of the basic techniques used to model three dimensional objects, both as surfaces and as volumes
- An acquaintance with the principles of interaction and of user interfaces

D. Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Apply various algorithms for 2D primitives and Transformations	K3/S3
CO2	Use the basics concepts of Illumination and various projection techniques	K3/S3
CO3	Experiment with Illumination model and basic representation of image	K3/S3
CO4	Explore the abstractions of models for specifying complex objects and introduction to Multimedia.	K3/S3
CO5	Illustrate about the types of compression and advanced multimedia environments	K3/S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of COs with Program outcomes and Programme Specific Outcomes:

[illegible]

CO3	2	3			2							2		
CO4	2	2			2							2		
CO5	2	2			2							2		

High-3; Medium-2; Low-1

F. Course contents

Unit 1 Review of Graphics Fundamentals

L-9 Hours

Basic raster graphical algorithm for 2D primitives, Line drawing algorithm: Bresenham's algorithm, Midpoint line algorithm, DDA algorithm, Circle drawing algorithm: Bresenham's algorithm, Midpoint circle algorithm, Ellipse drawing algorithm.

Unit 2 Transformation and Projection

L-9 Hours

2D,3D transformations: Window to Viewport transformations, Clipping algorithm, Bezier curve, b-spline curve, surfaces and Solid modeling. Parallel projection-Perspective projection, Visible surface determination techniques for visible line determination, Z-buffer algorithm, scanline algorithm, visible surfaces ray-tracing and recursive ray tracing, radiosity methods.

Unit 3 Shading and Representation

L-9 Hours

Illumination model, Specular reflection model, Shading models: Transparency, Shadows, Object reflection, Gouraud & Phong shading, Recursive ray tracing and path tracing. Light Sources – RGB, CMYK and HLS Color Models, Image representation and manipulation, Enlarging and Shrinking images, Texture mapping.

Unit 4 Advance Modelling and Multimedia

L-9 Hours

Procedural models, fractals models, Grammar-based models, multi-particle system, volume rendering, vsoftware applications and software environments – Media types – Analog and digital video, digital audio and animation - Memory storage – Basic tools – Authoring tools.

Unit 5 Elements in Multimedia

L-9 Hours

Building blocks text - hypertext - sound - sound cards - standards - image - image types – image compression - fractal and wavelet compressions - image file types - animation - capture and playback techniques. Multimedia environments, multimedia programming, advanced multimedia moving pictures - techniques realistic image synthesis, virtual reality - full motion digital video – video capture techniques – multimedia networks

Total: 45 Hours

G. Laboratory Experiments

Part - 1

- TASK 1** Using of Bresenham's algorithm to draw a line using Bresenham's algorithm.
Tools: Code Block
- TASK 2** Draw a rectangle using line drawing algorithm with C/C++ program.
Tools: Code Block
- TASK 3** Implementation of Circle drawing algorithms in C/Java: Bresenham's Algorithm, Mid-Point Algorithm
Tools: Code Block
- TASK 4** Develop a C/C++ program to draw a two-dimensional object and perform translation, rotation, and scaling transformations.
Tools: Code Block

- TASK5** Write a program to perform 2D Transformation on a line.
- TASK6** Write a program to perform shear transformation on a rectangle.
- TASK7** Write a program to draw a car using in build graphics function and translate it from bottom left corner to right bottom corner of screen.
- TASK8** Write a program to draw a cube using in build library function and perform 3D transformations
- i) Translations in x, y, z directions
 - ii) Rotation by angle 450 about z axis, rotation by 600 about y-axis in succession.
 - iii) Scaling in x-direction by a factor of 2, scaling in y- direction by a factor of 3
- TASK9** Write a C/C++ program to draw a cube using in build library function and perform 3D transformations
Tools: Code Block
- TASK10** To Implementa C/C++ program for implementation of Visualizing Projections of 3d Images.
Tools: Code Block
- TASK11** Implement the following fundamental operation on the given input image, such as
- i) How to read and write the image.
 - ii) Convert the given image into binary image.
 - iii) Convert the RGB image into gray scale image.
 - iv) Resize the given image into particular different pixel size.
 - v) Display pixel intensity values of the given image.
- Tools: Matlab, Open CV**
- TASK12** Implement a MATLAB program for Morphological Operations on Binary Images.
Tools: Matlab, Open CV

Part-2

Use Cases:

Use case 1: To create a primitives object and apply any transformation.

Use case 2: Using Flash/Maya perform different operations (rotation, scaling move etc..) on objects

Use case 3: Design and develop simple graphics programs using basic graphics functions defined in “graphics.h”

Use case 4: To Study various in build graphics functions in C library.

Use case 5: Design and implement game / animation clip / Graphics Editor using open source graphics library. Make use of maximum features of Object Oriented Programming.

Total:75 Hours

H.Learning Resources

i. Text Books:

1. David J.Eck, Hobart and William Smith, “Introduction to Computer Graphics”, Version 1.2, 2018. (Unit 1)
2. Hearn & Baker, “Computer Graphics C Version”, 2nd ed. Pearson Education, 2012.(Unit 2-3)
3. Foley, Van Dam, Feiner, and Hughes, “Introduction to Multimedia”: Principles and Practice, 2nd edition in C, Addison-Wesley(Unit 4-5).

ii. Reference Books:

1. DaveShreiner, Graham Sellers, John M. Kessenich, Bill M. Licea-Kane ,”OpenGL Programming Guide: The Official Guide to Learning OpenGL, Version 4.3”, 8th Edition, ARB working group.
2. Hearn and Baker, “Computer Graphics using open GL”, 3rd edition, Pearson Education, 2009.

iii. Online References:

1. “Computer Graphics”, Co-ordinated by IIT Guwahati Available from 6th May 2020 [Online]. Available: <https://nptel.ac.in/courses/106/103/106103224/>
2. “Fundamentals of Graphic Design”, Offered by CALARTS, Available from 22nd April 2021 [Online]. Available: <https://www.coursera.org/learn/fundamentals-of-graphic-design>
3. “Computer Graphics”, Available from Aug. 27, 2017 [Online]. Available: <https://www.slideshare.net/smrutisarangi2/computer-graphics-notes-79192682>

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS250	IDENTITY ACCESS MANAGEMENT	3	0	2	4

A. Preamble

This course focuses on managing and maintaining Identity of the users in an organization and operations. The course helps the learners to understand the access management and role engineering procedures and policies and their applicability over various contexts and situations. The identity and access management's necessity in a cloud environment are also dealt in the course along with its challenges and operations.

B. Prerequisite Courses

10211CS105/10211CS130 – Computer Networks/Fundamentals of Computer Networks

C. Course Objectives

Learners are exposed to

- Understand the basic concepts in identity and access management
- Identify the identity based restrictions created over various organization
- Utilize the access control principles and procedures to get acquainted with role engineering process
- Infer the policies and governance associated with identity and access management
- Identify the necessity of identity and access management in cloud environment

D. Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Understand the concepts and functionalities of identity and access management	K2/S3
CO2	Identify the standards and management of identities in IAM environment	K3/S3
CO3	Utilize the Access management controls and principles for end user access management	K3/S3
CO4	Infer the necessity of Policies and governance in identity and access management	K2/S3
CO5	Identify the importance of Cloud based Identity and access management and its standards	K3/S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of Cos with Program Outcomes and Programme Specific Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PS O1	PS O2	PS O3
CO 1		2												
CO 2		2	2	2	3									
CO 3		2	2	2	3									
CO 4		2	2		3									
CO 5		2			3									

High- 3; Medium-2; Low-1

F. Course Contents**Unit 1 Introduction****9 Hours**

Cybersecurity and its necessity – Context based Cyber security – Types of Cyber threats - Introduction to Identity and access management - IAM Concepts and terms - IAM Architecture - IAM Operational activities – User Identification, Authentication and Authorization –Roles in identity and access management – Identity vs Access management – Functional aspects of IAM – Key Service Components of IAM – IAM Risks - Drivers for Identity and access management

Unit 2 Identity Management**9 Hours**

Digital Identity and ID creation – Identity life cycle - Authentication: Security assertion markup language - Open ID - Open ID Connect Protocol Suite –Types of Digital authentication - Joiner, mover, leaver processes - Authorization: Access control lists - Role based, attribute based and identity based – Active Directory -Lightweight directory access protocol - Open Authentication- Extensible access control markup language –Password Management - Biometrics - Federated identity management - Tomcat Server – DB ServerAuth0 tool – Open VAS tool.

Unit 3 User Access Management**9Hours**

User management: Service provisioning markup language - User identity and attribute management –Role Engineering -Access Control Models: Discretionary access control – Mandatory access control - Role based access control –Rule based access control — Multi factor Authentication - Privileged access management– Provisioning management and its types - Kerberos authentication protocol for access management –Open SSL- Okta access management tool

Unit 4 IAM Policy Management**9 Hours**

Policy management in IAM - Types of policies: Identity based policy – Resource based Policy - Session Policy –Admin roles vs IAM policies –Admin and Break glass accounts - Effective Permission Boundaries – Difference between rule and policy - Role of IAM in UEM unified endpoint management –Data governance and protection –Auditing and logging - Compliance and Governance administration.

Unit 5 Identity and access management for Cloud

9 Hours

Cloud Identity and access management - Comparison of Enterprise and Consumer Authentication Standards and Protocols– IAM practices in cloud - Cloud Identity administration: Single Sign on – Federated SSO - Identity management as a service – Cloud Authorization management: Tag based access control - Replication and Disaster recovery – Availability Management in cloud – Access management in AWS Cloud.

Total: 45 Hours

G. Laboratory Experiments

Total: 30 Hours

Task1: Install and run the tomcat server and DB server and test whether the server is started and running.

Task 2: Create a user identity and manage the credential and its associated vulnerability using green bone security assistant in OpenVAS through Kali Linux Platform.

Task 3: Create and manage keys in ‘Openssl’ for certificate management including identification of certificate information and verification of SSL through Kali Linux.

Task 4: Create multiple user accounts with its credentials and manage to retrieve those created credentials through the command line interface password cracking built-in tool in Kali Linux and audit the passwords.

Task 5: Implement the firewall rules for access control management and create restrictions for ‘services’ based on their ‘names’ and ‘port numbers’ using kali Linux.

Task 6: Create new user accounts and groups and manage the access control/levels of authorization or ownership between different users under each created groups using kali Linux.

Task 7: Getting administrator rights using Burp Suite.

Authentication Bypass: Getting access to server-side script and changing the password and access the server-side applications.

Task 8: Identity management using Auth0

- i) Creation of customer managed policy
- ii) Permit users to manage their credentials

Task 9: Delegate access Management across AWS console using following roles.

- i) User identity creation

ii)User roles

iii)Admin roles

Task 10: Manage the access on AWS console and add the created users to different user groups based on which access restrictions will be created to other users using attribute-based access control. Check the Permissions of each user created, using ‘IAM users’ sign in link in AWS console to check the access restrictions over the instances created over S3, EC2 for validating the access restriction.

Task 11: Develop a web application to use Social Login using open Authentication to authenticate

Task 12: Develop a web application to authenticate using SAML protocol by creating an account with Okta and registering SAML application for authentication.

H. Learning Resources

i. Text Books:

1. Ertem O smanoglu, “Identity and Access Management: Business performance through connected Intelligence”, Second Edition, Syngress, O’Reilly ISBN: 9780124104334 2013.
2. Tim Mather, Subra Kumaraswamy, ShahedLatif, “Cloud Security and Privacy, An Enterprise perspective on risk and compliance”, O’Reilly ISBN: 978-0-596-80276-9, 2017.

ii. Reference Books:

1. Ken Hess, “Unified Endpoint management for Dummies”, IBM Limited edition Pearson Education, John Wiley jand Sons, 2017.
2. Atlice Labs, “Identity and Access Management”, by Atlice labs Whitepaper, 2017.

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS251	VIRTUALIZATION TECHNIQUES	2	0	2	3

A. Preamble

This course aims at molding the learner to understand the fundamentals of Cloud Computing and Virtualization Technologies, their features and operations. The course helps the learners to assess the applicability of different Virtualization platforms to make effective resource utilization in IT infrastructure solutions.

B. Prerequisite Courses

10211CS103 - Operating Systems /10212CS130-Fundamentals of Computer Networks

C. Course Objectives

Learners are exposed to

- The principles and applications of cloud computing and virtualization in enterprise information systems
- Installation procedures of VMware products and to run virtual machines of different configurations
- Microsoft Hyper-V for creating and managing cloud environments.
- Security issues found in Virtualized environment

D. Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Understand the need of cloud computing, models and benefits	K2/S3
CO2	Identify the various avenues of Virtualization and their Applications along with the different levels of implementation	K3/S3
CO3	Examine the Hardware support for virtualization – hypervisors and outcomes of virtualized Data centers	K3/S3
CO4	Construct and manage virtualized infrastructure using popular virtualization platforms.	K3/S3
CO5	Organize the techniques to address security concerns in virtualized environments:	K3/S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of COs with Program Outcomes and Programme Specific Outcomes

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1 1	PSO2 2	PSO3 3
CO1	2				3									
CO2	2				3									
CO3	2	1	2		3								1	
CO4	2	1	2		3								1	
CO5														

High- 3; Medium-2; Low-1

F. Course Contents**Unit – I Overview of Cloud Computing****L- 6 Hours**

NIST definition- History of Cloud Computing, Evolution of Cloud Computing, Traditional vs. Cloud Computing. Reason to move to Cloud Computing, Service models (IaaS, PaaS & SaaS). Cloud deployment models (Public, Private, Hybrid and Community Cloud), Benefits and Challenges of Cloud Computing.

Unit – II Virtualization and Hypervisors**L- 6 Hours**

Virtualization – Definitions – History: Virtual Memory Reasons for Moving to Virtualization- Characteristics – Benefits – Features, Implementation levels of Virtualization, **Hypervisor:** Types, Emulation: Basic Interpretation – Threaded Interpretation – Pre-Coded and Direct Interpretation – Binary Translation – Full and ParaVirtualization, Virtualization of CPU, Memory and I/O Devices – Storage Virtualization

Unit -III Virtual Machines in Data Centres**L 6 Hours**

System Architectures – Virtual Machine Basics – Taxonomy of Virtual Machines -Virtualization of CPU, Memory and I/O Devices – Storage Virtualization, Virtualization structures/tools and Mechanisms, Datacentre, Virtual Clusters, Cluster Management, Benefits of Virtualized Datacentres.

Unit - IV Virtualization Tools**L- 6 Hours**

Major Players and Products in Virtualization - Overview of VMware V Sphere, ESX/ ESXi, VMware workstation and Oracle virtual box, **VMware workstation:** Operations and Features, Benefits, **Hyper V:** Introduction to Hyper V Server Architecture - Hyper V hardware requirements - Guest operating systems specifications - Understanding of Virtual switches (Private, Internal, External) - Creating, Deleting Virtual Switches

Unit – V Virtual Machines Security**L- 6 Hours**

Top virtualization security issues: External attacks, Keeping snapshots on VMs, Sharing files between VM and host, or copy-pasting between host and remote console, VM sprawl, Security of offline & dormant VMs, Security of pre-configured (golden image) VM/active VMs, Lack of visibility and control over virtual networks

Total- 30 hours

G.Laboratory Experiments**Total- 30 hours****Part 1:**

Task1: Identify the prerequisites and enable virtualization in BIOS

Task2: Download and Install VMWare Workstation (WS)

Tools Required:

- VM Ware Workstation – Stable Version

Task3: Creating Virtual machines using the VM Ware WS console

Tools Required:

- VM Ware Workstation – Stable Version

Task4: Choose and Mounting ISO and Installing Windows 10 Pro OS & Server 2019 OS on the created VM

Tools Required:

- VM Ware Workstation – Stable Version

Task5: Model the migration features by managing virtual machines

Tools Required:

- VM Ware Workstation – Stable Version,

Task6: i. Download and Install Microsoft Hyper-V

ii. Configuring and managing Hyper-V Networking (create, delete, manage virtual switches)

Tools Required:

- Microsoft Server OS (Recent Stable Version) with Hyper –V, PowerShell, Network Connections Control Panel.

Task7: Managing Hyper-V Storage (VHD types (vhd, vhdx, vhd set)(fixed, dynamic, differential)

Tools Required:

- Microsoft Server OS (Recent Stable Version) with Hyper –V, PowerShell, Hyper-V Manager PowerShell cmdlets.

Task8: Configuring and managing Check points in Hyper-V

Tools Required:

Microsoft Server OS (Recent Stable Version) with Hyper –V, Checkpoint Management GUI.

Task9: i. Creating and Modifying Virtual machines using the Microsoft Hyper-V console

ii. Import and export Virtual machines

Tools Required:

- Microsoft Server OS (Recent Stable Version) with Hyper –V, System Center Virtual Machine Manager (SCVMM).

Part 2:**Use cases:**

Usecase 1: Mr. Maarison is in charge for establishing a lab environment using existing computers (PC), Workstations for training involving different scenarios. VM ware Workstation can be used to create a lab environment for this training purpose. With this he can create multiple virtual machines and configure them to simulate different scenarios.

Technologies used: VMware Workstation.

Usecase 2: Mr. Abu is in need to develop and test applications over different platforms. With Hyper-V, he can create multiple virtual machines running different operating systems. He can set the hardware specifications, including CPU, RAM, and storage space to make variant capable platforms. This can be useful for testing and development purposes.

Technologies used: Hyper-V

Usecase 3: M/S. Info Systems planning to allow its employees to work from their home. Hyper-V can be used to create a virtual desktop infrastructure (VDI) environment, where multiple users can connect to virtual machines to access their desktops and applications. This can be useful for remote workers, also for organizations with limited hardware resources.

Technologies used: Hyper-V: Virtual Machine Management, Remote Connection Protocol

Usecase 4: Mr. Wilson is about to model a scenario which stands against disasters. To test disaster recovery scenarios Hyper-V can be used. He can create a backup of a virtual machine and then restore it to a different host machine to ensure that it works as expected.

Technologies used: Hyper-V: Virtual Machine Management, Backup and Storage, Replication.

Total- 60 hours

G. Learning Resources

i. Text Books:

1. Rajkumar Buyya, Christian Vecchiola, S. Thamarai Selvi, "Mastering Cloud Computing", McGraw-Hill Education (India) Private Limited, 2017.
2. Chris Wolf, Erick M. Halter, Virtualization: From the Desktop to the Enterprise, A Press 2005
3. David Marshall, Wade A. Reynolds, Advanced Server Virtualization: VMware and Microsoft Platform in the Virtual Data Center, Auerbach Publications, 2006.

ii. Reference Books:

1. Thomas Erl, Zaigham Mahmood and Ricardo Puttini, "Cloud Computing: Concepts, Technology & Architecture", ServiceTech Press, 2014. ISBN-13: 978-0-13-338752-0.
2. "Pro Microsoft Hyper-V 2019: Practical Guidance and Hands-On Labs", Andy Syrewicze, Richard Siddaway, October 2018, Apress, ISBN: 9781484241165

iii. Online References:

1. Prof. Sowmya Kanti Ghosh, in Cloud Computing, NPTEL [Online], 2017. Available: <https://archive.nptel.ac.in/courses/106/105/106105167/>
2. <https://docs.vmware.com/en/VMware-Workstation-Pro/index.html>
3. <https://learn.microsoft.com/en-us/iis/web-hosting/installing-infrastructure-components/hyper-v-guides>

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS252	WINDOWS CLIENT OPERATING SYSTEM	2	0	2	3

A. Preamble

This course aims to equip students with a thorough comprehensive understanding of the Windows operating system and its key features. By means of a blend of theoretical lectures, practical exercises, and assignments, students will become proficient in navigating the Windows interface, managing files and folders, personalizing system settings, and resolving common problems. Upon completion of this course, students will be able to gain valuable technical skills, career opportunities, practical experience, certification preparation, and problem-solving skills students will have acquired the skills and expertise to utilize Windows proficiently, both in personal and professional settings.

B. Prerequisite Courses

- Computer Networks
- Operating Systems

C. Course Objectives

Learners are exposed to

- Navigating the Windows user interface and Managing files and folders, including creating, renaming, moving, and deleting files and folders.
- Managing network connections, including configuring and troubleshooting network settings.
- Implementing security measures, including user authentication, password policies, and firewalls
- Troubleshooting hardware issues, including diagnosing and replacing faulty hardware components.

C. Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Create and manage user accounts in Windows, including creating new accounts, modifying account settings, and managing user privileges.	K3/S3
CO2	Manage disk storage in Windows, including creating and formatting partitions, extending or shrinking volumes, and managing disk quotas.	K3/S3
CO3	Configure network settings in client operating systems, including IP addressing, subnetting, and DNS configuration.	K3/S3
CO4	Configure and manage user accounts and permissions to enforce access control and minimize security risks.	K3/S3
CO5	Develop skills in troubleshooting network connectivity and communication issues.	K3/S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of Cos with Program Outcomes and Programme Specific Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3		3		3			2						
CO2	3	3	3	3	3	2		2						
CO3	3	3	3	3	3	3	3	3				2		
CO4	2	3	3	3	3	3	3	3				2		
CO5	2		2		3	2	2							

High- 3; Medium-2; Low-1

E. Course Contents**Unit 1 Introduction to Windows Client Operating System and User Management 6 hours**

Overview of Windows operating systems: Windows 10 OS, Introduction to User Management in Windows Client: Overview of user accounts, groups, and permissions, Types of user accounts (local, domain, built-in), User account in GUI, User account in CLI, Powershell,.

Unit 2 Managing storage and device drivers in Windows Client Administration 6 hours

Introduction to Storage Management in Windows Client, Overview of disk storage types (HDD, SSD, NVMe), Basic disk concepts (partitions, volumes, file systems), MBR, GPT, Basic Disk and Dynamic Disk, File System Types

Unit 3 Network Management in Client Administration 6 hours

IP Configuration, Remote desktop, Remote Assistance, Quick Assist, Network Commands and trouble shooting, Sharing Folders, Permissions, Mapping Network Drive

Unit 4 Security in Client Administration 6 hours

Windows Security Fundamentals: Windows Firewall- Inbound and Outbound Traffic, Local Policy, Local Security Policy, Bit Locker Management: Enabling BitLocker, Key Management, Group Policy, Encryption, Event Viewer: Types of Event Logs, Filtering and Searching, Event ID and Description, Auditing

Unit 5 Troubleshooting 6 hours

Introduction to Trouble Shooting, Windows Client Architecture, Hardware, Software, Networking and Security Troubleshooting, Troubleshooting Tools, Best Practices of Troubleshooting, Overview of OS Disaster Recovery: Planning, Backup, Recovery, Testing and Maintenance

Total :30 hours

Total: 30 hours

F. Laboratory Experiments

Task 1. Configuring Installing and configuring a Windows client operating system in a virtual machine environment.

Task 2. Configuring user accounts, groups, and permissions to manage access control

Task 3. Configuring and managing Windows Firewall to allow and block traffic.

Task 4. Configuring and managing Group Policy to control system settings and user configurations.

Task 5. Configuring and managing network adapters and protocols to enable connectivity.

Task 6. Configuring and managing DNS and DHCP services to provide name resolution and IP address management.

Task 7. Configuring and managing virtual private network (VPN) connections for secure remote access.

Task 8. Configuring and managing backup and restore operations to protect data and system configurations.

Task 9. Configuring and managing Windows Update to ensure system security and performance updates.

Task 10. Creating and managing shared folders and permissions.

Task 11. Configuring and managing printers and print services.

Task 12. Configuring and managing remote management tools, such as Remote Assistance and Remote Desktop.

Task 13. Configuring and managing system monitoring and performance tools, such as Task Manager and Resource Monitor.

Task 14. Configuring and managing disk management, including disk partitions and file systems.

Task 15. Configuring and managing system security policies, such as password policies and account lockout policies.

Total: 60 hours

G. Learning Resources (in IEEE Format)

i. Text Books:

[1] E. Bott and C. Siechert, "Windows 10 Inside Out," 3rd ed. Redmond, WA, USA: Microsoft Press, 2019.

ii. Reference Books:

[1] M. Russinovich, A. Margosis, and A. Solomon, "Windows Internals, Part 1: System Architecture, Processes, Threads, Memory Management, and More," 7th ed. Redmond, WA, USA: Microsoft Press, 2017.

[2] W. Stanek, "Windows 10 Administrator's Pocket Consultant," 2nd ed. Redmond, WA, USA: Microsoft Press, 2016.

[3] B. Hill, "Windows 10 Plain & Simple," Redmond, WA, USA: Microsoft Press, 2015.

iii. Online References:

[1] Microsoft Community, "Windows Community," Microsoft, Available: <https://answers.microsoft.com/>.

[2] Microsoft, "Windows Support Center," Microsoft Support, Available: <https://support.microsoft.com/en-us/windows>.

[3] Microsoft, "Windows Blog," Microsoft, Available: <https://blogs.windows.com/>.

[4] Microsoft, "Windows Dev Center," Microsoft Developer, Available: <https://developer.microsoft.com/en-us/windows/>.

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS253	WINDOWS SERVER ADMINISTRATION OPERATING SYSTEM	2	0	2	3

A. Preamble

This course is designed to provide students with a comprehensive understanding of the Windows server administration operating system and its key features. Through a combination of lectures, hands-on exercises, and practical assignments, students will learn how to install, configure, and maintain various types of hardware and software, which often involves creating user accounts, carrying out backup and recovery functions, and always monitoring the performance of servers. By the end of this course, students will have developed the skills and knowledge necessary to use Windows effectively in a variety of personal and professional contexts.

B. Prerequisite Courses

10211CS103 - Operating Systems

C. Course Objectives

Learners are exposed to

- Familiarize students with the basic features and functions of the Windows operating system, including the user interface, file management, and system settings.
- Provide students with an understanding of the different versions of Windows and their respective capabilities and limitations.
- Introduce students to the security features of Windows and the best practices for keeping a Windows-based system secure from external threats.
- Teach students how to troubleshoot common issues that may arise when using Windows, such as driver errors, system crashes, and software compatibility issues.

D. Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Explore the configuration and management of the Windows server administration and power shell.	K3/S3
CO2	Apply Domain Control technique to store and manage the objects using AD DS and TCP/IP on window server.	K3/S3
CO3	Demonstrate with Filter, Restore, Backup and Common issues in windows server using DHCP DORA and DNS Server.	K3/S3
CO4	Implementation of group policy, files, print server, troubleshooting and also classify data using FSRM in windows server.	K3/S3
CO5	Use IIS and WDS to deploy and manage web application and network based installation through windows server and backup.	K3/S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of Cos with Program Outcomes and Programme Specific Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3		3		3			2						
CO2	3	3	3	3	3	2		2						
CO3	3	3	3	3	3	3	3	3				2		
CO4	2	3	3	3	3	3	3	3				2		
CO5	2		2		3	2	2							

High- 3; Medium-2; Low-1

F. Course Contents**Unit 1 Deploying & managing server 2019****6 Hours**

Windows Server 2019 Overview, Installing and Post-Installation Configuration of Windows Server 2019, Overview of Windows Server 2019 Management, Introduction to Windows PowerShell.

Unit 2 Introduction to AD DS**6 Hours**

Overview of AD DS, Overview of Domain Controllers, Installing a Domain Controller, User management, Group Management, Overview of TCP/IP, Understanding IPv4 & IPv6 Addressing

Unit 3 DNS and DHCP**6 Hours**

DHCP DORA Process, Configuring Scopes Reservation and Exclusion. Overview of DHCP Configuration files, DHCP Filters, DHCP Backup and restore, Common Issues in DHCP Server. Introduction to DNS, Name Resolution Methods- Hierarchy - Records. DNS Working and Query Process. Common issues in DNS Server

Unit 4 Implementing Group Policy, File and Print Services**6 Hours**

Introducing Group Policy, Implementing and administering GPOs, Scope and Group Policy processing (LSDOU), Troubleshooting the application of GPOs, Implementing administrative templates, Configuring Folder Redirection, Software installation, and scripts Configuring Group Policy preferences. Overview of FSRM, Using FSRM to Manage Quotas, File Screens and Storage Reports. Overview of DFS and DFS Replication

Unit 5 IIS, WDS and WDUS**6 Hours**

Overview of IIS and Components, Overview of WDS, WDS Server Roles, WDS Multicasting, PXE booting, Windows Images, Overview Of Patch management Process, Data Protection Overview, Implementing Windows Server Backup.

Total: 30 Hours**G. Laboratory Experiments**

- Task 1.** Add\remove roles using server manager, PowerShell
- Task 2.** Installing a Domain Controller, Promoting as a Forest
- Task 3.** Create, Delete, OU, Users

- Task 4.** Installing and Configuring DNS
- Task 5.** Verify Name resolution from client
- Task 6.** Installing DHCP Role, Authorizing DHCP Role.
- Task 7.** Creating, Activating and Deactivating Scope
- Task 8.** Configure DHCP filters, DHCP backup and restore
- Task 9.** Creating, editing, Linking Group Policy
- Task 10.** Creating password policies & account Lockout policies.
- Task 11.** Installing and Configuring IIS
- Task 12.** Installing & Configure WDS
- Task 13.** Configuring WDS Properties
- Task 14.** Deploying OS from WDS to WDS Client
- Task 15.** Back up files from client to server and folders restores them from server after Deleting.
- Task 16.** Backup and BareMetal recovery

Total: 30 Hours

H. Learning Resources (in IEEE Format)

i. Text Books:

1. Thomas, Orin. ‘Windows server 2019 inside out’. Microsoft Press, 2020.
2. Jordan Krause, “Mastering Windows Server 2019”, Packt Publishing Limited; 2nd edition (18 March 2019)

ii. Reference Books:

1. William R. Stanek, Jr. Stanek, William R.” Windows Server 2016: The Administrator's Reference”, Createspace Independent Pub (1 September 2016)
2. Jason Eckert, “Hands-On Microsoft Windows Server 2019”, (MindTap Course List), Cengage Learning; 3rd edition (June 5, 2020)

iii. Online References:

1. “Windows Server documentation” , Accessed on Apr. 04, 2023 [Online]. Available: <https://learn.microsoft.com/en-us/windows-server/>
2. “Microsoft Certified: Windows Server Hybrid Administrator Associate” Accessed on Apr. 04, 2023 [Online]. Available: <https://learn.microsoft.com/en-us/certifications/windows-server-hybrid-administrator/>

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS254	CLOUD FUNDAMENTALS USING AZURE	2	0	2	3

A. Preamble

This course aims at moulding the learner to understand cloud concepts, Azure services, Azure workloads and support. The course helps the learners to assess the benefits and considerations of using Azure cloud services in enterprise IT infrastructure solutions.

B. Prerequisite Courses

10211CS105/10211CS130 -Computer Networks/Fundamentals of Computer Networks

C. Course Objectives

Learners are exposed to

- Cloud computing models and the underlying concepts
- Architectural components of Azure
- Storage and Deployment tools in Microsoft Azure
- Various Cloud computing programming models in developing services with Azure.

D. Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K – Level
CO1	Illustrate the cloud computing models and underlying concepts.	K2/S3
CO2	Construct Architectural components of Azure	K2/S3
CO3	Experiment with Storage and Deployment tools in Microsoft Azure to design Cloud applications	K3/S3
CO4	Apply the concepts of Windows Azure to create and manage virtual machines	K3/S3
CO5	Develop services using various Cloud computing programming models with Azure.	K3/S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of Cos with Program Outcomes and Programme Specific Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	2				3									
CO2	2				3									
CO3	2	1	3		3								1	
CO4	2	1	3		3								1	
CO5	2	1	3		3								1	

High- 3; Medium-2; Low-1

F. Course Contents**Unit-I Introduction****L-6 Hours**

Cloud Computing – Characteristics of Cloud – Cloud Deployment Models – Cloud Service Models- Benefits - Shared responsibility model- consumption-based model- Reliability and predictability in cloud- security and governance in cloud- Cloud pricing models: Freemium, Pay Per Reservation, Pay per User, Subscription based Charging.

Unit-II Virtualization and Networking in Data Centers**L- 6 Hours**

Data Centers -Regions and availability zones – Cloud resources and Resource management groups - compute types: container instances - Azure storage services-Azure Migrate and Azure Data Box - Azure Virtual Networks, Azure DNS, Azure VPN Gateway-Azure ExpressRoute.

Unit-III Directory and Domain Services**L- 6 Hours**

Microsoft Azure Active Directory (Azure AD) and Azure AD Domain Services (Azure AD DS) Microsoft Defender for Cloud - External identities and guest access in Azure-Conditional Access in Microsoft Azure Active Directory-Authentication methods in Azure - Microsoft Entra -Role based access control (RBAC) – Microsoft Defender for Cloud.

Unit-IV Policy and Pricing**L- 6 Hours**

Azure Policy - Blueprints - Service Trust Portal - Factors affecting costs -Purpose of tags in Pricing- Compare: Pricing calculator and Total Cost of Ownership (TCO) calculator -Azure Site Recovery and Disaster Recovery- Azure Arc-Azure Cost Management and Billing tool.

Unit-V Azure Resources and Applications**L- 6 Hours**

Azure Cloud shell, CLI and Power Shell-Azure Resource Manager and templates (ARM templates) - Azure Kubernetes Services-Azure Service Health -Azure Monitor: Log Analytics, Azure Monitor alerts and Application Insights.

H.Laboratory Experiments

Total: 30 Hours

Part – 1

Task 1	Deploy VM in Availability set and load balancing the traffic through External Load Balancer to achieve efficient traffic distribution. Tools: Azure Subscription, Azure Portal, Vnet
Task 2	Deploy WebApps in two different regions and managing it through traffic manager to achieve high availability and global scalability. Tools: Azure Subscription, Azure Portal, Vnet, Subnet, Availability set
Task 3	Create and migrate database from on premise to SQL Azure to leverage the benefits of a managed and highly available database platform. Tools: Azure Subscription, Azure Portal,SQL SSMS,DMA
Task 4	Create an Azure Storage Account and Provisioning Azure Storage services. Tools: Azure Subscription, Azure Portal,Azure storage Account
Task 5	Create a VET-to-VNET virtual networks to establish secure and scalable network connectivity between virtual networks. Tools: Azure Subscription, Azure Portal, Vnet
Task 6	Deploy two VMs in Availability Set and load balancing it to increase the availability of application or service. Tools: Azure Subscription, Azure Portal, Vnet, subnet, Availability set, Load Balancer
Task 7	Creation of VMs with portal, ARM Templates and PowerShell to provide flexibility and automation in the deployment and management of VMs. Tools: Azure Subscription, Azure Portal, ARM Resource Manager, ARM Templates, Powershell
Task 8	Implement Azure AD Identity Protection for Authentication to enhance the security of your organization's resources by detecting and preventing identity-based attacks. Tools: Azure Subscription, Azure Portal, Azure Active Directory(AD)
Task 9	Create and Manage roles and devices in Microsoft Azure to control access to Azure resources effectively. Tools: Azure Subscription, Azure Portal, Azure Active Directory(AD),Azure RBAC
Task 10	Create a web based Campus Events Calendar application using Microsoft Azure. Tools: Azure Subscription, Azure Portal,Azure App Services,Web Application Framework
Task 11	Implement a data storage and management in Azure using Azure SQL Database for Student Information System (SIS). Tools: Azure Subscription, Azure Portal,Azure SQL Database: Azure SQL Database is a fully managed, cloud-based relational database service prov
Task 12	Create the own custom virtual machine with the required OS, Software and other components using Microsoft Azure. Tools: Azure Subscription, Azure Portal, Virtual Machine ,OS Image

Part - 2

Use Cases:

Use Cases 1: Loan application processing

Azure Logic Apps can be used to automate loan application processing, allowing banks to quickly process and approve loan applications. The system can automatically verify customer information, perform credit checks, and generate loan agreements, reducing the time and effort required to process loan applications. Azure Logic Apps can be used to build workflows that automate loan application processing. The system can automatically verify customer information, perform credit checks, and generate loan agreements. Logic Apps can integrate with other Azure services, such as Azure Functions, to provide additional functionality.

Technologies Used: Azure Logic Apps, HTTPConnector, Azure SQL Database Connector, Azure Blob Storage Connector, Azure Service Bus Connector, Azure Cognitive Services Connectors, Azure Active Directory Connector.

Use Cases 2: Online banking portal

An online banking portal can be built using Azure App Service, which provides a platform for building web and mobile applications. The portal can provide customers with access to their account information, transaction history, and other features such as bill pay and fund transfers. Azure Active Directory can be used for authentication and authorization, ensuring that only authorized users can access the portal.

Technologies Used: Azure App Service, Technology stack such as ASP.NET, Node.js, Python, or PHP.

Use Cases 3: Disaster response and Recovery efforts

Azure can be used to quickly set up and deploy critical systems and infrastructure in response to natural disasters or other emergencies. Suppose there is a natural disaster, such as a hurricane, that has impacted a region, causing widespread damage to buildings, power lines, and communication networks. Emergency responders need to quickly set up temporary shelters and provide basic services such as food, water, and medical care to affected individuals. By using Azure for disaster response and recovery efforts, emergency responders can quickly set up critical systems and infrastructure, communicate more efficiently, and allocate resources more effectively, ultimately helping to save lives and mitigate the impact of the disaster on affected communities.

Technologies Used: Azure Virtual Machines, AzureStorage, Azure Site Recovery, Azure Backup

Use Cases 4: Fraud detection and prevention

Azure Machine Learning can be used to build predictive models for fraud detection and prevention. By analyzing customer data and transaction history, the system can identify potential fraudulent activity and alert staff to take action. Azure provides several options for storing and managing data, including Azure SQL Database, Azure Cosmos DB, and Azure Blob Storage. These services can be used to store transaction data, customer profiles, and other data needed for fraud detection and prevention.

Technologies Used: Azure Machine learning, Azure Cognitive Service.

Total: 60 Hours

G. Learning Resources

i. Text Books:

1. Jim Boyce, “Microsoft certified Azure Fundamentals Study Guide”, Sybex – A Wiley Brand, 2021.
2. Michael Collier and Robin Shahan,” Fundamentals of Azure – Microsoft Azure essentials” , Microsoft Press, 2015.

ii. Reference Books:

1. Namit Tanasseri, Rahul Rai, “Microservices with Azure”, 1st Edition, Packet Publishing, UK, 2017.
2. Sriram Krishnan, “Programming: Windows Azure”, O’Reilly, 2010.
3. Enamul Haque, “Cloud Service Management and Governance: Smart Service Management in Cloud Era”, Enel Publications.

iii. Online References:

1. “Introduction to Microsoft Azure Fundamentals” Available: “[https://learn.microsoft.com/en-us/training/modules/describe-cloud compute /introduction-microsoft-azure-fundamentals](https://learn.microsoft.com/en-us/training/modules/describe-cloud-compute/introduction-microsoft-azure-fundamentals)”, [Online].
2. “Introduction to Azure Fundamentals”: Available: “<https://learn.microsoft.com/en-us/training/modules/intro-to-azure-fundamentals/>”, [Online].

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS255	ADMINISTERING MICROSOFT EXCHANGE SERVER	3	0	2	3

A.Preamble

This course provides with the knowledge and skills that are necessary to administer and manage a Microsoft Exchange Server 2016/2019 messaging environment. Specifically, this course will give students the skills to configure and manage Exchange Server organization, recipients, address policies and objects, as well as, perform messaging compliance and auditing. This training will also help students gain an understanding of how to deploy and manage Exchange Online in an Office 365 environment.

B.Prerequisite Courses

10212CS253 - Windows Server Administration Operating System

C.Course Objectives

Students are able to

- Elaborate on the utilization of management tools for the purpose of managing Exchange.
- Learn how to administer and support Exchange Server.
- Learn how to install and configure Exchange Server
- Manage Exchange Online.
- Implement and manage Exchange Online deployments.

D.Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Demonstrate the way of deploying and managing Exchange Server, including its architecture, storage, and management tools.	K2/S3
CO2	Identify the recipient object and use cmdlets and scripts for Managing Exchange Servers	K3/S3
CO3	Make use of Exchange Server to configure, manage and restore client access, outlook and mobile messaging services.	K3/S3
CO4	Utilize Exchange Server to configuring and managing messaging transport services, Antivirus and AntiSpam solutions.	K3/S3
CO5	Apply security in Microsoft Exchange Server by using RBAC.	K3/S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of COs with Program outcomes and Programme Specific Outcomes:

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO 1	PSO 2	PSO3
CO1	3		1		3				2	3	2			2
CO2	3		1	2	3				2	3	2			2
CO3	3		1		3			2	2	3	2			2
CO4	3		1		3				2		2			2
CO5	3		1		3		2	2			2			2

High- 3; Medium-2; Low-1

F.Course Contents**UNIT I Deploying and Managing Exchange Server****L-9**

Exchange Server: Overview- requirements-Architecture-Management Tools-Pros and cons. Deploying: Management Tools. Exchange Storage: Overview- architectures, Mailbox databases: Creation- properties.

UNIT II Managing recipients and Exchange servers**L-9**

Recipient objects: Management of user mailboxes- Types: User- Linked- Microsoft 365/Office 365- - Resource- System mailboxes, Distribution groups, Exchange Management Shell: Overview -Exchange Server using cmdlets and scripts.

UNIT III Implementing client connectivity**L-9**

Exchange Server: Management of client access services- Client Connectivity - Outlook on the web- mobile messaging, Managing high availability: Mailbox Databases-Client Access Services. Disaster Recovery: Server Backup-Server Recovery.

UNIT IV Configuring and managing message transport**L-9**

Message transport: Overview- Transport rules: components- properties- Rule storage and replication, Message security: Edge Transport server: Internet mail flow- Mail flow rules, AntiVirus- AntiSpam.

UNIT V Monitoring and troubleshooting**L-9**

Monitoring - Troubleshooting, Securing: Audit Logging, Using RBAC: Overview- Requirements- Service Principals- Application Roles, Exchange Online and Office365: Overview - Migration - Hybrid Environment.

TOTAL: 45 hours**G. Laboratory Experiments****Total: 30 Hours****LIST OF EQUIPMENT FOR A BATCH OF 30 STUDENTS:**

- TASK 1 Setup the environment for Microsoft Exchange Server
- TASK 2 Create mailbox databases to manage data's associated with mails.
- Implement and manage recipient objects to enable users to send and receive emails as well as other forms of communication within an organization.
- TASK 3
- Create and manage public folder mailboxes to store and share information within an organization.
- Configure address lists and policies to control the visibility and accessibility of email addresses and distribution groups for users and recipients
- Create and manage email address policies to assign multiple SMTP email addresses to recipients by using different combinations of the local part and domain values.
 - Create and manage the address lists and address book policies to segment users into specific groups to give them customized global address lists (GALs) in Outlook and Outlook on the web.
- TASK 4
- Manage Exchange Server and recipient objects by using Exchange Management Shell to provide administrators with a powerful and flexible command-line interface for managing their Exchange environment
- TASK 5
- Manage recipients using Exchange Management Shell.
 - Manage Exchange Server using Exchange Management Shell.
- Configuring client access services
- TASK 6
- Configuring namespaces for client access
 - Configuring certificates for client access
 - Configuring custom MailTips
- Deploying and configuring client access services on Exchange Server
- TASK 7
- Configuring Exchange Server for Outlook
 - Configuring Outlook on the web
 - Configuring Microsoft Exchange ActiveSync
- Implement DAGs
- TASK8
- Create and configure a DAG (Database Availability Group) to store relevant information about the DAG, such as server membership information and some DAG configuration settings.
- TASK9 Implement and testing high availability

- Deploy a high availability solution for Client Access services to ensure uninterrupted and reliable access to critical client-facing applications and services.
- Test the high availability configuration.
- Configure highly available mailbox databases.

TASK10 To create a copy (Backup) of the data stored on the server to recover the data in the event of a primary data failure.

- Backing up an Exchange Server mailbox database.

To restore the Exchange Server data in case of data loss.

- TASK11
- Restore Exchange Server mailbox
 - Restore a database availability group (DAG) member (optional)

To Configure and manage message transport to facilitate the routing and delivery of email messages between mailboxes and other messaging systems.

- TASK12
- Configure the message transport
 - Monitor and verifying message delivery
 - Configure a disclaimer transport rule
 - Configure a DLP policy for financial data

Configure message security in Exchange Server to send encrypted messages to people inside or outside your organization

- TASK13
- Configure and testing EdgeSync
 - Configure antivirus, antispam and malware protection features to prevent and address computer virus, worm, spyware, malware, and other types of malicious software.

Monitoring and troubleshooting Exchange Server to track performance and security issues

- TASK14
- Monitoring Exchange Server
 - Troubleshooting database availability
 - Troubleshooting Client Access servers

Securing and maintaining Exchange Server to protect Exchange from viruses, phishing, denial-of-service attacks, malware and spam.

- TASK 15
- Configure Exchange Server permissions
 - Configure audit logging
 - Maintaining Exchange Server

Managing Exchange Online

- TASK 16
- Implement and manage Exchange Online deployments

Stand alone systems with 16 GB RAM and 1TB Hard Disk 30 Nos.

H. Learning Resources

i. Text Book(s):

1. Edward Van Biljon, "Microsoft Exchange Server 2019 Administration Guide", BPB Publications, April 2021.
2. Clifton Leonard and David Elfassy, "Mastering Microsoft Exchange Server 2016", Sybex, December 2016.
3. Paul Robichaux, "Microsoft Exchange Server 2019: Administration and Support", O'Reilly Media, October 2019.

ii. Reference Book(s):

1. Jonas Andersson, "Microsoft Exchange Server 2016 PowerShell Cookbook", Packt Publishing, September 2016.
2. Tony Redmond, "Microsoft Exchange Server 2013 Inside Out", Microsoft Press, October 2013.

iii. Online References:

1. <https://learn.microsoft.com/en-us/exchange/exchange-server?view=exchserver-2019>

COURSECODE	COURSE TITLE	L	T	P	C
10212CS256	OFFICE 365 ADMINISTRATION	3	0	2	3

A.Preamble

This course focuses on the administration, configuration, troubleshooting and operations of the Microsoft Office 365 platform. It also makes us to learn how to adjust user and subscription settings, configure Exchange mailboxes and groups, set up SharePoint sites, manage Microsoft Teams, and perform advanced configurations, such as multifactor authentication and PowerShell scripting.

B.PrerequisiteCourses

10212CS253 - Windows Server Administration Operating System

C.Course Objectives

Learners are exposed to,

- Understand the basic concepts of Identity and Access Management, Zero Trust Security and user management.
- Learn how to develop and implement an effective security strategy.
- Equip students with the knowledge and skills needed to manage mobile devices and protect sensitive information in an organization.
- Equip students with the knowledge and skills needed to plan for and manage compliance requirements in Microsoft 365, including archiving and retention, content search and investigation, and data governance troubleshooting.

D.Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Understand the fundamentals of Microsoft 365 and learn about Role Based Access Control (RBAC)	K2/S3
CO2	Analyze and identify gaps in current Microsoft 365 security posture using Secure Score.	K3/S3
CO3	Compute Advanced Threat Analytics and learn to identify attacks using attack simulator in Microsoft 365.	K3/S3
CO4	Manage and configure devices with MDM and domains for MDM.	K3/S3
CO5	Create useful retention tags and retention policy.	K3/S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of COs with Program outcomes and Programme Specific Outcomes:

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO 1	PSO2	PSO 3
CO1	3						2	3		3				
CO2	3	3	3	2										2
CO3	3	3	3		3						2			2
CO4	3				2			3		2				2
CO5	3		3	2	2		2		2	2	2			2

High- 3; Medium-2; Low-1

F. Course Content**UNIT I User and Group Protection****L-9**

User and Group Protection: Identity and Access Management Concepts-Zero Trust Security-User Accounts -Administrator Roles and Security Groups - Password Management in Microsoft 365-**Identity Synchronization:**Azure AD Connect-Planning- Implementation- Identity Protection -Managing Synchronized Identities - Federated Identities-**Access Management:**Conditional access-Manage device access-Role Based Access Control (RBAC)-Solutions for external access.

UNIT II Security in Microsoft 365**L-9**

Threat vectors and data breaches-Security strategy, principles and solutions -Microsoft Secure Score.**Advanced Threat Protection:** Exchange Online Protection- Azure -Microsoft Defender- Advanced Office365 Threat Protection-Manage Safe Attachments and Links-**Threat Management:**Threat investigation and response-Azure Sentinel.

UNIT III Mobility and Information Protection**L-9**

Mobility: Plan for Mobile Application Management-**Mobile Device Management:**Plan, Deploy and Enroll Devices -**Information Protection:** Concepts - Azure protection-Advance Windows Information Protection.

UNIT IV Rights Management and Encryption**L- 9**

Rights Management and Encryption: Information Rights Management- S/MIME-Office 365 Message Encryption, **Data Loss Prevention:** An Overview- Create and Customize Policies- Policy Tips, **Cloud Application Security:** An Overview.

UNIT V Compliance in Microsoft 365**L-9**

Compliance in Microsoft 365: Plan for compliance requirements- Build ethical walls in Exchange Online- Manage Retention in Email-Troubleshoot Data Governance, **Archiving and Retention:** Archive-Retention and its policies -In-place Records Management in SharePoint-**Content Search and Investigation:** Overview-Audit Log Investigations-Advanced eDiscovery.

TOTAL: 45 hours

G.Laboratory Experiments**Total: 30 Hours****LIST OF EQUIPMENT FOR A BATCH OF 30 STUDENTS:**

- TASK 1** To Discover and Manage Azure Resources to create, update, and delete resources in your Azure account.
- Assign Directory Roles
 - Activate and Deactivate PIM Roles
 - Directory Roles (General)
 - PIM Resource Workflows
 - View audit history for Azure AD roles in PIM.
- TASK 2** Use Conditional Access to enable MFA to make the right decisions as well as to enforce effective organizational policies.
- MFA Authentication Pilot (require MFA for specific apps)
 - MFA Conditional Access (complete an MFA roll out)
- TASK 3** Use Microsoft Secure Score to measure an organization's security posture, with a higher number indicating more recommended actions taken.
- Improve your secure score in the Microsoft 365 Security Center.
- TASK 4** Manage Microsoft 365 Security Services to assist organizations to detect threats early and help minimize the impact of a breach.
- Implement ATP Policies.
- TASK 5** Using Attack Simulator
- Conduct a simulated Spear phishing attack to recognize and respond to these types of cyber threats.
 - Conduct simulated password attacks to assess the security posture of an organization's user accounts and identify potential vulnerabilities.
- TASK 6** Configure Azure AD for Intune to establish a secure and streamlined environment for managing and protecting devices, applications, and data within an organization.
- Enable Device Management to manage and secure the devices used by the employees to access O365 resources.
 - Create Intune Policies to manage and protect corporate data, applications, and resources on various types of devices, including Windows, iOS, Android, and macOS.
- TASK 7** Implement Azure Information Protection and Windows Information Protection
- Implement Azure Information Protection to enhance data protection and control by classifying, labelling, and protecting sensitive information in an organization
 - Implement Windows Information Protection to protect sensitive data on Windows devices by applying security controls and preventing data leakage or unauthorized access.
- TASK 8** Configure Advanced Threat Analytics to detect and prevent attacks.

- TASK 9** Configure Office 365 Message Encryption to protect sensitive information in emails and other communications by encrypting them and ensuring that only authorized recipients can access them.
- Validate Information Rights Management
- TASK 10** Implement Data Loss Prevention policies to protect an organization's sensitive and confidential data from being disclosed or lost.
- Manage DLP Policies
 - Test MRM and DLP Policies
- TASK 11** Compliance and Retention to ensure that an organization's data is managed, retained, and disposed of in a compliant and secure manner.
- Initialize Compliance
 - Configure retention tags and policies
- TASK 12** Manage Search and Investigation to ensure that an organization is in compliance with data protection regulations and to fulfill the rights of data subjects.
- Investigate your Microsoft 365 Data
 - Conduct a Data Subject Request

Standalone systems with 16 GB RAM and 1TB Hard Disk 30 Nos.

H. Learning Resources

i. Text Book(s):

1. Aaron Guilmette, Darryl Kegg, Ed Fisher, "Microsoft 365 Administration Inside Out", 3rd Edition, The Microsoft Press Store by Pearson, 2023.
2. Carpe, Thomas, "Mastering Office 365 Administration", Packt, 2018

ii. Reference Book(s):

1. Randy Nordell, Kathleen Stewart, Pat Graves, and Annette Easton, "Microsoft Office 365: In Practice, 2019 Edition", McGraw-Hill Education, 2019.
2. "Microsoft Office 365 Administration Cookbook", Packt Publishing, 2020

iii. Online References:

1. <https://learn.microsoft.com/en-us/microsoft-365/admin/admin-overview/admin-center-overview?view=o365-worldwide>

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS257	Data Center Hosting _WINDOWS OPERATING SYSTEM	2	0	2	3

A. Preamble

This course is designed to provide students with a comprehensive understanding of the Windows operating system and its key features. Through a combination of lectures, hands-on exercises, and practical assignments, students will learn how to navigate the Windows user interface, manage files and folders, customize system settings, and troubleshoot common issues. By the end of this course, students will have developed the skills and knowledge necessary to use Windows effectively in a variety of personal and professional contexts.

B. Prerequisite Courses

10211CS103-Operating Systems

C. Course Objectives

Learners are exposed to

- Familiarize students with the basic features and functions of the Windows operating system, including the user interface, file management, and system settings.
- Provide students with an understanding of the different versions of Windows and their respective capabilities and limitations.
- Introduce students to the security features of Windows and the best practices for keeping a Windows-based system secure from external threats.
- Teach students how to troubleshoot common issues that may arise when using Windows, such as driver errors, system crashes, and software compatibility issues.

D. Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Apply appropriate methods to manage users & groups and different methods for securing data and the Windows OS.	K3/S3
CO2	Implement IP network connectivity, name resolution using DNS , remote management and access for a Windows-based system.	K3/S3
CO3	Demonstrate the ability to implement security measures, different methods of encryption, monitoring and troubleshooting tools to diagnose and resolve performance issues in a Windows-based system.	K3/S3
CO4	Experiment with Backup , Restore and Troubleshooting Windows Server	K3/S3
CO5	Solve troubleshoot network connectivity issues and Manage and control Configuration Management Database (CMDB) to meet customer and ITIL	K3/S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of Cos with Program Outcomes and Programme Specific Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	1	1	1								2		2	
CO2	1	2	3	3	3			2			2		2	
CO3	1	2	3	3	3			2			2		2	
CO4	1	1	3	3	3			2			2		2	
CO5	1	2	3	2	3						2		2	

High- 3; Medium-2; Low-1

F.Course Contents**Unit 1 Introduction to Operating System, Authentication and Update Management 6 Hours**

Server OS and Client OS-Different Versions/Editions of Windows OS (2003, 2008, 2012, 2016,2019)- Windows Server 2016 Overview-Windows Server 2016 Management-Deployment Methods-different editions and features of Windows Server-Authentication-different methods for securing data. Drivers-Windows Servicing Model-Updating Windows-updating Windows using WSUS-Windows Update for Business-Monitoring & Patching Tools-SolarWinds Patch Manager-Update through Intune Portal

Unit 2 Remote, Disk, File and Security Management 6 Hours

Remote Access-Remote Management-working principle of name resolution- remote access Technologies-Managing Storage, Disks, Volumes and Storage Spaces- options and benefits of local, cloud, and virtual Storage-File Systems-Managing User Files- common security threats-mitigate common security threats-encryption methods-secure the device by Windows firewall- benefits of using IPSec-features of Windows Defender.

Unit 3 Performance & Monitor, Backup & Restore and Trouble shooting 6 Hours

Windows Architecture-Support and Diagnostic Tools-Monitoring and Troubleshooting Computer Performance-SysInternal Tools-key stages in troubleshooting-purpose and benefits of the various tools-File Recovery-Application Troubleshooting-methods of file recovery-Troubleshooting Windows Startup,Operating System Service Issues-recovering from startup issues-advanced startup options-failed services- locked account scenarios- Troubleshooting Device Driver Failures, Hardware, Physical Failures.

Unit 4 Storage management and Windows command 6 Hours

Remediate driver issues,Peripherals.storage technologies and Usage-different storage topologies, FC & iSCSI- RAID redundancy RAID 0, RAID 1, RAID 5,RAID 10-disk types-iSCSI logical units-iSCSI target instances-ATA basic disk, dynamic disk, mount points, file systems, mounting a virtual hard disk, distributed file system- Command shell-Networking Information- Lists Installed Drivers- Hardware Information- Server is Reachable-Scan and Repair System Files .

Unit 5DNS Management , Basic Troubleshooting, ADDS Service,IT-IMS and ITIL 6 Hours

Windows Services-Windows Services Control Manager-Differences between Windows Services and Regular Applications Windows Services and Regular Applications-DNS Client Service-Windows troubleshooting-Knowledge of local and remote computer system diagnostic tools-Active Directory Service -Domain Controllers-Protocols in AD-Active Directory, Identity, and Access-AD console-Tree- IT Infrastructure-Infrastructure Management-Remote Infrastructure Management-Server Hardware, Component-Network Devicesknowledge database, work instructions- Service Life Cycle-Process and Function-Processes and Function in Service Operations-Processes in Service Transition-ITIL Incident Change and Problem Management

Total: 30 Hours

G.Laboratory Experiments**Total: 30 Hours****Task 17.** Basics of VMware & Virtualization Technology.**Task 18.** Authentication Management

- a) Managing Users and Groups- Create a new user account, add it to a new group, and assign group permissions to a folder.
- b) Configuring User Account Control- Configure User Account Control (UAC) settings to improve security on a Windows Server.
- c) Implementing Device Registration- Implement device registration in Windows Server to improve security and control access to resources.
- d) Managing Domain Authentication- Manage domain authentication in Windows Server to improve security and control access to resources.
- e) Managing Local User and Microsoft Account Authentication- Manage local user and Microsoft account authentication in Windows Server to improve security and control access to resources.
- f) Managing password and account options- Manage password and account options in Windows Server to improve security and control access to resources.
- g) Managing Azure AD Authentication- Manage Azure AD Authentication to secure access to cloud-based resources.
- h) Configure user account control. - Configure User Account Control (UAC) in Windows Server to enhance security and prevent unauthorized changes.

Task 19. Configuration Management

- a) Register a device with a domain- Register a device with a domain in Windows Server to enable centralized management of devices and improve security.
- b) Configure and Customize the Windows Start Menu- Configure and customize the Windows Start Menu in Windows Server to improve usability and enhance user productivity.
- c) Common Configuration Options- Configure common configuration options in Windows Server to optimize performance and improve security.
- d) Advanced Configuration Methods- Configure advanced configuration methods in Windows Server to optimize performance and improve security.
- e) Managing Drivers and Devices- You are a system administrator responsible for managing a Windows Server environment with multiple client devices. You have received a request from a department manager to install a new printer on their department's client devices.
- f) Synchronizing settings between devices- Synchronize settings between devices in Windows Server to enable seamless access to user preferences and settings.

Ex: system administrator is responsible for managing a Windows Server environment with multiple client devices. Administrator has received a request from a user who regularly works on multiple devices and wants their settings and preferences to be synchronized across all their devices.

Task 20. Update Management

- a) Use the Windows control panel and setting app to configure settings. - Use the Windows Control Panel and Settings app to configure network settings on a client device in the Windows Server environment.

Ex: System administrator is responsible for managing a Windows Server environment with multiple client devices. Administrator has received a request from a user who needs to configure their network settings to access a new network resource.

- b) Perform tasks using Windows PowerShell- Use Windows PowerShell to disable a specific feature on all client devices in the Windows Server environment.

Ex: system administrator is responsible for managing a Windows Server environment with multiple client devices. Administrator has received a request from the IT security team to disable a specific feature on all client devices to address a security vulnerability.

- c) Configure client-side printing and managing print server properties- Configure client-side printing and manage print server properties in Windows Server to enable printing for a department's client devices.

Ex: system administrator is responsible for managing a Windows Server environment with multiple client devices Administrator has received a request from a department manager to configure a new printer on their department's client devices and manage the print server properties.

- d) Configure Windows update settings- Configure Windows update settings in Windows Server to ensure that all client devices in the environment are up-to-date with the latest security patches.

Ex: system administrator is responsible for managing a Windows Server environment with multiple client devices. Administrator has received a request from the IT security team to configure Windows update settings on all client devices to ensure they are up-to-date with the latest security patches.

- e) Configure Windows update using group policy- Configure Windows update settings using group policy in Windows Server to ensure that all client devices in the environment are up-to-date with the latest security patches.

Ex: system administrator is responsible for managing a Windows Server environment with multiple client devices. Administrator has received a request from the IT security team to configure Windows update settings on all client devices to ensure they are up-to-date with the latest security patches. However, Administrator need to configure the update settings using group policy to ensure consistency across all client devices.

Task 21. Remote Management

- a) Configure IP Network Connectivity- Configure IP network connectivity settings in Windows Server to enable a new client device to connect to the network.

Ex: network administrator is responsible for managing a Windows Server environment with multiple client devices. You have received a request from a department manager to configure network connectivity settings on a new client device that has been added to their department.

- b) Implement Name Resolution- Implement name resolution in Windows Server to ensure that all client devices can correctly resolve the FQDN of the new web-based application server.

Ex: You are a network administrator responsible for managing a Windows Server environment with multiple client devices. The IT department has recently deployed a new web-based application on a server in the environment. The application is accessed using a fully qualified domain name (FQDN), but some client devices are experiencing issues with resolving the name to the correct IP address.

- c) Implement Wireless Network Connectivity- Implement wireless network connectivity in Windows Server to enable client devices to connect securely to the new wireless network.

Ex: A network administrator is responsible for managing a Windows Server environment with multiple client devices. Recently, the IT department has deployed a new wireless network for employees to use. The task at hand is to configure the wireless network settings and ensure that client devices can connect securely.

- d) Configure IP network connectivity. - Configure IP network connectivity in Windows Server to ensure that the server can communicate with other servers and devices on the network.

Ex: A network administrator is responsible for managing a Windows Server environment with multiple client devices. The administrator discovers that one of the servers in the environment is unable to communicate with other servers or devices on the network. After

investigation, it is determined that the server's IP address settings are incorrect, and it is not properly configured for the network.

- e) Configure Windows for remote management and access- Configure a Windows Server to allow remote management and access using Remote Desktop Protocol (RDP). Ensure that all remote connections are encrypted using SSL/TLS encryption. Test remote access to the server using a remote device, and verify that the connection is encrypted. Additionally, configure firewall rules to allow RDP traffic through the server's firewall, and create user accounts with appropriate permissions to allow remote access to the server.

Task 22. File management

- a) Configuring and Managing File Access- In a Windows Server environment, create a shared folder and configure file access permissions to allow only certain users or groups to access the folder and its contents. Test the permissions by attempting to access the folder from a user account that does not have permission to do so.

Ex: As a Windows Server administrator, you have been tasked with configuring file access permissions for a shared folder that contains sensitive information. You need to ensure that only authorized users can access the folder and its contents. What steps would you take to configure and manage file access?

- b) Configuring and Managing Shared Folders- Create a shared folder on a Windows Server and configure permissions to allow specific users to access it. Test the folder access by logging in as a user with the appropriate permissions and verifying that they can read and write to the folder, while other users cannot access it.

Ex: As a network administrator is responsible for managing a Windows Server environment that includes several file servers with shared folders. One of the departments in the organization needs access to a shared folder on one of the file servers, but only specific users should have read and write permissions to the folder. How network administrator configures and manage the shared folder to ensure that only the designated users have access to it?

- c) Configuring and Managing Permissions and Shares- Configure a shared folder on a Windows Server and manage its permissions and shares. Create different user groups and assign them different levels of access to the shared folder. Test the access permissions by attempting to access the shared folder from different client devices using different user accounts.

Ex: IT administrator is responsible for managing the file server in your organization. The file server contains important documents and files that are

shared among different departments. IT administrator need to ensure that only authorized users have access to these files, and that the files are protected from unauthorized access or modifications. How IT administrator configure and manage permissions and shares for the files on the file server?

- d) Using Conditions to Control Access and Effective Permissions- Configure a shared folder named "Finance" on a Windows Server and set the permissions to allow only members of the "Finance" group to access it. Configure a separate shared folder named "Public" that is accessible to all users. Use conditions to control access to the "Finance" folder based on the user's group membership. Test the configuration by logging in as a user who is a member of the "Finance" group and a user who is not a member, and attempting to access the folders.

- e) Work Folders-

1. Install and configure Work Folders on a Windows Server 2016 or later server.
2. Create a new sync share for a department or group of users.
3. Configure the Work Folders sync settings for a user or group of users.

4. Test the Work Folders setup by syncing files between a work device and a personal device.
 5. Monitor Work Folders synchronization and troubleshoot any issues that arise.
- f) Synchronizing files with OneDrive- In the lab, you will be asked to configure OneDrive for Business synchronization settings, including selecting which folders to synchronize and setting up security options such as encryption and conditional access. You will also learn how to monitor synchronization status and troubleshoot common synchronization issues. Finally, you will practice configuring policies to control access to OneDrive files and prevent data leakage.
- Ex: IT administrator working at a company that has recently migrated to Microsoft 365. The management has asked administrator to configure OneDrive for Business to synchronize files between users' devices and the cloud. administrator need to ensure that the synchronization is secure and efficient, while also preventing unauthorized access to sensitive files.

Task 23. Performance and Monitor

- a) Monitoring Reliability and Performance- how to use the Performance Monitor and Resource Monitor to monitor system reliability and performance.
Ex: You are the system administrator for a company that has a Windows Server environment. One of the servers in the environment is experiencing slow performance, and users have reported issues accessing the applications hosted on the server. You need to identify the cause of the performance issues and optimize the server for better reliability and performance.
- b) Monitoring Events- how to use the Event Viewer tool to monitor events on a Windows Server.
Ex: As a system administrator, you are responsible for monitoring events on a Windows Server. One of the applications on the server is experiencing issues, and you suspect that it might be due to errors in the event log. You need to investigate the event log to find the root cause of the problem and resolve it.
- c) Use monitoring tools to establish a performance baseline- As an IT administrator, you have been tasked with identifying the cause of a slow network performance issue in your Windows Server environment. You need to establish a performance baseline and monitor the network to identify potential bottlenecks and areas for optimization. What monitoring tools and techniques can you use to establish a performance baseline and identify potential network performance issues?
- d) Optimize performance on Windows 10 clients. - In the lab, you have a Windows 10 client device that is running slow and needs to be optimized for better performance. Use the built-in performance optimization tools and techniques to identify and resolve performance issues, such as disabling unnecessary startup programs, defragmenting the hard drive, and adjusting power settings. Monitor the system performance using Task Manager and other performance monitoring tools to ensure that the changes have a positive impact on the system's overall performance.

Task 24. Backup Restore

- a) Using File History to Recover Files- Using a Windows 10 device, set up and configure File History to automatically back up important files. Then, intentionally delete a file and use File History to restore it.
Ex: You work as an IT administrator for a company with multiple Windows 10 devices. One of the employees accidentally deleted an important file and needs to recover it. You need to use the File History feature to restore the deleted file.
- b) Configure Windows Server to support individual file and system recovery.
You are the system administrator of a Windows Server environment. One of your users has reported that they have accidentally deleted an important file from their shared folder. You need

to restore the file as quickly as possible. What steps would you take to recover the deleted file using the individual file recovery feature in Windows Server?

- c) Recover a device using the Reset This PC function.

A support technician in a company that uses Windows 10 on all of its devices. One of the employees has reported that their laptop is not functioning properly and is experiencing various issues. Technician has diagnosed the problem and concluded that the best solution would be to reset the device to its factory settings using the Reset This PC function. What steps the technician will take to ensure that the employee's personal files are not lost during the reset process? How the technician will help the employee restore their files once the reset is complete?

- d) Solve application compatibility issues with the Application Compatibility Toolkit.

You are an IT administrator tasked with deploying a new application across your organization. After the deployment, you receive reports from some users that the application is not working properly and experiencing compatibility issues with their systems. You need to solve these compatibility issues to ensure the application is working as intended on all systems.

Task 25. Troubleshooting Windows Server

- a) Using Advanced Startup and Windows RE to recover from Boot Failures

system administrator is responsible for managing a Windows environment in a large organization. One of the computers on the network is having boot issues, and it is not starting up properly. Administrator suspect that the boot files may be corrupted or missing. Administrator need to use the advanced startup options and Windows Recovery Environment (Windows RE) to recover the system and resolve the boot issues.

- b) Recovering Windows using Reset This PC

You work as a technical support specialist for a company that has recently experienced some issues with one of its computers. The computer is running Windows 10, and the user has reported that it is running very slowly and has become unresponsive at times. You have tried various troubleshooting steps, but the issue persists. You decide to use the "Reset This PC" feature to try and fix the problem.

- c) Troubleshooting Hardware by Using Windows Memory Diagnostics

Technician is responsible for maintaining a large number of computers in a corporate environment. One of the computers is experiencing frequent crashes and freezes, which could be caused by faulty memory modules. Technician need to troubleshoot the issue using Windows Memory Diagnostics to identify and resolve the problem.

- d) Recovering Windows by using a Restore Point

You are a support technician for a small business. A user has reported that their computer has been acting up and they suspect that a recent software update may be the cause of the issue. You suspect that the update may have caused a conflict with the system's drivers, leading to stability issues. You want to try restoring the system to a previous state using a restore point to see if it resolves the issue. How would you go about doing this?

Task 26. Storage Management

- a) Enable the iSCSI initiator

As a system administrator, you have been tasked with setting up an iSCSI storage solution for your organization. You need to enable the iSCSI initiator on a Windows Server to connect to the iSCSI target. How would you do this?

- b) Configure iSCSI initiator

Your organization has recently decided to implement a new storage area network (SAN) using iSCSI technology. As part of your responsibilities, you need to configure the iSCSI initiator on the servers and ensure that they can connect to the new SAN.

c) Configuring iSCSI target

system administrator is responsible for managing a storage area network (SAN) in your organization. Technician has been asked to configure an iSCSI target to provide storage to a group of servers. The storage should be easily scalable and provide high availability.

Task 27. DNS Management

a) Installing and Configuring DNS Server

A network administrator is tasked with setting up a DNS server for your organization. Your organization has multiple internal networks and requires DNS resolution for internal resources. Additionally, your organization's website is hosted externally and needs to be accessible to users on the internet. Administrator need to install and configure the DNS server to meet these requirements.

b) creating DNS ZONE and Records

You have been tasked with setting up a new DNS server for your organization. One of your primary responsibilities is to create DNS zones and records for various services and resources on your network. You need to ensure that your DNS configuration is accurate and efficient, and that it provides reliable name resolution for all devices on your network. How would you approach this task, and what steps would you take to create and manage DNS zones and records?

Task 28. Installing a Domain Controller

You are an IT administrator responsible for setting up a new domain in your organization. Your organization has several departments with different access levels, and you need to set up a domain controller to manage user accounts and permissions. You decide to install a domain controller on a Windows Server 2019 machine. What steps will you take to install and configure the domain controller, and how will you ensure that it is properly configured to manage user accounts and permissions?

Task 29. Basic Troubleshooting Windows

a) Activating Windows

You have just installed Windows 10 on a new computer and need to activate it. You have a valid product key, but are unsure of the steps to take to activate Windows. What are the steps you should take to activate Windows?

b) Fix computer booting very slowly

You are a technical support specialist, and a user has reported that their computer is taking an unusually long time to boot up. The user has already tried restarting the computer

several times but still experiences the same issue. What steps would you take to diagnose and fix the problem?

c) Kill background apps to speed your PC

You have noticed that your computer has been running very slowly lately, and you suspect that background apps may be the cause. You have tried closing some of the apps manually, but the issue persists. What steps can you take to kill background apps and speed up your PC?

d) Fixing the Blue Screen of Death

You have a Windows 10 computer that is displaying the Blue Screen of Death (BSOD) error message. The error message suggests that there is an issue with the system's memory. How would you troubleshoot and fix this issue?

e) Install the latest updates for your computer.

A system administrator is responsible for maintaining a network of computers. Recently, administrator has noticed that some of the computers are running slowly and experiencing frequent crashes. Upon investigation, administrator find that many of the computers have outdated software and operating system versions. To resolve the issue, administrator decide to install the latest updates on all the computers.

Total: 60 Hours

H.Learning Resources

i. Text Books:

1. Thomas, Orin. ‘Windows server 2019 inside out’. Microsoft Press, 2020.
2. Jordan Krause, “Mastering Windows Server 2019”, Packt Publishing Limited; 2nd edition (18 March 2019)

ii. Reference Books:

1. William R. Stanek, Jr. Stanek, William R.”Windows Server 2016: The Administrator's Reference”, Createspace Independent Pub (1 September 2016)
2. Jason Eckert, “Hands-On Microsoft Windows Server 2019”, (MindTap Course List), Cengage Learning; 3rd edition (June 5, 2020)

iii. Online References:

1. “Windows Server documentation” ,Accessed on Apr. 04, 2023 [Online]. Available: <https://learn.microsoft.com/en-us/windows-server/>
2. “Microsoft Certified: Windows Server Hybrid Administrator Associate” Accessed on Apr. 04, 2023 [Online]. Available: <https://learn.microsoft.com/en-us/certifications/windows-server-hybrid-administrator/>
3. “Microsoft Windows Server 2016 Training for Beginners”. Accessed on:
4. Apr. 04,2023 [Online]. Available:<https://www.udemy.com/course/windows-server-2016/>

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS259	ADVANCED SWITCHING AND NETWORKING TECHNIQUES	2	0	2	3

A. Preamble

This course covers network fundamentals install, operate, configure, and verify basic IPv4 and IPv6 networks. It also includes network access, IP connectivity, IP services, configuration of DNS, DHCP configuration and Wireless networks, Ethernet LANS in detail. This course also lays the foundation to advanced Computer networking concepts.

B. Prerequisite Courses

10211CS105/10211CS130- Computer Networks /Fundamentals of Computer Networks

C. Course Objectives

Learners are exposed to

- Understand Network Fundamentals.
- Identify the basic need of Network Access.
- Implement various IP Services
- Implement VLAN

D. Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Design subnets and calculate the IP addresses to fulfill network requirements of an organization.	K3/S3
CO2	Evaluate the performance of Ethernet LAN and Wireless LAN through simulation.	K3/S3
CO3	Configure the VLAN Network and VLAN device settings.	K3/S3
CO4	Analyzing VTP Ethernet LAN Switching Configuration.	K3/S3
CO5	Assessing Network performance and using STP to track every connection in the specific network.	K3/S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of Cos with Program Outcomes and Programme Specific Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PS O1	PS O2	PS O3
CO 1	2	2	1									3		
CO 2	2	3	2	1								3		
CO 3	2	3	2	1								3		
CO 4	2	2	2									3		
CO 5	2	2	3	1								3		

High- 3; Medium-2; Low-1

F. Course Contents**Unit 1 Network Addressing and Introduction to Wireless Networks****L-6 Hours**

Introduction to IP Addressing and configuration-DHCP Configuration and Troubleshooting- Understanding DNS and troubleshooting basic issues related to DNS-Introduction to Wireless Network and Architecture- Components of a WLAN (WLC, Access Point, EndPoints)

Unit 2 Introduction to Ethernet**L- 6 Hours**

Introduction to Ethernet -Terminology used in Ethernet switches - Frames, Transparent Bridging, Address Learning, Traffic Filtering, Frame Flooding, Multicast Traffic, Configure and troubleshoot-EtherChannel Concept-Discovery Protocols & L2 and L3 EtherChannel Configuration

Unit 3 VLAN**L-6 Hours**

VLAN - Configuring VLANs - Creation of an Ethernet VLAN - Traditional Inter-VLAN Routing - Create VLANs on the switch - Assign the VLANs to switch port - Configure the IP addresses on the router - Router-on-a-Stick Inter - VLAN Routing- Configuring Inter - VLAN Routing - Multilayer Switch Inter - VLAN Routing

Unit 4 VLAN Trunking Protocol**L-6 Hours**

VLAN Trunking Protocol (VTP) - Requirements of VTP – modes - Configuration - Layer 2 switching - LLDP concepts - LLDP Service - Analyzing Ethernet LAN Switching

Unit 5 Spanning Tree Protocol**L- 6 Hours**

Introduction to Spanning-Tree - concepts in STP- Varieties of Spanning Tree Protocols-Spanning Tree Configuration- Calculation process of the STP algorithm - STP Operation- Step-by-Step Instructions - STP protocol packets.

Total: 30 Hours

Laboratory Task:**Total: 30 Hours****Task 1:** Study of basic network command and Network configuration commands.**Task 2:** Connect the computers in Local Area Network**Task 3:** Perform an initial configuration of a Cisco Catalyst 2960 switch for the given topology diagram**Task 4:** Configuring and Troubleshooting a Switched Network for the given topology diagram

- Establish console connection to the switch.
- Configure the host name and VLAN1.
- Use the help feature to configure the clock.
- Configure passwords and console/Telnet access.
- Configure login banners.
- Configure the router.
- Solve duplex and speed mismatch problems.
- Configure port security.
- Secure unused ports.
- Manage the switch configuration file.

Task 5: Connecting a Switch for the given topology diagram

- Connect a switch to the network.
- Verify the configuration on the switch

Task 6: Implementing an IP Addressing Scheme for the below topology diagram

- Subnet an address space based on the host requirements.
- Assign host addresses to devices.
- Configure devices with IP addressing.
- Verify the addressing configuration

Task 7: Configuring Ethernet and Serial Interfaces

- Configure a LAN Ethernet interface.
- Configure a WAN serial interface.
- Verify the interface configurations

Task 8: Configuring a Cisco Router as a DHCP Server for the given topology diagram**Task 9:** Configure a layer 3 EtherChannel for the two links between switches Dist1 and Dist2 so that IPv4 packets can be routed between the two switches, as follows:

- Configure the two links between the distribution switches as a static layer 3 EtherChannel with port-channel interface number 10.
- Make the EtherChannel a routed port instead of a switch port.
- Configure the IP addresses per the figure for the Port Channel interface

Task 10.Configure Port to VLAN Interface Settings on a Switch.**Task 11.**Configure a routed port for connecting to the firewall on the L3 switch.

Task 12. In a typical network some switches are configured as VTP servers and other switches are configured as VTP clients. Create a VLAN on a VTP server switch which is automatically advertised to all switches inside the same VTP domain.

Task 13. Configure STP on Catalyst Switches.

Task 14. Create the network with values shown below containing 3 interconnected Ethernet switches and 3 computers. Verify that from PC1, you can ping the IP address of the other five devices. Subnet mask of all devices should be 255.255.255.0.

Total: 60 Hours

G. Learning Resources

i. Text Books:

1. Richard Deal, “CCNA Cisco Certified Network Associate Study Guide”, DOI: 10.1036/0071497285.
2. Todd Lammle, “CCNA Routing and Switching Guide”, Sybex, 2016.
3. Behrouz Forouzan, “Data Communications and Networking”, Tata McGraw Hill, 5th Edition, 2021.

ii. Reference Books:

1. Kurose and K.W. Ross, “Computer Networking: A Top-Down Approach Featuring the Internet”, Addison-Wesley, 2017.

iii. Online References:

1. <https://www.gratisexam.com/cisco/200-120/Cisco.Actualtests.200-120.v2014-02-12.by.GillBeast.299q.pdf>
2. <http://examsforall.com/exams/cisco/200-120/Cisco.Actualtests.200-120.v2013-12-09.by.Watson.314q.vce/#.ZD94AXZBzic>
3. <http://ppdi.stmik-banjarbaru.ac.id/data.bc/24.%20Mc%20Graw%20Hill%20Series/McGraw.Hill.Cisco.Certified.Network.Associate.Study.Guide.Ex.pdf>

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS260	ADVANCED ROUTING TECHNIQUES AND SECURITY	2	0	2	3

A. Preamble

This course covers network configuration, operation, and examining IPv4 and IPv6 networks. It also includes network access, IP connectivity, IP services, network security, automation, and programmability.

B. Prerequisite Courses

10211CS105/10211CS130 - Computer Networks / Fundamentals of Computer Networks

C. Course Objectives

Learners are exposed to

- Examine the Network Configuration.
- Design and optimize the various routing protocols.
- Implement various IP connectivity and services.
- Implement the remote access and authentication protocols.

D. Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Explain the role and function of Routing Configuration.	K3/S3
CO2	Configure IP Routing.	K3/S3
CO3	Interpret the IGRP and EIGRP.	K3/S3
CO4	Dynamic Routing Protocols.	K3/S3
CO5	Border Gateway Protocol.	K3/S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of Cos with Program Outcomes and Programme Specific Outcomes

COs	PO1	PO2	PO3	PO4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PSO 1	PSO 2	PSO 3
CO1	2	2	1									3		
CO2	2	3	2	1								3		
CO3	2	3	2	1								3		
CO4	2	2	2									3		
CO5	2	2	3	1								3		

High- 3; Medium-2; Low-1

F.Course Contents**Unit 1 Routing Configuration****L - 6 Hours**

Introduction: The Internal Components of a Cisco Router and Switch, The Router and Switch Boot Sequence, Configuring DHCP, DHCP Relay, Network Time Protocol, Configuring IP Routing, Static Routing, Default Routing, Dynamic Routing, Routing Protocol Basics, Routing Information Protocol.

Unit 2 IGRP and EIGRP**L-6 Hours**

IGRP (Interior Gateway Routing Protocol) - Configuration, Timers, Loop Avoidance Mechanisms. EIGRP (Enhanced Interior Gateway Routing Protocol) - Neighbors, Topology Table, Packet Types.

Unit 3 Dynamic Routing Protocols**L- 6 Hours**

OSPF (Open Shortest Path First) - Neighbors, Designated Routers, Network Types and Metric. Border Gateway Protocol (BGP) – Peers, Neighbors, Synchronization, Routing Table.

Unit 4 Introduction to Security**L-6 Hours**

Introduction: Security Trends; Overview of Firewall–Needs for Firewall and Types; Infrastructure security- Port Security, Access Control List, Device Hardening.

Unit 5 Remote Access and Authentication**L- 6 Hours**

Introduction - Remote Access–Secure socket layer–VPN–IPSEC–Port Address Translator –Network Address Translator (NAT) –Access policies; Authentication: Authentication- Radius, AD, TACACS, Directory Access Protocol (DAP), Lightweight directory access protocol (LDAP), DAP, Link layer discovery protocol (LLDP).

G. Laboratory Task:**Total: 30 Hours**

Task 1: Study of network command and configuration commands.

Task 2: Interpreting Ping and Traceroute Output.

Task 3: Examining and observing Static and Dynamic Routing.

Task 4: Implementing and configuring a Default Route.

Task 5: Configuring RIP.

Task 6: The topology below is running EIGRP. You are required to troubleshoot and resolve the EIGRP issues between the various routers. Use the appropriate show commands to troubleshoot the issues.

Task 7: The topology below is running OSPF. You are required to troubleshoot and resolve the OSPF issues between the various routers. Use the appropriate show commands to troubleshoot the issues.

Task 8: After HostA pings HostB, which entry will be in the ARP cache of HostA to support this transmission.

Task 9: Implement Firewall in The Given Network to Enhance Security.

Task10: Implement The ACLs and verify that the appropriate traffic is permitted or denied.

Task11: Implement the Static NAT.

Task 12.Implement the Static PAT.

Task 13.Implement The VPNtunnels in the network using routers.

G. Learning Resources (in IEEE Format)

i.Text Books:

- 1.Behrouz Forouzan, “Data Communications and Networking”, Tata McGraw Hill, 5th Edition,2021.
- Kurose and K.W. Ross, “Computer Networking: A Top-Down Approach Featuring the Internet”, Addison-Wesley, 2017

Reference Books:

1. Richard Deal, “CCNA Cisco Certified Network Associate Study Guide”, DOI: 10.1036/0071497285.
2. Todd Lammle, “CCNA Routing and Switching Guide”, Sybex, 2016.

iii. Online References:

1. <https://www.gratisexam.com/cisco/200-120/Cisco.Actualtests.200-120.v2014-02-12.by.GillBeast.299q.pdf>
2. <http://examsforall.com/exams/cisco/200-120/Cisco.Actualtests.200-120.v2013-12-09.by.Watson.314q.vce/#.ZD94AXZBzic>
3. <http://ppdi.stmik-banjarbaru.ac.id/data.bc/24.%20Mc%20Graw%20Hill%20Series/McGraw.Hill.Cisco.Certified.Network.Associate.Study.Guide.Ex.pdf>

COURSECODE	COURSE TITLE	L	T	P	C
10212CS262	STORAGE MANAGEMENT	2	0	2	3

A.Preamble

This course aims at molding the learner to understand the various networking components and the storage management techniques. The course helps the learners to expertise in simulating backup and recovery methods and also acquire the knowledge of virtualization.

B. Prerequisite Courses

10212CS253 - Fundamentals of Computer Networks

C.Course Objectives

Learners are exposed to,

- Understand the basics of networking protocols and server storage techniques.
- Simulate Illustratethe various cloud based storage allocation and reallocation principles using EMC SAN Simulator.
- Implement the server storage technique using NetApp simulator.
- Summarize various Backup & Recovery and Virtualization strategies.

D.Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Implement storage management protocols SAN and NAS	K2/S3
CO2	Implement the methods for the Provisioning and deprovisioning of host and Pool of network storage Using EMC SAN Simulator.	K3/S3
CO3	Simulate the various cloud based storage techniques using NetApp Simulator.	K3/S3
CO4	Describe various storage backup and recovery strategies.	K3/S3
CO5	Infer the concepts of Data Center Virtualization.	K3/S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of COs with Program outcomes and Programme Specific Outcomes:

[illegible]

CO2	3													
CO3		2		2						2				
CO4	3	3	2		2	2	2				2		2	2
CO5	3	3	2		2	2	2				2		2	2

High- 3; Medium-2; Low-1

F.Course Content

Unit 1: SAN and NAS Technologies

L-6 Hours

Storage (SAN & NAS) - Storage Fundamentals- NAS Protocols- SAN Protocols- RAID concepts - Types of disks -storage health check - storage vendors - storage architecture - vendor coordination - hardware components

Unit 2 –EMC SAN

L-7 Hours

EMC SAN - Unity - allocation of storage both SAN & NAS - deprovisioning of storage host creation - pool creation

Unit 3 NetApp

L-4 Hours

Aggr - volume - qtree - quota - vservers - subnet - lif - cifs - nfs -iscsi -lun mgmt - snapshot & restore - snapmirror

Unit 4: Introduction to Backup and Restore

L-8 Hours

Introduction to Backup and recovery, Backup server monitoring, Disable & enable agent, backup Removal tasks - configurations, library, media agent, servers & zoning for media server Restore & Retention extend request, Backup info & status request, Different type of backup failures - CompTIA Linux+ Backup & Restore- Backup & Recovery: Enterprise Backup Strategies, Windows Client Backup and Recovery Tools, Business Continuity & Disaster Recovery, Backup & Recovery Strategies.

Unit 5: Virtualization

L-5 Hours

VM-Memory expansion & Disk expansion-VM Planned move & datastore addition - Certification Checks - ESXiHost Restart--Resource allocation – increase & decrease - Data Center Virtualization: vSphere & ESXi - Data Center Virtualization: vCenter Server.

LAB TASKS:

Total-30 Hours

1.Perform the following system health checks issues before they arise

- CPU or memory Health
- Connectivity Heal
- Host Health

2. Construct a RAID array and Implement the System (aggr)

3. Configure vserver according to the type of pool

4. Creation of aggregate nad volumes

5. Modifying the volumes and options

6. Implementation of NAS protocols- NFS

6.a subnet creation

6.b lif creation

6.c export policy creation

6.d export policy rule creation

7. Implementation of NAS protocols- CIFS

7.a subnet creation

7.b lif creation

7.c cifs share creation

8. Implementation of SAN protocols - LUN Mapping

8.a network config

8.b volume creation

8.c iscsi creation

8.d lun creation

8.e lun mapping

8.1 On windows host

8.1.a. configure iscsi initiator

8.1.b. discover portal

9. Snapshot

9.a creating snapshot

9.b deleting snapshot

9.c resizing snapshot

10. Understand the Snapmirror concepts (replication)

Mandatory Training and Certification

- DCA-ISM - Information Storage and Management, Certification Number - DEA-1TT5 (Exam price \$230)
- DELLEMC Storage Softwares , <https://www.dell.com/en-in/dt/hands-on-labs/index.htm#tab0=3>
 - PowerStore T - Administration and Management
 - PowerScale OneFS – Getting Started
 - PowerMax – Provisioning, Data Reduction and Role Based Access Control
 - Dell EMC PowerFlex - Overview and Operations including Replication
 - CloudIQ
 - <https://www.dell.com/en-in/dt/storage/unity.htm#tab0=0> – Unity Simulators
 - iDRAC9 for PowerEdge Servers
 - PowerProtect Cyber Recovery
 - Data Protection Suite - Avamar and Networker 19.4
-
- Storage NETAPP simulators - https://kb.netapp.com/Advice_and_Troubleshooting/Data_Storage_Software/ONTAP_OS/Where_can_the_NetApp_ONTAP_9_Simulator_be_downloaded
- Virtualization Free Labs - VMware - - <https://www.vmware.com/in/products/vsphere/vsphere-hol.html>
- Nutanix - <https://www.nutanix.com/products/community-edition>

Online Resources

- Series of NETAPP storage free YouTube videos
<https://www.youtube.com/watch?v=wIW9rbeTdmQ> - www.flackbox.com
- VMware Some Free YouTube Trainings
https://www.youtube.com/watch?v=_wRRXEZkeXo
https://www.youtube.com/watch?v=a_VyvJGeEcQ
<https://www.youtube.com/watch?v=beqAIHSbkfc>
<https://www.youtube.com/watch?v=xxbcE7PMd5I>
<https://www.youtube.com/watch?v=iqkQXaFr-to>
<https://www.youtube.com/watch?v=tEfOP96vJRc>
<https://www.youtube.com/watch?v=gudC4JQyyGg>

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS263	IIS WEB SERVER TECHNOLOGIES	2	0	2	3

A. Preamble

The objective of this course is to provide basic concepts about IIS and its administration to create the interactive client-server integrated design of web applications. This course also teaches the enhanced security features using client certificate mapping, IP security, Request filtering and URL authorization. This course offers Microsoft's own cloud for logging and troubleshooting web applications using integrated services. Finally, the creation of an application server and SSL Certificate in Microsoft's IIS will be discussed.

B. Pre-requisite course

10211CS103 Operating Systems

10211CS105 Computer Networks

C. Course Objectives:

Learners are exposed to:

- Enable and configure IIS web server application
- Create web server and client, migration, authentication with advance controls
- Install and Manage SSL Certificate for Secure the Application server
- Monitor and optimize the performance of IIS server and client console
- Troubleshoot IIS console environment using Microsoft's Log

D. Course Outcomes:

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Design web server application by enabling the IIS Manager in Windows platform and also create client console using commands	K3/S3
CO2	Build an interactive client-server environment to fetch the active web page migration using Web Site Deployment console application.	K3/S3
CO3	Create, manage and configure SSL Certificate in IIS web server application to secure client server communication using ASP.NET Web page handler.	K3/S3
CO4	Make use of monitoring connection, components and respective action for optimizing the IIS web server application.	K3/S3
CO5	Apply Amazon Elastic Compute Cloud (Amazon EC2) to run application programs in AWS public cloud and Troubleshoot Amazon EC2 resources using Troubleshooting Console Environment	K3/S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of COs with Program outcomes and Programme Specific Outcomes:

Cos	PO 1	PO2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO1 0	PO1 1	PSO 1	PSO 2	PSO3
CO1		2	3	2	3	1							3	
CO2		2	3	2	3	1							3	2
CO3		2	3	2	3	3							3	2
CO4		2	3	2	3	3							2	2
CO5		2	1	2	3	1							2	2

High- 3; Medium-2; Low-1

F. Course Contents**UNIT 1 IIS Introduction and Administration****6 Hours**

Introduction to IIS: IIS overview - IIS Architecture - System Management Overview. **IIS Administration:** System Management Overview - Installation and Configuration - Application Deployment - Security Management - Monitoring and Performance - SSL Configuration - Log rotation - Backup and Restoration - IIS vulnerability fixes

UNIT 2 Website Creation, Migration and Authentication with Advance Controls **6 Hours**

Creating website & Migration: Process of creating first website - Binding Explanation - Website migration. **Application Pools:** Application pool identities - General - Advanced Settings – CPU Advanced Settings - process model - process orphaning - recycling - Default application pool settings. **Authentication:** Introduction about authentication in IIS - Anonymous authentication - Anonymous with Impersonation - Basic and digest authentication - Windows authentication - Forms authentication.

UNIT 3 IIS Extensions and Certificate Management**6 Hours**

Microsoft IIS Extensions & Certificates: HTTPS Overview - HTTP/2 Overview - Features of HTTP/2 - Installing SSL Certificate - Backup - Restore SSL certificates - SSL certificate types - Trusting the self-signed certificate - SSL certificate authentication.

Certificate Management: CSR generation - Certificate installation - Certificate renewal process.

UNIT 4 IIS Monitoring, Optimization and Operations**6 Hours**

IIS Monitoring: Worker Processes monitoring - Request failed tracing - Advanced logging module - Performance monitoring tool. **IIS Performance Optimization:** Configure Logging – Disable Debugging - Threads per processor - Queue length - Application pool settings - Output cache - IIS Server Services - Restarting IIS Services.

Standard Operations Procedures: IIS services Start and Stop - Log Management.

UNIT 5 IIS Troubleshooting Console Environment**6 Hours**

Troubleshooting: Known issues and solutions: Known errors and associated solutions while working as IIS admin - Troubleshooting different types of errors in IIS management console.

Total: 30 Hours**Use Case1 : Websites Hosting**

Hosting websites on-premises or in the cloud, Microsoft IIS is the best choice for deploying websites using Microsoft technology such as ASP.NET pages. The websites interact with the customers requirements, such as access control, timesheet, and so on; this helps to automatically create and configure a website is a differential of the solution. This will be used to satisfy the untrained end user to install and configure a web application. However, Microsoft IIS has great features for advanced users, allowing for example, to host several sites on the same server and segregate them for internal publication such as on an intranet, as well as for external publication on the Internet.

Tools: IIS Web server and AWS Console

Use Case 2 :Business web application hosting

To build a business web application to support operations which are hosted on Windows servers using IIS. The utilisation of IIS is probably considered fairly basic with it running HTTP based HTML websites using winforms and re-write rules. IIS is being used as a centralized web farm management tool for Business web application hosting. IIS will deploy and manage web sites and applications across large farms of web servers from a centralized location. Microsoft IIS makes it easy to centrally deploy and manage the configuration and content across a bunch of different web servers.also iis is used to manage several internal monitoring websites that keep a track a of different threads.

It allows us to host a variety of these websites across the business to manage different applications within the service. The business problem is addresses is that it allows us to easily host and change these monitoring websites to suit our business needs. Because Installation, usage, configuration is easy, without any lack of functionality

Tools :HTML and IIS internet web application server

Use Case 3 :Server- client deliverables hosting communication

In small projects, Microsoft IIS acts as the client deliverables. also Server- client deliverables projects require IIS to implement for overall project hosting and delivery. Because Microsoft IIS is very easy to enable and uninstall, and its GUI is very informative. The Microsoft internet web server facilitates implementation of AWS cloud seamless. It is very useful due to its simplicity for small projects which primarily require Cloud requirements in AWS..

Tool :Microsoft IIS on Window Server 2012 R2 and AWS

LAB EXPERIMENTS**Total: 30 Hours**

TASK 1: Enable the IIS in Windows platform, create the Server 2012 (GUI and Core Edition) and Enable the Web Server.

TASK 2:

- a) Create and verify the default IIS using command and Managing and Configuring the IIS Web Server.
- b) Deploy the IIS Web Server Application using command and manage the security.

TASK 3: Implement how to backup and restore the SSL Certificate.

TASK 4:

- a) Create a website using IIS and make forms authentication.
- b) Migrate website from one IIS to another IIS

TASK 5: Configuring the advanced setting in CPU process model and process orphaning.

TASK 6: Configuring the application pool for a site or application using IIS.

TASK 7: SSL certificate creation with implementing authentication.

TASK 8: Implement CSR Certificate generation and renewal process.

TASK 9: IIS services start and stop along with log management.

TASK 10: Workers process monitoring based on advanced logging modules by using monitoring tools.

TASK 11: Tune the value of Thread Per Processor Limit and make sure you are disabling the IIS Web Server.

TASK 12: Troubleshooting different types of errors in IIS management console.

Total: Theory + Lab= 60 Hours**H. Learning Resources****i. Text Books:**

1. Mike Volodarsky, ,Olga Londer“, ,Brett Hill Internet Information Services (IIS) 7.0 Resource Kit“, 2008. [Unit 1 - 5]

ii. Reference Books:

1. Anwarul Haque , “Mastering Microsoft IIS Paperback”, 2018. [All Units]

iii. Online Resources:

1. “IIS Overview and Guide Material”, Accessed on: May. 6, 2021 [Online], Available: <https://learn.microsoft.com/en-us/iis/>

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS264	APPLICATION SERVER TECHNOLOGIES	2	0	2	3

A. Preamble

The objective of this course is to provide basic concepts about Apache Tomcat Server and its administration to create the interactive client-server integrated design of web applications. This course also teaches the enhanced security features using SSL certificate management mapping, security, JDBC Connectivity, Authentication and authorization. This course offers to implement the concepts of Jmeter application, certificate generation, log management of overall Apache Tomcat Administration will be discussed.

B. Pre-requisite course

10211CS212 - Web and Mobile Application Development

C. Course Objectives:

Learners are exposed to:

- Enable and configure the concept of HTML and JSP for Apache Web application.
- Configure the war file to deploy web applications.
- Use of JDBC connectivity to perform backend database connectivity for an Application Server.
- Monitor the security operations over Apache tomcat server and also control overall administration process flow.
- Troubleshoot and implement the various mechanism over jmeter application, logs and service packs.

D. Course Outcomes:

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Design the structure of web application using HTML and JSP.	K3/S3
CO2	Build and configure war file to deploy web application.	K3/S3
CO3	Make use of JDBC connectivity to connect with Apache Tomcat Server.	K3/S3
CO4	Apply Java security concepts to perform authentication & authorization in a web application.	K3/S3
CO5	Implement the concepts like Jmeter application for testing, Certificate Generation, log management and service packs for developing the web application.	K3/S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of COs with Program outcomes and Programme Specific Outcomes:

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1		2	3		3	1							3	
CO2		3	3		2	1							3	
CO3		3	2		3	2							2	2
CO4		3	3		3	2							3	3
CO5		3	3		3	2							3	3

High- 3; Medium-2; Low-1

F. Course Contents**UNIT 1 Apache Tomcat Installation and Directories****L- 6 Hours**

Apache Tomcat Overview: Apache Software Foundation Introduction-Apache Tomcat licensing. **Installing Apache Tomcat:** Download and install a JVM. **Apache Tomcat Directories:** Apache Tomcat Directory Structure, Common and shared Directories-The bin, The conf, logs. Server, work, temp, webapps Directory- **Structure of the Web Application:** Directory Structure-Deploying HTML and JSP Pages and Web Context-JSP Code-Deploying a Web App.

UNIT 2 Apache Tomcat Configuration, Management and Advanced Features 6 Hours

Configuring Apache Tomcat: Configuration of Apache Tomcat and Techniques-conf/web.xml file-Component Architecture-Server.xml File-Modifying server.xml. **Apache Tomcat Manager:** Deploying and Managing Web Application using the Apache Tomcat Manager-Starting and looking at Apache Tomcat Manager-Applications, Deploy, Server Information-Creating and Deploying a WAR File-Java Management Extensions (JMX) on Apache Tomcat-Apache Tomcat integration with Ant.

UNIT 3 JDBC Connectivity and Connecting Tomcat to Apache**L-6 Hours**

JDBC Connectivity: Overview of JDBC - JDBC Drivers - Connecting Database and Connection Pools-Defining a Data Source in Tomcat and Defining a Data Source Manually Modifying the Deployment Descriptor-Detecting and closing leaked database connections. **Connecting Tomcat to Apache:** HTTP Connectors- Web Server Advantages - Connector Protocols-Start Tomcat- Worker-Load balancing applications creation across multiple Tomcat Instances - Issues-Server affinity via sticky sessions-Shared session storage-Replicated sessions via Tomcat Clustering-Shared Hosting.

UNIT 4 Apache-Tomcat Security and Apache-Tomcat Administration**L-6 Hours**

Apache Tomcat Security: Configuration-Security Considerations-File System Security and Filesystem Recommendations-Java Security Manager-Granting Permissions to Java Apps-Java Permissions-Security Manager Tips and Security Realms- Different Types of Realms and Configuring a Realm-Users and Roles - authentication and authorization. **Apache-Tomcat Administration:** System Management Overview-Installation and Configuration -Application Deployment-Security Management-Monitoring

and Performance-SSL Configuration- Log Rotation-Security Management-Perform failover procedures

UNIT 5 Load Testing and Certificate Management

L-6 Hours

JMeter Application- Load Testing with JMeter- JMeter Installation- Add Report Listener-Run the Test Plan-Viewing the Results. **Certificate Management:** CSR generation-Certificate installation - Certificate renewal process.**Standard Operations Procedures:** Apache-Tomcat services Start and Stop-Log Management. **Applying service packs:** A brief overview of Patch process - installation of fix packs-Rollback of fix packs-Perform Apache-Tomcat upgrades/migrations.

Total: 30 Hours

G.LAB EXPERIMENTS

Total: 30 Hours

TASK 1:

- Create a web application with web.xml configuration file
- Mapping a Servlet to a URL using web-app, resource-ref, env-entry, ejb-ref elements

TASK 2: Deploy a web application using the Apache Tomcat Manager:

- Use the Apache Tomcat Manager web application to deploy a web application.
- Access the Apache Tomcat Manager at <http://localhost:8080/manager/html> and enter your Apache Tomcat login credentials.
- Upload the WAR file or directory containing the web application files and deploy it using the Apache Tomcat Manager interface.
- Verify that the application is running by accessing it at the URL <http://localhost:8080/<app-name>>, where <app-name> is the name of your web application

TASK 3: Deploy a WAR file:

- Create a web application as a WAR (Web Application Archive) file and deploy it in Tomcat. Copy the WAR file to the webapps directory in the Apache Tomcat installation directory.
- Apache Tomcat will automatically extract the WAR file and deploy the web application.
- Verify that the application is running by accessing it at the URL <http://localhost:8080/<app-name>>, where <app-name> is the name of your web application.

TASK 4: Develop a simple web server with Jakarta Project

TASK 5: Installation of MySQL and Configuration of Apache Tomcat to Connect to a Database and Deploy a JDBC Driver

TASK 6: Configuration of JNDI Resources and JavaMail Session and understand the working of environment and resource elements.

TASK 7: Configuration of an AJP Connector:

- Create the mod_jk.conf file
- Configure httpd.conf

- Enable virtual hosting as a standalone server using jk2 or webapp and create separate JVM for each host.

TASK 8: Create and Add Own Policies, enable the Java Security Manager and Create a Data Source Realm with the protection of passwords and FORM-based authentication

TASK 9: Configuration of Apache Tomcat for SSLin applications

TASK 10: Perform Backup, Restoration, disaster recovery and Apache-Tomcat vulnerability fixes

TASK 11: Create a Thread Group:

- Open JMeter
- Create a new Test Plan.
- Add a Thread Group to the Test Plan, and configure the Thread Group with the number of threads, ramp-up period, and loop count that you want to use for your load test.

TASK 12: Create and run the test plan by configuring the Thread group and HTTP Request and view the results

TASK 13: Identification of Known errors and associated solutions while working as Apache-Tomcat admin and troubleshoot different types of error codes for Apache-Tomcat issues.

TASK 14: Troubleshoot Apache Tomcat with unknown website authentication.

TASK 15: Database connection troubleshooting with Apache Tomcat.

Part 2:

Use Case1 : Websites Hosting

Hosting websites on-premises or in the cloud, Microsoft IIS is the best choice for deploying websites using Microsoft technology such as ASP.NET pages. The websites interact with the customers requirements, such as access control, timesheet, and so on; this helps to automatically create and configure a website is a differential of the solution. This will be used to satisfy the untrained end user to install and configure a web application. However, Microsoft IIS has great features for advanced users, allowing for example, to host several sites on the same server and segregate them for internal publication such as on an intranet, as well as for external publication on the Internet.

Tools: Apache Tomcat Webserver

Use Case 2 : Business web application hosting

To build a business web application to support operations which are hosted on Windows servers using IIS. The utilisation of IIS is probably considered fairly basic with it running HTTP based HTML websites using winforms and re-write rules. IIS is being used as a centralized web farm management tool for Business web application hosting. IIS will deploy and manage web sites and applications across large farms of web servers from a centralized location. Microsoft IIS makes it easy to centrally deploy

and manage the configuration and content across a bunch of different web servers. also it is used to manage several internal monitoring websites that keep a track a of different threads.

It allows us to host a variety of these websites across the business to manage different applications within the service. The business problem is addresses is that it allows us to easily host and change these monitoring websites to suit our business needs. Because Installation, usage, configuration is easy, without any lack of functionality

Tools :Apache Tomcat Web Server, HTML

Use Case 3 :Server- client deliverables hosting communication

In small projects, Microsoft IIS acts as the client deliverables. also Server- client deliverables projects require IIS to implement for overall project hosting and delivery. Because Microsoft IIS is very easy to enable and uninstall, and its GUI is very informative. The Microsoft internet web server facilitates implementation of AWS cloud seamless. It is very useful due to its simplicity for small projects which primarily require Cloud requirements in AWS..

Tool :Apache Tomcat Web Server,

Total: Theory+Lab= 60 Hours

H. Learning Resources

i.Text Books:

1.Aleksa Vukotic, James Goodwill,Apache Tomcat 7”, 2011. [Unit 1 - 5]“

ii.Reference Books:

1.Tanuj Khare, “Apache Tomcat 7 Essentials”, 2012. [All Units]

iii.Online Resources:

1. “Apache Tomcat®”, Accessed on: Nov. 18, 2023 [Online], Available: [https://tomcat.apache.org/download-10.cgi /](https://tomcat.apache.org/download-10.cgi/)
2. “Apache Tomcat Server from Beginners to Advanced”, Accessed on: Nov. 20, 2023 [Online], Available: <https://www.udemy.com/course/apache-tomcat-for-beginners-and-advanced/>

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS265	WEBSHERE APPLICATION SERVER	2	0	2	3

A. Preamble

This course covers the essentials to installing and administering Websphere Application Server. Websphere Application Server covers vital topics like choosing your server roles, IBM server environments, server modernization and migration efforts, storage options, identity and access, and virtualization. This course also highlights the benefits of upgrading to the Websphere Application and shows how to automate installation, Efficiency for cloud-native applications, Application modernization, and Automated operations and security. This course is to learn the essentials of Websphere Application Server and looks at configuring and deploying to WebSphere Application server environments,

B. Pre-requisite course

10212CS264 - Application Server Technologies

C. Course Objectives:

Learners are exposed to:

- Enable and configure WebSphere Application Server and its Building blocks
- Install and Manage Control-M Automation API and modern application release process
- Monitor and optimize the performance of Red Hat - Jboss Application Server
- Troubleshoot AutoSys Automation platforms and their service delivery
- Create and managing Unix Commands used in the WebSphere Application Server

D. Course Outcomes:

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Design web server application by enabling the WebSphere application server in windows platform and also create database connections using system management administration	K3/S3
CO2	Build an interactive Profiles creation with WAS Configuration tools using Console and Command Line Administration.	K3/S3
CO3	Utilize SSL Certificate in WebSphere Application Server to secure client server communication using SSL configurations	K3/S3
CO4	Make use of monitoring standard operations procedures, and deployment manager procedure for troubleshooting the WebSphere application server.	K3/S3
CO5	Apply Applying service packs to upgrade WAS and Troubleshoot WAS Admin Issues using Installation, Rollback of ifix/fix packs	K3/S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of COs with Program outcomes and**Programme Specific Outcomes:**

Cos	PO 1	PO2	PO3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PSO 1	PSO 2	PSO 3
CO1	3	2	3	3	3				2	2	2		3	3
CO2	3	2	3	3	3				2	2	2		3	3
CO3	3	2	3	3	3				2	2	2		3	3
CO4	3	2	3	3	3				2	2	2		3	3
CO5	3	2	3	3	3				2	2	2		3	3

High- 3; Medium-2; Low-1

F. Course Contents**UNIT 1 Introduction to WebSphere Application Server****06 Hours**

WAS Introduction: IBM WebSphere family and edition - WebSphere Application Server (WAS) - WAS Architecture - operational modernization components - **WAS Administration:** System Management Administration - Database Connections. **Introduction to WAS Management:** hardware and software prerequisites - WebSphere Application server installation.

UNIT 2 WAS Administration and Session Management**06 Hours**

WAS System Management - System Management Overview – System Management functionality - problem understanding- Support Assistant.**Installation and Configuration tools** – WAS Installation - WAS Configuration - Managing Profiles. **WAS Administration** - Console Administration - Command Line Administration - WebSphere Processes Administration.**Managing Database Connections:** WAS support policy – troubleshootingpool issues in WAS database connections. **Session Management:** Session configuration – Session identifiers - Session properties - Session security

UNIT 3 Security Management in WebSphere Application Server**06 Hours**

WebSphere Application Server Application Deployment: Installation process –updating process - enabling process. **Security Management:** Security in applications - Security in resources - WebSphere Application Server Performance Monitoring. **SSL Configuration:** Secure Sockets Layer (SSL) configurations – outbound and inbound connections - client certificate - server certificate – Trust store - Handshaking protocol – Client/server authentication - **Log rotation** - **Port Assignment:** Default Port Assignments– **Security Management in Port Assignment:** Manage security configurations - Security Management topologies - **Backup and Restoration**

UNIT 4 WebSphere Application Server Standard Operations**06 Hours**

Standard Operations Procedures: configurations - Application Migration Toolkit - WebSphere Application Server Migration planning - Troubleshooting migration. **Deployment**

Manager Procedure: Deployment Manager Start and Stop - Node agent Start and Stop - Application Server Start and Stop - Web Server Start and Stop - Deployment Manager Migration Restrictions

UNIT 5 WAS Admin Issues Handling

06 Hours

Applying service packs: WAS Patches/Upgrade process overview - Installation Manager -
Installation of ifix/fix packs: Download interim fixes – Install the interim fix – Procedure – configurations – command line. Install the fix – Procedure – configurations – command line.
Rollback of ifix/fix packs: Uninstalling interim fixes – Procedure - configurations – command line. Uninstalling fixes – Procedure - configurations – command line.

H. LAB EXPERIMENTS

Total: 30 Hours

UNIT 1 (LAB)

TASK 1: Downloading WebSphere Application server from the IBM Cloud server using Guest user

TASK 2: WebSphere Application Server Database Connections Procedure using Console and Command-Line

TASK 3: Practical study about Security Management Monitoring and Performance result for SSL Configuration and Port Assignment

UNIT 2 (LAB)

TASK 4: WAS Installation and handling configuration tools

TASK 5: Practical study of efficient and errorless WAS Configuration and its procedural aspects

TASK 6: Managing Profiles for WebSphere Application Server with Errorless Connections Procedure.

TASK 7: Practical study of WebSphere Application Server updating process and enabling process

TASK 8: Secure Sockets Layer (SSL) configurations based on outbound and inbound SSL connection attributes, which include:

A. Client Certificate SSL configurations

B. Server Certificate SSL configurations

TASK 9: Practical Steps to modify the existing SSL configuration by enabling the Handshaking protocol and verify the following

- A. Trust manager control of certificate trust decisions
- B. Key manager control of certificate identities
- C. Certificate revocation checking with the trust manager

TASK 10: Write and study about Console and Command-Line for the following:

- A. Default Port Assignments
- B. Backup and Restoration

UNIT 4 (LAB)

TASK 11: Practical study of WebSphere Application Server's Application Migration Toolkit configurations

TASK 12: working principles for WebSphere Application Server Migration Troubleshooting

TASK 13: Write and study about Console and Command-Line for the following

- A. Deployment Manager Start and Stop
- B. Node agent Start and Stop
- C. Application Server Start and Stop
- D. Web Server Start and Stop

UNIT 5 (LAB)

TASK 14: Downloading Installation Manager and Installing and Rollback the following

- A. Installing ifix/fix packs and configuring using the command line.
- B. Rollback of ifix/fix packs using the command line

TASK 15: Working principles of Admin Issues Handling by listing Known issues and resolving with the associated solutions using the command line.

Total: Theory + Lab= 60 Hours

Use Case1 : Websphere Application server Installation and Configuration

To build a Websphere Application server, Install WebSphere Application Server 9.0.5 and also use the Identity Manager Service Center while WebSphere Application Server Network Deployment 9.0.5. using IBM® Installation Manager. Also Create clusters with WebSphere Application Server 9.0.5 for balance workload. For this, user must create two server clusters in WebSphere Application Server environment. One cluster hosts the Identity Manager application and cluster is used as a messaging service.

Tools: Websphere Application server

Use Case2 :Installation and configuration of IBM HTTP Server and WebSphere Web Server plug-in

Install the IBM HTTP Server and the WebSphere Web Server plug-in on the same computer that has the deployment manager. Also, install the IBM HTTP Server and the WebSphere Web Server plug-in on a separate computer for additional security and load balancing. In this concern, if any Performance issues arise, user must create WebSphere Application Server performance tuning tasks to make WebSphere Application Server initial configuration actions can take to ensure that WebSphere Application Server runs correctly.

Tools: IBM HTTP Server and WebSphere Application server

Use Case 3: SSL Configuration and Port Assignment

In the context of SSL Configuration and Port Assignment in smaller projects, users need to do the following step to improve the configuring SSL for secure communication, followed by the installation and configuration of the Identity Manager Server. Subsequently, the IBM Global Security Kit (GSKit) is to be utilized to generate the necessary certificates. Additionally, the installation and configuration of the Identity Manager Server, along with the required supported middleware, including the directory server, should be conducted. In the case of a cluster configuration where the directory server is situated on a separate computer, it is crucial to verify the correct operation of the initial configuration. It is recommended to consult the IBM Security Identity Manager Configuration Guide for further information. The IBM Global Security Kit (GSKit) is provided as part of the IBM Security Directory Server, which is installed during the initial configuration.

Tools :IBM® Global Security Kit (GSKit) and WebSphere Application server

I. Learning Resources

i. Text Books:

1. A. Black, M. Everett, D. Draeger, K. Miller, R. Iyer, K. McGuinness, D. Patel, M. Herescu, T. Gissel, M. Betancourt, M. Casile, Y. Tang, A. Beaubien, “IBM WebSphere Application Server for Distributed Platforms and z/OS: An Administrator's Guide”, IBM Press. 2005 [Unit 1 - 5]

ii. Reference Books:

1. IBM Redbooks, “WebSphere Application Server V6 Scalability and Performance Handbook”, IBM 2005. [All Units]
2. IBM Redbooks, “ WebSphere Application Server V6 System Management & Configuration Handbook”, IBM 2005. [All Units]
3. Carla Sadtler, “IBM WebSphere Application Server V5.1 system management and configuration : WebSphere handbook series”, IBM International Technical Support Organization, WebSphere handbook series.; ITPro collection.; IBM redbooks 2005. [All Units]

iii. Online Resources:

1. “WebSphere Hybrid Edition”, Accessed on: JAN, 2024 [Online], Available: <https://www.ibm.com/docs/en/WebSphere-hybrid>

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS301	WEBSphere MQ	0	0	2	1

A. Preamble

The course provides a comprehensive overview of WebSphere MQ, a market-leading messaging integration middleware product. This course is focused on providing a reliable, scalable, secure, and high-performance transport mechanism using MSSC commands, Queue Manager Configuration, Channel Triggering, and Troubleshooting. Additionally, the course covers various interfaces and services for configuring, managing, and monitoring operations, as well as log management and channel troubleshooting in WebSphere MQ.

B. Pre-requisite course

10212CS264 - Application Server Technologies

C. Course Objectives:

Learners are exposed to:

- Enable applications to send messages to a queue, where the receiving application can process them.
- Basic Configuration of WebSphere, Queue Manager and Channels, Remote Queue, Security Policy, Queue Handling
- Use of Backup and Restore Queue Manager in an Application
- Monitor the WebSphere MQ Channels intercommunication and triggering with other platforms.
- Troubleshoot and maintain the Log management.

D. Course Outcomes:

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Build and configure the WebSphere MQ for the given application	K3/S3
CO2	Design the Queue manager and handle various types of messages and properties	K3/S3
CO3	Apply regular backup and recovery procedures for MQ configurations for data.	K3/S3
CO4	Organize Interprocess Communication to manage message channels for inter-queue manager	K3/S3
CO5	Experiment the log management, diagnose, and resolve the common issues.	K3/S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of COs with Program outcomes and Programme Specific Outcomes:

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	2				3									2
CO2	2	2	3		3									2
CO3	2	2	3		3									2
CO4	2	2	3		3									2
CO5	2	2	3		3									2

High- 3; Medium-2; Low-1

F. Course Contents

TASK 1: Study on Message Queue overview, Architecture and managing Queue Manager Objects.

TASK 2: WebSphere MQ Installation and software prerequisites for MQ installation and MQ Administration.

TASK 3: Implementation of MQSC commands

1. Queue Managers
2. Clusters
3. Channels
4. Queues
5. Name lists
6. Process definitions
7. Authentication Information Object.
8. Disaster recovery/ failover procedures and Reset channel procedure

TASK 4: Creation, Configuration, Dumping, and Restoring a Queue Managers configuration and MQ objects

TASK 5: Study and implement Dead Letter Queue Handling (DLQ).

1. Invoking the DLQ Handlers
2. Display the DLQ handler rules Table
3. Operation on rules table.

TASK 6: Import/Export and Backup messages from the queue in WebSphere MQ

TASK 7: Queue manager intercommunication and client connections in WebSphere MQ

TASK 8: Troubleshooting error in Queue manager intercommunication and client connections

TASK 9: Study and Implementation of Channel triggering process in the message queue.

TASK 10: Demonstrate messaging with WebSphere MQ with the following process

1. Send/Receive a message from the queue manager

2. Cold start recovery of queue manager

TASK 11: Implement the Standard Operations Procedures for Queue Manager Stop/Start

TASK 12: Study and Implementation of different types of Logging.

TASK 13: Build the WebSphere MQ infrastructure with Log Management

TASK 14: Implement the messaging activity with WebSphere MQ

TASK 15: Demonstrate how to implement Channel troubleshooting with WebSphere MQ

TASK 16: Demonstrate to Applying service packs in

A. Patch process

B. Certificate validation policies

TASK 17: Demonstrate the standard operations procedures for the Installation of fix packs

TASK 18: Demonstrate the Standard Operations Procedures for the Rollback of fix packs

TASK 19: Demonstrate the Standard Operations Procedures for Performing version upgrade

TASK 20: Study on overview of MQ reason.

Use Case 1 : Healthcare Information Exchange

A hospital needs to exchange patient information securely between different departments (radiology, pharmacy, billing) and external partners (insurance companies, laboratories). Ensure secure and reliable exchange of sensitive patient information. Deploy WebSphere MQ to create a secure messaging infrastructure that encrypts and delivers messages between the hospital's internal systems and external partners. Configure security settings to comply with healthcare regulations like HIPAA.

Use Case 2 : Customer Support Ticketing System

A global company needs to manage customer support tickets and ensure they are routed to the appropriate support teams based on region and issue type. Efficiently route and manage support tickets. Implement WebSphere MQ to create a messaging system that routes support tickets to the appropriate teams. Use message selectors and channel triggering to ensure tickets are automatically assigned based on predefined criteria (e.g., region, issue type).

Use Case 3 : Financial Transactions Processing

A financial institution needs to process millions of transactions daily with high reliability and security. The system must ensure that transactions are processed in order and without loss. Implement a reliable messaging system to handle transaction messages between various banking applications. Use WebSphere MQ to set up a message queue that ensures all transaction messages are delivered and processed in sequence. Configure queue managers to handle load balancing and failover to maintain high availability.

G. Learning Resources

i. Text Books:

1. "WebSphere MQ Primer: An Introduction to Messaging and WebSphere MQ" by Chris Lovett and Timothy J. Parkin
2. "IBM WebSphere MQ V7.1 and V7.5 Features and Enhancements" Cezar Aranha, ,Craig BothBarry Dearfield · 2013
3. "IBM WebSphere MQ V7.0 Features and Enhancements" by Saida Davies, T.Rob Wyatt, et al.

ii. Reference Books:

1. IBM WebSphere MQ A Complete Guide Kindle Edition by Gerardus Blokdyk (Author)
Format: Kindle Edition
2. "IBM MQ V8 Features and Enhancements" by Valerie Lampkin, Saida Davies, et al.

iii. Online Resources:

1. "IBM MQ Documentation", <https://www.ibm.com/docs/en/ibm-mq/9.1>
2. "IBM WebSphere Application Server V9 - IBM Training"
<https://www.ibm.com/training/path/ibm-websphere-application-server-v9-2118>

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS267	MIDDLEWARE TECHNOLOGIES	1	0	2	2

A. Preamble

This course delivers comprehensive course on Unix Commands, Operating System Administration, Workload Scheduler Concepts, BMC Control-M Administration, Database Concepts, Data Backup Strategies, and Data Disaster Recovery. This course is designed to equip you with essential skills and knowledge in managing and optimizing IT infrastructures, ensuring smooth operation and resilience against potential disruptions. Throughout this course, you will delve into the foundational principles and advanced techniques required to effectively administer operating systems, automate workload scheduling, and implement robust data protection strategies. Students will be well-equipped to design, implement, and manage middleware solutions that enable efficient and secure communication and data exchange between diverse systems and applications.

B. Pre-requisite course

10212CS264 - Application Server Technologies

D. Course Objectives:

Learners are exposed to:

- Understand and effectively utilize essential Unix commands for system management and troubleshooting.
- Effectively plan, automate, and manage complex workflows and job scheduling in IT environments.
- Install, configure, and utilize BMC Control-M for efficient workload scheduling and management.
- Understand database concepts, database languages, and SQL queries for effective data manipulation and management.
- Optimize resource utilization and scheduling efficiently.
- Implement effective backup and recovery strategies

E. Course Outcomes:

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Apply essential Unix utility commands to perform basic system operations.	K3
CO2	Implement scheduling techniques to manage server workloads.	K3
CO3	Utilize SSH and Control-M Configuration Manager (CCM) functions to manage and secure server communications.	K3
CO4	Make use of fundamental database concepts to understand and manage data effectively.	K3

CO5	Develop and execute backup plans using internal resources, ensuring regular data backups and quick restoration capabilities.	K3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of COs with Program outcomes and Programme Specific Outcomes:

Cos	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PS O1	PS O2	PSO 3
CO1	2	2	3	3	3				2	2	2		3	2
CO2	2	2	3	3	3				2	2	2		3	2
CO3	2	2	3	3	3				2	2	2		3	2
CO4	2	2	2	3	3				2	2	2		3	2
CO5	2	2	2	3	3				2	2	2		3	2

High- 3; Medium-2; Low-1

F. Course Contents

UNIT 1 Unix Commands & Operating System Administration

3 Hours

Unix commands - Utility commands - directories commands - File creation - listing of files and folder command -File Permission - File operation - Introduction to server architecture - Unix Command Line & GUI - GNU & Unix Commands - Getting Started with PowerShell - Introduction to Server Storage.

Unit 2: Introduction to Workload Scheduler Concepts

3 Hours

Server Overview: Server Architecture - Scheduler - Scheduling Terminology - Types of Schedules - Workload automation - Tools for Workload Scheduling: BMC Control-M – Installation – Control-M Architecture - Benefits of Scheduling.

Unit 3: BMC Control-M

3 Hours

Control-M Administration: Overview of Control-M - Resource Planning - Create and edit new and existing jobs - Manage job definitions and calendars - modify job parameters & Restart jobs - View and Modify – NDP – Job Monitoring - Log Management - Time Management and - Workload management – Control-M/Server security configuration – SSH – CCM Functions

Unit 4: CCM Functions, Job Forecasting and Database Concepts

3 Hours

Database concepts - Database Languages - SQL Concepts and Queries - Keys in Database. Data Backup: Resilience strategies - Backup Considerations - Techniques to protect and store sensitive data - Data Classifications - Data Breach and its cost - Business Impact Assessment - Assessing Risk- Planning for Continuity - Service Levels.

Unit 5: Data Disaster recovery**3 Hours**

Data Protection Solutions - Roles and Responsibilities - Recovery time and recovery point - Storage solutions, Replication technology - Backup and recovery strategies: In-house, Outsourced and Cloud.

Total: 15 Hours**G. Lab Experiments****Total: 30 Hours**

Task 1 : Enter these commands at the UNIX prompt, and try to interpret the output: echo, passwd, date, who, cal, man, uname, pwd, which, whereas and also give proper command for the following scenarios

- a. Change back into your home directory.
- b. Make subdirectories called work and play.
- c. Delete subdirectory called work.
- d. Copy file /etc/passwd into your home directory.
- e. Move it into the subdirectory play.

Task 2: Create a one or more files and directories using unix commands and reports the following information on the file:

- a. File type
- b. Number of links
- c. Time of last access
- d. Read, write, and execute permission
- e. Fetch unique data from a file
- f. Sort File Content

Task 3: Use the vim Editor for the following scenarios

- . Create File, Make Changes and Close file without saving
 - a. Insert data to file
 - b. Insert data in new line
 - c. Save content of file

Task 4: Use the Command Line Interface to perform the following operations

- a. Changing file permission
- b. Changing ownership of file/directory
- c. Show the difference between Grep/ sed/ awk commands

Task 5: Use powershell cmdlets to perform following operations

- a. Create a new Folder named “Lab” and change the location to the Lab folder
- b. Create a new CSV file and enter the students’ details.
- c. Read the CSV file content and display the student’s name.

Task 6: Installation and getting started with workload scheduler

Task 7: Simulate Workload Automation Job Schedule Changes using workload scheduler

Task 8: Create and run an application server in workload scheduler

Task 9: Create the n number of process and added to your Process Library with the same specifications as the original jobs.

Manage the Scheduler jobs in this environment and perform modification in calendars.

Task 10: Create a job depending on your business needs, you can complete the following actions:

- a. Run a job immediately.
- b. Move the workload to the Dynamic Workload Console to run your job with more complex scheduling conditions.
- c. Monitor your Job from the admin control.

Task 11: Demonstrate how to view and modification of system parameters in process and log management.

Task 12: Implementation of DDL and DML commands of SQL with suitable examples

DDL commands:

- a. CREATE
- b. ALTER
- c. DROP
- d. RENAME
- e. TRUNCATE

DML COMMANDS:

- a. INSERT
- b. UPDATE
- c. DELETE
- d. SELECT

Task 13: Implement the Aggregate functions along with filtering and Sorting on table in database.

Task 14: Implementation of Database Backup & Recovery Commands.

- a. Rollback
- b. Commit
- c. Save point

Task 15: Make a database on cloud platform

- a. Create a database on Relational Database Service (RDS) of Amazon Web Services

- (AWS)
- b. Building Resilience through Cloud Databases

Total: Theory + Lab= 45 Hours

Use Case 1 : Scheduling Optimization

You are tasked with optimizing the scheduling of various batch jobs on your company's server. The goal is to automate as many tasks as possible to reduce manual intervention and increase efficiency.

- Understand the server architecture and its scheduling capabilities.
- Learn and use scheduling terminology to communicate effectively with your team.
- Implement different types of schedules (time-based, event-based) for various tasks.
- Use workload automation techniques to streamline processes.
- Install BMC Control-M and configure it to manage the server's workload.
- Describe the benefits of scheduling to your management to justify further investment in automation tools.

Use Case 2 :Data Breach

Your company has experienced a data breach, and you need to implement a comprehensive data protection and disaster recovery plan to prevent future incidents and minimize downtime.

- Apply data protection solutions to safeguard sensitive information.
- Define roles and responsibilities for data protection within your organization.
- Establish recovery time and recovery point objectives to guide your recovery efforts.
- Evaluate and implement appropriate storage solutions to ensure data availability.
- Use replication technology to enhance data redundancy and availability.
- Develop backup and recovery strategies, considering in-house, outsourced, and cloud options.
- Perform a business impact assessment to understand the effects of data loss.
- Assess risks and implement mitigation strategies.
- Plan for continuity to ensure business operations can resume quickly after a disaster.
- Set and measure service levels to ensure data management and recovery meet organizational requirements.

Use Case 3 : System Upgradation

Your company is planning a major system upgrade, and you need to ensure that all scheduled jobs and database operations continue to run smoothly during and after the transition.

- Use CCM functions to configure Control-M for the upcoming changes.
- Perform job forecasting to predict the impact of the upgrade on scheduled jobs.
- Understand database concepts to manage the underlying database supporting your applications.
- Write SQL queries to retrieve and manipulate data as needed.
- Implement data backup strategies to ensure data resilience during the upgrade.
- Classify data to determine the appropriate protection measures.
- Assess the risk of data breaches and plan for continuity in case of system failure.

- Define service levels to ensure that critical operations have the required performance and availability.

H. Learning Resources

i. Text Books:

1. Kerrisk, M. (2010). *The Linux Programming Interface*. No Starch Press. ISBN 978-1-59327-220-3. [Unit 1-2]
2. Ding, Q. (2021). *Control-M: A Practical Guide* (1st ed.). Packt Publishing. [Unit 3]
3. Price, J. (2023). *Learning SQL: Master SQL Fundamentals*. O'Reilly Media. [Unit 4-5]

ii. Reference Books:

1. Stevens, W. R., & Rago, S. A. (2013). *Advanced Programming in the UNIX Environment*. Addison-Wesley Professional.
2. Forta, B. (2023). *SQL in 10 Minutes, Sams Teach Yourself*.

iii. Online Resources:

1. “Control-m Documentation”, Accessed on: June, 2024 [Online], Available: <https://www.bmcsoftware.es/it-solutions/control-m.html>
2. “Database Creation”, Accessed on: May, 2024 [Online], Available: <https://aws.amazon.com/free/database/>

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS268	DATA BASE ADMINISTRATION USING MS SQL SERVER	2	0	2	3

A. Preamble

This course trains the student to control and regain the databases in System Architecture. It encompasses how to administrate, mitigate, monitor, control risk in database administration, configure in the cloud, migrate from one to another, etc. It adds to the existing capability of programming practice.

B. Prerequisite Courses

10211CS207- Database Management Systems

C. Course Objectives

Learners are exposed to

- overview of database systems and explains how to install SQL Server 2019 and its components.
- “Transact-SQL Language,” which is intended for end users and application programmers of the Database Engine.
- all objectives of Database Engine system administration.

D. Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Planning the installation of the SQL server and explaining using front-end tools for database engines.	K3/S3
CO2	Using the MSSQL objects, database objects, and Transact-SQL (T-SQL).	K3/S3
CO3	Managing Database Security and concurrency control.	K3/S3
CO4	Administrating SQL Server.	K3/S3
CO5	Performing Backup and Recovery techniques, automating system administration.	K3/S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of Cos with Program Outcomes and Programme Specific Outcomes

COs	P O 1	P O 2	P O 3	P O 4	P O 5	P O 6	P O 7	P O 8	P O 9	PO 10	P O1 1	PS O1	PS O2	PS O3
CO1	3				3								3	
CO2	3	3	3		3								3	3
CO3	3	3	3		3								3	
CO4	3				3								3	3
CO5	3	2	3		3								3	

High- 3; Medium-2; Low-1

F. Course Contents**Unit 1 RDBMS Overview and Installation****4 Hours**

Relational Database Systems- Introduction- data base design- normal forms- ER model- Syntax conversions- SQL server edition and management tools-Planning installation and installation of SQ server on Windows or Ubuntu- Front-End Tools for Data Base(DB) Engine- SQL Server Management Studio(SSMS)- using SSMS with the DB Engine-Authoring activities using SSMS. (chapters-1,2,3)

Unit 2 MSSQL Objects and T-SQL**8 Hours**

SQL components- Basic Objects- Data Types- Transact-SQL(T-SQL) Functions- NULL values- Database Objects: Create, Modify, Remove- Queries- Select statement with its clauses and functions- sub-queries- join operator- correlated subqueries- table expressions- Modification of table contents (DML):-insert, Update, Delete, other T-SQL modification statements and clauses- Stored procedures- user-defined functions- system catalog- general interfaces- Proprietary Interfaces- Indices- types- T-SQL and Indices- Guidelines for creating and using Indices. (chapters-4,5,6,7,8,9,10)

Unit 3 Microsoft SQL Security and SQL Server database**6 Hours**

Views- DDL Statements and views- DML Statements and Views- Security System of DB Engine- Encrypting data- Authentication- Schemas-Data Base Security- Managing DB Security using SSMS- Managing DB Security using T-SQL statements- Roles-Authorization- Change Tracking- Concurrency control-Transaction logs- editing- locking-isolation levels. Triggers- Create, modify, delete- application areas of DML: Triggers- After- instead of- Application areas of DDL Triggers- DB Level triggers- Server-Level Triggers. (Chapters: 10,11,12,13,14)

Unit 4 SQL Server: System Administration**6 Hours**

System Databases- Master DB, Model DB, TempDB, MSDB- Disk storage- Data Pages- Properties, types- Editing information of storage- Parallel Processing- Utilities- BCP, SQLCMD, MSSQL-CLI, SQLSERVER- DBCC Commands- Policy-based Management. (Chapter-15)

Unit 5 Backup, Recovery, and System Availability**6 Hours**

Reasons for Data Loss, Backup Methods- Full Database, Differential, Transaction Log, File or File Group- Performing Database Backup- Backing Up using T-SQL statements, Backing Up using SSMS, Determining which databases to Backup- performing Database Recovery- Automatic, Manual- Recovery models- System Availability- using a stand by server, using RAID Technology, Database Mirroring, Failover Clustering, Log Shipping, AlwaysOn- Comparison of High-Availability Components- Automating System Administration Tasks- Starting SQL server Agent- Creating Jobs and operators- Alerts. (Chapters-16, and 17)

Total: 30 Hours

Laboratory Tasks: (Tool: MS SQL Server2019, SSMS 2019) **30 hours**

Task-01: Planning and installation of MS SQL server CO1, S3

Task-02: Planning and installation of SQL Server Management Studio CO1, S3

Task-03: DDL Operations on SQL databases. CO2, S3

Task-04: Writing DML queries for single-row functions on SQL databases. CO2, S3

Task-05: Writing advanced queries for multi-row functions, sub-querying. CO2, S3

Taks-06: Writing T-SQL queries using modification clauses. CO2, S3

Task-07: Program a stored procedure or function on business scenarios. CO2, S3

Task-08: Using catalog views, perform find-outs for the following. CO2, S3

a. Finding OS path and filename of the sample database

b. Finding integrity constraints that are defined on a table

Task-09: Implement DB Security on the Database using TDE and AE methods and perform managing security using SSMS. CO3, S3

Task-10: Implement DB Security on the Database using TDE and AE methods and perform managing security using T-SQL Statements. CO3, S3

Task-11: Outperform the large data in SQL server using indices, storage, DMF. CO4,S3

Task-12: Generate a policy that disables the use of CLR. CO4, S3

Task-13: Implement Backup methods on the databases and find outs for the below CO5,S3

a. Using TSQL statements

b. Using SSMS

Task-14: Implement Automation System Administering tasks. CO5, S3

G.Learning Resources (in IEEE Format)

i. Text Books:

1. Dusan Petkovic, “ Microsoft SQL Server 2019- ABeginner’s Guide”, 7th edition, Mc Graw Hill, 2020.

ii. Reference Books:

1. Kellyn Gorman et al, “ Introducing Microsoft SQL server 2019”, First Edition, Packt Publishing Ltd, 2019.

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS269	ORACLE DATABASE ADMINISTRATION	2	0	2	3

A. Preamble

This course trains the student to control over the databases in Oracle DataBase System Architecture. It encompasses how to administrate, mitigate, monitor, risk control in database administration, configure in cloud, migration from one to another, etc. It addends the existing capability of programming practice.

B. Prerequisite Courses

10211CS207

Database Management Systems

C. Course Objectives

Learners are exposed to

- Exercise SGA instance management, file & tablespace management, perform user administration & security and table & index management.
- Install and configure a working Oracle database, create tablespaces and files, manage security and user access, and create tables, index-organized tables, cluster tables, managing indexes, bitmap indexes, function-based indexes and bitmap join indexes.migration from Oracle database to cloud-based infrastructure.

D. Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Perform data retrieval using SQL.	K3/S3
CO2	Report aggregated data from single and multiple databases using SQL.	K3/S3
CO3	Employ database using Oracle.	K3/S3
CO4	Articulate Oracle database architecture with manageable schema objects.	K3/S3
CO5	Manage data movement, backup, security with database.	K3/S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of Cos with Program Outcomes and Programme Specific Outcomes

COs	P O1	P O2	P O3	P O4	P O5	P O6	P O7	P O8	P O9	P O10	P O11	PS O1	PS O2	PS O3
CO1		3	3		3								2	2
CO2	2	3	3		3								2	2
CO3	1	3	3		3								2	2

CO4	2	2	2	1	3								2	2
CO5	1		3		3								2	2

High- 3; Medium-2; Low-1

F. Course Contents

Unit 1 Basic SELECT and Interactive SQL

6 Hours

Full Syntax of SELECT Command- capabilities- execution- query estimation-Limiting rows- sorting rows in filtering- Interactive SQL- Single-Row Functions- customized output- Using conversion functions and Conditional Expressions.

Unit 2 Reporting Aggregated Data and Using Joins, SubQueries

6 Hours

Identifying Group Functions- use- GROUP BY clause- Include or Exclude-Having clause- Displaying data from Multiple Tables using equijoins and non-equijoins- self-join- Outer joins- Cartesian product- cartesian join- SubQuery- types of problems- fixing subqueries- single-row and multiple row subqueries- Set Operators- Using set operator to combine multiple queries- Union- UnionAll- Intersect- IntersectAll- Controlling Order of rows- Order By Clause.

Unit 3 Data Manipulation and Data Definition

6 Hours

Data Manipulation Language- statement- INSERT INTO- INSERT-SELECT- UPDATE- DELETE- Control Transactions- Data Definition Language- Categorize the main database objects- Reviewing Table Structure-CREATE- CREATE-SELECT- CreateX- List of data types-work- creating a simple table- constraints- types of constraints at the time of table creation- Constrains manipulation after table creation-ALTER- TRUNCATE-RENAME- schema working- schema objects.

Unit 4 Oracle Database Architecture, Environment and Managing Objects

6 Hours

Memory Structures- Process Structures- Overview of Storage Structures- Tools identification for Administering Oracle Databases- Planning Oracle Database Installation- Thin Client- Thick Client- Network Installation- Install Software- Creating database by Database Configuration Assistant (DBCA)- setting initialization parameters- Stages of Storage Startup – Shutdown- Using Alert Log files and trace files- Using data dictionary and dynamic performance views- Configuring and manage Oracle Network- Managing Schema Objects- Create and Modify tables- Managing Constraints revisited- Create Indexes- Create and use of Temporary tables – views.

Unit 5 Security, Maintenance, Backup Management and moving data

6 Hours

Transactions and UNDO data- Managing UNDO- Security- Principle of least privilege- Standard Database Auditing- Use and Manage Optimizer Statistics- Use and Manage Automatic Workload Repository (AWR)- Manage alerts and thresholds- Automatic Memory Management- Troubleshoot invalid and unusable objects- Identifying types of failure in Oracle databases- Flash Recovery area- configure ARCHIVELOG mode- Creating consistent database

backups and backup without Shutdown- creating incremental Backups- Automate database backups- Moving Data: General Architecture- Oracle Data Pump: Overview, Benefits – SQL Loader: Overview, Review- Data Pump Export and Import Clients- Data Pump Interfaces and Modes-Data Pump Import Transformations- Maintenance- Auto Tasks- Database Patching.

Total: 30 Hours

G.Laboratory Tasks:

Total: 30 Hours

Task-01: Data Retrieval using SQL-Select Statement (Tool: Oracle19c, OEM, Toad)

- i) Basic SELECT usage Queries

Task-02: Data Limiting, Substitute Restriction, Sort Output (Tool: Oracle19c, OEM, Toad)

- i) SQL data retrieval queries without using LIMIT clause
- ii) SQL data retrieval queries with using LIMIT Clause
- iii) Interactive SQL using ampersand operator
- iv) Interactive SQLqueries for sorting filtered results at runtime

Task-03: Conditional Expressions and Functions in SQL- DATE, NUMBER and CHARACTER (Tool: Oracle19c, OEM, Toad)

- i) SQL data retrieval using character functions
- ii) SQL data retrieval using number functions
- iii) SQL data retrieval using date functions
- iv) Queries using to_char conversions
- v) Queries using to_number conversions
- vi) Queries using to_date conversions
- vii) Conditional expressions in SELECT statement.

Task-04: Reporting Aggregation using Group functions(Tool: Oracle19c, OEM, Toad)

- i) Queries using GROUP BY clause
- ii) Include and exclude rows using by HAVING clause.

Task-05: Joins and Subqueries (Tool: Oracle19c, OEM, Toad)

- i) Queries for data retrieval from multiple tables using Equi-Joins, self-joins
- ii) Queries for data retrieval from multiple tables using Outer Joins
- iii) Controlling order of rows using ORDER BY clause

Task-06: Subqueries and Set Operators (Tool: Oracle19c, OEM, Toad)

- i) Data retrieval using subqueries
- ii) Combining multiple queries using set operators-UNION, UNION ALL, INTERSECT, INTERSECT ALL, DIVIDE, MINUS

Task-07: Usage of schema objects with DML and DDL (Tool: Oracle19c, OEM, Toad)

- i) Using CREATE, Create-Select, creating tables
- ii) Using Alter, Modifying database schemas and objects
- iii) Using Constraints by create-add, alter, delete
- iv) Row manipulation using INSERT INTO, INSERT FROM, UPDATE-SET

Task-08: Plan, install, manage, configure Oracle database (Tool: Oracle19c, OEM, Toad)

- i) Planning and installation of Oracle Environment- Thin client and Thick client
- ii) Creating Database using DBCA
- iii) Managing Instances – Startup and shutdown progress.

Task-09: Tracing instance, alert log, DDL log using ADRCI utility (Tool: Oracle 19c/ADRCI)

- i) Deploying Net services, log services and
- ii) interpreting alert log files, trace files, view alert log file, DDL logging, Tracing instances, namespaces, timestamp validation in logs using listener control

Task-10: Viewing Dynamic Performance and Oracle Net (Tool: OracleNetlog, LSNRCTL)

- i) Dynamic Performance views (Netlog)
- ii) Naming Methods: Easy Connect, Local naming
- iii) Naming Methods: Directory Naming, LSNRCTL and NETCA

Task-11: Mitigating Logical vs Physical Storage (Tool: OEM Express)

- i) System Table space, Sysaux tablespace, Temp Tablespace
- ii) Undo and users tablespaces
- iii) Segment Information, Extent Information, DB block and Data File information
- iv) Creating a table space using EM Express
- v) Extend Allocation-Automatic or Uniform
- vi) Logging: Force Logging, Nologging

Task-12: Handling Segment Space management and Table Full Error (Tool: EMExpress, Netlog)

- i) Automatic vs manual space management
- ii) Moving, copying and renaming database files
- iii) Simulating TableFull Error

Task-13: Database Maintenance and performance management (Tool: Oracle19c, OEM, Toad)

- i) Using Optimizer statistics
- ii) Using Automatic Workload Repository (AWR)
- iii) Managing Alerts and thresholds
- iv) Auto task
- v) Database Patching

Task-14: Consistent Database Backups (Tool: Oracle19c, OEM, Toad)

- i) Creating consistent database backup

- ii) Backing up without shutting down
- iii) Creating incremental backup
- iv) Automating database backup

Task-15: Perform Moving Data (Tool: Oracle19c, Data Pump, OEM, Toad)

- i) Using Data Pump Export and Import to move data between Oracle Databases

Total: 60 Hours

Part- B

Implementing Use case/ case study (Part-B) satisfying the following.

- Retrieving Data Using the SQL SELECT Statement, Restricting and Sorting Data, Single-Row functions, Conversion functions, Reporting Aggregated Data Using the Group Functions, Displaying Data from Multiple Tables, Using Subqueries to Solve Queries, Using the Set Operators, Manipulating Data
- Using DDL Statements to Create and Manage Tables
- Exploring the Oracle Database Architecture
- Preparing, creating, managing the Database Environment, Configuring the Oracle Network Environment, Managing Schema Objects, Managing Undo Data, Implementing Oracle Database Security, Database Maintenance, Performance Management,
- Backup and Recovery Concepts, Performing Database Backups, Moving Data, Maintenance Task, Data Guard, Standby Tools, Oracle Support, Known issues and solutions.

Use Case-01: Large Advertising & Marketing Services Company @

The project: build a secure Oracle APEX environment on Oracle Cloud for the company's internet-facing applications.

Starting point: one virtual machine running a pre-created database with APEX installed
End point: the full Apache/Tomcat/ORDS/APEX/database stack, SSL enabled.

As with any cloud provider, it takes a while to understand how to use the cloud infrastructure, particularly the networking and security. Once you have got past that, you can start work. First, the Oracle Enterprise Linux default configuration isn't good enough. You need to install a graphical desktop and also various RPMs and update OpenSSL. The firewall needs some configuration too.

Then the usual: download and install Tomcat; download ORDS, configure and deploy to Tomcat; download the Apache http server plus the APR libraries and compile. Need to be added an OpenVPN server for SQL*Net and ssh access, so that only the Apache listening ports would be exposed to the internet. Everything configured for automatic start on boot up.

Use case-02: Foremost Farms @

Foremost Farms is a milk solids processing business owned by the dairy farm families who supply our milk. That's what it means to be a cooperative – and it's why our farmer-members have a vested interest in providing the highest-quality milk possible. They own Foremost Farms.

Project: Research running Oracle Data Appliance (ODA) in Virtualized Environments. This project was about investigating the ODA virtualized architecture and understanding how best to use it with single instance databases. Not because of RAC, because of licences. One can run ODA as “bare metal” or “virtualized”. Typically, people who installed ODA more than, say, two years ago run bare metal: the two ODA nodes run normal Linux and the databases reside on ASM disc groups. This is all straightforward and should present no surprises to the DBA or the SA. However, most people installing now follow Oracle's recommendation and run it virtualized. The configuration can strike you as bizarre the first time you see it. The nodes run Xen Linux (Oracle's version, of course) and everything happens in virtual machines. The databases reside on ACFS file systems, and are created from ACFS snapshots of preconfigured databases. The networking may take a while to get your head around. In general, the virtualized platform is probably the way to go, but you do need to be a bit of a wizard with Grid Infrastructure and Xen to understand what is going on. There are many GI facilities that can be used for fault tolerance and failover without RAC, which is not something that is very well documented. Perhaps, it should be.

@ internet sources

Total: 60 hours

G.Learning Resources (in IEEE Format)

i. Text Books:

1. Tanveer A, “Oracle 19c Database Administration: Oracle Simplified” Kindle Edition, 2020.

ii. Reference Books:

Ravinder Gupta, “Oracle Database 19c DBA By Examples: Installation and Administration”, Independently published (June 26, 2021), ISBN-13 : 979-8469226970.

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS270	INTRODUCTION TO SYSTEM PROGRAMMING, z/OS COMMANDS AND PANEL ON IBM Z	2	0	2	3

A. Preamble

This course is intended to provide the foundational skills required to launch a career in the enterprise platform that powers over 68% of worldwide transactions. Companies around the world are searching for these skills to run their critical system workloads. In this course, the student will be introduced to various operating systems, mainframe applications, and get real-time hands-on experience on an IBM Z server. The skills acquired on completion of this course allows the learner to pursue a career as a mainframe application developer, system programmer, system administrator, or a DBA Practitioner.

B. Prerequisite Courses

10211CS103-Operating Systems

C. Course Objectives

Learners are exposed to

- IBM z/OS Mainframe
- z/OS Commands and Panels on IBM Z
- System Programming on IBM Z

D. Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Explain mainframe hardware components and the structured nature of the mainframe I/O architecture.	K2/S3
CO2	Demonstrate z/OS Commands and Panels on IBM Z.	K2/S3
CO3	Describe the responsibilities of JES and JCL.	K2/S3
CO4	Explain System Programming Components.	K2/S3
CO5	Explain the z/OSMF and UNIX System Services.	K2/S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of Cos with Program Outcomes and Programme Specific Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PS O1	PS O2	PS O3
CO1					3							3	
CO2	2	3	3	3	3							3	3
CO3	2	3	3	3	3							3	
CO4				3	3							3	3
CO5		2	3	3	3							3	3

High- 3; Medium-2; Low-1

F. Course Contents**Unit 1 Introduction to IBM z/OS****L-6 Hours**

Mainframe Environment: Computer Architecture - Mainframe Architecture – Virtualization - Mainframe Operating System - Diving Deeper into z/OS - Mainframe Infrastructure - IBM Z components - Mainframe Features - IBM Z Configuration Setup – Millicode - HMC and SE – Sysplex - Mainframe Security - z/OS Security - Security Addendum - Transaction Level Security - Crypto Express Key Management – IOCDS - Activation Profiles - IPLing an Operating System - z/OS MVS System Initialization Logic Initial Program Load (IPL).

Unit 2 Introduction to z/OS Commands and Panels on IBM Z**L-6 Hours**

Introduction to Commands and Panels - Introduction to TSO and ISPF - z/OS 2.3 ISPF User Guide - Creating, editing, and displaying members in ISPF - Working with Data Sets - Allocating a data set in TSO - Working with data sets - TSO Commands.

Unit 3 Introduction to JES and JCL**L-6 Hours**

System Programming - JES and JCL - Submitting and Viewing JCL - Lab Familiarization - Getting Connected - Terminal and Password Issues - Access lab environment - Submit a job - JCL exercises - JCL Procedures

Unit 4 System Programming Components**L-6 Hours**

VSAM - z/OS Components - z/OS System Libraries - Application Infrastructure - Language Environment - Generation data groups – RAIM - IBM DB2 12 for z/OS – Networking Utilities

Unit 5 z/OSMF and UNIX System Services**L-6 Hours**

z/OSMF Interactive Content - Lab - Using z/OSMF - IBM z/OS Management Facility V2R3 - UNIX System Services - USS file systems - ISHELL and hierarchical file system - USS Processes and Permissions

Total: 30 Hours**G. Lab Experiments:****Total: 30 Hours****z/OS Commands and Panels****Task 1. System Familiarization**

- Section 1: Gathering information and accessing the lab image
- Section 2: The logon screen
- Section 3: ISPF Primary Menu
- Section 4: ISPF Settings
- Section 5: ISPF Settings: Action bar choices
- Section 6: View a data set
- Section 7: Exit ISPF
- Section 8: Log Data Set Disposition
- Section 9: Logoff

Task 2. Allocate new data sets

- Section 1: Lab exercise assistance
- Section 2: Allocate a Physical Sequential data set
- Section 3: Allocate a Partitioned Data Set
- Section 4: Allocate a Partitioned Data Set Extended

- Task 3. ISPF editor primary commands
 - Section 1: Create a PDS member
 - Section 2: Edit the LITLGAME member
 - Section 3: Ending the exec
- Task 4. ISPF editor line commands
 - Section 1: Create a new member and copy data
 - Section 2: Copy / overlay commands
 - Section 3: Insert lines
 - Section 4: Add and repeat lines
- Task 5. Copy, move, rename, delete data sets and members
- Task 6. Data set lists
- Task 7. Using TSO/E commands
 - Section 1: Accessing the TSO command line

- Section 2: TSO commands

JES and JCL

- Task 8. System Familiarization
 - Section 1: Gathering information and accessing the lab image
 - Section 2: Exit ISPF
 - Section 3: Logoff
- Task 9. Submit a job
 - Section 1: Create a data set
 - Section 2: JCL and viewing help
 - Section 3: Rename a data set member
- Task 10. JCL exercises
 - Section 1: Copy a selected member of a PDS
 - Section 2: Copy an entire data set
 - Section 3: Copy and replace a selected member
 - Section 4: Perform a securecompress
 - Section 5: Compress a data set in place
 - Section 6: Copy a PDS and a PDSE data set
 - Section 7: Copy in-stream data to PS
 - Section 8: Print a Sequential data set to SYSOUT
 - Section 9: Print a PDS member
 - Section 10: Create a PDS member
- Task 11. Procedures
 - Section 1: Override DD-statements of an in-stream procedure
 - Section 2: Run cataloged procedure
 - Section 3: Create cataloged procedure
 - Section 4: Use of DFSORT (ICEMAN)
 - Section 5: Create PS from in-stream
 - Section 6: Conditional processing
 - Section 7: Route a job to INTRDR using IEBGENER
- Task 12. z/OS Management Facility
 - Section 1: Accessing the z/OS Management Facility
 - Section 2: z/OSMF interface layout
 - Section 3: TSO interface
 - Section 4: Links
 - Section 5: Setting up a z/OS console
 - Section 6: Using the z/OS console

- Section 7: Entering z/OS commands and interpreting the output
- Section 8: Identifying the system configuration
- Section 9: The DEVSERV command
- Task 13. ISHELL and hierarchical file system
 - Section 1: ISHELL introduction
 - Section 2: Introduction to the file system
 - Section 3: Customize ISHELL
 - Section 4: UNIX directory list
 - Section 5: Create a hierarchical file system
 - Section 6: Managing files
 - Section 7: Additional UNIX activities
 - Section 8: Introduction to the file system

G. Learning Resources

i. Text Books:

1. Mike Ebbers, John Kettner, Wayne O'Brien, Bill Ogden, "Introduction to the New Mainframe z/OS Basics", IBM Redbooks, Third Edition, 2011

ii. Online References:

1. Introduction to IBM z/OS. Accessed on: Apr. 18, 2023 [Online]. Available: <https://learn.ibm.com/course/view.php?id=6815>
2. Introduction to z/OS Commands and Panels on IBM Z. Accessed on: Apr. 18, 2023 [Online]. Available: <https://learn.ibm.com/course/view.php?id=7419>
3. Introduction to System Programming on IBM Z. Accessed on: Apr. 18, 2023 [Online]. Available: <https://learn.ibm.com/course/view.php?id=7512>

COURSECODE	COURSE TITLE	L	T	P	C
10212CS276	GENERATIVE AI	1	0	2	2

A. Preamble

The Generative AI course introduces students to the principles and applications of deep learning, focusing on generative models. The course covers the rise of deep learning, large language models (LLMs), prompt engineering, Langchain, and diffusion models. It integrates theoretical knowledge with practical lab exercises to enhance learning and application.

B. Prerequisite Courses

10210MA103 Probability, Statistics and Queuing theory

C. Course Objectives

Students will be able to:

- Understand the fundamentals of deep learning and its applications. (K2)
- Describe large language models and their architectures. (K1)
- Apply principles and techniques of prompt engineering. (K3)
- Explore Langchain and its components. (K3)
- Explain diffusion models and their use in generative AI. (K2)

D. Course Outcomes

Upon successful completion of the course, students will be able to:

CO No's	Course Outcomes	K-Level
CO1	Comprehend the fundamentals and rise of deep learning	K2/S3
CO2	Apply knowledge of large language models in various NLP tasks	K3/S3
CO3	Develop and utilize prompt engineering techniques	K3/S3
CO4	Implement Langchain architecture for generative AI applications.	K3/S3
CO5	Apply diffusion models to progressively refine noise into high-quality images and videos	K3/S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of COs with Program Outcomes and Program Specific Outcomes

CO's	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO 1	PSO 2	PSO 3
CO1	3	3		2	1							3		1
CO2	3	3	3	2	2							3	2	2
CO3	1	2	3	2	3							1	2	2
CO4	1	2	2	2	3							1	1	2
CO5	3	3	2	2	3							3		2

3- High; 2-Medium; 1-Low

F. Course Contents

Unit I: Rise of Deep Learning

L - 3 Hours

AI vs ML vs DL, Need for DL, Power of Neural Networks, Tools for sequence models: RNNs, Transformers, attention, Deep vision by CNNs, Deep Generative Modeling, Responsible and ethical use of Deep Learning tools.

Unit II: Large Language Models (LLMs)

L - 3 Hours

History of NLP, Tokenization and embeddings, Positional encoding of Transformers, Text generation process, LLM model architecture: BERT, GPT, Training LLMs: Base, Instruction Tuned, Fine Tuning, Limitations of LLMs, API calling of OpenAI, Hugging Face, Google, Meta models.

Unit III: Prompt Engineering

L - 3 Hours

System-user-assistant format, Principles of prompting, chaining prompts, using delimiters to avoid prompt injections, Iterative prompt development, Zero and few shots prompting, Chain-of-thoughts prompting, Program-aided language prompting, ReAct prompting, Building a chatbot using prompt engineering (use cases: healthcare, education, e-commerce).

Unit IV: Langchain

L - 3 Hours

Langchain architecture, LangChain Expression Language (LCEL), Lang Graph, Runnable interface, Chat models, Prompt templates, Example selectors, Output parsers, Chat history, Document loaders, Text splitters, embedding models, Vector stores, Retrievers, Agents, Callbacks, RAG using langchain and chromadb.

Unit V: Diffusion Models

L - 3 Hours

Variational Autoencoder, Generative Adversarial Networks, Diffusion Process: Forward Noising, Reverse Denoising, Stable Diffusion models, U-Net, Text encoder, Video generation, generating images from natural language using Hugging Face API, Beyond images: molecular design

G. Laboratory Experiments (30 Hours)

Task 1	Introduction to Python & NLP, familiarizing with NLTK and Transformers. Tools: Python, NLTK, Transformers
Task 2	Create a text generator using any Hugging Face LLM that can generate realistic sentences or paragraphs. Tools: Python, Hugging Face Transformers
Task 3	Build a language translation tool using an LLM that can translate given text from one language to another with high accuracy. Tools: Python, Hugging Face Transformers
Task 4	Create a sentiment analysis tool that can analyze the sentiment of text data and classify it as positive, negative, or neutral. Tools: Python, Hugging Face Transformers
Task 5	Develop a text summarization tool that can generate concise summaries of long articles and documents. Tools: Python, Hugging Face Transformers

- Task 6** Create a recipe generator that can generate new and unique recipes based on user preferences and dietary restrictions.
Tools: Python, Hugging Face Transformers
- Task 7** Develop a legal research tool that can search and retrieve relevant case details from the database based on user queries.
Tools: Python, Hugging Face Transformers
- Task 8** Develop a story generator using any Hugging Face LLM that can generate creative and engaging storylines and plot twists.
Tools: Python, Hugging Face Transformers
- Task 9** Build a painting generator using a model like Stable Diffusion that can create unique oil paintings.
Tools: Python, Stable Diffusion, TensorFlow
- Task 10** Develop a logo generator using a model like Stable Diffusion that can create unique and professional logos for businesses or organizations.
Tools: Python, Stable Diffusion, TensorFlow
- Task 11** Develop a program that generates random modern Disney-style avatars or character images with different facial features, hairstyles, and accessories.
Tools: Python, Stable Diffusion, TensorFlow

Part – 2

Use Cases:

- Use Case 1: Generate Cricket Match Summaries
- Use Case 2: Automated Art Creation for E-commerce
- Use Case 3: Interactive Storytelling for Children's Education
- Use Case 4: Brand Identity Creation for Startups
- Use Case 5: Personalized Recipe Suggestions for Health Apps

H. Learning Resources

i. Text Books:

1. Foster D, *et al.* "Generative deep learning ", O'Reilly Media, Inc, 2022.
2. Dhamani N, *et al.* "Introduction to Generative AI", Simon and Schuster, 2024.
3. Phoenix J, Taylor M. "Prompt Engineering for Generative AI", O'Reilly Media, Inc, 2024.
4. Omar Sanseviero, *et al.* "Hands-On Generative AI with Transformers and Diffusion Models", O'Reilly Media, Inc, 2024.

ii. Reference Books:

1. Ian Goodfellow, Yoshua Bengio, and Aaron Courville. "Deep Learning", 2021.
2. Delip Rao and Brian McMahan, "Natural Language Processing with Transformers", 2021.
3. David Foster. "Generative Deep Learning: Teaching Machines to Paint, Write, Compose, and Play", 2020.

iii. Online References:

- "Generative AI with Large Language Models" (online): Coursera.
- "Generative AI and Language Models" (online): NPTEL.

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS277	Deep Learning Techniques	3	0	2	4

A.Preamble

In this course students will be gain knowledge on how to build and train neural network architectures such as Convolutional Neural Networks, Recurrent Neural Networks, LSTMs, and Autoencoders. Understand the capabilities, challenges, and consequences of deep learning compared to Machine learning and prepare students in the development of leading-edge AI technology.

B.Prerequisite Course

10212CS223 - Machine Learning Techniques

C.Course Objectives

Students are able to

- Understand basic concepts of deep learning Framework
- Understand CNN, RNN and GAN architecture models
- Perform model optimization and deployment

D.Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K – Level/ S - Level
CO1	Explore data processing techniques using various deep learning frameworks for classification.	K3/S3
CO2	Build and train Convolutional Neural Network (CNN) model for real time applications.	K3/S3
CO3	Use various Convolutional Neural Network architectures for model building.	K3/S3
CO4	Demonstrate Recurrent Neural Networks and LSTM Networks	K3/S3
CO5	Apply attention mechanisms and transformer models for sequence and vision-based tasks.	K3/S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E.Correlation of COs with Program outcomes and Programme Specific Outcomes:

Cos	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	3	3		2						3		
CO2	3	2	2	2	2	2						3	2	
CO3	3	3	3	3	2	2						3	2	
CO4	3	3	3	3	2	2						2	2	
CO5	3	3	3	3	3	2						2	2	

3- High; 2-Medium; 1-Low

F.Course Contents**Unit 1- Data Processing and Deep Learning Framework****L-9 Hours**

Introduction to Data Handling – Data processing and Normalization using Deep Learning libraries and Data Augmentation Techniques.vDeep Learning Frameworks – PyTorch, TensorFlow and Keras. Deep Learning Parameters – Epoch, Learning rate, Batch size, Open-source deep learning tools.

Unit 2 – DeepConvolutional Neural Networks (CNN)**L-9 Hours**

Deep learning Layers – Convolutional Layer- Convolutional Properties and operations, Pooling Layer – Role and its types. Activation Layer – Significance and its types. Optimizers – Stochastic gradient, RMSProp, Ada Delta and Adam optimizers. Loss Function – Mean Square Error (MSE), Binary Cross Entropy, Categorical cross Entropy and Sparse Categorical cross Entropy.

Unit 3- Convnet Architecture and Transfer Learning**L-9 Hours**

Introduction to Convnet - Architectures – AlexNet, VGG, Inception, ResNet - Training a Convnet: weights initialization, batch normalization, hyperparameter optimization, Addressing Overfitting and Underfitting issues. Transfer Learning, Advance architectures – Generative Adversarial Networks (GANs), Autoencoders, One shot learning, Siamese Networks and Capsule networks.

Unit 4 – Recurrent Neural Networks (RNN)**L-9 Hours**

Introduction toRecurrent Neural Networks (RNN)–RNN architecture, Building RNN step by step, Gated Recurrent Unit (GRU), Long-Short Term Memory (LSTM), LSTM architecture, Bi-LSTM.Language Models- Word embedding, Word-Level RNNs.

Unit 5- Transformers**L-9Hours**

Introduction to Attention Mechanisms – Limitations of RNNs and LSTMs in sequential modeling– Types of attention: additive and multiplicative. Self-Attention and Multi-Head Attention – Scaled dot-product attention – Multi-head attention – Positional encoding Transformer Architecture – Components of encoder and decoder. Pretrained Transformer models: BERT, GPT – Fine-tuning vs. pretraining – Vision Transformers (ViT).

Total: 45 Hours**G.Laboratory Experiments:****Total:30 Hours****Part – 1**

- Task 1** Load a dataset (e.g., MNIST or CIFAR-10), perform data normalization, and visualize the effect of normalization on sample images.
(Framework: TensorFlow/Pytorch).
- Task 2** Apply data augmentation techniques (such as rotation, flipping, scaling) on images using TensorFlow or PyTorch libraries
(Framework: TensorFlow/Pytorch).
- Task 3** Implement a simple CNN for image classification with convolutional, pooling, and activation layers.
(Framework: TensorFlow/Pytorch).
- Task4** Compare training results using different optimizers and loss functions on the CNN model.
(Framework: TensorFlow/Pytorch).

- Task5** Load a pretrained CNN model and perform transfer learning on a new dataset.
(Framework: TensorFlow/Pytorch).
- Task6** Build a simple **Generative Adversarial Network (GAN)** to generate image. Visualize and compare real vs. generated/reconstructed images.
(Framework: TensorFlow/Pytorch).
- Task7** Build and train an RNN model for sequence prediction (e.g., character-level text generation).
(Framework: TensorFlow/Pytorch).
- Task8** Implement an LSTM or Bi-LSTM model for sentiment analysis using a text dataset.
(Framework: TensorFlow/Pytorch).
- Task9** Implement scaled dot-product attention and multi-head attention mechanisms on sample input sequences.
(Framework: TensorFlow/Pytorch).
- Task10** Fine-tune a pretrained Transformer models on a text classification task and evaluate its performance.
(Framework: TensorFlow/Pytorch).

Part-2

Use Cases:

1. Medical Image Diagnosis and Report Generation
2. Autonomous Vehicle Perception and Decision Making
3. Smart Customer Support Chatbot with Visual Input
4. Financial Fraud Detection and Reporting
5. E-commerce Product Recommendation and Review Analysis

Total:75 Hours

H.Learning Resources

i. Text Books:

1. I Goodfellow, Y Bengio and A Courville, "Deep Learning",1st Edition MIT Press, 2017. [Unit 1-5]

ii. Reference Books:

1. Venkatesan, Ragav, and Baoxin Li. "Convolutional neural networks in visual computing: a concise guide". CRC Press, 2017.[Unit-2,3]
2. Nikhil Buduma., "Fundamentals of Deep Learning", O'Reilly Media, 2017. [Unit-4,5]

iii. Online References:

1. "Deep Learning Specialization". Accessed on: April 20, 2021 [Online]. <https://www.coursera.org/specializations/deep-learning>.
2. "Deep Learning Nanodegree", Accessed on: April. 16, 2021 [Online]. Available: <https://www.udacity.com/course/deep-learning-nanodegree--nd101>

COURSE CODE	COURSE TITLE	L	T	P	C

10212CS278	Quantum Machine Learning	3	0	2	4

A. Preamble

Quantum computing represents a paradigm shift in computational science, offering unprecedented capabilities to solve problems intractable for classical computers. This course is designed to equip students with the foundational knowledge, technical skills, and critical thinking required to engage with this transformative field. As quantum technologies rapidly advance from theoretical breakthroughs to real-world applications in cryptography, optimization, and machine learning this course bridges theory and practice, preparing learners to contribute to academia, industry, and innovation.

B. Prerequisite Courses

- 10210MA101- Linear Algebra for Computing
- 10210MA103- Probability, Statistics and Queuing theory

C. Course Objectives

Learners are exposed to:

- Equip students with mathematical foundations for quantum computing.
- Introduce quantum gates, circuits, and logic operations.
- Explore fundamental and advanced quantum algorithms.
- Examine quantum machine learning techniques and applications.
- Provide hands-on experience with quantum programming frameworks (e.g., Qiskit, Cirq).

D. Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level / S - Level
CO1	Apply core mathematical tools (Dirac notation, linear algebra) to solve quantum state evolution and measurement problems.	K3/S3
CO2	Implementing advanced mathematical frameworks (tensor products, group theory) to model and analyze multi-qubit quantum systems.	K3/S3
CO3	Design and implement quantum circuits using fundamental gates to achieve specific computational or error-correction tasks.	K3/S3
CO4	Execute basic and advanced quantum algorithms (e.g., Shor's, Grover's) to demonstrate computational advantages over classical approaches.	K3/S3
CO5	Deploy quantum machine learning algorithms (e.g., QSVM, QAOA) to optimize real-world classification or optimization problems.	K3/S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Creat S1-Imitate, S2-Perform, S3-Precision, S4-Articulation, and S5-Naturalization		

E. Correlation of COs with Program outcomes and Programme Specific Outcomes:

Cos	PO1	PO ₂	PO3	PO ₄	PO5	PO ₆	PO7	PO8	PO9	PO10	PO11	PSO ₁	PSO2	PSO3

CO1	3	2	2	2	2	2					2	3	2	2
CO2	3	3	2	2	2						2	3	3	2
CO3	2	2	3	2	3						2	3	2	3
CO4	2	3	3	3	3	2					2	3	3	3
CO5		2	3	2	3	3	2				2	3	2	3

1-Low,2-Medium,3- High

F. Course Content

UNIT I Mathematical Foundations of Quantum Computing L-9 Hours

Introduction - Linear Algebra: Vector spaces - Inner products - Matrices - Eigenvalues and eigenvectors - Complex Numbers and Probability - Complex Hilbert spaces - Born rule - quantum measurement - Dirac Notation: Bras, kets and operator representations - Qubit States: Superposition - entanglement - Bloch sphere representation - Postulates of Quantum Mechanics: State vectors - Time evolution - Composite systems.

UNIT II Advanced Mathematical Foundations L- 9 Hours

Hilbert Spaces: Infinite-dimensional spaces - Orthonormal bases - Tensor Products: Multi-qubit systems - Matrix product states - Partial trace and entanglement - Group Theory in QC: Pauli matrices - SU(2) group and symmetry in quantum systems - Quantum Entropy: Von Neumann entropy - Quantum information theory.

UNIT III Quantum Gates and Logics L-9 Hours

Single-Qubit Gates: Pauli-X/Y/Z, Hadamard - Phase and T gates - Multi-Qubit Gates: CNOT, SWAP, Toffoli, and controlled-phase gates - Quantum Circuits: Circuit diagrams, universality of gate sets - Quantum Error Correction: Shor code, surface codes, stabilizer formalism - Quantum Logic: Reversible computing, quantum parallelism.

UNITIV Quantum Algorithms L- 9 Hours

Basic Algorithms: Deutsch-Jozsa - Bernstein-Vazirani - Simon's algorithm - Quantum Fourier Transform (QFT): Phase estimation - Period finding - Advanced Algorithms: Shor's (factoring) - Grover's (search) - HHL (linear systems) - Quantum Simulation: Variational Quantum Eigensolver (VQE) - Quantum chemistry applications - Quantum Supremacy: NISQ devices -Random circuit sampling - Quantum Walk algorithm.

UNIT V Quantum Machine Learning Algorithms L- 9 Hours

Basic QML: Quantum Support Vector Machines (QSVM) - Quantum PCA - Advanced QML: Quantum Neural Networks (QNN) - Quantum Boltzmann Machines - Hybrid Algorithms: Quantum Approximate Optimization Algorithm (QAOA) - Quantum Generative Adversarial Networks (QGANs) – Quantum Reinforcement learning - Applications: Drug discovery - Financial modeling and Optimization - Challenges: Noise - Scalability and hybrid classical - Quantum workflows.

Total: 45 Hours

G. Laboratory Experiments

Total: 30 Hours

Part - 1

TASK 1 Apply Born rule to calculate measurement probabilities for superposition states.
Tools: Python, NumPy, Matplotlib.

- TASK 2** Construct Pauli-X/Y/Z matrices and apply them to qubit states.
Compute eigenvalues/eigenvectors of quantum operators.
Tools: NumPy, SymPy.
- TASK 3** Build entangled states (Bell states) using tensor products.
Measure entanglement entropy for bipartite systems.
Tools: IBM Qiskit, NumPy.
- TASK4** Verify commutation/anti-commutation relations of Pauli matrices.
Decompose single-qubit gates into Euler angles.
Tools: SymPy, Cirq.
- TASK5** Implement a CNOT gate circuit.
Simulate quantum teleportation using Qiskit.
Tools: IBM Qiskit.
- TASK6** Encode a logical qubit into 9 physical qubits.
Simulate noise and recovery using Qiskit's noise models.
Tools: Qiskit Aer (noise simulation).
- TASK7** Code the algorithm for a 2-qubit oracle.
Tools: PennyLane, IBM Qiskit.
- TASK8** Implement Grover's algorithm for a 3-qubit database.
Tools: Cirq, Matplotlib.
- TASK9** Implement a QSVM on the Iris dataset using PennyLane.
Tools: PennyLane, scikit-learn.
- TASK10** Implement the QAOA algorithm to solve the Max-Cut problem using a hybrid quantum-classical approach.
Tools: Qiskit Optimization, PyTorch.

Part - 2

USE CASES:

Use case 1: Accelerating Drug Discovery and Molecular Modeling

Use case 2: Climate Modeling

Use case 3: Quantum-Enhanced Machine learning models

Use case 4: Portfolio Optimization in Finance

Use case 5: Quantum Generative models for data synthesis

Total: 75 Hours

H. Learning Resources

i. Textbooks

1. Eleanor G. Rieffel & Wolfgang H. Polak, "Quantum Computing: A Gentle Introduction", Second Edition, MIT Press, 2020.
2. Michael A. Nielsen & Isaac L. Chuang, "Introduction to Quantum Mechanics", Cambridge University Press, 2010.
3. Chris Bernhardt, "Quantum Computing for Everyone", MIT Press, First Edition, 2019.
4. N. David Mermin, "Quantum Computer Science: An Introduction", Cambridge University Press, 2007.

ii. Reference Books

1. Maria Schuld & Francesco Petruccione, “Quantum Machine Learning: Foundations and Applications”, Springer, 2024.
2. Hassi Norlén, “Quantum Computing in Practice with Qiskit and IBM Quantum Experience”, Second Edition, Packt Publishing, 2020.

iii. Web Resources

1. IBM Quantum Experience - [Online] Available: <https://quantum.ibm.com/>
2. Google Quantum AI - [Online] Available: <https://quantumai.google/>
3. Microsoft Quantum Development Kit (QDK) - [Online] Available: <https://learn.microsoft.com/en-us/azure/quantum/>
4. Quantum Katas - [Online] Available: <https://github.com/microsoft/QuantumKatas>

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS279	Application Development and Maintenance	3	0	2	4

(This course is offered under General Elective Category. All School students can opt the course.)

A. Preamble

This course provides basic concepts about SDLC, Unix commands and shell operations, Object Oriented Programming, JavaScript technologies to create the interactive client-side design of web applications. This course introduces DBMS concepts, to share a common set of models, design paradigms via a Structured Query Language (SQL). More specifically, this course also deals with UI/UX concepts to develop a pleasant user interface and deploy the application using Cloud AWS services.

B. Prerequisite Course

10210CS101 - Problem Solving using C.

C. Course Objectives

Learners are exposed to:

- Make use of basic SDLC, Unix Commands.
- Practice Java basic constructs for real world problems.
- Build efficient client-side validation using javascript.
- Acquire knowledge on basic SQL queries and designing database schema
- Learn and build the workflow of UI/UX
- Deploy the application on AWS cloud

D. Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Interpret the Software Development Life Cycle (SDLC) stages, Unix basics commands and shell operations, and OOP.	K2
CO2	Articulate proficiency in exception handling, inheritance, and API fundamentals.	K3
CO3	Design the interactive and dynamic webpage using JavaScript.	K3
CO4	Model the Relation Database schema and perform CRUD operations.	K3
CO5	Chart UI/UX and deploy the applications using AWS cloud.	K3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of COs with POs:

COs	P O 1	P O 2	P O 3	P O 4	P O 5	P O 6	P O 7	P O 8	P O 9	P O 10	P O 11	P S O 1	P S O 2	P S O 3
CO1	3	3	3	3	2						3			
CO2	3	3	3	3	2						3			
CO3	3	3	3	3	2						3			
CO4	3	3	3	3	2						2			
CO5	3	3	3	3	2						2			

3 - High; 2 - Medium; 1 - Low

F. Course Contents**Unit 1 Introduction to SDLC, UNIX/OS and OOP Concepts****L-9 Hours**

SDLC: Requirements gathering - System design - Implementation - Testing - Various types of testing - Maintenance - Documentation - Project Management - Agile. UNIX/OS: Unix Basics - Unix shell - File Operations - Process Management - Networking and Communication - System Administration - Shell Customization. OOPs Concept: Benefits and Multiple Features of OOP - Object Oriented Programming Vs Procedural Programming - Class - Cohesion and Coupling in design process - Popular OOP Languages.

Unit 2 Exception Handling, Inheritance & APIs**L-9 Hours**

Exception Handling - Categories of Exceptions - Hierarchy of Exception Class - Exception Types - Checked and Unchecked Exceptions. Inheritance and its Types - Creating Subclass - Method Overriding - Super keyword - Polymorphism - instanceof operator - Abstract Vs Non-Abstract classes - Interface. **API:** Introduction - API Basics - Types of API Protocols - Styles and Benefits of API Development.

Unit 3 Scripting Language - Javascript**L-9 Hours**

Basic Web Development Concepts - **Javascript:** Introduction to Javascript - Syntax and Basic Concepts - Document Object Model (DOM) Manipulation - Arrays and Objects - Asynchronous JavaScript - Error Handling and Debugging - ECMAScript 6 (ES6)+ Features.

Unit 4 Database Fundamentals: RDBMS, SQL**L-9 Hours**

DBMS: History of Data and its definition - Various formats to recognize databases - DBMS through MySQL Workbench - DBMS Types and its features - Requirements of DBMS - Roles, job-related tasks.

RDBMS: Introduction and its uses - popular RDBMS and their uses - Oracle RDBMS versions and features - SQL Server RDBMS, history and its features - MySQL RDBMS, history and its features - MySQL Workbench - Normalization - E.F. Codd.

SQL: Introduction - Connectivity - Reserved words - Data Types - SQL Commands: DDL - Alter tables - drop and truncate - table constraints - constraint violation.

Unit 5 Fundamentals of UI/UX, Generative AI and AWS Cloud Practitioner L-9 Hours

UI/UX: Introduction - importance - UI Vs UX - Future of UI/UX - User prevention on errors - flowcharts of UI/UX - journey maps of UX design - micro interactions - age-responsive design - interstitial anxiety - VR interfaces Vs normal interfaces - Modern concepts of UI/UX.

Generative AI: Definition - Working principle - Model Types - Applications.

AWS Cloud Practitioner: Introduction - AWS services - SDKs Vs APIs - AWS Management Console - Infrastructure as Code (IaC) - Deployment and Management - Serverless Computing - Data Storage and Analytics.

Total: 45 Hours

G.Laboratory Experiments

Total: 30 Hours

Part – 1

Task 1: Working with Unix OS Commands.

[OS: UNIX, LINUX, IDE: Terminal]

Task2. Develop a Simple Java application with class (Object), Methods and Packages.

[Package: java.io, java.util, IDE: Eclipse]

Task 3: Inheritance & Interfaces

- Demonstrate Method overloading, constructor and Method Overriding.
- Develop a code for user-defined interface and pre-defined interface concept

[Package: java.io, java.util, IDE: Eclipse]

Task 4: Exception Handling Mechanism in Java

- Construct a program to handle user defined and predefined exceptions
- Design the code with performing Checked and Unchecked Exceptions

[Package: java.io, java.lang, IDE: Eclipse]

Task 5: Conceptual Design after FTR

Using basic database design methodology and ER modeler, design Entity Relationship Diagram by satisfying the following sub tasks:

- Identifying the entities.
- Identifying the attributes.
- Identification of relationships, cardinality, type of relationship.
- Reframing the relations with keys and constraints.

[Database: MySQL]

Task6: Generating design of another traditional database model

Creating Hierarchical /Network model of the database by enhancing the sound abstract data by performing following tasks using forms of inheritance:

- Identify the specificity of each relationship, find, and form surplus relations.
- Check is-a hierarchy/ has-a hierarchy and performs generalization and/or specialization relationship.
- Find the domain of the attribute and perform a check constraint to the applicable.
- Rename the relations.
- Perform SQL Relations using DDL, DML, DCL commands

[Database: MySQL]

Task7. Construct a dynamic webpage using JavaScript.

[IDE: Visual Studio]

Task8. Manipulating the DOM with javascript

[IDE: Visual Studio]

Task 9: UI/UX Design

- Design a UI for the home page using figma tool.
- [Tool: figma]

Task 10: AWS Cloud Practitioner

Deployment and Management

- a) Choose a sample web application.
 - b) Deploy the application on an EC2 instance.
 - c) Outline the steps taken and any challenges faced during deployment.
- [Services: AWS Cloud]**

Part– 2 Use cases:**Use case-1: Develop Online learning platform**

An educational technology company is planning to develop an online learning platform to offer programming courses to students. The platform aims to provide a seamless learning experience.

Technologies Used: JavaScript, UI/UX principle, Generative AI, AWS Cloud services.

Use case-2: Tracking the student marks statement:

In the college, every month the assessment test will be scheduled and results will be updated in the portal. Create the web page with username and password of each student and with admin controls. The student can track the test marks by applying their login credentials in the server page and get the exact marks, when the test is attended and when the marks are updated in the portal and result status. Demonstrate the web page with frontend and backend control to track the mark statements and design the interface to make it user friendly.

Technologies Used: JavaScript, UI/UX principle, Generative AI, AWS Cloud services.

Total: 75 Hours

G.Learning Resources**i.Text Books:**

1. "Software Engineering: A Practitioner's Approach" by Roger S. Pressman is published by McGraw-Hill Education. September 2021 , 9th edition. [Unit - 1]
2. "UNIX and Linux System Administration Handbook" is published by Pearson and the 5th edition was published in 2017. [Unit - 1]
3. "Introduction to Java Programming and Data Structures, Comprehensive Version" (12th Edition) By Y. Daniel Liang, Pearson in 2018. [Unit - 1 & 2]
5. "APIs: A Strategy Guide" by Daniel Jacobson, Greg Brail, and Dan Woods, O'Reilly Media, Inc. December 2011. [Unit - 2]
6. "JavaScript: The Definitive Guide, 7th Edition." by David Flanagan. Released May 2020. Publisher(s): O'Reilly Media, Inc. [Unit - 3]
7. "Database System Concepts" by Abraham Silberschatz, Henry F. Korth, and S. Sudarshan 7th edition McGraw-Hill Education. [Unit - 4]
8. "Don't Make Me Think: A Common Sense Approach to Web Usability" by Steve Krug 3rd edition by New Riders. [Unit - 5]
9. "AWS Certified Cloud Practitioner Study Guide" 2nd Edition by Sybex. [Unit - 5]

ii.Reference Books:

1. H.M. Deitel and P.J. Deitel, "Java How to Program", Pearson Prentice Hall Seventh edition, 2018.
2. E. Balaguruswamy, "Programming in java", Sixth Edition, Tata McGraw Hill, 2019.

3. Mary Delamater, “Murach's JavaScript and jQuery (4th Edition) Paperback – Illustrated”, Mike Murach& Associates Inc, August 26, 2020.

4. Elmasri Ramez, Navathe S, “Fundamentals of Database System”, Seventh Edition, Pearson, 2017.

5. The Design of Everyday Things: Revised and Expanded Edition by Don Norman.

6. Learning Amazon Web Services (AWS): A Hands-On Guide to the Fundamentals of AWS Cloud by Mark Wilkins.

iii. Online References:

1. “Java Tutorial”, June 20, 2023 [Online], Available: <https://www.javatpoint.com/java-tutorial>

2. “Programming in Java”, Accessed on: June 20, 2023 [Online], Available: <https://nptel.ac.in/courses/106/105/106105191/>

3. “UI / UX Design Tutorial – Wireframe, Mockup & Design in Figma “
https://www.youtube.com/watch?v=c9Wg6Cb_YIU

4. Get started with Amazon EC2 Linux instances
https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/EC2_GetStarted.html

5. “ AWS EC2 Instances | How to Create AWS EC2 Instance | AWS EC2 Tutorial For Beginner” <https://www.youtube.com/watch?v=LZXWIF5udYs>

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS280	Low-Code Development	1	0	2	2

A.Preamble

This course is designed to develop skills to design and analyze the latest low-code development using mendix platform and allow them to experiment and build a working solution. This enables them to build a project prototype to showcase their skills and add to their work portfolio.

B.Prerequisite Courses

10210CS101 - Problem Solving using C

C.Course Objectives

Students undergoing this course are expected to

- Develop skills in selecting the appropriate data mining algorithm for solving practical problems and identify an organizations information processing requirement
- Understand the standard development life cycle (SDLC) and use Agile methods to develop an information system that meets the business needs
- Develop database structures ,UML diagrams with workflow processes and business needs
- Build the user interfaces ,experiences and implement simple security with user role access

D.Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Understand the role of Mendix low code development and application life cycle management	K2
CO2	Build the domain model and creation of object entities and understating of elements in the microflow	K3
CO3	Develop the process automation model and association based data retrieval.	K3
CO4	Apply regular expression in the micro flow model and merging multiple flows for data validation and consistency	K3
CO5	Identify the security settings for user, data, microflows, entity access in all the levels	K3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E.Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO 1	PO2	PO3	PO 4	PO 5	PO6	PO7	PO8	PO9	PO10	PO11	PS O1	PS O2	PSO 3

CO1		2	2		3								2	
CO2		2	2		3								2	
CO3		2	2		3								2	
CO4		2	2		3								2	
CO5		2	2		3								2	

3- High; 2-Medium; 1-Low

E. Course Content

UNIT 1 Introduction

L-3 Hours

Introduction to mendix platform and Low Code Development, Installation procedure of mendix Studio, Application Lifecycle Management, Project methodology-Waterfall and Agile - Agile Process, Project Roles-Scrum Process, Wireframe.

UNIT 2 Data entity and Microflows

L-3 Hours

Dynamic data, Entities and attributes-Attribute types, entities ,Objects and object creation. The Domain Model-Steps to add entities to domain model, Associations in the Domain model. Microflows- Need for a microflow, Elements of a microflow, Mendix Assist Logic Bot. Studio Mendix studio pro-Exploring mendix Studio Pro, The Team Server, Information Entity, Nested data.

UNIT 3 Process Automation and Retrieval

L-3 Hours

Automation of Process flow, Triggering a Microflow from an Input Widget, Microflow Expressions, Microflows scope, Retrieving data, Association based data retrieval, analyze error, decisions in microflow , Stored vs Calculated Attributes, Object event handlers

UNIT 4 Data validation and consistency

L-3 Hours

Checking Data validation and consistency of data ,validation Rules and types, Regular Expressions, Validation in Microflow, Merge Multiple Flows, Deleting Objects- Cascading Delete

UNIT 5 Data Security and Mobile app development

L-3 Hours

Data Security and level, User and Module Roles, Access rules-page, Microflow access, Entity access, Mobile apps, PWA and its properties, Nanoflows -The nanoflow commons module, Sharing information via mobile social channels-update security and Publishing the project

Total : 15 Hours

E. Laboratory Experiments

Total : 30 Hours

Part - 1

- TASK 1** Development of Project Life Cycle and Wireframe
- TASK 2** Project Initiation, Project planning, Project Management, Teams Developed Development Methodologies(Waterfall & Agile)
- TASK 3** Analyzing business needs & Requirements gathering
- TASK 4** Creation of User Stories and importing into mendix studio
- TASK 5** Design Thinking-Design, Develop Wireframes and project brainstorming to start the creation process
- TASK 6** Data Modeling-Develop Entity Relationship Diagrams(ERDs) and build back end database using low- code
- TASK 7** Development of User Interface Design for sprints

- TASK 8** Advanced User interface and Design with Business Logic and Flows -Develop Business logic and flow of information with security and permission
- TASK 9** **Designing and Refining**
- i) System Design & System Design Variations Implementation
 - ii) Transition to the new system
 - iii) Object Oriented Analysis & Design
 - iv) Managing Systems Implementation
- TASK 10** Project Presentation

Total : 45 Hours

Part-2

Use cases:

- Learning App Development
- Travel Management Application
- Health Care Management System
- Hospital Management System
- Upi Application
- Contacts Application
- Online Supermarket Application

Stories for Use Case: LEARNING APP DEVELOPMENT

Problem Statement

Jimmy is the director of the LearnNow company, which offers various trainings. He uses Excel to manage the training courses that the company offers. As the company is growing, it's becoming more and more difficult to keep a clear overview of all the courses.

The Solution

Jimmy has decided to use an app to keep track of everything. He wants to be able to easily register new participants, enter new courses into the system, and quickly schedule them in his agenda. The trainers will also benefit from a clear overview of when they have to teach each class.

Learning App Development - Sprints

1. As an administrator, I want to be able to easily access and manage my data, so I can run my company more efficiently. view and manage :- Courses,- Locations,- Teachers,- Trainees
2. Have a homepage that gives access to pages where information can be seen.
3. Connect the buttons to pages that show the information in the system.
4. Create detail pages to add and edit information.
5. Add the pages to the navigation menu.
6. As an administrator, I want to be able to schedule a training event at any time, so that the teachers can see their schedule, even from their phone.
7. Add the Training Event entity to the Domain Model
8. Extend the homepage with a button that goes to a training event overview page.
9. Create an overview page for the training events.
10. Create a detail page to manage the training events.
11. Add the page to the navigation menu.
12. As an administrator, I want to be able to schedule a training event directly from the

homepage, so that I spend as little time as possible on this.
13. As an administrator, I want to be able to schedule a training event from the course overview page with the course name already selected, so that I spend as little time as possible on this.
14. As an administrator, I want to be able to add registrations to a training event, so that my teachers know who their students will be.
15. As an administrator, when scheduling a training event I want the end-date of the event to be automatically calculated, preventing me from making calculation errors.
16. Schedule a training event directly from the homepage.
17. Schedule a training event from the Course overview page, with the selected course prefilled.
18. Automatically calculate the end date of a training event, based on the start date and the duration.
19. As a teacher, I want to be able to see the number of trainees attending a training event at any time, so that I know what resources I need.
20. As an administrator, I want my data to be valid and consistent, so my database doesn't become messy.
21. As an administrator, I want to be able to delete training events that don't have registrations, so I can't accidentally delete a training event that people have already paid for to attend.
22. As an administrator, I want to be able to delete Trainees and all their Registrations, so that I can remove them from the system if they no longer wish to be a LearnNow student.
23. As an Administrator, I want the app to be secure, so I comply with the rules of data protection.
24. As a trainee, I want to view upcoming training events including their geolocation, so that I can choose the closest one
25. As an administrator, I would like trainees to be able to share information about training events, in order to promote offerings of the LearnNow app

Total: 45Hours

G. Learning Resources (in IEEE Format)

i. Online Recourses:

1. "Mendix Documents", Accessed on: Apr. 13, 2022 [Online] Available : <https://docs.mendix.com/>
2. "Studio 9 Guide", Accessed on: Apr. 13, 2022 [Online] Available: <https://docs.mendix.com/studio/>
3. "Mendix - Developer Portal GuideStudio Pro 9 Guide", Accessed on: Apr. 13, 2022 [Online] Available: <https://docs.mendix.com/developerportal/>
4. "Studio Pro 9 Guide", Accessed on: Apr. 13, 2022 [Online] Available: <https://docs.mendix.com/refguide/>

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS281	Industry Internet of Things	3	0	2	4

A. Preamble

This Course covers Intelligent IoT, which enables new business models and digital transformation. Intelligent IOT gives an informed tour to IoT solutions for smart healthcare, one of the most essential uses of IoT focusing on wearable healthcare sensors, body sensors, sophisticated pervasive healthcare systems, and big data analytics designed to help people lead better lifestyles.

B. Prerequisite Course

10211CS130- Fundamentals of Computer Networks

C. Course Objectives

Students are able to

- Introduce the basics of the Internet of Things (IoT).
- Understand sensors, communication, and networking in IoT.
- Explore IoT architecture, models, and tools like Arduino and Raspberry Pi.
- Study real-world IoT applications across various domains.
- Apply AI and IoT in industrial automation and smart systems.

D. Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K – Level/ S - Level
CO1	Understand general concepts of Internet of Things and Recognize various devices, sensors and applications	K2
CO2	Infer various M2M and IoT architectures	K3
CO3	Compare various design concepts to develop IoT solutions	K3
CO4	Identify Intelligent IoT solutions using sensors, actuators and Devices	K3
CO5	Build Intelligent IOT based applications for Industrial automation and other Sectors.	K3

Knowledge Level (Based on revised Bloom's Taxonomy)

K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create

E. Correlation of COs with Program outcomes and Programme Specific Outcomes:

Cos	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PSO1	PSO 2	PSO 3
CO1	2	2	2	-	-	-	-	-	-	-	-	-	-	-
CO2	2	2	2	-	-	-	-	-	-	-	-	-	-	-
CO3	2	2	2	-	-	-	-	-	-	-	-	-	-	2
CO4	-	2	2	-	-	-	-	-	-	-	-	-	-	2
CO5	2	3	2	-	-	-	-	-	-	-	-	-	-	2

3- High; 2-Medium; 1-Low

F.Course Contents**UNIT 1 INTRODUCTION TO IOT****L-9 Hours**

Sensing, Actuation, Networking basics, Communication Protocols, Sensor Networks, Machine-to-Machine Communications, IoT Definition, Characteristics. IoT Functional Blocks, Physical design of IoT, Logical design of IoT, Communication models & APIs.

UNIT 2 M2M TO IOT**L-9 Hours**

The Vision-Introduction, From M2M to IoT, M2M towards IoT-the global context, A use case example, Differing Characteristics. Definitions, M2M Value Chains, IoT Value Chains, An emerging industrial structure for IoT. An Architectural Overview–Building architecture, Main design principles and needed capabilities.

UNIT 3 IOT REFERENCE ARCHITECTURE**L-9 Hours**

An IoT architecture outline, standards considerations. Reference Architecture and Reference Model of IoT, Getting Familiar with IoT Architecture, Various architectural views of IoT such as Functional, Information, Operational and Deployment. Constraints affecting design in IoT world- Introduction, Technical design Constraints.

UNIT 4 DOMAIN SPECIFIC APPLICATIONS OF IOT & TOOLS**L- 9 Hours**

IoT tools - IOT implementation using Python. Introduction to Arduino and Raspberry Pi - Implementation of IoT with Arduino and Raspberry. Home automation, Intelligent Algorithms for IOT, Industry applications, Surveillance applications, Other IoT application - Cloud Computing, Fog Computing, Connected Vehicles, Data Aggregation for the IoT in Smart Cities, Privacy and Security Issues in IoT.

UNIT 5 INDUSTRIAL AUTOMATION USING AI & IOT**L- 9 Hours**

Road to Industry 4.0: Internet of Things (IoT) and Industrial Internet of Things (IIoT) and Internet of Services, smart manufacturing, smart devices and products, smart logistics, smart cities, predictive analytics. Case studies: Moisture and Humidity Control, Weather forecasting. [Edge Computing]

Total: 45 Hours**Laboratory Experiments****Total : 30 Hours****Part - 1****TASK 1**

Title: Sensor Data Acquisition and Actuation

Description: Implementation of sensor reading (e.g., temperature or light) and actuator control using Arduino or Raspberry Pi.

Language: Python

TASK 2

Title: IoT Networking and MQTT Communication

Description: Implement basic publish-subscribe communication using MQTT protocol between IoT devices.

Language: Python

TASK 3

Title: REST API for IoT Device Communication

Description: Develop a simple RESTful API using Flask to send and receive sensor data.

Language: Python

TASK 4**Title:** M2M Communication Simulation**Description:** Implement basic serial/I2C communication between two microcontrollers to demonstrate M2M communication.**Language:** Python / Arduino C**TASK 5****Title:** Design and Analysis of IoT Architecture**Description:** Create and analyze an IoT system architecture for a smart city or health application.**Tool:** Lucidchart / Draw.io**TASK 6****Title:** IoT Application Development with Raspberry Pi**Description:** Implement a real-time IoT application such as home automation or smart surveillance.**Language:** Python**TASK 7****Title:** Cloud and Fog Computing in IoT**Description:** Simulate sensor data upload to cloud and perform basic fog layer processing locally on edge devices.**Language:** Python**TASK 8****Title:** Data Aggregation and Visualization for Smart Cities**Description:** Collect and visualize multi-sensor data using Python plotting libraries (matplotlib/plotly).**Language:** Python**TASK 9****Title:** IoT Security - Data Encryption**Description:** Implement basic encryption (AES/Hashing) on sensor data before transmission.**Language:** Python**TASK 10****Title:** Industrial Automation with AI + IoT**Description:** Implement a predictive maintenance or humidity control use case using sensor input and simple ML model.**Language:** Python, **Tool:** scikit-learn**Part 2****Use cases****Use Case 1:** Smart Home Automation

- **Description:** Control lights, fans, and appliances using sensors (temperature, motion) and actuators.
- **IoT Concepts:** Sensing, Actuation, REST APIs, MQTT, Home automation.

Use Case 2: Smart Agriculture (Soil Moisture Monitoring & Irrigation)

- **Description:** Measure soil moisture using a sensor and automatically trigger irrigation using actuators.
- **IoT Concepts:** Sensor network, Data acquisition, Edge computing, Automation.

Use Case 3: Connected Vehicles

- **Description:** Monitor vehicle diagnostics (fuel, location, tire pressure), and send alerts.

- **IoT Concepts:** M2M communication, Cloud storage, APIs, Real-time data visualization.

Use Case 4: Remote Health Monitoring

- **Description:** Track heart rate, temperature, and SpO2 using wearable sensors and alert doctors remotely.
- **IoT Concepts:** Sensor networks, Data encryption, REST APIs, Fog Computing.

Use Case 5: Industrial IoT for Predictive Maintenance

- **Description:** Monitor vibration, temperature of machinery and predict failures using ML.
- **IoT Concepts:** Edge computing, Predictive analytics, Industrial automation.

Use Case 6: Smart City Traffic Management

- **Description:** Use sensors and cameras to monitor and control traffic signals dynamically.
- **IoT Concepts:** Computer vision, Data aggregation, Smart city architecture.

Use Case 7: IoT Security in Smart Homes

- **Description:** Use encrypted transmission and device authentication to secure smart home data.
- **IoT Concepts:** Privacy & security, Encryption algorithms, Network protocols.

H.Learning Resources

i. Text Books:

1. Farshad Firouzi, Krishnendu Chakrabarty, Sani Nassif, , “"Intelligent Internet of Things", From Device to Fog and Cloud”, Springer, 2020
2. Jan Holler, Vlasios Tsiatsis, Catherine Mulligan, Stefan Avesand Stamatis Karnouskos, David Boyle, “From Machine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligence”, 1st Edition, Academic Press, 2014.

ii. Reference Books:

1. Simon Monk, “Programming the Raspberry Pi: Getting Started with Python”, January 2012, McGraw Hill Professional
2. Massimo Banzi, “Getting Started with Arduino”, First Edition, February 2009, O'Reilly Media, Inc

iii. Online References:

1. “Edge AI for IOT Developers” Dec. 12, 2009. Accessed on: Jun. 28, 2021 [Online]. <https://www.udacity.com/course/intel-edge-ai-for-iot-developers-nanodegree--nd131>

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS282	Robotics & Automation	3	0	2	4

A.Preamble

Robotics and Automation course is designed to make students to impart basic knowledge of Robotics and automation concepts, robot vision, image processing sensors and implementation. Students are also exposed to gain knowledge over robot design and control through programming for industry applications.

B.Prerequisite Course

10211CS213-Python Programming

C.Course Objectives

Students are able to

- Understand the Fundamental concepts of robotics and automation
- Understand the concepts of robotic sensors, control and planning
- Understand the algorithmic approach towards designing intelligent systems.
- Learn about robotic automation in Industry 4.0.

D.Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K – Level/ S - Level
CO1	Apply the fundamentals concepts of Robotics and automation	K3/S3
CO2	Implement various robotic vision processing application using robotic sensors	K3/S3
CO3	Use the concepts of robotics motion control and kinematics that form the structure of robotic systems.	K3/S3
CO4	Experiment with localization and path planning techniques	K3/S3
CO5	Build AI and robotic process automation applications related to industry automation.	K3/S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of COs with Program outcomes and Programme Specific Outcomes:

Cos	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO1 0	PO1 1	PSO1	PSO 2	PSO 3
CO1	2	2	2										2	
CO2	2	2	2	2									2	2

CO3	3	2	2	2	2								2	2
CO4	3	2	2	2	2								2	2
CO5		2	2	2										2

3- High; 2-Medium; 1-Low

F. Course Contents

UNIT 1 Introduction to Robotics And Automation

L-6 Hours

Introduction to Robotics- History, Robot features, Artificial Intelligence and Robotics - Agent-Task environment model, Application of AI in robotics (unmanned systems, defense, medical, industries, etc.). Introduction to Automation- Definition, automation principles and strategies, scope of automation, socio-economic consideration, Information processing in manufacturing industry, Production concepts and automation strategies.

UNIT 2 Sensors and Robotic Vision Processing

L-6 Hours

Introduction to sensors, Types of Sensors - Internal sensors and External sensors, Sensor data interpretation and sensor fusion. Robot vision processing - Image acquisition techniques, camera transfer characteristics, Raster scan, Image capture time, Image representation, picture coding techniques. Robot Control through Vision sensors, Robot vision locating position, Robot guidance with vision system. Vision applications in robotics.

UNIT 3 Robotic Control and Kinematics

L-6 Hours

Robot Control: Fundamental principles, Classification, Position, path and speed control systems, adaptive control, Types of control- open loop and closed loop. Kinematics systems - Definition of mechanisms and manipulators, Degrees of Freedom, Parameters of links and joints, Kinematic chains, Trajectory planning and control.

UNIT 4 Localization and Intelligent Path Planning

L-6 Hours

Introduction to Localization and Mapping in robotics, Techniques - Extended Kalman Filter, Particle filter, Simultaneous localization and mapping (SLAM). Intelligent Path planning – Map representation, Adaptive Approaches - Reinforcement learning for control, Model Based learning approaches to control Learning maps and Evolutionary approaches.

UNIT 5 Robotics and Automation in Industry 4.0

L-6 Hours

Introduction - Artificial Intelligence and Industry 4.0, Mobile robots and Industry 4.0 – Intelligent Features, Advanced user interface. Robotic Process Automation (RPA) , RPA Tools - UiPath , Kofax, Automation Anywhere, WinAutomation, AssistEdge, Automagica.

Total: 30 Hours

Laboratory Experiments**Total : 30 Hours****Part - 1**

- TASK 1** Implementation of a basic Robotic Automation program.
Language: Python
- TASK 2** Implementation of vision control sensor.
Language: Python
- TASK 3** Implement robotic pick and place process.
Language: Python
- TASK4** Implement robotic computer vision programming.
Language: Python , Tool: OpenCV
- TASK5** Implement robotic Trajectory planning and analysis.
Language: Python
- TASK6** Create a mobile robot to perform Forward kinematics and validation.
- TASK7** Localization and Mapping in robotics using SLAM.
Language: Python
- TASK8** Implement path planning algorithm for robotics.
Language: Python
- TASK9** Implementation of degrees of freedom.
Language: Python
- TASK10** Implementation of RPA for industry application.
Tool: Open-source RPA

Part-2**Use Cases:**

Use Case 1: Consider an environment where your robot can be used to find the distance from an obstacle using robotic vision. Apply distance or proximity sensors to detect the obstacles and to avoid collisions. Use vision sensors to acquire 3D data of the environment for object detection and collision avoidance.

Use Case 2: Consider an environment where robotic navigation can be performed using 2D vision sensors and image processing libraries like OpenCV to detect the presence of objects.

Use Case 3: Use mobile robots and apply SLAM technique to build up a map of an unknown environment or update a map within a known environment by tracking the current location of a robot to perform robotic trajectory and also estimate localization your mobile robot using the map.

Use Case 4: Consider robot arm DoF (Degree of Freedom) to make the robot arm reach the target component from any state. The arm should not go below the ground. The target component is always placed above the ground and within the reach of the robotic arm.

Use Case 5: For an industry perform create a bot to automate sending of E-mail, where the user just needs to provide their login credentials and the user email they want to send and rest of the work will be done by the bot.

Total: 60 Hours

H.Learning Resources

i. Text Books:

1. M. P. Groover, Ashish Dutta, “Industrial Robotics”, Tata McGraw-Hill Education Pvt. Ltd, 2012. [Unit 1-5]

ii. Reference Book(s):

1. John J. Craig, “Introduction to Robotics”, Pearson Education Asia, 2009. [Unit -1]
2. Lentin Joseph, Learning Robotics Using Python, Packt, 2015. [Unit 2-4]
3. Mikell P Groover & Nicholas G Odrey, Mitchel Weiss, Roger N Nagel, Ashish Dutta, “Industrial Robotics, Technology programming and Applications”, McGraw Hill, 2012. [Unit 5]

iii. Online Resource(s):

1. “Mobile Robots”. Accessed on: May, 27, 2021[Online]. Available: https://onlinecourses.nptel.ac.in/noc21_me44/preview
2. Mechanisms and Robotics Lab (ernet.in). Available: <http://vlabs.iitkgp.ernet.in/mr/#>
3. Webots: robot simulator. Available: <https://cyberbotics.com/> [Open-source simulator]

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS283	Industrial Automation	3	0	0	3

A. Preamble

Industrial Automation became a must part of every industry. Automation helps industries to have increased quality with high productivity, accuracy, and precision to industrial processes. The course will provide students, an opportunity to learn the industrial automation techniques and address engineering problems related to process industries. After completion of this course, the students will be industry ready for deployment in the field of Automation.

B. Prerequisite Course

10212CS211-Artificial Intelligence Techniques

C. Course Objectives

Students are able to

- Introduce the basics of the Internet of Things (IoT).
- Understand sensors, communication, and networking in IoT.
- Explore IoT architecture, models, and tools like Arduino and Raspberry Pi.
- Study real-world IoT applications across various domains.
- Apply AI and IoT in industrial automation and smart systems.

D. Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K – Level/ S - Level
CO1	Understand Industrial Automated production lines, work part transfer mechanism and buffer storage analysis.	K2
CO2	Select the right material handling system for a given application.	K2
CO3	Apply testing strategies on different industry based applications	K3
CO4	Demonstrate a given engineering problems in the framework of different AI methods	K3
CO5	Apply the concept of AI in Robotic Applications.	K3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of COs with Program outcomes and Programme Specific Outcomes:

Cos	PO 1	PO2	PO3	PO4	PO 5	PO 6	PO 7	PO 8	PO9	PO1 0	PO 11	PSO 1	PSO 2	PSO 3
CO1	2	-	-	-	-	-	-	-	-	-	-	-	-	-
CO2	2		-	-	-	-	-	-	-	-	-	-	-	-
CO3	-	2	2	2	-	-	-	-	-	-	-	2	-	-
CO4	-	2	2	2	-	-	-	-	-	-	-	-	-	2
CO5	-	2	2	-	-	2	-	-	-	-	-	-	-	2

3- High; 2-Medium; 1-Low

F.Course Contents**Unit 1 Introduction****L-9 Hours**

Definition, automation principles and strategies - scope of automation - Architecture of Industrial Automation system, socio-economic consideration, low cost automation - Production concepts and automation strategies - Fixed Automation: Automated Flow lines, Methods of Work part Transport. Transfer Mechanism - Continuous transfer, intermittent transfer - Indexing mechanism Automation for Machining Operations, Design and Fabrication Considerations. Analysis of Automated Flow Lines: General Terminology and Analysis.

Unit 2 Material Handling**L-9 Hours**

Concepts of material handling, principles and considerations in material handling systems design, conventional material handling systems - industrial trucks, monorails, rail guided vehicles, conveyor systems, cranes and hoists, advanced material handling systems, automated guided vehicle systems, automated storage and retrieval systems(ASRS), Product identification system: Barcode, RFID etc. Introduction to Automation Plant design software

Unit 3 Automated Inspection and Testing**L-9 Hours**

Automated Inspection Principles and Methods, Sensor Technologies for Automated Inspection, Coordinate Measuring Machines, Other Contact Inspection Methods, Machine Vision, Other optical Inspection Methods. Image representation, object recognition and categorization, depth measurement, visual inspection. Key performance Indicators.

Unit 4 Industrial AI Applications**L-9 Hours**

AI in Robotics : Applications in unmanned systems, defense, medical, Industrial AI in Monitoring, optimization and control. AI applications in Industry Automation using -natural language processing, AI in retail and ecommerce, Healthcare industries, logistics and transportation, manufacturing industries, Robotic perception, localization, mapping-configuring space, planning uncertain movements, dynamics and control of movement, Ethics and risks of artificial intelligence in robotics.

Unit 5 Robotics in Automation**L-9 Hours**

Application of Robots in welding, Spray painting, assembly operation, cleaning, robot for underwater applications. Gripper force analysis and gripper design for typical applications, design of multiple degrees of freedom, active and passive grippers - Factors influencing the choice of a robot, robot performance testing- Impact of robot on industry and society.

Total: 45 Hours**H.Learning Resources****i. Text Books:**

1. M.P. Groover, “Automation, Production Systems and Computer Integrated Manufacturing”, 5 th Edn., Pearson Education, 2016
2. Deb S R.and DebS., —”Robotics Technology and Flexible Automation”, Tata McGraw Hill Education Pvt. Ltd, 2010

ii. Reference Book(s) [IEEE Format]:

1. Nicholas Odrey , Mitchell Weiss , Mikell Groover, Roger Nagel & Ashish Dutta, "Industrial Robots - Technology, Programming and Applications", McGraw Hill, New York, 2017

iii. Web Resource(s) [IEEE Format]:

- 1.https://onlinecourses.nptel.ac.in/noc20_me39/preview
- 2.<https://www.mindsmapped.com/course-category/industrial-automation/>

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS284	Business Intelligence	3	0	2	4

A. Preamble

This course introduces the concepts, processes, and technologies used in Business Intelligence (BI). It focuses on transforming raw data into meaningful insights to support strategic and operational decision-making. The course covers BI architectures, data warehousing, ETL, OLAP, dashboards, reporting tools, and modern trends including self-service BI and augmented analytics.

B.Prerequisite Course : Nil

B. Course Objectives

Learners are exposed to:

- Understand fundamental BI concepts, framework, and lifecycle.
- Design and develop data warehouse models with ETL processes.
- Apply OLAP operations for multidimensional analysis.
- Use BI tools for dashboards, reporting, and performance measurement.
- Explore modern BI trends such as self-service BI, predictive analytics, and real-time BI.

C. Course Outcomes

Upon the successful completion of the course, students will be able to:

COs	Course Outcomes	K - Level
CO1	Explain the fundamentals and architecture of Business Intelligence systems	K2
CO2	Design data warehouse schemas and implement ETL processes	K3
CO3	Apply OLAP operations for data analysis and decision support	K3
CO4	Develop dashboards and reports using BI tools	K3
CO5	Compare traditional BI with modern BI approaches including predictive and real-time analytics	K3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

D. Correlation of COs with Program Outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3		3	3								3		
CO2		2	2	2	3							2	3	
CO3	2		3		3	2							3	
CO4	2	3		3	3	3						3	3	
CO5	2	2		3		3						3		

E. Course Contents

UNIT I – INTRODUCTION TO BUSINESS INTELLIGENCE

L-9 Hours

Introduction – Evolution of BI – BI Framework – BI Lifecycle – BI Applications – Benefits and Challenges – BI vs Data Analytics vs Data Science – Decision Support Systems- **Case Study:** *Retail Sales Performance Analysis*

UNIT II – DATA WAREHOUSING & ETL

L-9 Hours

Data Warehousing Concepts – Data Warehouse Architecture – OLTP vs OLAP – Star Schema, Snowflake Schema, Fact and Dimension Tables – ETL Process – Data Quality, Cleansing, and Transformation. **Case Study:** *Banking Data Warehouse Implementation*

UNIT III – OLAP & DATA MODELLING

L-9 Hours

OLAP Models: MOLAP, ROLAP, HOLAP – OLAP Operations: Slice, Dice, Drill-Down, Roll-Up, Pivot – Multidimensional Modeling – Metadata Management – KPI and Metrics. **Case Study:** *Telecom Customer Usage Analysis*

UNIT IV – BI TOOLS, REPORTING & DASHBOARDS

L-9 Hours

BI Tools Overview (Power BI, Tableau, Qlik) – Dashboard Design Principles – Data Visualization Best Practices – Scorecards – Interactive Reports – Real-Time Reporting – Case Study with Business Metrics. **Case Study:** *Healthcare Dashboard for Hospital Management*

UNIT V – ADVANCED TRENDS IN BI

L-9 Hours

Self-Service BI – Cloud BI – Mobile BI – Real-Time BI – Predictive Analytics in BI – AI-Enabled BI (Augmented Analytics, Natural Language Querying) – Future of BI. **Case Study:** *E-Commerce Real-Time and AI-Driven B*

F. Learning Resources

i. Text Books:

1. Ramesh Sharda, Dursun Delen, and Efraim Turban, *Business Intelligence and Analytics: Systems for Decision Support*, Pearson, 10th Edition, 2020.
2. Larissa T. Moss & Shaku Atre, *Business Intelligence Roadmap: The Complete Project Lifecycle for Decision-Support Applications*, Addison Wesley, 2019.

ii. Reference Books:

1. Foster Provost & Tom Fawcett, *Data Science for Business*, O'Reilly, 2020.
2. Rick Sherman, *Business Intelligence Guidebook: From Data Integration to Analytics*, Elsevier, 2015.

iii. Web Resources

1. "Business Intelligence Basics," Accessed Dec 2025.
2. "Power BI Documentation," Accessed Dec 2025.
3. "Fundamentals of OLAP and Data Warehouse," Accessed Dec 2025.

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS285	Drone Technology	3	0	2	4

A. Preamble

This course introduces the fundamentals of Unmanned Aerial Vehicles (UAVs), covering drone types, components, aerodynamics, sensors, and flight control systems. Students learn the basics of drone assembly, calibration, and safe operation using simulators and field practice. The course also explores key applications of drones in agriculture, surveillance, mapping, logistics, and disaster management, along with essential DGCA regulations and safety guidelines. By the end of the course, students will understand how drones work, how to operate them safely, and how they are applied in real-world scenarios.

B. Course Objectives

Learners are exposed to:

- Introduce the foundational knowledge of drone concepts, classifications of Unmanned Aerial Vehicles (UAVs) and industry relevance.
- Introduce students to the key hardware components, onboard systems, and overall architecture of UAVs.
- Develop skills in drone programming, software interfacing, and system-level interaction.
- Enable learners to explore and evaluate real-world commercial applications of drone technology.
- Build awareness of operational safety, legal frameworks, ethical considerations, and cyber security practices in UAV usage.
- Explore real-world applications of drones across various industries and emerging advanced technologies.

C. Course Outcomes

Upon the successful completion of the course, students will be able to:

COs	Course Outcomes	K - Level
CO1	Understand the fundamental concepts, types, and business potential of drone technologies.	K2
CO2	Identify and explain the main hardware components and system architecture of drones.	K3
CO3	Apply programming knowledge to control drones and interact with onboard systems using software tools.	K3
CO4	Analyze commercial applications of drones in industries such as agriculture, logistics, surveillance, and mapping.	K3
CO5	Demonstrate awareness of safety rules, ethical concerns, legal regulations, and cyber security in drone usage.	K3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

C. Correlation of COs with Program Outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO1 0	PO1 1	PSO 1	PSO 2	PSO 3
CO1	3	2	2	2								2	2	
CO2	3	2	2	2								2	2	
CO3	3	3	3	2	2							2	2	
CO4	3	3	2									2	2	
CO5	3	3	3									2	2	

D. Course Content**Unit I Introduction to Drone Technology****L- 9 Hours**

Basic concepts in drone technology - History and Evolution of Drones - Drone Fundamentals and Industry Terms - UAV, UAS, RPAS - Broad Applications of Drones - Technology Risks and Challenges: Technical risks (signal loss, crashes) - Environmental limitations - Regulatory issues - Drone Opportunities and the Future.

Unit II Drone Hardware and System Architecture**L-9 Hours**

Overview of drone parts - Drone Hardware and Components: Types of frames and propellers - GPS, sensors, and power systems-Battery types and capacity-Microcontrollers and integration - Drone Project Cycle in Practice: Written proposal - Client interaction and interview - Site planning and design - Post-construction monitoring - Data Types Captured by Drones: 2D and 3D visualization - Point cloud generation - Contour maps, mesh overlays - Thermal and orthorectified imagery - Drone Data Analysis Tools.

Unit III Drone Programming and Control Systems**L- 9 Hours**

Basics of drone software - Common programming languages - Software to control drones - Writing simple programs to fly drones - Getting data from sensors - Flying drones in different modes - Programming for GPS and camera functions.

Unit IV Drone Commercial Applications**L- 9 Hours**

UAV Deployment Scenarios: Civilian, tactical, and military uses - Fixed and rotary deployment zones - Logistics and operational setup - Interoperability in the field - Sensor types for applications - Commercial Use Cases - Emergency and Disaster Management: Firefighting and surveillance - Flood mapping and rescue support - Search & rescue operations - Public health delivery during epidemics - Post-disaster insurance assessments - Security and Surveillance Applications: Border security and patrolling - Maritime reconnaissance - Urban traffic surveillance - Crowd monitoring in events - Disaster area monitoring.

Unit V Safety, Ethics, Cyber security and Regulations**L- 9 Hours**

Drone Aviation Rules and Standards: DGCA regulations in India - Licensing and operation zones - NO-FLY zone classification - Design for Reliability and Maintenance - Ethical and Legal Responsibilities: Data collection ethics - Privacy and surveillance boundaries -Commercial drone legalities - Cyber security in UAV Systems: GPS spoofing, jamming, hijacking - Secure communication protocols - Onboard data encryption - Anti-tampering mechanisms - Cyber security certifications and tests.

E. Laboratory Experiments**Total: 30 Hours****Part – 1****Task 1: Exploring Drone Simulation Environment**

- Use a drone simulation tool (Mission Planner / QGroundControl / AirSim) to understand the basic interface.
- Load a sample UAV model and inspect key parameters such as frame type, battery, motors, and sensors.

Task 2: Virtual Drone Assembly and Component Configuration

- Configure drone components including motors, ESCs, and propellers using virtual assembly tools.
- Verify component orientation, rotation direction, and propeller classification in the simulator.

Task 3: Flight Controller Tuning (PID Control)

- Adjust PID parameters for Roll, Pitch, and Yaw in the simulator.
- Observe and compare the effect of PID variations on drone stability during hovering

Task 4: Sensor Data Analysis (IMU & GPS)

- Visualize IMU sensor readings such as accelerometer, gyroscope, barometer, and magnetometer data.
- Simulate drone movement and record variations in sensor data for analysis.

Task 5: Manual Flight Operation in Simulation

- Practice basic manual controls such as take-off, landing, altitude hold, and directional movement.
- Perform controlled flight patterns like square path, circular path, and figure-8 movement.

Task 6: Autonomous Waypoint Navigation

- Create an autonomous mission using waypoint planning tools.
- Configure waypoint parameters (altitude, speed, delay) and simulate complete autonomous flight.

Task 7: Obstacle Avoidance and Path Planning

- Use built-in simulator sensors (LiDAR/Optical Flow) to detect obstacles.
- Plan and execute a path that avoids obstacles and test semi-autonomous navigation behavior.

Task 8: Drone Mapping and Survey Mission

- Simulate an aerial mapping mission using grid-based or terrain-following survey patterns.
- Generate a preview of stitched images or orthomosaic results using mapping simulation tools.

Task 9: Delivery Drone Route Simulation

- Define a package delivery route including pickup point, intermediate waypoints, and delivery zone.
- Analyze route efficiency, flight time, and simulated battery consumption.

Task 10: Drone Cyber security and Geofencing

- Configure geofencing boundaries and simulate no-fly zones in the software.
- Trigger boundary violations and observe drone response (auto-return, hover, or land).

Simulation Tools Recommended

- Mission Planner
- QGroundControl
- AirSim
- Gazebo
- Real Flight / Liftoff FPV
- MATLAB UAV Toolbox

Part – 2**Use cases:****1. Precision Agriculture – Optimizing Crop Spraying Route**

In modern precision agriculture, drones are increasingly used to spray fertilizers and pesticides across large agricultural fields. However, manual flight paths often result in uneven coverage, excessive chemical usage, and time loss. Through simulation tools such as Mission Planner or QGroundControl, agricultural flight patterns can be pre-planned and optimized before deployment. The simulator allows users to design grid-based spraying routes, adjust altitude, spray width, and overlap percentages, and visualize the final coverage pattern. By testing these parameters virtually, the operator can ensure uniform application, reduce resource wastage, and

improve operational efficiency in real-world farming conditions.

2. Search and Rescue in Disaster Zones

During natural disasters such as floods, landslides, or earthquakes, drones play a vital role in surveying affected areas to locate survivors. However, flying in unstable or obstructed environments poses risks to both operators and drones. Simulation platforms like AirSim allow emergency teams to recreate virtual disaster zones with rubble, collapsed structures, and varying weather conditions. Thermal and RGB camera simulations help test victim-detection techniques and optimal scanning altitudes. By practicing these missions virtually, rescue teams can refine their strategies, improve detection accuracy, and carry out faster and safer real-world search and rescue operations.

3. Traffic Monitoring and Accident Analysis

Urban traffic authorities use drones to monitor traffic flow, detect congestion, and capture accident scenes for analysis. Conducting these flights directly over busy highways can be difficult and hazardous. Simulation tools such as Gazebo or AirSim enable engineers to replicate real road networks and test drone flight patterns safely. Operators can experiment with different altitudes, camera angles, and monitoring intervals to determine the most effective surveillance setup. By validating monitoring strategies through simulation, agencies can ensure smoother implementation of real-world drone operations without disrupting ongoing traffic.

4. Wildlife Monitoring and Anti-Poaching Patrol

Drones assist forest departments in monitoring wildlife, tracking animal movement, and preventing illegal poaching activities, especially during the night. Simulators like AirSim can recreate forest environments with varying terrains, lighting conditions, and animal movement patterns. Operators can test night-vision cameras, infrared sensors, and autonomous patrol routes to determine optimal monitoring strategies. The virtual environment allows safe evaluation of drone performance without disturbing wildlife or risking equipment, ultimately enhancing real-world conservation and anti-poaching efforts.

5. Drone Swarm Coordination for Large Area Surveillance

Security agencies and defense organizations increasingly rely on drone swarms for border surveillance and large-area coverage. Coordinating multiple drones manually is complex and risky. Simulation environments such as Gazebo or MATLAB UAV Toolbox allow engineers to model swarm behavior, communication patterns, collision avoidance techniques, and formation flights. Through simulation, different swarm strategies can be tested without risking multi-drone crashes. This enables development of reliable surveillance patterns and ensures seamless deployment of swarm systems in real-world security missions.

F. Learning Resources

i. Textbooks:

1. Beard, R. W., & McLain, T. W. (2012). Small unmanned aircraft: theory and practice.

- Princeton University Press.
2. Valavanis, K. P. (2015). Handbook of unmanned aerial vehicles. 1. Springer.
 3. Paul Gerin Fahlstrom, & Gleason, T. J. (2012). Introduction to UAV systems. John Wiley & Sons.
 4. Jha, A. R. (2017). Theory, design, and applications of unmanned aerial vehicles. CRC Press, Taylor & Francis Group.

ii. Reference Books:

1. Singh, C., & Gatti, R. R. (2024). Drone Applications for Industry 5.0. IGI Global.
2. Gupta, S. K., Kumar, M., Nayyar, A., & Mahajan, S. (2024). Unmanned Aircraft Systems. John Wiley & Sons.

Total: 75 Hours

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS286	Linux Device Drivers	2	0	2	3

A. Preamble

This course imparts comprehensive knowledge and practical skills in Linux Device Driver (LDD) development for embedded systems. It covers kernel programming, character device drivers, interrupt handling, deferred work mechanisms, synchronization techniques and integration with Sysfs and the Linux device model. The course is designed to provide industry-oriented, application-driven training, enabling students to build real-time drivers for embedded platforms. The content is aligned with Wipro Centre of Excellence (CoE) – Next-Gen Associates (NGA) Skilling: Linux Device Drivers (LDD), ensuring that learners acquire competencies expected by industry for next-generation embedded solutions.

B. Prerequisite

10211CS103 – Operating Systems

C. Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Develop the kernel modules to program the basic kernel concepts.	K3, S3
CO2	Implement character device drivers supporting file operations and user–kernel communication.	K3, S3
CO3	Identify the interrupt handling mechanisms with appropriate synchronization techniques in drivers.	K3, S3
CO4	Apply deferred work mechanisms such as tasklets, work queues and timers in driver development.	K3, S3
CO5	Utilize advanced Linux device drivers with Sysfs and the Linux device model for embedded applications.	K3, S3

Knowledge Level (Based on revised Bloom's Taxonomy)

K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create S3- Apply

D Correlation of Cos with Program Outcomes and Programme Specific Outcomes:

Cos	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PSO 1	PSO 2	PSO 3
CO1	3	2	2	2						2		3	2	
CO2	3	2	3	2	2					2		3	3	
CO3	3	3	3	3	2					2		3	3	
CO4	3	2	3	3	2					2		3	3	
CO5	3	2	3	2	3					2		3	3	

3- High; 2-Medium; 1-Low

E.Course content**UNIT I – Linux Kernel & Module Programming****L-6 Hours**

Kernel architecture: user vs kernel space - kernel modules: insmod, rmmod, modprobe - module programming basics: init, exit, printk - kernel build system, make files for modules.

UNIT II – Character Device Drivers**L-6 Hours**

Major and minor numbers, registering device numbers - file operations: open, read, write, release, ioctl - kernel buffer management: copy_to_user, copy_from_user – case studies: char driver.

UNIT III – Kernel Interrupts and Synchronization Primitives**L-6 Hours**

Interrupt handling flow in linux - request_irq (), free_irq () - shared IRQs, interrupt context vs process context - synchronization: spinlocks, semaphores, atomic variables.

UNIT IV – Asynchronous and Delayed Execution In Kernel**L-6 Hours**

Bottom halves, tasklets and work queues - kernel timers and time management - delayed execution in drivers - case studies: interrupt + work queue driver.

UNIT V – Device Management and Debugging**L-6 Hours**

Sysfs, kobjects, udev interaction - linux device model: buses, devices, drivers - exporting attributes to user space - debugging drivers: dmesg, debugfs, and ftrace basics.

Total: 30 Hours**F. Lab Tasks****Total: 30 Hours**

S. No	CO Mapping	Practical Exercises	Dave's Taxonomy Level
1	CO1	Compile and load a basic kernel module (Hello Kernel).	S2
2	CO1	Demonstrate module parameters and printk logging.	S3
3	CO2	Develop a simple char driver with open, read, write operations.	S3
4	CO2	Demonstrate a program using copy_to_user/copy_from_user.	S3
5	CO3	Implement an interrupt handler for a simulated device event.	S3

6	CO3	Demonstrate shared interrupts using request_irq().	S3
7	CO4	Implement deferred work using tasklets.	S3
8	CO4	Implement deferred work using workqueues.	S3
9	CO5	Export kernel attributes to Sysfs.	S3
10	CO5	Demonstrate kobject creation and manipulation.	S3
11	CO1-CO5	Mini Project	S4

Total :60 Hours

G. Learning Resources

i. References

1. Robert Love, “Linux Kernel Development”, Addison-Wesley, 2010.
2. Greg Kroah-Hartman, “Linux Kernel in a Nutshell”, O'Reilly Media, 2010.
3. Jonathan Corbet et al, “Linux Device Drivers”, 3rd Edition, O'Reilly, 2005.

ii. Online Resources

1. Standards documents applicable to the Linux kernel: <https://kernel.org/doc/>
2. The Linux Documentation Project: <https://tldp.org/>
3. Explore Linux v6.16.7 source code: <https://elixir.bootlin.com/linux/latest/source>

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS287	Linux System Programming	2	0	2	3

A. Preamble

This course introduces the fundamentals of Linux System Programming (LSP) for embedded systems. It covers system calls, file operations, process management, inter-process communication Inter Process Communication (IPC), signals, multithreading and memory management. Emphasis is placed on practical application of concepts through programming exercises that build strong foundations for embedded development. The course content is aligned with Wipro Center of Excellence – Next-Gen Associates (NGA) Skilling: Linux System Programming (LSP), ensuring that students acquire industry-relevant skills bridging academic learning and professional practice.

B.Prerequisite

10211CS103 – Operating Systems

C.Course Outcomes

On successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Explain the Linux architecture, user/kernel space and toolchain usage.	K2, S3
CO2	Make use of Linux system calls for file handling for I/O process.	K3, S3
CO3	Utilize the project management calls for system process creation and scheduling the signals.	K3, S3
CO4	Construct the IPC mechanisms using pipes, shared memory and message queues.	K3, S3
CO5	Develop the synchronization and multithreading using POSIX threads and basics of debugging.	K3, S3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create S3- Apply		

D. Correlation of Cos with Program Outcomes and Programme Specific Outcomes:

Cos	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PSO 1	PSO 2	PSO 3
CO1	3	2	2	2	2					2		3	2	
CO2	3	3	3	2	2					2		3	3	
CO3	3	3	3	3	2					2		3	3	
CO4	3	3	3	3	2					2		3	3	
CO5	3	3	3	3	3					2		3	3	

3- High; 2-Medium; 1-Low

E. Course Content**UNIT I – Linux Architecture & Toolchain****L-6 Hours**

Linux architecture: user space versus kernel space - GNU Compiler Collection (GCC) tool chain: preprocessor, compiler, assembler, and linker - make files and build automation basics – command line essentials for embedded development.

UNIT II – File I/O System Calls**L-6 Hours**

Low level file I/O: open, read, write, close, lseek - file descriptors, error handling (errno) - directory operations, stat, fcntl - concept of Virtual File System (VFS).

UNIT III – Process Management System Calls**L-6 Hours**

Process creation: fork, exec family, wait, exit - process hierarchy, PID, PPID - daemon processes and scheduling - signals: handling, signal (), sigaction (), kill ().

UNIT IV – IPC Mechanisms**L-6 Hours**

Pipes: unnamed and named, FIFOs - message queues - shared memory: shmget, shmat, shmdt, shmctl - semaphores for synchronization.

UNIT V – Threads and Advanced Programming**L-6 Hours**

POSIX threads (pthreads): creation, joining, detaching - synchronization: mutex, condition variables - multithreaded producer-consumer problem - memory management: malloc, mmap, brk - error debugging and GDB (GNU Debugger) basics.

Total: 30 Hours**F. List of Tasks****Total: 30 Hours**

S. No	Practical Exercises	CO Mapping	Dave's Taxonomy Level
1	Explore Linux file system by writing a program to list directory contents using system calls.	CO1	S2
2	Construct a Makefile to automate the build process of a simple C program.	CO1	S3
3	Develop a file copy utility using open, read, write system calls.	CO2	S3
4	Create a process management program using fork(), exec() and wait() system calls.	CO2	S3
5	Implement inter-process communication using unnamed pipes.	CO3	S3
6	Design communication between processes using named pipes (FIFOs).	CO3	S3
7	Build a multithreaded program using POSIX pthreads.	CO4	S3
8	Apply synchronization by developing a producer-consumer solution with pthreads & mutex.	CO4	S3

9	Demonstrate signal handling using <code>sigaction()</code> system call.	CO5	S3
10	Utilize shared memory allocation with <code>mmap()</code> for inter-process communication.	CO5	S3
11	Integrate concepts in a Mini Project (system calls, processes, IPC, threads, signals, memory).	CO1-CO5	S4

Total: 30 Hours

G. Learning Resources

i. References

1. Robert Love, “Linux System Programming”, Second Edition, O'Reilly, 2014.
2. W. Richard Stevens, “Advanced Programming in the UNIX Environment”, Pearson Education, 2013.
3. Michael Kerrisk, “The Linux Programming Interface”, No Starch Press, US, 2010.

ii. Online Resources

1. Linux manual pages: <https://man7.org/linux/man-pages/>
2. The Linux Documentation Project: <https://tldp.org/>
3. Standards documents applicable to the Linux kernel: <https://kernel.org/doc/>

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS288	Salesforce Ecosystem and Application Development	3	0	2	4

A. Preamble

This course introduces students to Salesforce as a leading CRM platform, covering its ecosystem, declarative tools, programmatic development, and application lifecycle. Students explore Salesforce administration, platform development, Lightning Web Components (LWC), and JavaScript fundamentals aligned with Salesforce certifications. Hands-on labs include real-world projects and Salesforce Superbadges.

B. Prerequisite Courses

NIL

C. Course Objectives

Learners are exposed to

- Understand Salesforce cloud architecture and multi-tenant platform.
- Configure and administer Salesforce using declarative tools.
- Develop applications using Apex, SOQL, SOSL, and LWC.
- Apply JavaScript skills for Salesforce development.
- Implement real CRM use cases with best practices and Superbadges.

D. Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Use Salesforce navigation, ecosystem features, and basic org settings to perform essential CRM tasks effectively.	K3
CO2	Build data models, custom fields, and reporting tools to support organizational needs in Lightning Experience.	K3
CO3	Implement identity controls, permissions, data security, and Service Cloud features to manage users and business processes.	K3
CO4	Develop Apex logic, triggers, and automation processes while debugging and testing backend components.	K3
CO5	Construct Lightning Web Components using JavaScript, CLI tools, and Salesforce data access methods for interactive user interfaces.	K3

E. Correlation of Cos with Program Outcomes and Programme Specific Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3				2						2	3		
CO2	3	2	2		2							3	2	
CO3	3	2	2		3			2				3	2	
CO4	3	3	3	2	3							3	3	
CO5	2	1	3		3			2	2			2	1	

High - 3, Medium - 2, Low – 1

F. Course Contents**Unit I: Introduction to Salesforce and Navigation Essential L – 9 Hour**

Salesforce Values, Guide to Trailhead, Trailhead Playground Management, Salesforce for Nonprofits Basic, Salesforce Ecosystem, Agentforce Key Components, Company-Wide Org Settings, List Views, Setup, Scrum and Kanban at Salesforce, Salesforce CRM.

Unit II: Salesforce Associate L – 9 Hour

Salesforce Basics, Salesforce Platform Basics, Data Modeling, Lightning Experience Productivity, Picklist Administration, Custom Fields, App Tools and Reporting, Lightning Knowledge Setup and Customization, Reports & Dashboards, Joined Reports, AppExchange Basics.

Unit III: Salesforce Administrator L – 9 Hour

Identity Basics, Permission Set Groups, Data Security, Duplicate Management, Campaign Basics, Collaboration & Cloud Services, Chatter Administration for Lightning Experience, Service Cloud for Lightning Experience.

Unit IV: Salesforce Developer L – 9 Hour

Apex Fundamentals, Apex & .NET Basics, Apex Basics & Database, Apex Triggers & Automation, Record-Triggered Flows, Approve Records with Approval Processes, Async Processing & Debug Tools, Asynchronous Apex, Developer Console Basics, Find and Fix Bugs with Apex Replay Debugger, Visualforce Basics, Secure Server-Side Development, Org Development Model, Apex Testing

Unit V: JavaScript Developer L – 9 Hour

JavaScript Skills for Salesforce Developers, Get Started with Modern JavaScript Development, Coding for Web Accessibility, Command-Line Interface, Lightning Web Components Basics, Lightning Web Components for Visualforce Developers, Lightning Web Components for Aura Developers, Lightning Web Components and Salesforce Data, Lightning Web Components Tests.

Total: 45 Hours**G. Laboratory Experiments****Total: 30 Hours**

Task 1: Login to Salesforce Playground and locate the core CRM objects Leads, Accounts, Contacts, and Opportunities. Open each object and capture screenshots of recorded data and navigation paths.

Task 2: Customize the Lightning Home page in Salesforce Playground using Lightning App Builder by adding components such as Recent Records, Key Deals, and Tasks, and save the updated layout.

Task 3: Create a new custom object in Salesforce Playground along with at least three custom fields of different data types (Text, Number, Picklist) and display the object in the App Launcher.

Task 4: Generate a report in Salesforce Playground for Accounts or Contacts, apply relevant filters, group the data, and produce the final summarized report view.

Task 5: Create a new user in Salesforce Playground, assign a profile and permission settings, and validate the access configuration for the created user.

Task 6: Build a dashboard in Salesforce Playground using an existing report and add at least three visual components such as charts or gauges, then record the completed dashboard view.

Task 7: Create and activate a Record-Triggered Flow in Salesforce Playground and demonstrate its execution by updating a record and verifying the automatic change.

Task 8: Execute SOQL queries in the Developer Console within Salesforce Playground to retrieve specific records and insert new records using Apex, then confirm the data update.

Task 9: Write and test an Apex Trigger in Salesforce Playground that performs an update or action when a record is created or modified, and provide verification using debug logs.

Task 10: Develop a Lightning Web Component in Salesforce Playground that displays formatted information in a Lightning Card, deploy the component, and capture its displayed output.

Total: 75 Hours

H. Learning Resources

i. Text Book(s):

1. Paul Battisson, "Learning Salesforce with Apex", BPB Publisher, Second Edition, 2022.
2. Mohith Shrivatsava, "Salesforce Essentials for Administrators", Packt Publishing Limited, 2014.

ii. Reference Book(s):

1. James A. Hutcherson, "Mastering Salesforce Flow for Enterprises: An end-to-end guide to creating, troubleshooting, and using Salesforce Flows", BPB Publisher, 2024.

ii. Online Reference(s):

1. "Salesforce Trailhead", Accessed on: Apr. 20, 2024 [Online], Available: <https://trailhead.salesforce.com/>

COURSE CODE	COURSE TITLE	L	T	P	C

10212CS291	Foundations of Mainframe Computing	3	0	2	4

A. Preamble

This course is intended to provide the foundational skills required to launch a career in the enterprise platform that powers over 68% of worldwide transactions. Companies around the world are searching for these skills to run their critical system workloads. In this course, the student will be introduced to various operating systems, mainframe applications, and get real-time hands-on experience on an IBM Z server. The skills acquired on completion of this course allows the learner to pursue a career as a mainframe application developer, system programmer, system administrator, or a DBA Practitioner.

B. Prerequisite Courses

Operating systems and Computer Organization and Architecture

C. Course Objectives

Learners are exposed to

- IBM z/OS Mainframe
- z/OS Commands and Panels on IBM Z
- System Programming on IBM Z

D. Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Explain mainframe hardware components and the structured nature of the mainframe I/O architecture.	K2
CO2	Demonstrate z/OS Commands and Panels on IBM Z.	K3
CO3	Describe the responsibilities of JES and JCL.	K3
CO4	Explain System Programming Components.	K2
CO5	Explain the z/OSMF and UNIX System Services.	K2
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Course Contents

Unit 1 Introduction to IBM z/OS

5 Hours

Mainframe Environment: Computer Architecture - Mainframe Architecture – Virtualization - Mainframe Operating System - Diving Deeper into z/OS – z/OS workload types (Batch, Online, CICS, IMS) -Mainframe Infrastructure - IBM Z components - Mainframe Features - IBM Z

Configuration Setup – Millicode - HMC and SE – Sysplex & **Parallel Sysplex**- Mainframe Security - z/OS Security - Security Addendum - Transaction Level Security - Crypto Express Key Management – IOCDS - Activation Profiles - IPLing an Operating System - z/OS MVS System Initialization Logic Initial Program Load (IPL).

Unit 2 Introduction to z/OS Commands and Panels on IBM Z

5 Hours

Introduction to Commands and Panels - Introduction to TSO and ISPF - z/OS 2.3 ISPF **Navigation** - Creating, editing, and displaying members in ISPF – z/OS Data Set Types - Allocating a data set in TSO - Working with data sets - TSO Commands – **ISPF Enhancements**

Unit 3 Introduction to JES and JCL

5 Hours

System Programming – JES and JCL – **JES2 functionality** & Job Flow, **Submitting JCL via ISPF**, **Zowe CLI and SDSF for Job Monitoring** - Common Utilities- IEFBR14, IEBGENER, IEBCOPY, SORT, JCL Procedures, Error Diagnostics -Abend codes.

Unit 4 System Programming Components

7 Hours

VSAM – z/OS Components – z/OS System Libraries - Application Infrastructure - DFSMS - Non-SMS versus SMS authorization - **PARMLIB, PROCLIP, LINKLIST** - Language Environment - Generation data groups – RAIM -**Application Infrastructure -CICS**, IBM DB2 12 for z/OS – Networking in z/OS, **Introduction to z/OSMF REST APIs , Dataset APIs, Job APIs, console APIs.**

Unit 5 z/OSMF and UNIX System Services

8 Hours

Introduction to z/OSMF - IBM z/OS Management Facility V2R3 - UNIX System Services - USS file systems - ISHELL and hierarchical file system - USS Processes and Permissions, **Zowe CLI**, **Introduction to Ansible**, Ansible for z/OS Overview , Ansible Collections

Total: 30 Hours

Lab Exercises: (30 hours)

z/OS Commands and Panels

Exercise 1. System Familiarization

Section 1: Gathering information and accessing the lab image

Section 2: The logon screen

Section 3: ISPF Primary Menu

Section 4: ISPF Settings

Section 5: ISPF Settings: Action bar choices

Section 6: View a data set

Section 7: Exit ISPF

Section 8: Log Data Set Disposition

Section 9: Logoff

Exercise 2. Allocate new data sets

Section 1: Lab exercise assistance

Section 2: Allocate a Physical Sequential data set

Section 3: Allocate a Partitioned Data Set

Section 4: Allocate a Partitioned Data Set Extended

Exercise 3. ISPF editor primary commands

Section 1: Create a PDS member

Section 2: Edit the LITLGAME member

Section 3: Ending the exec

Exercise 4. ISPF editor line commands

Section 1: Create a new member and copy data

Section 2: Copy / overlay commands

Section 3: Insert lines

Section 4: Add and repeat lines

Exercise 5. Copy, move, rename, delete data sets and members

Exercise 6. Data set lists

Exercise 7. Using TSO/E commands

Section 1: Accessing the TSO command line

Section 2: TSO commands

JES and JCL

Exercise 1. Submit a job

Section 1: Create a data set

Section 2: JCL and viewing help

Section 3: Rename a data set member

Exercise 2. JCL exercises

Section 1: Copy a selected member of a PDS

Section 2: Copy an entire data set

Section 3: Copy and replace a selected member

Section 4: Perform a securecompress

Section 5: Compress a data set in place

Section 6: Copy a PDS and a PDSE data set

Section 7: Copy in-stream data to PS

Section 8: Print a Sequential data set to SYSOUT

Section 9: Print a PDS member

Section 10: Create a PDS member

Exercise 3. Procedures

Section 1: Override DD-statements of an in-stream procedure

Section 2: Run cataloged procedure

Section 3: Create cataloged procedure

Section 4: Use of DFSORT (ICEMAN)

Section 5: Create PS from in-stream

Section 6: Conditional processing

Section 7: Route a job to INTRDR using IEBGENER

Exercise 4. z/OS Management Facility

Section 1: Accessing the z/OS Management Facility

Section 2: z/OSMF interface layout

Section 3: TSO interface

Section 4: Links

Section 5: Setting up a z/OS console

Section 6: Using the z/OS console

Section 7: Entering z/OS commands and interpreting the output

Section 8: Identifying the system configuration

Section 9: The DEVSERV command

Exercise 5. ISHELL and hierarchical file system

Section 1: ISHELL introduction

Section 2: Introduction to the file system

Section 3: Customize ISHELL

Section 4: UNIX directory list

Section 5: Create a hierarchical file system

Section 6: Managing files

Section 7: Additional UNIX activities

Section 8: Introduction to the file system

Exercise 6 : Zowe CLI

- Login
- Dataset List
- Job Submits
- Job Output

Exercise 7 Working with Ansible Playbooks for z/OS

- Dataset creation using Ansible
- Submitting JCL using Ansible
- Copying data sets
- Editing members
- USS file operations
- Job monitoring via Ansible

Exercise 8 Ansible with z/OSMF REST APIs

- API Authentication
- Endpoint overview
- Using Ansible to:
- Create datasets
- Submit a job
- Read spool content
- Create USS files
- Issue z/OS operator commands

Learning Resources**i. Text Books:**

1. Mike Ebbers, John Kettner, Wayne O'Brien, Bill Ogden, "Introduction to the New Mainframe z/OS Basics", IBM Redbooks, Third Edition, 2011

ii. Online References:

1. Introduction to IBM z/OS. Accessed on: Apr. 18, 2023 [Online]. Available: <https://learn.ibm.com/course/view.php?id=6815>
2. Introduction to z/OS Commands and Panels on IBM Z. Accessed on: Apr. 18, 2023 [Online]. Available: <https://learn.ibm.com/course/view.php?id=7419>
3. Introduction to System Programming on IBM Z. Accessed on: Apr. 18, 2023 [Online]. Available: <https://learn.ibm.com/course/view.php?id=7512>

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS292	Applied Programming Skills	3	0	2	4

A. Preamble

This course develops problem-solving and programming skills using Java. It focuses on Data Structures and Algorithms concepts, encouraging students to implement, test, and optimize solutions for real-world computational problems. Students will learn to design efficient programs using arrays, linked lists, stacks, queues, trees, graphs, and advanced algorithmic techniques.

B. Prerequisite Course:

Nil

C. Course Objectives

Learners are able to:

- Understand and use linear data structures for efficient data processing.
- Apply stack and queue to solve real world problems.
- Implement tree and heap structures and learn programming skills.
- Utilize suitable data structures and algorithms to solve graph problems effectively.
- Handle real world programming scenarios using algorithm design techniques.

D. Course Outcomes

Upon the successful completion of the course, students will be able to:

COs	Course Outcomes	K Level
CO1	Use array and linked list to process competitive programming problems.	K3
CO2	Implement stack and queue-based patterns to solve problems.	K3
CO3	Apply tree and heap structures to handle hierarchical data and priority-based computations.	K3
CO4	Utilize graph and set-based structures to analyze connectivity, traversal, and relationship-driven problems.	K3
CO5	Demonstrate algorithmic solutions using programming strategies for various programming problems.	K3

E. Correlation of COs with Program Outcomes and Programme Specific Outcomes

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	2	1	1	1	-	-	-	-	-	2	3	2
CO2	3	3	2	2	1	1	-	-	-	-	-	2	3	2
CO3	3	3	3	2	2	1	-	-	-	-	-	3	3	2
CO4	3	3	3	3	2	1	-	-	-	-	-	3	3	3
CO5	2	3	3	3	3	2	-	-	1	2	1	2	3	3

3- High; 2-Medium; 1-Low

F. Course Contents**Unit I: Linear Data Structures for Competitive Input Processing****L - 9 Hours**

Two Pointers – Sliding Window – Prefix Sum – Difference Array – Binary Search – Quick Sort and Merge Sort, Greedy pairing – Hashing for frequency/count – In-place array manipulation – Linked list fast/slow pointer – Merge two lists – Reverse nodes in k-group – Detect/remove cycle – Intersection of linked lists.

Unit II: Stack and Queue Based Competitive Problem Modeling**L - 9 Hours**

Monotonic Stack – Next Greater Element pattern – Valid parentheses variants – Min Stack design – Stack simulation problems – Queue using stacks – Sliding window maximum using deque – FIFO processing with queue – Circular queue simulation – Task scheduling with queue.

Unit III: Hierarchical and Priority-Oriented Data Structures**L-9 Hours**

Tree – Tree Traversal – Print Birdseye View of a Tree - Path sum - Root-to-leaf problems – Diameter/height problems – Lowest Common Ancestor – Kth smallest/largest in Binary Search Tree – Heap - Top-K pattern – K-way merge using heap.

Unit IV: Network and Relationship-Driven Data Structures**L - 9 Hours**

Graph – Breadth First Search –Depth First Search – Connected components – Cycle detection (directed/undirected) – Topological sorting– Shortest path in unweighted graphs – Grid traversal (islands, flood fill) – Union Find (Disjoint Set Union) – HashSet patterns – Frequency map and sorting by keys/values.

Unit V: Algorithm Design Techniques**L - 9 Hours**

Divide and Conquer pattern – Merge sort for counting inversions – Quick select (Kth element) – Dynamic Programming – One Dimensional (House robber, Climbing stairs) – Two Dimensional (grid paths) – Subset Dynamic Programming (0/1 Knapsack style) – Greedy interval scheduling – Backtracking (Subsets, Permutations, Combinations).

TOTAL: 45 Hours**G. Laboratory Experiments****Part A****Task 1: (Level: Easy)**

Given an array of intervals, merge all overlapping intervals.

Approach / Explanation: Sort by start. Iterate, merging when $\text{current.start} \leq \text{last.end}$.

Sample Input: 4

1 3
2 6
8 10
15 18

Sample Output:

1 6
8 10
15 18

Task 2: (Level: Easy)

Insert a new interval into a list of non-overlapping, sorted intervals, merging if necessary.

Approach / Explanation: Add all intervals ending before new.start , merge overlaps, then append the rest.

Sample Test Case:

Sample Input:

3
1 3
6 9
12 16
4 10

Sample Output:

1 3
4 10
12 16

Task 3: (Level: Easy)

You are given an array `arr[]` of size $n - 1$ that contains distinct integers in the range from 1 to n (inclusive). This array represents a permutation of the integers from 1 to n with one element missing. Your task is to identify and return the missing element.

Examples:

Input: `arr[] = [1, 2, 3, 5]`

Output: 4

Explanation: All the numbers from 1 to 5 are present except 4.

Input: `arr[] = [8, 2, 4, 5, 3, 7, 1]`

Output: 6

Explanation: All the numbers from 1 to 8 are present except 6.

Input: `arr[] = [1]`

Output: 2

Explanation: Only 1 is present so the missing element is 2.

Constraints:

$1 \leq \text{arr.size()} \leq 10^6$

$1 \leq \text{arr}[i] \leq \text{arr.size()} + 1$

Task 4: (Level: Easy)

Write a function that finds and returns the Nth prime number. The number N will be passed as input to the function

constraint: $1 \leq N \leq 1000$, where N is the position of the prime number

Sample Test Case:

Sample Input:

10

Sample Output:

29

Task 5: (Level: Medium)

Given the head of a singly linked list, the task is to remove a cycle if present. A cycle exists when a node's next pointer points back to a previous node, forming a loop. Internally, a variable “**pos**” denotes the index of the node where the cycle starts, but it is not passed as a parameter. The terminal will print true if a cycle is removed otherwise, it will print false.

Examples:

Sample Input 1: head = 1 -> 3 -> 4->1, pos = 0

Sample Output 1: true

Explanation: The linked list has a loop is present in the list, and it is removed.

Sample Input 2: head = 1 -> 8 -> 3 -> 4, pos = 0

Sample Output 2: false

Explanation:

The Linked list does not contains any loop.

Sample Input 3: head = 1 -> 2 -> 3 -> 4->1, pos = 0

Sample Output 3: true

Explanation: The linked list has a loop is present in the list, and it is removed.

Constraints:

$$1 \leq \text{size of linked list} \leq 10^5$$

Task 6: (Level: Hard)

Library Management System

Scenario: A city-library chain wants to digitize its inventory and borrower system.

Internal tasks:

1. Design and implement data structures for book inventory (arrays or linked lists) and borrower records.
2. Implement borrow/return operations using a queue of pending returns and stack for recent returns.
3. Implement search (linear & binary) for books by title/ISBN, and sort books by popularity (using quick sort or merge sort).

Task 7: (Level: Easy)

Given a string s containing '()[]{}', determine if it is valid.

Approach / Explanation: Use a stack; push openings and check matching closings.

Sample Test Case:

Sample Input1:

()[]{}

Sample Output2:

True

Sample Input2:

(]

Sample Output2:

False

Task 9: (Level: Easy)

You are given a string S that contains lowercase and uppercase English letters along with brackets '(' and ')'. Your task is to:

1. Remove all brackets '(' and ')' from the string.
2. Reverse the resulting string.
3. Print the final reversed string.

Input Format:

- The first line contains a string S ($1 \leq |S| \leq 10^5$) consisting of lowercase and uppercase letters and brackets '(' and ')'.

Output Format:

- Print the final string after removing brackets and reversing.

Sample Test Case:

Sample Input:

a(bc)d

Sample Output:

dcba

Task 11: (Level: Easy)

Given an integer array `nums` and integer `k`, return the `k` most frequent elements.

Approach / Explanation: Count with hashmap; use bucket sort (frequency -> list of values) or a min-heap of size `k`.

Sample test Case:**Sample Input 1:**

1 1 1 2 2 3

2

Sample Output 1: 1 2

Task 12: (Level: Easy)

Given two strings `s` and `t`, determine if `t` is an anagram of `s`.

Approach / Explanation: Count characters with fixed-size array.

Sample Test Cases:**Sample Input 1:**

listen

silent

Sample Output 1:

true

Sample Input 2:

Rat

tar

Sample Output 2:

False

Task 13: (Level: Medium)

You are given a binary tree with `N` leaf nodes, each leaf node containing a positive integer representing its data size.

You need to combine all leaf nodes into a single file using the minimum total cost.

The cost of merging two leaf nodes is equal to the sum of their sizes.

Each time you merge two files, the resulting file is pushed back to the set of available files.

Your task is to find the minimum total cost to merge all the files into one. This problem is analogous to constructing an Optimal Binary Merge Tree (OBMT) — or equivalently, the Huffman coding tree — where merging with minimal total cost is achieved using a min-heap (priority queue).

Explanation

Let's say you have leaf sizes:

[5, 3, 8, 2]

Pick the two smallest $\rightarrow$ 2 and 3 $\rightarrow$ merge cost = $2 + 3 = 5$.

New list $\rightarrow$ [5, 5, 8]

Pick the two smallest $\rightarrow$ 5 and 5 $\rightarrow$ merge cost = $5 + 5 = 10$.

New list $\rightarrow$ [8, 10]

Pick the two smallest $\rightarrow$ 8 and 10 $\rightarrow$ merge cost = $8 + 10 = 18$.

Total cost = $5 + 10 + 18 = 33$

Minimum total merge cost = 33

Input Format

N

a1 a2 a3 ... aN

N $\rightarrow$ number of leaf nodes ($1 \leq N \leq 10^5$)

a_i $\rightarrow$ integer size of each leaf ($1 \leq a_i \leq 10^6$)

Output Format

Minimum total cost

Sample Input1

4

5 3 8 2

Sample Output1

33

Sample Input2

3

10 20 30

Sample Output2

90

Explanation

Merge 10 & 20 $\rightarrow$ cost = 30 $\rightarrow$ new list [30, 30]

Merge 30 & 30 $\rightarrow$ cost = 60

Total = 30 + 60 = 90

Task 14: (Level: Hard)

Real-Time Stock Price Tracker & Analyzer

Scenario: A fintech startup needs to monitor live stock price streams, track top movers and send alerts.

Internal tasks:

1. Use an array or list to store incoming price updates and apply sorting (e.g., merge sort or quick sort) for top gainers/losers.
2. Use a min-heap or max-heap priority queue to keep track of top N stocks efficiently as data updates.
3. Maintain a BST for quick look-up of any given stock's history, and provide find-min/find-max operations.

Task 15: (Level: Hard)

Chatbot with Real-Time Response Analysis

Scenario: A customer-service chatbot that processes incoming messages and provides responses, tracking priority issues.

Internal tasks:

1. Use queue (FIFO) to process incoming user messages; use stack for rollback or to manage recent conversations.
2. Use hash map to map keywords $\rightarrow$ responses; implement search and update operations.
3. Use priority queue to elevate urgent queries (e.g., containing “urgent”, “error”) and ensure they are handled first.

Task 16: (Level: Hard)

Given a string A, we may represent it as a binary tree by partitioning it to two non-empty substrings recursively. **Given two strings A and B of the same length, determine if B is a scrambled string of S.**

Below is one possible representation of A = “great”:

```

great
 /  \
gr   eat
 /\  /\
g r e at
   /\
   a t

```

To scramble the string, we may choose any non-leaf node and swap its two children.

For example, if we choose the node “gr” and swap its two children, it produces a scrambled string “rgeat”.

```
rgeat
```

```

    /  \
   rg  eat
  /\   /\
 r g e at
    /\
    a t

```

We say that “rgeat” is a scrambled string of “great”.

Similarly, if we continue to swap the children of nodes “eat” and “at”, it produces a scrambled string “rgtae”.

```

    rgtae
   /\   /\
  rg  tae
 /\   /\
r g ta e
    /\
    t a

```

We say that “rgtae” is a scrambled string of “great”.

Input Format:

The first argument of input contains a string A.

The second argument of input contains a string B.

Output Format:

Return an integer, 0 or 1:

=> 0 : False

=> 1 : True

Constraints:

$1 \leq \text{len}(A), \text{len}(B) \leq 50$

Sample Input 1:

A = " rgtae "

B = "great"

Sample Output 1:

1

Sample Input 2:

A = "hello"

B = "hel"

Sample Output 2:

0

Task 17: (Level: Hard)

Online Voting / Polling System

Scenario: A website runs polls or elections and wants to manage votes, handle unique voters, show results.

Internal tasks:

1. Use HashSet to track unique voters (prevent duplicates); use HashMap to map candidates → vote count.
2. Use quick/merge sort to sort candidates by vote count to display top winners.
3. Use a priority queue to track upcoming or most-popular polls; manage insertion/deletion of polls dynamically.

Task 18: (Level: Hard)**Minimum Cost to Connect All Cities**

You are given N cities (numbered 1 to N) and M bidirectional roads.

Each road connects two cities u and v and has a construction cost w.

Your task is to find the minimum total cost required to connect all cities — that is, every city should be reachable from every other city. If it's not possible to connect all cities, print -1. This problem represents finding the Minimum Spanning Tree (MST) of a graph.

Explanation:

- We can use Kruskal's Algorithm with Disjoint Set Union (DSU) to efficiently find the MST.
- Sort all edges by their cost.
- Use DSU to add edges one by one (avoid cycles).
- Keep adding edges until all cities are connected.

Input Format

N M

u1 v1 w1

u2 v2 w2

...

uM vM wM

N → number of cities ($1 \leq N \leq 10^5$)

M → number of roads ($1 \leq M \leq 2 \times 10^5$)

ui, vi → cities connected by the i-th road

wi → cost of that road

Output Format

Minimum total cost to connect all cities. If it's not possible to connect all, print -1.

Sample Input

4 5

1 2 5

1 3 10

2 3 4

2 4 11

3 4 2

Sample Output

11

Explanation

The edges in the MST are:

(3–4) → 2

(2–3) → 4

(1–2) → 5

Total cost = 2 + 4 + 5 = 11

Task 19: (Level: Easy)

Write a method to print the characters which occur a minimum number of times in the array. In case of a dilemma, print the char which is alphabetically earlier.

Consider the input as “ABCC”.

The output is ‘A’ as it occurs a minimum number of times.

Please note that ‘A’ and ‘B’ occur the same number of times but ‘A’ is earlier than ‘B’.

Sample Test Cases:**Sample Input:**

ABCC

Sample Output:

A

Part - B**Task 20: (Level: Hard)****Social Media Friend Recommendation Engine**

Scenario: A social network needs to suggest new friend connections to users.

Internal tasks:

1. Model users and friendships as a graph (adjacency matrix or adjacency list).
2. Use BFS/DFS to compute “friends-of-friends” and mutual connections.
3. Use a priority queue (heap) to rank suggestions by strength of connection (mutual friend count) and display top K.

Task 21: (Level: Hard)**Campus Navigation & Way-finding System**

Scenario: A university wants an app that helps students navigate between buildings, labs, facilities.

Internal tasks:

1. Model the campus as a graph (nodes = buildings/rooms, edges = paths) using adjacency matrix/list.
2. Implement BFS/DFS to explore reachable locations from a given start point; compute shortest unweighted path.
3. Use Java Collections (List, Set, and Map) to maintain visited nodes, path history, and map node IDs to building names; compare performance.

Task 22: (Level: Hard)**Travel Planner / Route Finder**

Scenario: A travel-app that finds routes between cities, suggests trip plans.

Internal tasks:

1. Represent cities and routes as a graph; store routes (edges) and cities (nodes).
2. Use BFS/DFS to find reachable cities; maybe implement shortest path in unweighted graphs.
3. Sort possible routes by number of stops or cost using sorting algorithms and present top options.

Task 23: (Level: Hard)**Contact Management & Search System**

Scenario: A company wants a digital phonebook/contact system with fast search and sorting.

Internal tasks:

- Use array or linked list to store contacts; implement CRUD operations (add, delete, update, and retrieve).
- Sort contacts by name or last update using insertion/selection/quick sort; implement search via binary and linear search.
- Enhance with a hash map (HashMap) for lookup by phone number; analyze the time complexity differences between list vs. hash-map lookup.

Task 24: (Level: Hard)**Maze / Labyrinth Solver**

Scenario: A game or robotics simulation navigates a maze from start to exit.

Internal tasks:

- Model the maze as a 2D-array (matrix) where walls = blocked cells, open cells = nodes.
- Use BFS/DFS to find a path from start to finish (or all possible paths).
- Use backtracking to explore and mark dead-ends, count the number of unique paths; compare stack vs. recursion implementations.

Task 25: (Level: Hard)**Expense Tracker / Budget Management**

Scenario: A personal finance app where users log expenses, sort and filter them, get analytics.

Internal tasks:

1. Use array/list to store expense entries (date, category, and amount).
2. Sort expenses by date or amount using quick sort/merge sort; implement search for entries by category/date.
3. Use HashMap category → total amount;
use tree structure (TreeMap) to maintain sorted categories; present top expense categories via priority queue.

Task 26:(Level: Hard)**File / Data De-duplication Tool**

Scenario: A cloud storage provider wants to detect duplicate files and save space by de-duplication.

Internal tasks:

1. Use hash map (file-hash $\rightarrow$ list of files) to detect duplicates; support insert and delete.
2. Use sorting algorithms to sort files by size or last-modified date for efficient checking.
3. Use tree or priority queue to list largest duplicates removed, track space saved; implement find-max and delete operations.

Task 27:(Level: Hard)

Multiplayer Online Game Leader board & Event System

Scenario: An online game maintains player scores, sorts leader boards, and handles events and matchmaking.

Internal tasks:

- Use array/list for players and their scores; implement quick sort/merge sort for leader board.
- Use priority queue to manage event scheduling or matchmaking, where top players get priority in events.
- Use graph structure for matchmaking (players = nodes, edges = connections or past matches); use BFS/DFS to find potential match clusters or teams.

Task 28 : (Level: Easy)

An automobile company manufactures both a two wheeler (TW) and a four wheeler (FW). A company manager wants to make the production of both types of vehicle according to the given data below:

- 1st data , Total number of vehicle (two – wheeler + four - wheeler)=v
- 2nd data, Total number of wheels=w

The task is to find how many two-wheelers as well as four-wheelers need to manufacture as per the given data. Print “INVALID INPUT”, if inputs did not meet the constraints.

Constraints:

- $2 \leq W$
- $W \% 2 = 0$
- $V < W$

Sample Test Cases:

Sample Input:

200 540

Sample Output: 130 70

Task 29: (Level: Easy)

Given weights and values of n items, put these items in a knapsack of capacity W to get the maximum total value in the knapsack.

Input: An integer n , the number of items. Two lists: weights and values. An integer W , the maximum capacity of the knapsack.

Output: Print the maximum total value. **Constraints:** $1 \leq n \leq 1000$ $1 \leq W \leq 10^5$

Approach/Explanation: Use dynamic programming to solve the 0/1 knapsack problem.

Sample Test Case:**Sample Input:**

$n = 3$

values = [60, 100, 120]

weights = [10, 20, 30]

$W = 50$

Sample Output:

220

Task 30: (Level: Hard)**Data Compression Tool using Huffman Coding**

Scenario: A software firm needs to compress text files using efficient encoding.

Internal tasks:

1. Given character frequency count, build Huffman tree via greedy algorithm using priority queue (heap).
2. Traverse the Huffman tree (preorder/inorder/postorder) to assign codes to characters.
3. Implement encoding and decoding of text; compare size of encoded vs. original; analyze time/space complexity.

TOTAL: 30 Hours

I. Learning Resources**i. Text Books:**

1. M. A. Weiss, Data Structures and Algorithm Analysis in Java, 3rd ed., Pearson Education, New Delhi, India, ISBN 978-8131760635.
2. Dan S. Myers, Data Structures and Algorithms in Java: A Project-Based Approach, 1st Edition, Cengage Learning, 2024
3. Michael T. Goodrich, Roberto Tamassia, Michael H. Goldwasser, Data Structures & Algorithms in Java, 6th Edition, Wiley India, 2023 (Indian Edition).
4. Oswald Campesato, Data Structures & Algorithms in Java, 2nd Edition, Packt Publishing, 2022.

5. Narasimha Karumanchi , Data Structures & Algorithms Made Easy in Java, CareerMonk Publications, 2023.
6. M. T. Goodrich, R. Tamassia, M. H. Goldwasser, and S. Banerjee, Data Structures and Algorithms in Java, 6th ed., Indian Adaptation, Wiley India Pvt. Ltd., 2022, ISBN 978-9354247934.

ii. Reference Books:

1. Y. Daniel Liang, Introduction to Java Programming and Data Structures, Comprehensive Version, 12th Edition, Pearson Education, 2023.
2. S. K. Srivastava & Deepali Srivastava, Data Structures and Algorithms in Java (Comprehensive), BPB Publications, 2026.

iii. Online Resources:

1. “Competitive Programming - A Complete Guide”, [Online] Last Updated: 22.08.2025. Available: <https://www.geeksforgeeks.org/dsa/competitive-programming-a-complete-guide/>
2. w3schools.com, Explore Java programming language, [Online] Last Updated: 22.08.2025. Available: [w3schools.com](https://www.w3schools.com/)
3. Takeuforward.org Assessed on: Dec. 20, 2025 [Online], Available: <https://www.takeuforward.org>
4. Leetcode.com Assessed on: Dec. 20, 2025 [Online], Available: <https://www.leetcode.com>
5. Codechef.com Assessed on: Dec. 20, 2025 [Online], Available: <https://www.codechef.com>

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS293	Microprocessors and ARM Microcontrollers	2	0	2	3

A.Preamble

Microprocessors are a core course in the stream of Computer Science & Engineering. The objective of this course is to explore the concepts of hardware and software part of a microprocessor. It includes the architecture of 8086 Microprocessor and latest ARM processor architecture. This course covers ARM processor programming with interfacing techniques and equivalent assembly language code for high level language instructions like 'C'.

B.Prerequisite Course

10211CS201–Digital Electronics

C.Course Objective

Learners are exposed to:

- Explain the architecture and operation of the Intel 8086 microprocessor.
- Develop simple assembly language programs using the instruction set and addressing modes of the Intel 8086.
- Describe the architecture and features of the ARM Cortex-M processor.
- Write assembly programs using ARM instruction sets and addressing modes.
- Implement basic peripheral interfacing using GPIO, timers, and communication interfaces with the TM4C123 Microcontroller.

D.Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Know the architecture of 8086 microprocessor and its basic configuration.	K2
CO2	Develop assembly language programs for the given problems using 8086 processor.	K3
CO3	Understand ARM processor architecture and the principle of its debug system.	K2
CO4	Develop programs for the problems specified using ARM processor instructions.	K3
CO5	Establish interfacing of I / O devices with ARM processor to develop real time applications.	K3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of COs with Program outcomes and Programme Specific Outcomes:

Cos	P O1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO1 0	PO1 1	PSO1	PSO 2	PSO 3
CO1	3	3												
CO2	3	2			3							2		
CO3	3	2												
CO4	3	2			3							3		
CO5	3		3		3							3		

3- High; 2-Medium; 1-Low

F. Course Contents**Unit 1 Microprocessor Architecture – 8086****L-6Hours**

Introduction – Types of computers – Evolution of Microprocessor – 8086 Microprocessor architecture – 8086 Programming method — program development tools – Interrupts and interrupt service procedures - 8086 signals – Basic configurations – System connections – Timing – Troubleshooting – Assembler directives - Introduction to OpenRISC Architecture.

Unit 2 Programming In 8086**L-6Hours**

Assembly language programming – Addressing modes – Instruction set: Data transfer instructions – arithmetic and logical instructions – Branch instructions — Loop instructions – While – Do programs – Strings – Procedures – Macros.

Unit 3 Arm Processor Architecture**L-6Hours**

Cortex – M architecture – Block diagram – pipelined architecture – memory address map – bus system and bus matrix – memory and peripherals: memory endianness – bit banding – system stack architecture – debug system: AHB access port, FPB, DWT, ITM, ETM, TPIU – conditional execution – interrupts: exception and interrupt handling.

Unit 4 Arm Processor Programming**L-6Hours**

Addressing modes – ARM instruction sets: thumb instruction set – ADD and MOV data processing instructions – LDR and STR memory access instructions – data processing instructions – bitfield instructions – branch and control instructions – recursive functions If-Then conditional instructions – reset sequence – instruction encoding.

Unit 5 Arm Processor Interfacing**L-6Hours**

TM4C123 microcontroller block diagram – GPIO features – peripherals on the memory map – GPIO pin configuration – LED and switch interfacing – keypad interfacing – I/O synchronization – polling – interrupt driven – processor interrupt configuration – interrupt based interfacing – Timer: timer as I/O device – external event counter – input capture device – general purpose timer modules – serial communication interface – I²C interface – SPI interface – CAN interface.

Total: 30Hours (L)

G. Laboratory Tasks**Part – 1****Task 1: 16bit arithmetic and logical operations using 8086 instruction set.**

Construct assembly language programs for addition, subtraction, multiplication and division.

Task 2: String operations using 8086 instruction set

Develop programs for string concatenation and conversion of uppercase letters into lowercase letters and vice versa.

Task 3: Smallest and largest number using 8086 instruction set

Realize a program that is able to identify the smallest and largest number from the given set of data.

Task 4: Ascending and descending order using 8086 instruction set

Write assembly language program to arrange the given number from smallest to largest and vice versa.

Task 5: Data transfer using ARM processor

Employ an assembly language program which is able to transfer data from ARM processor register to memory or memory to ARM processor.

Task 6: Count of negative numbers in an array using ARM processor instructions

Formulate a logic to identify the list of negative numbers from the given numbers and develop the program using ARM processor instruction set.

Task 7: Factorial of positive integer N using ARM processor instructions

Calculate the factorial of a given numbers using ARM processor instructions.

Task 8: GCD algorithm using ARM processor instructions

Identify the greatest common divisor of given numbers by executing program developed using ARM processor instructions.

Task 9: Seven segment LED display

Control the display of an LED through an ARM processor.

Task 10: Key board and display interfacing using ARM processor kit development board

Realize the display of the information typed using a keyboard and processed by an ARM processor.

Task 11: UART device driver for data transmission

Experiment the process of data transfer that can be established between two ARM processor boards.

Task 12: Temperature measurement using ARM processor kit development board

Construct a data acquisition mechanism of any physical parameter like temperature using an ADC which is in-built in an ARM processor.

Part-2**Use Cases:****Use Case 1:**

Develop an application which counts the number of customers entering into a supermarket for purchasing using 8086 microprocessor. IR sensor shall be used to identify the customer entering into the supermarket. 8086 Processor recognizes the output of IR sensor as a digital input and a suitable ALP is able to initiate the count as and when it receives a logic '1'.

Use Case 2 :

Implement a scheme to measure the distance between two cars that helps to avoid collision due to over speed. GP2D120 sensor can be used to measure the distance between two objects. Output of the sensor is analog signal. A/D converter in an ARM CORTEX processor recognizes this output as an input to it. Suitable Keil 'C' coding implemented in the ARM processor converts the analog signal into digital and displays the result through LCD display.

Use Case 3 :

Interpret the given digital signal, develop a Keil 'C' program to implement in an ARM CORTEX processor for identifying the frequency of the signal and display its value in an LCD.

Use Case 4 :

Design a Keil 'C' program to generate a PWM signal based on the input value given by the user. The program must be implemented in an ARM CORTEX processor and the output shall be displayed by changing the ON / OFF condition of an LED. Timer in the ARM processor can change the width of the pulse depending the input applied to it. Frequency of the timer is the frequency of the ARM processor.

Total: 30 Hours

Total: 60 Hours

G.Learning Resources**i. Text Books:**

1. Douglas V. Hall, "Microprocessors and Interfacing", TMH, Revised second edition, 2005. [Unit 1-2]

2. Muhammad Tahir, Kashif Javed, “ARM Microprocessor Systems: Cortex-M Architecture, Programming, and Interfacing”, CRC Press, Taylor & Francis group, 2017. [Unit 3-5]

ii. Reference Books:

1. Andrew N. Sloss, “ARM system developer’s guide – Designing and optimizing system software”, ELSEVIER, 2004.
2. Charles M. Gilmore, “Microprocessors: Principles and applications”, TMH, Second edition, 1995.
3. Jonathan W. Valvano, “Embedded microcomputer systems: Real time interfacing”, CENGAGE Learning, Third edition, 2012.
4. Steve Furber, “ARM system-on-chip architecture”, Pearson education, Second edition, 2015.

iii. Online References:

1. “Microprocessor interfacing”, Accessed on Sep 20, 2022 [Online], Available: https://onlinecourses.nptel.ac.in/noc20_ee11/preview.
2. “Microprocessors and Digital systems”, Accessed on Sep 21, 2022 [Online], Available: <https://archive.org/details/microprocessorsd0000hall/page/n3/mode/2up>.
3. “ARM processor architecture”, Accessed on Sep 21, 2022 [Online], Available: https://www.cs.ccu.edu.tw/~pahsiung/courses/ese/notes/ESD_03_ARM_Architecture.pdf.

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS234	TIME SERIES AND FORECASTING	3	0	2	4

A. Preamble

The course will provide a detailed knowledge in time series analysis. The topics include exploratory data analysis, storage of temporal data, ARMA/ARIMA models and some advanced models, Machine learning and Deep learning techniques to classify and cluster time-series data.

B. Prerequisite Course

10210MA103- Probability, Statistics and Queuing theory

C. Course Objectives

Learners are exposed to:

- Understand and analyse the fundamental components of time series data, including trend, seasonality, smoothing techniques, and transformations, to prepare data for effective modelling and forecasting.
- Explore, simulate, and manage time series data using statistical tests, correlation analysis, simulation techniques, and appropriate temporal data storage systems for reliable analysis and experimentation.
- Develop and apply stationary and non-stationary time series models such as ARMA, ARIMA, SARIMA, state-space models, and exponential smoothing methods to model and forecast temporal data.
- Design, evaluate, and validate machine learning and deep learning–based time series models, and assess their forecasting performance using appropriate error metrics, backtesting strategies, and real-world case studies.

D. Course Outcomes

Upon the successful completion of the course, students will be able to:

Cos No	Course Outcomes	K Level
CO1	Utilize time series principles to process temporal data involving trend, seasonality, and noise components.	K3
CO2	Implement data sourcing, cleansing, transformation, and encoding procedures to prepare time series datasets for exploratory studies.	K3
CO3	Execute classical forecasting techniques and stationarity tests on both stationary and non-stationary time series.	K3
CO4	Employ machine learning and deep learning approaches to derive features and handle classification or clustering of time-dependent data	K3

CO5	Apply validation, backtesting, and error measurement methods to verify forecasting performance in practical applications.	K3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of COs with Program Outcomes and Programme Specific Outcomes

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	1	-	2	-	-	-	-	-	-	3	1	-
CO2	3	2	2	1	2	-	-	-	-	-	-	2	2	-
CO3	3	3	2	2	2	-	-	-	-	-	-	3	1	-
CO4	2	2	3	-	3	-	-	-	-	-	-	2	2	1
CO5	2	2	2	2	2	-	-	-	-	-	-	2	1	2

3- High; 2-Medium; 1-Low

F. Course Contents

Unit I: Introduction to Time Series Analysis

L - 9 Hours

Examples and Objectives of Time Series Analysis – Time series models: zero-mean model – trends – seasonality – harmonic regression – Smoothing with a finite moving average filter – Exponential smoothing – Smoothing by elimination of high-frequency components – Trend elimination by differencing.

Unit II: Exploratory Data Analysis, Simulation, and Storing

L - 9 Hours

Data Sourcing- Data Cleaning – Handling Missing values- Outlier Treatment- Types of Analysis (Univariate, Bivariate, Multivariate)- Numerical Analysis- Feature Binning – Feature Encoding- Differencing techniques.

Unit III: Stationary and Non-Stationary Time Series Model

L – 9 Hours

Auto-Correlation Function (ACF) & Partial Auto-Correlation Function (PACF)- ARMA model – ARIMA – SARIMA – Linear Trend Models -Quadratic Trend Models- Logarithmic Trend Models (Log Linear, Log Quadratic)- Dickey–Fuller test- InfluxDB database- Autoregressive models and Moving Average models- the Holt-Winters Seasonal algorithm.

Unit IV: Machine Learning and Deep Learning for Time Series Data

L- 9 Hours

Generating and Selecting Features for a Time Series, General Considerations When Computing Features: The Nature of the Time Series - Domain Knowledge - External Considerations – Selecting and Generating Features for Classification - Gradient boosted trees - Generating Clustering Features from the Data - Temporally Aware Distance Metrics for Clustering: Fréchet distance - Pearson correlation - Longest common subsequence - Deep Learning for Time Series: CNN, RNN.

Unit V: Performance Analysis**L- 9 Hours**

Measuring Error: How to Test Forecasts - Model-Specific Considerations for Back testing - Estimating Uncertainty - Predicting Multiple Steps Ahead - Recursive Approach to Distant Temporal Horizons - Multitask Learning Applied to Time Series - Model Validation Gotchas –

Performance Considerations in Fitting and Serving Time Series Models – Time series Applications and Case studies.

TOTAL: 45 Hours**G. Laboratory Experiments****TOTAL: 30 Hours****Part A**

- TASK 1** The time series data set has some fluctuations/aberrations. Apply exponential smoothing technique to the data set.
Tools: python
- TASK 2** Consider the data set given to you. It has some trend and seasonal components. Apply filter to this time series data and discuss the results.
Tools: python
- TASK 3** The given time series data set has trend components. Remove the trend using differencing technique.
Tools: python
- TASK 4** Check if a particular time-series data is stationary or not. Apply Augmented Dickey–Fuller test using python language. Apply rolling windows for the given time-series data using python language.
Tools: python
- TASK 5** Plot the autocorrelation function for the given set of time-series data using python language. Use partial autocorrelation for the given data as well.
Tools: python
- TASK 6** Create a simulation of the time-series data using python language. Also, read and write time-series data to InfluxDB database.
Tools: python
- TASK 7** Implement Autoregressive models and Moving Average models to the given time-series data
Tools: python
- TASK 8** Compare the results produced by Autoregressive Integrated Moving Average model and Vector Autoregression model.
Tools: python
- TASK 9** Write a program that uses the Durbin-Levinson algorithm to do time series analysis.
Tools: python
- TASK 10** Consider the time series data set given to you. Do stationary analysis using the innovations algorithm.
Tools: python
- TASK 11** For the given data set, choose which model is suitable to use: ARMA, ARIMA, SARIMA.

Tools: python

- TASK 12** Do multivariate time series analytics using ARAR algorithm for the given data set. **Tools: python**
- TASK 13** Apply both the Holt-Winters algorithm and the Holt-Winters Seasonal algorithm to the datasets given to you.
Tools: python
- TASK 14** Check the performance of different distance metrics on clustering of time-series data **Tools: python**
- TASK 15** Apply feed forward networks, CNN, RNN in deep learning techniques to the given time-series data. Compare its performance.

Tools: python**Part B****Use Cases:**

- 1) Periodicity analysis of LH release in isolated pituitary gland by hidden frequency analysis
- 2) On the forecasting of freight transportation by time series analysis.
- 3) The water flow prediction in Ganga River (or any other river).
- 4) Forecast weekly air passenger traffic in major domestic airlines.
- 5) Individual household electric power consumption.

TOTAL: 75 Hours**H. Learning Resources****i. Text Books:**

1. Aileen Nielsen, “Practical Time Series Analysis - Prediction with Statistics and Machine Learning”, O’Reilly publications, First Edition, 2019. (UNIT II, IV, V)
2. Peter J. Brockwell, Richard A. Davis, “*Introduction to Time Series and Forecasting*”, Second Edition, Springer, 2016 (UNIT I, III)

ii. Reference Books:

1. Robert H. Shumway, David S. Stoffer, “*Time Series Analysis and Its Applications - With R Examples*”, Fourth Edition, Springer, 2016
2. William.W.S.Wei, “*Time Series Analysis – Univariate and Multivariate Methods*”, Second Edition, Pearson, 2006
3. James D. Hamilton, “*Time Series Analysis*”, Princeton University Press, First Edition, 1994

iii. Online Resources:

1. “Pandas python library”, pandas.pydata.org, 2021 [Online]
2. “Scikit-learn ML library”, scikit-learn.org/stable, 2021 [Online]

COURSE CODE	COURSE TITLE	L	T	P	C
10212CS233	SOCIAL MEDIA ANALYTICS	3	0	2	4

A. Preamble

This course Social Media Analytics focuses on extracting meaningful insights from large-scale social media data. The course introduces social network structures, text preprocessing, and data modeling techniques. Students learn topic modeling, recommendation systems, and predictive analytics using machine learning. Emphasis is given to practical implementation with real-world social media datasets. The course prepares students to solve real-time data-driven problems in diverse application domains.

B. Prerequisite Course

10210MA103- Probability, Statistics and Queuing theory

C. Course Objectives

Learners are exposed to:

- Understand the basic concepts of social networks and social media analytics.
- Learn methods for collecting and preprocessing social media data.
- Study topic modeling techniques for analyzing textual social media content.
- Understand recommendation system approaches used in social media platforms.
- Gain exposure to prediction and evaluation techniques using real-world social media data.

D. Course Outcomes

Upon the successful completion of the course, students will be able to:

COs	Course Outcomes	K Level
CO1	Utilize graph-theoretic concepts to represent and interpret social network structures and interaction patterns.	K3
CO2	Implement text and activity-based preprocessing techniques to extract meaningful features from social media data.	K3
CO3	Execute topic modeling approaches to derive latent thematic structures from high-dimensional textual content	K3
CO4	Employ recommendation system techniques to model user–item interactions and generate personalized suggestions	K3

CO5	Perform prediction and evaluation methods to estimate movie ratings using collaborative and supervised learning approaches.	K3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of COs with Program Outcomes and Programme Specific Outcomes

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO 1	PSO2	PSO 3
CO1	3	2			2							3		
CO2	2	3			3							2	3	
CO3	3	2		1	2							3	1	
CO4	2	2	3		3						1	2	3	1
CO5	1	3		2	3					1			2	3

3- High; 2-Medium; 1-Low

F. Course Contents

UNIT 1 Introduction to Social Media Analytics

L-9 Hours

Aspects of Networks – Graph Theory - Triadic Closure - The Strength of Weak Ties - Closure, Structural Holes, and Social Capital - Betweenness Measures and Graph Partitioning - The Girvan-Newman Method - Computing Betweenness Values – Homophily - Mechanisms Underlying Homophily: Selection and Social Influence

UNIT 2 Social Media Data Pre-Processing

L-9 Hours

Defining Content: Focus on Text and Unstructured Data - Creating features from text: The basics of natural language processing - The basic statistics of term occurrences in text - Using Content Features to Identify Topics – Inter-Event Times (IET) – IET with memory lessness – Deviations from memory lessness - Periodicities/Burstiness in User Activities

UNIT 3 Topic Modeling

L-9 Hours

The popularity of topics - How diverse are individual user's interests? - Extracting Low-Dimensional Information from High-Dimensional Text - Topic Modeling - Unsupervised Topic Modeling - Supervised Topic Modeling - Relational Topic Modeling

UNIT 4 Social Media Recommendations

L-9 Hours

Online Social Media Services: Search Engines - Content Engagement - Interactions with the real world - Interactions with People; Problem Formulation - Learning and Mapping - Matrix Factorization - Learning, Training - Under-and Overfitting - Regularizing in Matrix Factorization - Non-Negative Matrix Factorization and Sparsity

UNIT 5 Movie Ratings Prediction

L-9 Hours

Demonstration of movie ratings - Interpreting the Learned Stereotypes - Exploratory Analysis - Prediction - Evaluation - Overview of methodologies - Nearest neighbor-based approaches, Approaches based on supervised learning - Predict movie ratings with logistic regression

Total : 45 Hours

G. Laboratory Experiments

Total : 30 Hours

Part A

TASK 1	Use Corpus Readers to retrieve social media data from the Internet to a plain-text file and/or HTML file and/or a XML file. Tools: python
TASK 2	Read and pre-process social media data to change unstructured (or semi-structured) data to its structured form. Initially understand the attributes in the social media data, then apply techniques like removing stop words, stemming, etc. Tools: python
TASK 3	Create distance matrix for posts using only the stemmed terms (previous Task) that appear in their text bodies. Run the hierarchical agglomerative clustering model on the post dissimilarities. Tools: python
TASK4	Plot the dendrogram of cluster merges. Show how to interactively select subtrees of the dendrogram and create word clouds for them. Tools: python
TASK5	Run hierarchical agglomerative clustering algorithm to work with the dendrogram of cluster merges to determine what kinds of clusters we had at some point. Tools: python
TASK6	Fit Latent Dirichlet Allocation (LDA) model to the corpus data and inspect the most important terms. Spot-checking the model results for 10 randomly chosen documents. Tools: python
TASK7	Train the LDA model on the political blog dataset. Link the topics to the response variable. Tools: python
TASK8	The data set given corresponds to purchasing history or watching movies or any other social media data. Understand the attributes and the contents in the data set. Read the data set and provide recommendations to the user. Evaluate your model using RMSE and recall metrics. Tools: python
TASK9	Predict the target class in the social media data using logistic regression and analyse its performance using different metrics. Tools: python

- TASK10** Understand the personal social media API (like Twitter API, Facebook API, or any other website). Analyze personal social network using several graph techniques (friend graph network, node properties, network communities, and so on). Experiment on how recommendation system is executed in these social medias.
Tools: python

PART – B

Use Cases:

- 1) Marketing strategy of Facebook website
- 2) Tie Strength on Twitter and Trend of topics in Twitter
- 3) Trend/Sentiment analysis and recommendation systems on Foursquare application.
- 4) Netflix (or Amazon Prime) Recommendation Systems
- 5) IMDb Rating Prediction

Total : 75 Hours

H. Learning Resources

i. Text Books:

1. David Easley and Jon Kleinberg, “Networks, Crowds and Markets: Reasoning about a Highly Connected World”, Cambridge University Press, 2010 (Unit I)
2. Gabor Szabo, Gungor Polatkan, P. Oscar Boykin and Antonios Chalkiopoulos, “Social Media Data Mining and Analytics”, Wiley, First Edition, 2018 (Unit II, III, IV, V)

iii. Reference Books:

1. Raghav Bali, Dipanjan Sarkar and Tushar Sharma, “*Learning Social Media Analytics with R*”, Packt Publishing, First Edition, 2017
2. Alex Goncalves, “*Social Media Analytics Strategy*”, Apress, First Edition, 2017

iii. Online Resources:

1. “NLTK documentation”, 2021 [Online] www.nltk.org,
2. “Scikit-learn ML library”, 2021 [Online] scikit-learn.org/stable

PROGRAM ELECTIVE

(Self Learning Courses)

Academic Year: 2024-2025
Semester : Summer

COURSE CODE	COURSE TITLE	W	H	C
10212CS401	SOCIAL NETWORKS	12	-	3

CourseCategory:

Independent Learning (Self - Learning Course)

Course Contents

Week 1: Introduction

Week 2: Handling Real-world Network Datasets

Week 3: Strength of Weak Ties

Week 4: Strong and Weak Relationships (Continued) & Homophily

Week 5: Homophily Continued and +Ve / -Ve Relationships

Week 6: Link Analysis

Week 7: Cascading Behaviour in Networks

Week 8: Link Analysis (Continued)

Week 9: Power Laws and Rich-Get-Richer Phenomena

Week 10: Power law (contd..) and Epidemics

Week 11: Small World Phenomenon

Week 12: Pseudocore (How to go viral on web)

CourseProvider :	NPTEL
Professor :	Prof. Sudarshan Iyengar
University :	IIT Ropar

Academic Year: 2024-2025
Semester : Summer

COURSE CODE	COURSE TITLE	W	H	C
10212CS402	DESIGN & IMPLEMENTATION OF HUMAN-COMPUTER INTERFACES	12	-	3

CourseCategory:

Independent Learning (Self - Learning Course)

Course Contents

Week 1: Introduction

Week 2: Identification of usability requirements I

Week 3: Identification of usability requirements II

Week 4: Usable interface design

Week 5: Rapid usability evaluation

Week 6: Converting design to system I

Week 7: Converting design to system II

Week 8: System implementation I

Week 9: System implementation II

Week 10: System implementation III

Week 11: Empirical usability evaluation

Week 12: Conclusion

CourseProvider : NPTEL

Professor : Prof. Samit Bhattacharya, IIT
Guwahati

Academic Year: 2024-2025
Semester : Summer

COURSE CODE	COURSE TITLE	W	H	C
10212CS403	INTRODUCTION TO INDUSTRY 4.0 AND INDUSTRIAL INTERNET OF THINGS	12	-	3

CourseCategory:

Independent Learning (Self - Learning Course)

Course Contents

Week 1 : Introduction: Sensing & actuation, Communication-Part I, Part II, Networking-Part I, Part II

Week 2 : Industry 4.0: Globalization and Emerging Issues, The Fourth Revolution, LEAN Production Systems, Smart and Connected Business Perspective, Smart Factories

Week 3 : Industry 4.0: Cyber Physical Systems and Next Generation Sensors, Collaborative Platform and Product Lifecycle Management, Augmented Reality and Virtual Reality, Artificial Intelligence, Big Data and Advanced Analysis

Week 4 : Cybersecurity in Industry 4.0, Basics of Industrial IoT: Industrial Processes-Part I, Part II, Industrial Sensing & Actuation, Industrial Internet Systems.

Week 5 : IIoT-Introduction, Industrial IoT: Business Model and Reference Architecture: IIoT-Business Models-Part I, Part II, IIoT Reference Architecture-Part I, Part II.

Week 6 : Industrial IoT- Layers: IIoT Sensing-Part I, Part II, IIoT Processing-Part I, Part II, IIoT Communication-Part I.

Week 7 : Industrial IoT- Layers: IIoT Communication-Part II, Part III, IIoT Networking- Part I, Part II, Part III.

Week 8 : Industrial IoT: Big Data Analytics and Software Defined Networks: IIoT

Analytics - Introduction, Machine Learning and Data Science - Part I, Part II, R and Julia Programming, Data Management with Hadoop.

Week 9 : Industrial IoT: Big Data Analytics and Software Defined Networks: SDN in IIoT-Part I, Part II, Data Center Networks, Industrial IoT: Security and Fog Computing: Cloud Computing in IIoT-Part I, Part II.

Week 10 : Industrial IoT: Security and Fog Computing - Fog Computing in IIoT, Security in IIoT-Part I, Part II, Industrial IoT- Application Domains: Factories and Assembly Line, Food Industry.

Week 11 : Industrial IoT- Application Domains: Healthcare, Power Plants, Inventory Management & Quality Control, Plant Safety and Security (Including AR and VR safety applications), Facility Management.

Week 12 : Industrial IoT- Application Domains: Oil, chemical and pharmaceutical industry, Applications of UAVs in Industries, Real case studies :

Case study - I : Milk Processing and Packaging Industries

Case study - II: Manufacturing Industries - Part I

Case study - III : Manufacturing Industries - Part II

Case study - IV : Student Projects - Part I

Case study - V : Student Projects - Part II

Case study - VI : Virtual Reality Lab

Case study - VII : Steel Technology Lab

CourseProvider : NPTEL

Professor : Prof Sudip Misra

University : IIT Kharagpur

Academic Year: 2024-2025

Semester : Summer

COURSE CODE	COURSE TITLE	W	H	C
10212CS404	REINFORCEMENT LEARNING	12	-	3

CourseCategory:

Independent Learning (Self - Learning Course)

Course Contents

Week 1 : Introduction

Week 2 : Bandit algorithms – UCB, PAC

Week 3 : Bandit algorithms –Median Elimination, Policy Gradient

Week 4 : Full RL & MDPs

Week 5 : Bellman Optimality

Week 6 : Dynamic Programming & TD Methods

Week 7 : Eligibility Traces

Week 8 : Function Approximation

Week 9 : Least Squares Methods

Week 10 : Fitted Q, DQN & Policy Gradient for Full RL

Week 11 : Hierarchical RL

Week 12 : POMDPs

CourseProvider : NPTEL

Professor : Prof. Balaraman Ravindran

University : IIT Madras

Academic Year: 2024-2025
Semester : Summer

COURSE CODE	COURSE TITLE	W	H	C
10212CS405	RESPONSIBLE & SAFE AI SYSTEMS	12	-	3

CourseCategory:

Independent Learning (Self - Learning Course)

Course Contents

Week 1 & 2:

- AI Capabilities Improvement in last 5-10 years
- Imminent risks from AI Models: Toxicity, bias, goal misspecification, adversarial examples etc.
- Long-term risks from AI Models: Misuse, Misgeneralization, Rogue AGI
- Principles of RAI - Transparency; Accountability; Safety, Robustness and Reliability; Privacy and Security; Fairness and non-discrimination; Human-Centred Values; Inclusive and Sustainable development, Interpretability
- Recap of Deep Learning Techniques, Language/Vision Models
- AI Risks for Gen models
- Adversarial Attacks – Vision, NLP, Superhuman Go agents

Week 3 & 4:

- ML Poisoning Attacks like Trojans
- Implications for current and future AI safety
- Explainability
- Imminent and Long-term potential for transparency techniques

- Mechanistic Interpretability

- Representation Engineering, model editing and probing
- Critiques of Transparency for AI Safety

Week 5 & 6:

- Privacy & Fairness in AI

Week 7 & 8:

- Metrics and Tools for RAI - measuring bias/fairness, adversarial testing, explanations (Lime/SHAP/GradCam), audit mechanisms
- Regulation landscape - DPDP act (India), GDPR (EU), EU AI act, US presidential declaration, Ethical approvals, informed consent, participatory design, future of work, Indian context
- What is AGI? When could it be achieved?
- Instrumental Convergence: Power Seeking, Deception etc.

Week 9 & 10:

- RAI in Legal domain
- RAI in Health care domain
- RAI in Education domain
- A few other domains
- Policy issues in RAI

Week 11 & 12:

- Couple of panel discussion with industry practitioners, academic, government (possibly), and others.
- Fireside chat with eminent personalities
- Recorded Paper reading discussion

CourseProvider : NPTEL

Professor : Prof Ponnurangam Kumaraguru

Prof Balaraman Ravindran
Prof Arun Rajkumar

University : IIIT Hyderabad
IIT Madras

Academic Year: 2024-2025
Semester : Summer

COURSE CODE	COURSE TITLE	W	H	C
10212CS406	DEEP LEARNING FOR COMPUTER VISION	12	-	3

CourseCategory:

Independent Learning (Self - Learning Course)

Course Contents

Week 1: Introduction and Overview:

- Course Overview and Motivation; History of Computer Vision; Image Representation; Linear Filtering, Correlation, Convolution; Image in Frequency Domain
- (Optional) Image Formation; Image Sampling

Week 2: Visual Features and Representations:

- Edge Detection; From Edges to Blobs and Corners; Scale Space, Image Pyramids and Filter Bank; SIFT and Variants; Other Feature Spaces
- (Optional) Image Segmentation, Human Visual System

Week 3: Visual Matching:

- Feature Matching; From Points to Images: Bag-of-Words and VLAD Representations; Image Descriptor Matching; From Traditional Vision to Deep Learning
- (Optional) Hough Transform; Pyramid Matching

Week 4: Deep Learning Review:

- Neural Networks: A Review; Feedforward Neural Networks and Backpropagation; Gradient Descent and Variants; Regularization in Neural Networks; Improving Training of Neural Networks

Week 5: Convolutional Neural Networks (CNNs):

- Convolutional Neural Networks: An Introduction; Backpropagation in CNNs; Evolution of CNN Architectures for Image Classification; Recent CNN Architectures; Finetuning in CNNs

Week 6: Visualization and Understanding CNNs:

- Explaining CNNs: Visualization Methods; Early Methods (Visualization of Kernels; Backprop-to-image/Deconvolution Methods); Class Attribution Map Methods (CAM, Grad-CAM, Grad-CAM++, etc); Going Beyond Explaining CNNs

- (Optional) Explaining CNNs: Recent Methods

Week 7: CNNs for Recognition, Verification, Detection, Segmentation:

- CNNs for Object Detection; CNNs for Segmentation; CNNs for Human Understanding: Faces
- (Optional) CNNs for Human Understanding: Human Pose and Crowd; CNNs for Other Image Tasks

Week 8: Recurrent Neural Networks (RNNs):

- Recurrent Neural Networks: Introduction; Backpropagation in RNNs; LSTMs and GRUs; Video Understanding using CNNs and RNNs

Week 9: Attention Models:

- Attention in Vision Models: An Introduction; Vision and Language: Image Captioning; Self-Attention and Transformers
- (Optional) Beyond Captioning: Visual QA, Visual Dialog; Other Attention Models

Week 10: Deep Generative Models:

- Deep Generative Models: An Introduction; Generative Adversarial Networks; Variational Autoencoders; Combining VAEs and GANs
- (Optional) Beyond VAEs and GANs: Other Deep Generative Models

Week 11: Variants and Applications of Generative Models in Vision:

- GAN Improvements; Deep Generative Models across Multiple Domains; Deep Generative Models: Image Application
- (Optional) VAEs and Disentanglement; Deep Generative Models: Video Applications

Week 12: Recent Trends:

- Few-shot and Zero-shot Learning; Self-Supervised Learning; Adversarial Robustness; Course Conclusion
- (Optional) Pruning and Model Compression; Neural Architecture Search

Course Provider : NPTEL

Professor : Prof. Vineeth N Balasubramanian

University : IIT Hyderabad

Academic Year: 2024-2025

Semester : Summer

COURSE CODE	COURSE TITLE	W	H	C
10212CS408	SOFTWARE TESTING	12	-	3

CourseCategory:

Independent Learning (Self - Learning Course)

Course Contents**Week 1:** Techniques and algorithms for test case design: Graphs based testing- structural coverage criteria.**Week 2:** Graphs based testing: Data flow coverage criteria**Week 3:** Graphs based testing: Data flow coverage criteria**Week 4:** Graphs coverage for source code, design elements and requirements**Week 5:** Techniques and algorithms for test case design: Logic based testing- Predicates, logic based coverage criteria**Week 6:** Specification based logic coverage, logic coverage on finite state machines**Week 7:** Input space partitioning: Input domain modeling, combination strategies criteria**Week 8:** Syntax based testing: Coverage criteria based on syntax, mutation testing**Week 9:** Test case design (as learnt above) applied to object-oriented applications**Week 10:** Test case design (as learnt above) applied to web applications**Week 11:** Symbolic testing**Week 12:** Concolic testing, Conclusion**CourseProvider** : NPTEL**Professor** : Prof. Meenakshi D'souza**University** : IIIT Bangalore

Academic Year: 2024-2025

Semester : Winter

COURSE CODE	COURSE TITLE	W	H	C
10212CS411	Deep Learning for Natural Language Processing	12	-	3

Course Category:

Independent Learning (Self - Learning Course)

Course Outcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Explore Natural Language Processing and shallow and Deep Neural Networks.	K3
CO2	Build and train Convolutional Neural Network (CNN) model for real time applications.	K3
CO3	Use various Recurrent Neural Network architectures using GRU, LSTM.	K3
CO4	Apply Various Large Language Models and Unique Architecture Differences.	K3
CO5	Perform Parameter Efficient Fine-Tuning (PEFT) using LoRA and QLoRA.	K3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

Correlation of COs with Program outcomes and Programme Specific Outcomes:

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	3	3	3		2	2					3	
CO2	3	2	2	2	2	2	2					3	2
CO3	3	3	3	3	2	2	2					3	2
CO4	3	3	3	3	2	2	2					2	2
CO5	3	3	3	3	3	2	2					2	2

Course Contents

Week 1:

- Introduction to NLP: What is Natural Language Processing? A brief primer on word and sentence level tasks and n-gram language Model.

Week 2: Introduction to Deep Learning

- Shallow and Deep Neural Networks
- Representation Learning

Week 3: Word Representations

- Word2Vec
- Glove
- fastText,
- Multilingual representations with emphasis on Indian Languages

Week 4: Recurrent Neural Networks

- RNN LMs
- GRUs, LSTMs, Bi-LSTMs
- LSTMs for Sequence Labeling
- LSTMs for Sequence to Sequence

Week 5: Attention Mechanism

- Sequence to Sequence with Attention
- Transformers: Attention is all you need

Week 6: Self-supervised learning (SSL), Pretraining

- Designing SSL objectives
- Pretrained Bi-LSTMs: ELMO
- Pre-trained Transformers: BERT, GPT, T5, BERT

Week 7:

- Applications: Question Answering, Dialog Modeling, Text Summarization
- Multilingual extension with application to Indian languages

Week 8: Instruction Fine-tuning, FLAN-T5, Reinforcement Learning through Human Feedback (RLHF)

Week 9: In-context learning, chain-of-thought prompting. Scaling Laws. Various Large Language Models and unique architectural differences

Week 10: Parameter Efficient Fine-tuning (PEFT) - LoRA, QLoRA

Week 11: Handling Long Context, Retrieval Augmented Generation (RAG)

Week 12: Analysis and Interpretability, ethical considerations

Course Provider : NPTEL
Professor : Prof. Pawan Goyal
University : IIT Kharagpur

Academic Year: 2024-2025

Semester : Winter

COURSE CODE	COURSE TITLE	W	H	C
10212CS412	Machine Learning for Engineering and science applications	12	-	3

Course Category:

Independent Learning (Self - Learning Course)

Course Outcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Explain the mathematical and computational foundations of machine learning, including linear algebra, probability, and optimization.	K2
CO2	Apply machine learning models such as linear regression, logistic regression, and neural networks to solve real-world problems.	K3
CO3	Describe the architecture and operations of advanced neural networks, including CNNs, RNNs, LSTMs, and GRUs.	K2
CO4	Implement classical machine learning techniques like Bayesian Regression, SVM, Random Forests, k-Means, and Expectation Maximization for specific applications.	K3
CO5	Develop generative and probabilistic models using advanced techniques such as Monte Carlo methods, autoencoders, and GANs to address complex tasks.	K3

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	2	3			3				2					2
CO2	3	2	2		3				2			2		
CO3	2	2	2		3				3	3				
CO4	3	2	3	3	2				3				3	
CO5	2	3	3	2	3				2			2		3

Course Contents**Week 1:** Mathematical Basics 1 – Introduction to Machine Learning, Linear Algebra**Week 2:** Mathematical Basics 2 - Probability

Week 3: Computational Basics – Numerical computation and optimization, Introduction to Machine learning packages

Week 4: Linear and Logistic Regression – Bias/Variance Tradeoff, Regularization, Variants of Gradient Descent, MLE, MAP, Applications

Week 5: Neural Networks – Multilayer Perceptron, Backpropagation, Applications

Week 6: Convolutional Neural Networks 1 – CNN Operations, CNN architectures

Week 7: Convolutional Neural Networks 2 – Training, Transfer Learning, Applications

Week 8: Recurrent Neural Networks RNN, LSTM, GRU, Applications

Week 9: Classical Techniques 1 – Bayesian Regression, Binary Trees, Random Forests, SVM, Naïve Bayes, Applications

Week 10: Classical Techniques 2 – k-Means, kNN, GMM, Expectation Maximization, Applications

Week 11: Advanced Techniques 1 – Structured Probabilistic Models, Monte Carlo Methods

Week 12: Advanced Techniques 2 – Autoencoders, Generative Adversarial Network

Course Provider : NPTEL
Professor : Dr Balaji Srinivasan
 Dr Ganapthy Krishnamurthi
University : IIT Madras

Academic Year: 2024-2025

Semester : Winter

COURSE CODE	COURSE TITLE	W	H	C
10212CS413	Quantum Algorithms and Cryptography	12	-	3

Course Category:

Independent Learning (Self - Learning Course)

Course Outcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Understand the basics of Quantum Information, Entanglement, No Cloning, Quantum Parallelism	K1
CO2	Analyzing the Quantum Algorithms and Cryptography Algorithms	K2
CO3	Understand the concept of Public key encryption and fully homomorphic encryption	K1
CO4	Applying concept of the Quantum key distribution, Quantum one time pad, Quantum public key encryption and Quantum fully homomorphic encryption	K3

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3		3	3							3		
CO2	3	3	3	3	3	2						3		
CO3	3	2		3	3							3		
CO4	3	3	3	3	3	2						3		

Course Contents**Week 1:** Basics of Quantum Information**Week 2:** Entanglement, No Cloning, Quantum Parallelism.**Week 3:** Quantum Algorithms: Deutsch-Jozsa, Simons, Bernstein-Vazirani**Week 4:** Introduction to Cryptography, principles of cryptographic design, building cryptography from RSA, Discrete Log

Week 5: Key exchange, Symmetric and public key encryption, Random Oracle Model, RSA and Elgamal encryption

Week 6: Boolean Fourier Analysis, Grover's Algorithm, Quantum Fourier Transform

Week 7: Shor's Algorithm, Hidden subgroup problem

Week 8: Post Quantum Crypto: Introduction to lattices, Useful Lattice Problems. Learning with Errors and Short Integer Solution problem. Connection to dihedral hidden subgroup problem.

Week 9: Public key encryption and fully homomorphic encryption

Week 10: Quantum key distribution, Quantum one time pad

Week 11: Quantum public key encryption

Week 12: Quantum fully homomorphic encryption

Course Provider : NPTEL

Professor : PROF. SHWETA AGRAWAL

University : IIT Madras

Academic Year: 2024-2025

Semester : Winter

COURSE CODE	COURSE TITLE	W	H	C
10212CS414	Foundation of Cloud IoT Edge ML	8	-	2

Course Category:

Program Elective (Self - Learning Course)

Course Outcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Understand the fundamentals of Cloud Computing, its limitations for low-latency use cases, and how Edge Computing addresses these challenges in supporting real-time IoT applications.	K2
CO2	Understand IoT edge platforms such as Azure IoT Hub and AWS IoT Platform, focusing on their role in secure and efficient data processing and communication.	K2
CO3	Apply containerization and orchestration techniques, such as Docker and Kubernetes, in Edge Computing to improve the security and scalability of IoT applications.	K3
CO4	Understand the principles of distributed IoT systems, including time ordering, clock synchronization, and secure storage system design.	K2
CO5	Apply MQTT and Kafka in building IoT pipelines, ensuring efficient and secure data transfer in edge-based IoT systems.	K3

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	2	2	2						2	3	2	
CO2	3	3	2	2	3				2	2	2	2	3	
CO3	2	3	3	2	3	2	2	2		3	3	3	1	

CO4	3	2	2	3	2					2	2	2	2	
CO5	2	3	3	2	3	2		2	2	3	3	3	2	

(H-3, M-2, L-1)

Course Contents**Week 1:** Introduction to Cloud and its limitations to support low latency use cases**Week 2:** Edge computing to support IoT applications such as self-driving cars, etc**Week 3:** Introduction to IoT Edge platforms such as Azure IoT hub, AWS IoT platform**Week 4:** Introduction to docker container and kubernetes in edge computing**Week 5:** Concepts of distributed systems in IoT applications such as time ordering and clock synchronisation, distributed snapshot, etc**Week 6:** Edge Design of IoT storage system like key value store**Week 7:** Introduction to MQTT and Kafka for end-to-end IoT pipeling**Week 8:** Use Cases of Machine Learning for IOT in predictive maintenance, image classifier, and self-driving cars.

Course Provider : NPTEL
Professor : Prof. Rajiv Misra
University : IIT, Patna

Academic Year: 2024-2025
Semester : Winter

COURSE CODE	COURSE TITLE	W	H	C
10212CS415	Applied Linear Algebra in AI and ML	12	-	2

Course Category:

Independent Learning (Self - Learning Course)

Course Outcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Understand and explain fundamental linear algebra concepts such as vectors, matrices, eigenvalues, eigenvectors, and matrix decompositions used in AI and ML.	K3
CO2	Apply linear algebra techniques to represent and solve problems in machine learning, including transformations, dimensionality reduction, and optimization.	K3
CO3	Analyze and interpret the role of linear algebra in AI/ML algorithms such as regression, principal component analysis (PCA), and neural networks.	K3

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO6	PO 7	PO 8	PO 9	PO1 0	PO1 1	PO1 2
CO1	3	2	—	1	1	—	—	—	—	—	—	2
CO2	3	3	2	2	2	—	—	—	—	—	—	2
CO3	2	3	1	3	2	—	—	—	—	—	—	3

Course Contents

Week 1: Vectors, operations on vectors, vector spaces and subspaces, inner product and vector norm, linear dependence and independence, Matrices, linear transformations, orthogonal matrices

Week 2: System of linear equations, existence and uniqueness, left and right inverses, pseudo inverse, triangular systems

Week 3: LU decomposition and computational complexity, rotators and reflectors, QR decomposition, Gram Schmidt Orthogonalization

Week 4: Condition number of a square matrix, geometric interpretation, norm of matrix, sensitivity analysis results for the system of linear equations

Week 5: Linear least squares, existence and uniqueness, geometrical interpretation, data fitting with least squares, feature engineering, application to Vector auto-regressive models, fitting with continuous and discontinuous piecewise linear functions

Week 6: Application of least squares to classification, two-class and multi-class least squares classifiers, Polynomial classifiers, application to MNIST data set

Week 7: Multi-objective least squares, applications to estimation and regularized inversion, regularized data fitting and application to image de-blurring, constrained least squares, application to portfolio optimization

Week 8: Eigenvalue eigenvector decomposition of square matrices, spectral theorem for symmetric matrices

Week 9: SVD, relation to condition number, sensitivity analysis of least squares problems, variation in parameter estimates in regression

Week 10: Multicollinearity problem and applications to principal component analysis (PCA) and dimensionality reduction, power method, application to Google page ranking algorithm

Week 11: Underdetermined systems of linear equations, least norm solutions, sparse solutions, applications in dictionary learning and sparse code recovery, inverse eigenvalue problem, application in construction of Markov chains from the given stationary distribution

Week 12: Low rank approximation (LRA) and structured low rank approximation problem (SLRA), application to model order selection in time series, alternating projections for computing LRA and SLRA

Course Provider : NPTEL

Professor : PROF:SWANAND KHARE
University : IIT Kharagpur

Academic Year: 2024-2025
 Semester : Winter

COURSE CODE	COURSE TITLE	W	H	C
10212CS416	Artificial Intelligence: Knowledge Representation and Reasoning	12	-	3

Course Category: Independent Learning (Self - Learning Course)

Course Outcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Apply symbolic reasoning using propositional logic and direct proofs.	K3
CO2	Demonstrate proficiency in first-order logic, including unification and resolution methods.	K3
CO3	Implement forward and backward chaining and resolution refutation to derive conclusions.	K3
CO4	Use Prolog and logic programming for building rule-based systems with Horn clauses and Rete algorithm.	K3
CO5	Design knowledge representation models using semantic networks, frames, and description logics.	K3

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	3	3	2	1	-	-	-	-	-	-	2	2
CO2	3	3	3	2	1	-	-	-	-	-	-	2	2
CO3	3	3	3	2	1	-	-	-	-	-	-	2	2
CO4	3	3	3	3	1	-	-	-	-	-	-	2	2
CO5	3	3	3	3	1	-	-	-	-	-	-	2	2

Course Contents

Week1: Introduction. History and Philosophy.

Week2: Symbolic Reasoning. Truth, Logic, and Provability.

Week3: Propositional Logic. Direct Proofs. The Tableau Method.

Week4: First Order Logic. Universal Instantiation. The Unification Algorithm.

Week5: Forward and Backward Chaining. The Resolution Refutation Method.

Week 6: Horn Clauses and Logic Programming. Prolog.

Week7: RuleBasedSystems. TheOPS5Language. TheRete Algorithm.

Week8: RepresentationinFirstOrderLogic. ConceptualDependency.

Week9: Frames. DescriptionLogics and the WebOntologyLanguage

Week 10: Taxonomies and Inheritance. Default Reasoning.

Week11: Circumscription. Auto-epistemicReasoning. EventCalculus

Week12: EpistemicLogic. Knowledgeand Belief.

Course Provider	:	NPTEL
Professor	:	Prof. Deepak Khemani
University	:	IIT Madras

Academic Year: 2024-2025

Semester : Winter

COURSE CODE	COURSE TITLE	Course Provider	W	H	C
10212CS417	Introduction to Machine Learning	NPTEL	12	-	3

Course Category:

Open Elective (Self - Learning Course)

Course Outcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Understand and explain fundamental concepts of Machine Learning, including supervised and unsupervised learning techniques, model evaluation, and key algorithms.	K2
CO2	Apply appropriate Machine Learning algorithms such as regression, classification, and clustering to solve real-world problems using standard tools and datasets.	K3
CO3	Analyze and evaluate the performance of Machine Learning models using suitable metrics, and interpret the results for decision-making.	K3

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO 1	PO 2	PO 3	PO 4		PO 5	PO6	PO 7	PO 8	PO 9	PO1 0	PO1 1	PO1 2	PSO 1	PSO 2	PSO 3

CO1	3	2	–	1		2	–	–	–	–	–	–	2			
CO2	3	3	2	2		3	–	–	–	1	1	–	2			
CO3	2	3	1	3		2	–	–	–	–	1					

Course Contents:

Week 0: Probability Theory, Linear Algebra, Convex Optimization - (Recap)

Week 1: Introduction: Statistical Decision Theory - Regression, Classification, Bias Variance

Week 2: Linear Regression, Multivariate Regression, Subset Selection, Shrinkage Methods, Principal Component Regression, Partial Least squares

Week 3: Linear Classification, Logistic Regression, Linear Discriminant Analysis

Week 4: Perceptron, Support Vector Machines

Week 5: Neural Networks - Introduction, Early Models, Perceptron Learning, Backpropagation, Initialization, Training & Validation, Parameter Estimation - MLE, MAP, Bayesian Estimation

Week 6: Decision Trees, Regression Trees, Stopping Criterion & Pruning loss functions, Categorical Attributes, Multiway Splits, Missing Values, Decision Trees - Instability Evaluation Measures

Week 7: Bootstrapping & Cross Validation, Class Evaluation Measures, ROC curve, MDL, Ensemble Methods - Bagging, Committee Machines and Stacking, Boosting

Week 8: Gradient Boosting, Random Forests, Multi-class Classification, Naive Bayes, Bayesian Networks

Week 9: Undirected Graphical Models, HMM, Variable Elimination, Belief Propagation

Week 10: Partitional Clustering, Hierarchical Clustering, Birch Algorithm, CURE Algorithm, Density-based Clustering

Week 11: Gaussian Mixture Models, Expectation Maximization

Week 12: Learning Theory, Introduction to Reinforcement Learning, Optional videos (RL framework, TD learning, Solution Methods, Applications)

Course Provider : NPTEL

Professor : PROF. BALARAMAN RAVINDRAN
Department of Computer Science and
Engineering IIT Madras

University : IIT Madras

Academic Year: 2024-2025
Semester : Winter

COURSE CODE	COURSE TITLE	Course Provider	W	H	C
10212CS418	Natural Language Processing	NPTEL	12	-	3

Course Category:

Open Elective (Self - Learning Course)

Course Outcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Understand and explain fundamental concepts of Natural Language Processing, including text preprocessing, linguistic structures, and language models.	K2
CO2	Apply NLP techniques such as tokenization, stemming, parsing, and machine learning-based methods to process and analyze textual data.	K3
CO3	Analyze and evaluate NLP models using appropriate metrics and tools for tasks such as text classification, sentiment analysis, and information extraction.	K3

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	–	1	2	–	–	–	–	–	–	2
CO2	3	3	2	2	3	–	–	–	1	1	–	2

CO3	2	3	1	3	2	–	–	–	–	1	–	3

Course Contents:**Week 1: Introduction and Basic Text Processing****Week 2: Spelling Correction, Language Modeling****Week 3: Advanced smoothing for language modeling, POS tagging****Week 4: Models for Sequential tagging – MaxEnt, CRF****Week 5: Syntax – Constituency Parsing****Week 6: Dependency Parsing****Week 7: Distributional Semantics****Week 8: Lexical Semantics****Week 9: Topic Models****Week 10: Entity Linking, Information Extraction****Week 11: Text Summarization, Text Classification****Week 12: Sentiment Analysis and Opinion Mining****Course Provider : NPTEL****Professor : Prof. Pawan Goyal****University : IIT Kharagpur**

Academic Year: 2024-2025
Semester : Winter

COURSE CODE	COURSE TITLE	Course Provider	W	H	C
10212CS419	Computer Vision and Image Processing - Fundamentals and Applications	NPTEL	12	-	3

Course Category:

Open Elective (Self - Learning Course)

Course Outcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Understand and explain fundamental concepts of image processing and computer vision, including image representation, filtering, enhancement, and feature extraction techniques.	K2
CO2	Apply image processing and computer vision algorithms such as segmentation, edge detection, object recognition, and transformation techniques using appropriate tools and frameworks.	K3
CO3	Analyze and evaluate computer vision models and image processing techniques for real-world applications such as object detection, medical imaging, and pattern recognition.	K3

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	–	1	2	–	–	–	–	–	–	2

CO2	3	3	2	2	3	–	–	–	1	1	–	2
CO3	2	3	1	3	2	–	–	–	–	1	–	3

Course Contents:

Week 1: Introduction to Computer Vision and Basic Concepts of Image Formation: Introduction and Goals of Computer Vision and Image Processing, Image Formation Concepts.

Week 2: Fundamental Concepts of Image Formation: Radiometry, Geometric Transformations, Geometric Camera Models.

Week 3: Fundamental Concepts of Image Formation: Camera Calibration, Image Formation in a Stereo Vision Setup, Image Reconstruction from a Series of Projections.

Week 4: Image Processing Concepts: Image Transforms.

Week 5: Image Processing Concepts: Image Transforms, Image Enhancement.

Week 6: Image Processing Concepts: Image Filtering, Colour Image Processing, Image Segmentation

Week 7: Image Descriptors and Features: Texture Descriptors, Colour Features, Edges/Boundaries.

Week 8: Image Descriptors and Features: Object Boundary and Shape Representations.

Week 9: Image Descriptors and Features: Interest or Corner Point Detectors, Histogram of Oriented Gradients, Scale Invariant Feature Transform, Speeded up Robust Features, Saliency

Week 10: Fundamentals of Machine Learning: Linear Regression, Basic Concepts of Decision Functions, Elementary Statistical Decision Theory, Parameter Estimation, Clustering for Knowledge Representation, Dimension Reduction, Linear Discriminant Analysis.

Week 11: Applications of Computer Vision: Artificial Neural Network for Pattern Classification, Convolutional Neural Networks, Autoencoders.

Week 12: Applications of Computer Vision: Gesture Recognition, Motion Estimation and Object Tracking, Programming Assignments.

Course Provider : NPTEL
Professor : PROF. M.K. BHUYAN
University : IIT Guwahati

Academic Year: 2024-2025
Semester : Winter

COURSE CODE	COURSE TITLE	Course Provider	W	H	C
10212CS420	Deep Learning	NPTEL	12	-	3

Course Category:

Open Elective (Self - Learning Course)

Course Outcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Understand and explain the fundamental concepts of Deep Learning, including neural network architectures, activation functions, optimization techniques, and backpropagation.	K2
CO2	Apply deep learning models such as Artificial Neural Networks (ANN), Convolutional Neural Networks (CNN), and Recurrent Neural Networks (RNN) to solve real-world problems using appropriate frameworks.	K3
CO3	Analyze and evaluate the performance of deep learning models using suitable metrics, and optimize models for applications such as image recognition, natural language processing, and prediction tasks.	K3

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO6	PO 7	PO 8	PO 9	PO1 0	PO1 1	PO1 2
CO1	3	2	–	1	2	–	–	–	–	–	–	2
CO2	3	3	2	2	3	–	–	–	1	1	–	2
CO3	2	3	1	3	2	–	–	–	–	1	–	3

Course Contents:

Week 1: (Partial) History of Deep Learning, Deep Learning Success Stories, McCulloch Pitts Neuron, Thresholding Logic, Perceptrons, Perceptron Learning Algorithm

Week 2: Multilayer Perceptrons (MLPs), Representation Power of MLPs, Sigmoid Neurons, Gradient Descent, Feedforward Neural Networks, Representation Power of Feedforward Neural Networks

Week 3: FeedForward Neural Networks, Backpropagation

Week 4: Gradient Descent (GD), Momentum Based GD, Nesterov Accelerated GD, Stochastic GD, AdaGrad, RMSProp, Adam, Eigenvalues and eigenvectors, Eigenvalue Decomposition, Basis

Week 5: Principal Component Analysis and its interpretations, Singular Value Decomposition

Week 6: Autoencoders and relation to PCA, Regularization in autoencoders, Denoising autoencoders, Sparse autoencoders, Contractive autoencoders

Week 7: Regularization: Bias Variance Tradeoff, L2 regularization, Early stopping, Dataset augmentation, Parameter sharing and tying, Injecting noise at input, Ensemble methods, Dropout

Week 8: Greedy Layerwise Pre-training, Better activation functions, Better weight initialization methods, Batch Normalization

Week 9: Learning Vectorial Representations Of Words

Week 10: Convolutional Neural Networks, LeNet, AlexNet, ZF-Net, VGGNet, GoogLeNet, ResNet, Visualizing Convolutional Neural Networks, Guided Backpropagation, Deep Dream, Deep Art, Fooling Convolutional Neural Networks

Week 11: Recurrent Neural Networks, Backpropagation through time (BPTT), Vanishing and Exploding Gradients, Truncated BPTT, GRU, LSTMs
Week 12: Encoder Decoder Models, Attention Mechanism, Attention over images

Course Provider : NPTEL

Professor : PROF.SUDARSHAN IYENGAR

University : IIT Ropar

Course Provider : NPTEL
Professor : Prof. Mitesh M. Khanna
University : IIT Madras

Academic Year: 2024-2025
 Semester : Winter

COURSE CODE	COURSE TITLE	Course Provider	W	H	C
10212CS421	Machine Learning for Soil and Crop Management	NPTEL	12	-	3

Course Category:

Open Elective (Self - Learning Course)

Course Outcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Comprehend the applications of ML and DL in agriculture	K2
CO2	Apply Principal Component Analysis (PCA) and regression models in agricultural contexts.	K3
CO3	Understand and apply diffuse reflectance spectroscopy in crop and soil analysis.	K2
CO4	Understand digital soil mapping techniques and their agricultural applications	K2
CO5	Apply UAVs combined with ML for enhanced agricultural monitoring	K3

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2		2	3	2					3	3	3	3
CO2	3	3	2	3	3	2		2	2		3	3	3	3
CO3	3	3	3	3	3	3		2	2	2	2	2	2	
CO4	3	3	2	3	3	2					2	2	2	

CO5	3	3	2	3	3	3		2	2	2	3	1		

Course Contents:

Week 1: General Overview of ML And DL Applications In Agriculture

Week 2: Basics of Multivariate Data Analytics

Week 3: Principal Component Analysis and Regression Applications in Agriculture

Week 4: Applications of Classification and Clustering Methods in Agriculture

Week 5: Diffuse Reflectance Spectroscopy: Basics and Applications For Crop And Soil

Week 6: Use Of ML For Portable Proximal Soil and Crop Sensors

Week 7: ML And DL for Soil and Crop Image Processing

Week 8: UAV and ML Applications in Agriculture

Week 9: Hyperspectral Remote Sensing and ML Applications in Agriculture

Week 10: Digital Soil Mapping – General Overview

Week 11: Digital Soil Mapping with Continuous Variables

Week 12: Digital Soil Mapping with Categorical Variables

Course Provider : NPTEL

Professor : Prof. Somsubhra Chakraborty

University : IIIT Kharagpur

Academic Year: 2024-2025

Semester : Winter

COURSE CODE	COURSE TITLE	W	H	C
10212CS422	Algorithms in Computational Biology and Sequence Analysis	12	-	3

Course Category:

Independent Learning (Self - Learning Course)

Course Outcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Apply the knowledge of molecular biology and genetics to use bioinformatics tools, databases, and algorithms for analyzing biological sequences.	K3
CO2	Demonstrate the application of string-matching algorithms such as suffix trees, suffix arrays, and Burrows-Wheeler Transform (BWT) for solving exact matching problems in genomic data.	K3
CO3	Implement dynamic programming and heuristic alignment techniques to perform pairwise sequence alignment and evaluate the alignment significance using statistical measures.	K3
CO4	Apply graph algorithms, including de Bruijn graphs and overlap graphs, to solve genome assembly problems and construct evolutionary trees using multiple sequence alignment.	K3
CO5	Use Hidden Markov Models (HMMs) and large language models to classify biological sequences, predict genes, and interpret genomic variations through pangenome graphs.	K3

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	1	2	1								1

CO2	3	3	2	3	2								2
CO3	2	3	3	2	3								2
CO4	3	2	3	3	2								1
CO5	2	3	2	3	3								2

Course Contents

Week 1: Introduction

- Brief review of the fundamentals of molecular biology and genetics.
- Examples of widely used software, algorithms, databases.

Week 2: Strings and exact matching

- Suffix trees, suffix arrays, BWT, and their applications

Week 3: Pairwise Sequence Alignment

- Classic dynamic programming ideas for pairwise sequence alignment.

- Statistical measures of alignment significance.

Week 4: Heuristic Sequence Alignment

- Mathematical ideas underlying heuristic sequence aligners.
- Applications of sequence alignment for mutation finding and disease diagnosis.

Week 5: Genome reconstruction using graph algorithms

- de Bruijn Graphs, Overlap graphs, Shortest common superstring

Week 6: Evolutionary tree construction

- Multiple sequence alignment – formulations, optimal and approximation algorithms
- Classical and contemporary algorithms for inferring evolutionary trees.

Week 7: Sequence models and classification

- Gene finding. Hidden Markov models
- Large language models for biological sequences

Week 8: Pangenome graphs

- Gene finding. Hidden Markov models
- Large language models for biological sequences

Week 9: Discussion of research papers

Week 10: Discussion of research papers(Cont.,)

Week 11: Discussion of research papers(Cont.,)

Week 12: Discussion of research papers(Cont.,)

Course Provider : NPTEL
Professor : Prof. Chirag Jain
University : IISc Bangalore

Academic Year: 2024-2025

Semester : Winter

COURSE CODE	COURSE TITLE	Course Provider	W	H	C
10212CS423	Introduction to Large Language Models (LLMs)	NPTEL	12	-	3

Course Category:

Open Elective (Self - Learning Course)

Course Outcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Understand the fundamental principles of NLP, including its pipeline and key applications.	K2
CO2	Apply statistical language models and smoothing techniques for text processing tasks	K3
CO3	Develop and implement neural architectures, including deep learning models and PyTorch, for NLP tasks.	K3
CO4	Analyze and utilize advanced word representations (Word2Vec, GloVe) and tokenization strategies for better text understanding.	K3
CO5	Design and implement advanced neural language models, including RNNs, LSTMs, GRUs, Transformers, and attention mechanisms.	K2

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO6	PO 7	PO 8	PO 9	PO1 0	PO1 1	PSO 1	PSO 2	PSO 3
CO1			3	3		3	2	2	2	2	2			
CO2	2					2	2	2	2	2	2	2	2	
CO3	2	2				3	3	2	3	2	2	2	2	
CO4	2	2				3	2	2	2	2	2	2	2	2

CO5	2	2				3	3	3	3	3	3			2

Course Contents:**Week 1**

- Course Introduction
- Introduction to NLP (NLP Pipeline, Applications of NLP)

Week 2

- Introduction to Statistical Language Models
- Statistical Language Models: Advanced Smoothing and Evaluation

Week 3

- Introduction to Deep Learning (Perceptron, ANN, Backpropagation, CNN)
- Introduction to PyTorch

Week 4

- Word Representation
 - Word2Vec, fastText b. GloVe
- Tokenization Strategies

Week 5

- Neural Language Models
 - a. CNN, RNN b. LSTM, GRU
 - Sequence-to-Sequence Models, Greedy Decoding, Beam search
- Other Decoding Strategies: Nucleus Sampling, Temperature Sampling, Top-k Sampling
- Attention in Sequence-to-Sequence Models

Week 6

- Introduction to Transformers
 - a. Self and Multi-Head Attention b. Positional Encoding and Layer Normalization
- Implementation of Transformers using PyTorch

Week 7

- Transfer Learning: ELMo, BERT (Encoder-only Model)
- Transfer Learning: GPT (Decoder-only Model), T5 (Encoder-decoder model)
- Introduction to HuggingFace

Week 8

- Instruction Fine-tuning
- In-context Learning and Prompting Techniques
- Alignment with Human Feedback (RLHF)

Week 9

- Parameter-efficient Adaptation (Prompt Tuning, Prefix Tuning, LoRA)
- An Alternate Formulation of Transformers: Residual Stream Perspective
- Interpretability Techniques

Week 10

- Knowledge graphs (KGs)
 - a. Representation, completion
 - b. Tasks: Alignment and isomorphism
 - c. Distinction between graph neural networks and neural KG inference

Week 11

- Open-book question answering: The case for retrieving from structured and unstructured sources; retrieval-augmented inference and generation
- Retrieval augmentation techniques
 - a. Key-value memory networks in QA for simple paths in KGs
 - b. Early HotPotQA solvers, pointer networks, reading comprehension
 - c. REALM, RAG, FiD, Unlimiformer
 - d. KGQA (e.g., EmbedKGQA, GrailQA)
 - **Week 12**
- Overview of recently popular models such as GPT-4, Llama-3, Claude-3, Mistral, and Gemini
- Ethical NLP – Bias and Toxicity
- Conclusion

Course Provider : NPTEL
Professor : Prof. Tanmoy Chakraborty
 Prof. Soumen Chakraborti
University : IIT Bombay

Academic Year: 2024-2025
Semester : Winter

COURSE CODE	COURSE TITLE	W	H	C
10212CS424	Deep Learning for Visual Computing	12	-	3

Course Category:

Independent Learning (Self - Learning Course)

Course Outcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Understand the fundamental concepts of deep learning and their application in visual computing tasks such as image classification and object detection.	K2
CO2	Design and implement deep learning models (e.g., CNNs, transfer learning models) for solving real-world computer vision problems.	K3
CO3	Analyze and evaluate the performance of deep learning models using appropriate metrics and optimize them for improved accuracy and efficiency.	K3

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	-	-	1	-	-	-	-	1	-	2
CO2	2	3	3	1	3	-	-	-	1	1	-	-
CO3	2	3	2	2	2	-	-	-	-	1	-	2

Course Contents

Week 1: Introduction to Visual Computing and Neural Networks

Week 2: Multilayer Perceptron to Deep Neural Networks with Autoencoders

Week 3: Autoencoders for Representation Learning and MLP Initialization

Week 4: Stacked, Sparse, Denoising Autoencoders and Ladder Training

Week 5: Cost functions, Learning Rate Dynamics and Optimization

Week 6: Introduction to Convolutional Neural Networks (CNN) and LeNet

Week 7: Convolutional Autoencoders and Deep CNN (AlexNet, VGGNet)

Week 8: Very Deep CNN for Classification (GoogLeNet, ResNet, DenseNet)

Week 9: Computational Complexity and Transfer Learning of a Network

Week 10: Object Localization (RCNN) and Semantic Segmentation

Week 11: Generative Models with Adversarial Learning

Week 12: Recurrent Neural Networks (RNN) for Video Classification

Course Provider : NPTEL
Professor : Prof. Debdeep Sheet
University : IIT Kharagpur

COURSE CODE	COURSE TITLE	W	H	C
10212CS425	Essentials of Data Science With R Software-1: Probability and Statistical	12	-	3

CourseCategory:

Independent Learning (Self - Learning Course)

CourseOutcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Understand data science, probability theory, and random variables.	K2
CO2	Apply probability distributions for real-world data problems.	K3
CO3	Explain sampling distributions, convergence, and CLT.	K2

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PSO1	PSO2
CO1	3	3	-	-	-	-	-	3	-
CO2	-	-	3	-	3	-	-	3	3
CO3	-	3	-	-	-	3	-	3	-

Course Contents

Week 1: Introduction to data science, basic calculations with R Software and probability theory

Week 2: Probability theory and random variables

Week 3: Random variables and Discrete probability distributions

Week 4: Continuous probability distributions

Week 5: Sampling distributions and Functions of random variables

Week 6: Convergence of random variables, Central limit theorems and Law of large numbers

Week 7: Statistical inference and point estimation

Week 8: Methods of point estimation of parameters

Week 9: Point and confidence interval estimation

Week 10: Confidence interval estimation and test of hypothesis

Week 11: Test of hypothesis

Week 12: Test of hypothesis for attributes and other tests

Course Provider : NPTEL
Professor : Prof. Shalabh
University : IIT Kanpur

Academic Year: 2024-2025
 Semester : Winter

COURSE CODE	COURSE TITLE	W	H	C
10212CS426	Essentials of Data Science With R Software-2: Sampling Theory and Linear Regression Analysis	12	-	3

Course Category:

Independent Learning (Self - Learning Course)

Course Outcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Understand data science, probability theory, and random variables.	K2
CO2	Apply advanced sampling techniques like stratified sampling and bootstrap in R.	K3
CO3	Understand concepts of regression models and their importance in data science.	K2
CO4	Perform simple and multiple regression analysis using R software.	K3
CO5	Apply LASSO regression for variable selection and model optimization in R.	K3

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO6	PO 7	PO 8	PO 9	PO1 0	PO1 1	PSO 1	PSO 2	PSO 3
CO1			3	3		3								
CO2	2					2						2	2	
CO3	2	2	2		2	3						2	2	
CO4	2	2			2	3						2	2	2
CO5	2	2	2		2	3								2

Course Contents

Week 1: Introduction to data science and Calculations with R Software

Week 2: Basic Fundamentals of Sampling

Week 3: Simple Random Sampling

Week 4: Simple Random Sampling with R

Week 5: Stratified Random Sampling

Week 6: Stratified Random Sampling with R

Week 7: Bootstrap Methodology with R

Week 8: Introduction to Linear Models and Regression and Simple linear regression Analysis

Week 9: Simple Linear Regression Analysis with R

Week 10: Multiple Linear Regression Analysis

Week 11: Multiple Linear Regression Analysis with R

Week 12: Variable Selection using LASSO Regression

Course Provider : NPTEL

Professor : Prof. Shalabh

University : IIT Kanpur

Academic Year: 2024-2025
Semester : Winter

COURSE CODE	COURSE TITLE	Course Provider	W	H	C
10212CS427	Data Science for Engineers	NPTEL	8	-	2

Course Category:

Open Elective (Self - Learning Course)

Course Outcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Understand and explain fundamental concepts of data science, including data collection, preprocessing, exploratory data analysis, and statistical foundations.	K2
CO2	Apply data science techniques such as data visualization, regression, classification, and clustering using appropriate tools and programming frameworks.	K3

CO3	Analyze and interpret data to derive meaningful insights and evaluate models for decision-making in engineering applications.	K3

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	–	1	2	–	–	–	–	–	–	2
CO2	3	3	2	2	3	–	–	–	1	1	–	2
CO3	2	3	1	3	2	–	–	–	–	1	–	3

Course Contents:

Week 1: Course philosophy and introduction to R

Week 2: Linear algebra for data science 1. Algebraic view - vectors, matrices, product of matrix & vector, rank, null space, solution of over-determined set of equations and pseudo-inverse) 2. Geometric view - vectors, distance, projections, eigenvalue decomposition .

Week 3: Statistics (descriptive statistics, notion of probability, distributions, mean, variance, covariance, covariance matrix, understanding univariate and multivariate normal distributions, introduction to hypothesis testing, confidence interval for estimates)

Week 4: Optimization

Week 5: 1. Optimization 2. Typology of data science problems and a solution framework

Week 6: 1. Simple linear regression and verifying assumptions used in linear regression 2. Multivariate linear regression, model assessment, assessing importance of different variables, subset selection

Week 7: Classification using logistic regression

Week 8: Classification using kNN and k-means clusterin

Course Provider : NPTEL
Professor : PROF. RAGUNATHAN
 RENGASAMY
 PROF. SHANKAR
 NARASIMHAN
University : IIT Madras

Academic Year: 2024-2025
Semester : Winter

COURSE CODE	COURSE TITLE	W	H	C
10212CS428	Data Analytics with Python	12	-	3

Course Category:

Independent Learning (Self - Learning Course)

Course Outcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Apply Python programming to perform basic data analysis and implement data preprocessing techniques.	K3
CO2	Apply probability concepts, sampling methods, and sampling distributions to make data-driven inferences.	K3
CO3	Conduct hypothesis testing and perform two-sample testing, including ANOVA, to compare datasets.	K3

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO6	PO 7	PO 8	PO 9	PO1 0	PO1 1	PO1 2	PSO 1	PSO 2
CO1	3	2	1	2	3									2
CO2	2	3	2	3	2									2
CO3	2	3	2	3	2									2

Course Contents

Week 1 : Introduction to data analytics and Python fundamentals
Week 2 : Introduction to probability

		Week 3 :	Sampling and sampling distributions
Week 4	:		Hypothesis testing
Week 5	:		Two sample testing and introduction to ANOVA
Week 6	:		Two way ANOVA and linear regression
Week 7	:		Linear regression and multiple regression
Week 8	:		Concepts of MLE and Logistic regression
Week 9	:		ROC and Regression Analysis Model Building
Week 10	:		Test and introduction to cluster analysis
Week 11	:		Clustering analysis
Week 12	:		Classification and Regression Trees (CART)
Course Provider	:		NPTEL
Professor	:		Prof. A Ramesh
University	:		IIT Roorkee

Academic Year: 2024-2025
Semester : Winter

COURSE CODE	COURSE TITLE	W	H	C
10212CS429	Business Intelligence & Analytics	12	-	3

Course Category:

Independent Learning (Self - Learning Course)

Course Outcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Understand and explain fundamental concepts of Business Intelligence and Analytics, including data warehousing, data mining, and decision support systems.	K3
CO2	Apply BI tools and analytical techniques such as data visualization, OLAP operations, and predictive analytics to transform data into actionable insights.	K3
CO3	Analyze business data to support strategic decision-making and evaluate the effectiveness of analytics solutions in real-world scenarios.	K3

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	–	1	2	–	–	–	–	–	–	2
CO2	3	3	2	2	3	1	–	–	1	2	–	2
CO3	2	3	2	3	2	2	–	–	–	2	1	3

Course Contents

Week 1: Introduction to Business Intelligence & Analytics (BIA), drivers of BIA, types of analytics: descriptive to prescriptive, vocabulary of business analytics, course plan and resources

Week 2: Technical architecture of BIA, case analysis of AT&T Long distance, fundamentals of data management, OnLine Transaction Processing (OLTP), design process of databases

Week 3: Relational databases, normalisation, SQL queries, ShopSense case of management questions, data warehousing, OnLine Analytical Processing (OLAP), data cube

Week 4: Descriptive analytics, and visualization, customer analytics, survival analysis, customer lifetime value, case study
Week 5: Data mining process, introduction to statistical learning, data pre-processing, data quality, overview of data mining techniques, case study using regression analysis

Week 6: Introduction to classification, classification techniques, scoring models, classifier performance, ROC and PR curves

Week 7: Introduction to decision trees, tree induction, measures of purity, tree algorithms, pruning, ensemble methods

Week 8: Tree implementation in Python: problem of targeted mailing

Week 9: Cluster analysis, measures of distance, clustering algorithms, K-means and other techniques, cluster quality

Week 10: A store segmentation case study using clustering, implementation in Python, profiling clusters, cluster interpretation and actionable insights, RFM sub- segmentation for customer loyalty

Week 11: Machine learning, Artificial Neural Networks (ANN), topology and training algorithms, back propagation, financial time series modelling using ANN, implementation in Python

Week 12: Text mining, process, key concepts, sentiment scoring, text mining using R-the case of a movie discussion forum, summary

Course Provider : NPTEL
Professor : PROF. SAJI K MATHEW
University : IIT Madras

Academic Year: 2024-2025
 Semester : Winter

COURSE CODE	COURSE TITLE	W	H	C
10212CS430	Predictive Analytics - Regression and Classification	12	-	3

Course Category:

Independent Learning (Self - Learning Course)

Course Outcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Understand and explain the fundamental concepts of predictive analytics, including regression and classification techniques, assumptions, and model building principles.	K3
CO2	Apply regression and classification algorithms such as linear regression, logistic regression, decision trees, and ensemble methods to solve real-world prediction problems.	K3
CO3	Analyze and evaluate predictive models using appropriate performance metrics, and interpret results for effective decision-making.	K3

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO1	3	2	–	1	2	–	–	–	–	–	–	2
CO2	3	3	2	2	3	–	–	–	1	1	–	2
CO3	2	3	1	3	2	–	–	–	–	1	–	3

Course Contents

Week 1: Introduction: Introduce the course and the objective of the course. I will provide a broad overview of the course and landscape of the predictive models. In our experience, we see that predictive analytics is crucial in the industry and business and part of the daily life of a data scientist in the corporate sector. Least Squares method: We introduce the concept of simple and

multiple linear regression as regression hyper-plane. Then we discuss the concept of least square methods.

Week 2: Normal Equations: We discuss the results that the normal equations will always have at least one solution. If the system is of full rank, then the least square method will give us analytically solvable unique solutions. Also, we discuss why "mean absolute deviation" does not have an analytical solution. Gauss Markov theorem: Here, we introduce the underlying assumptions of linear regression models. We discuss how the Gauss Markov theorem is developed under the assumptions of the homogeneity and independence of the residuals. We also discuss the concept of mean squared error (MSE). How the MSE and prediction accuracy are related?

Week 3: The geometry of Regression Model and Feature Engineering Here, we discuss some examples. Also, we discuss the "basis expansion" in mathematical statistics, known as "feature engineering" in ML. With feature engineering, we put the original data into a higher dimension. We hope we find a good fit for a linear hyperplane in a higher dimension, explaining the non-linear relationship between the feature space and the target variable. Statistical Inference of Regression Coefficient: Here, we discuss the sampling distribution of regression coefficients and statistical inference of regression coefficient. We discuss how the t-test can be conducted to test whether a predictor/feature has a statistically significant effect on the dependent or target variables.

Week 4: Checking Model Assumptions We discuss how to check the model assumptions like (1) Independence, (2) Homogeneity or (3) Normality using statistical tests. We discuss how Bartlett's rank test can be used to check randomness in residuals. How can we apply the Breusch-Pagan Test to check homogeneity. Then we discuss how the Kolmogorov-Smirnov test can be used to check the normality. Model Comparison with R-squared, RMSE, AIC or BIC We discuss, how we compare two or several models using selection criteria, such as RMSE, R-squared, adjusted-R-Squared, AIC or BIC and select the best model among the set of possible models.

Week 5: Model Complexity and Bias-Variance tradeoff In this part, we tried to understand the issues of model complexities. We discuss the concept of what we mean by a complex model. Sometimes complex models help us to achieve high prediction accuracy. However, we often achieve it as a cost of the interpretability of the model. We try to capture the model complexity concerning the concept of bias variance tradeoff. Finally, we discuss how we can achieve a parsimonious model by minimising MSE. Feature selection and Dimension Reduction Here, we discuss the stepwise feature selection or variable selection algorithms. We discuss how the

feature selection technique can reduce the problem's dimension and achieve an interpretable and parsimonious regression model.

Week 6: Multicollinearity and Variance Inflation Factor Here, we discuss the problem of multicollinearity in the regression problem. We discuss how the correlation between strong or more features induces a strong correlation between the OLS estimators of coefficients. This makes the sampling distribution a very strong elliptical shape due to tight correlation. As a result, the standard error increases significantly, and the confidence interval becomes very large. We discuss how the Variance Inflation Factor (VIF) can be used to measure the contribution of each feature towards the multicollinearity problem. Regularization with LASSO, Ridge and Elastic Net In the second part of Lecture 6, we discuss what is "Ill-posed problems". We discuss how Tikhonov Regularization reconstructs the unconstrained minimization of the OLS method into constraint minimization. We discuss how the L2 penalty corresponds to the Ridge solution and the L1 penalty corresponds to LASSO solutions. We discuss why LASSO is a continuous subset selection and one should use LASSO feature selection over stepwise feature selection. Ridge Regression with Python

Week 7: Regression Analysis with Python Regression Analysis with R Regression Analysis with Julia

Week 8: Major Applications of Regression Models Capital Asset Pricing Model Here, we present a very celebrated application of the statistical regression model in quantitative finance, known as the Capital Asset Pricing Model (CAPM). The CAPM is often used to evaluate whether an asset is overpriced, underpriced, or fairly priced. Bootstrap Regression Here, we discuss the concept of Bootstrap statistics and nonparametric bootstrap regression. We discuss the two algorithms of Bootstrap regression: (1) Residual Bootstrap Regression and (2) Paired Bootstrap Regression . Time Series Forecasting with Regression Model Here, we focus on how the regression model technique can be implemented for long-term and short-term forecasting. We develop a long-term forecasting model by modelling the trend and seasonality as a function of time. On the other hand, we develop the short-term forecasting model using the Auto-regressive model. We used the AirPassengers dataset. Granger Causal model. Here, I introduce an excellent application of the regression model, the Granger Causal model. Correlation does not imply Causation. In practice establishing Causation from the data is very difficult. However, the Granger causal regression model tries to answer the question of causality with limited capability.

Week 9: Logistic Regression In this video, we introduce the concept of logistic regression for binary class classification problems. MLE of coefficient of Logistic Regression Here, we discuss how to estimate the regression coefficient of logistic regression.

Week 10: Fit Logistic Regression with optim function in R Fit Logistic Regression with glm function in R Fit Logistic Regression with sklearn in Python Fit Logistic Regression in Julia

Week 11: Logistic Regression and Inference In this lecture, we discuss how we can make statistical inferences for the logistic regression model. Discriminant Analysis In this lecture, we discuss the concept of discriminant analysis, LDA, and QDA.

Week 12: • Multinomial Logit Regression Generalised Linear Regression Poisson Regression Negative Binomial Regression

Course Provider : NPTEL
Professor : PROF. SOURISH DAS
University : CMI

Academic Year: 2024-2025

Semester : Winter

COURSE CODE	COURSE TITLE	W	H	C
10212CS431	COMPUTER NETWORKS AND INTERNET PROTOCOL	12	-	3

Course Category:

Independent Learning (Self - Learning Course)

Course Outcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Understand the fundamental concepts of computer networks, including network architectures, protocols, and layered models such as OSI and TCP/IP.	K2
CO2	Analyze the functioning of Internet protocols (IP, TCP, UDP, HTTP, DNS) and evaluate their role in reliable data communication across networks.	K2
CO3	Apply networking principles to design, configure, and troubleshoot basic network systems using appropriate tools and techniques.	K2

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	-	-	-	-	-	-	-	1	-	2
CO2	3	3	2	2	-	-	-	-	-	1	-	2
CO3	2	3	3	2	2	-	-	-	1	2	1	2

Course Contents

Week 1 : Introduction to Computer Networks – History, Circuit Switching and Packet Switching

Week 2 : TCP/IP Protocol Stack – Basic Overview

Week 3 : Application Layer Services (HTTP, FTP, Email, DNS)

Week 4 : Transport Layer Primitives – Connection

Establishment and Closure

Week 5 : Flow Control and Congestion Control at the Transport Layer

Week 6 : Transmission Control Protocol – Basic Features, TCP Congestion Control

Week 7 : Network Layer Primitives – IP Addressing

Week 8 : IP Routing – Intra Domain Routing Protocols, Inter Domain Routing Protocols (BGP)

Week 9 : IP Services – SNMP, ARP

Week 10 : Data Link Layer Service Primitives – Forwarding, Flow Control, Error Control

Week 11 : Media Access Control - Channel Access Protocols, Framing

Week 12 : End to End Principles of Computer Networks

CourseProvider : NPTEL

Professor : Prof. Soumya Kanti Ghosh, Prof.
Sandip Chakraborty

University : IIT Kharagpur

Academic Year: 2024-2025

Semester : Winter

COURSE CODE	COURSE TITLE	W	H	C
10212CS432	Blockchain and its Applications	12	-	3

Course Category:

Independent Learning (Self - Learning Course)

Course Outcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Explain the fundamentals of blockchain technology, cryptographic hash functions, and digital signatures.	K2
CO2	Apply blockchain consensus models, including permissionless and permissioned types, to design secure systems.	K3
CO3	Develop smart contracts using Ethereum and Hyperledger Fabric for both permissionless and permissioned blockchains.	K3

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	-	-	-	-	-	-	-	1	-	2
CO2	3	3	3	2	1	-	-	-	-	1	-	2
CO3	2	3	3	2	3	-	-	-	1	2	1	2

Course Contents**Week 1:** Introduction to Blockchain Technology and its Importance**Week 2:** Basic Crypto Primitives I – Cryptographic Hash**Week 3:** Basic Crypto Primitives II – Digital Signature**Week 4:** Evolution of the Blockchain Technology**Week 5:** Elements of a Blockchain**Week 6:** Blockchain Consensus I – Permissionless Models**Week 7:** Blockchain Consensus II – Permissioned Models**Week 8:** Smart Contract Hands On I – Ethereum Smart Contracts (Permissionless Model)**Week 9:** Smart Contract Hand On II – Hyperledger Fabric (Permissioned Model)

Week 10: Decentralized Identity Management**Week 11:** Blockchain Interoperability**Week 12:** Blockchain Applications

Course Provider : NPTEL
Professor : Prof. Sandip Chakraborty
 : Prof. Shamik Sural
University : IIT Kharagpur

Academic Year: 2024-2025
 Semester : Winter

COURSE CODE	COURSE TITLE	W	H	C
10212CS433	Cryptography and Network Security	12	-	3

Course Category:

Independent Learning (Self - Learning Course)

Course Outcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Explain the fundamental concepts of cryptography, including symmetric and asymmetric encryption techniques, hash functions, and authentication mechanisms.	K2
CO2	Analyze various network security protocols and mechanisms (such as SSL/TLS, IPsec, firewalls, and intrusion detection systems) to ensure secure communication.	K3
CO3	Apply cryptographic algorithms and security practices to design and implement secure systems, addressing real-world security threats and vulnerabilities.	K3

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	-	-	-	-	-	-	-	1	-	2
CO2	3	3	2	2	1	-	-	-	-	1	-	2
CO3	2	3	3	2	3	-	-	-	1	2	1	2

Course Contents

Week 1: Introduction to cryptography, Classical Cryptosystem, Block Cipher.

Week 2: Data Encryption Standard (DES), Triple DES, Modes of Operation, Stream Cipher.

Week 3: LFSR based Stream Cipher, Mathematical background, Abstract algebra, Number Theory.

Week 4: Modular Inverse, Extended Euclid Algorithm, Fermat's Little Theorem, Euler Phi-Function, Euler's theorem.

Week 5: Advanced Encryption Standard (AES), Introduction to Public Key Cryptosystem, Diffie-Hellman Key Exchange, Knapsack Cryptosystem, RSA Cryptosystem.

Week 6: Primarily Testing, ElGamal Cryptosystem, Elliptic Curve over the Reals, Elliptic curve Modulo a Prime.

Week 7: Generalized ElGamal Public Key Cryptosystem, Rabin Cryptosystem.

Week 8 : Message Authentication, Digital Signature, Key Management, Key Exchange, Hash Function.

Week 9 : Cryptographic Hash Function, Secure Hash Algorithm (SHA), Digital Signature

Standard (DSS).

Week 10: Cryptanalysis, Time-Memory Trade-off Attack, Differential and Linear Cryptanalysis.

Week 11: Cryptanalysis on Stream Cipher, Modern Stream Ciphers, Shamir's secret sharing and BE, Identity-based Encryption (IBE), Attribute-based Encryption (ABE).

Week 12: Side-channel attack, The Secure Sockets Layer (SSL), Pretty Good Privacy (PGP), Introduction to Quantum Cryptography, Blockchain, Bitcoin and Cryptocurrency.

Course Provider : NPTEL
Professor : Prof. Sourav Mukhopadhyay
University : IIT Kharagpur

Academic Year: 2024-2025
 Semester : Winter

COURSE CODE	COURSE TITLE	W	H	C
10212CS434	Foundations of Cryptography	12	-	3

Course Category:

Independent Learning (Self - Learning Course)

Course Outcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Understand and explain the fundamental concepts of cryptography, including classical ciphers, modern cryptographic techniques, and security principles.	K2
CO2	Apply cryptographic algorithms such as symmetric and asymmetric encryption, hash functions, and digital signatures to ensure data confidentiality and integrity.	K3
CO3	Analyze and evaluate the security of cryptographic systems and identify potential vulnerabilities and attacks in real-world scenarios.	K3

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	-	-	-	-	-	-	-	1	-	2
CO2	2	3	2	1	2	-	-	-	-	1	-	-
CO3	2	3	1	2	-	-	-	-	-	1	-	2

Course Contents

Week 1: Course Overview, Symmetric-key Encryption, Historical Ciphers, Perfect Security and Its Limitations

Week 2: Computational Security, Semantic Security and Pseudorandom Generators (PRGs)

Week 3: Stream Ciphers, Provably-secure Instantiation of PRG, Practical Instantiation of PRG, CPA-security and Pseudo-random Functions (PRFs)

Week 4: CPA-Secure Ciphers from PRF, Modes of Operations of Block Ciphers, Theoretical Constructions of Block Ciphers and Practical Constructions of Block Ciphers

Week 5: DES, AES and Message Authentication Codes (MAC)

Week 6: Information-theoretic Secure MAC, Cryptographic Hash Functions, Ideal-Cipher Model, Davies-Meyer construction and Merkle-Damgård Paradigm

Week 7: Birthday Attacks on Cryptographic Hash Functions, Applications of Hash Functions, Random Oracle Model and Authenticated Encryption

Week 8: Generic Constructions of Authenticated Encryption Schemes, Key-exchange Problem, One-way Trapdoor Functions and Cyclic Groups

Week 9: Discrete-Logarithm Problem, Computational Diffie-Hellman Problem, Decisional Diffie-Hellman Problem, Elliptic-Curve Based Cryptography and Public-Key Encryption

Week 10: El Gamal Encryption Scheme, RSA Assumption, RSA Public-key Cryptosystem, KEM-DEM Paradigm and CCA-security in the Public-key Domain

Week 11: CCA-secure Public-key Hybrid Ciphers Based on Diffie-Hellman Problems and RSA-assumption, Digital Signatures, RSA Signatures and Schnorr Identification Scheme

Week 12: Schnorr Signature, Overview of TLS/SSL, Number Theory, Interactive Protocols and Farewell

Course Provider : NPTEL
Professor : Prof. Ashish Choudhury
University : IIIT Bangalore

Academic Year: 2024-2025
Semester : Winter

COURSE CODE	COURSE TITLE	W	H	C
10212CS435	Secure Computation : Part I	12	-	3

Course Category:

Independent Learning (Self - Learning Course)

Course Outcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Understand the fundamental principles of cryptography as well as distributed computing	K1
CO2	Analyzing secure Computation: motivation and real-world examples, various dimensions, recalling relevant	K2
CO3	Understand the concept of Secret sharing, Linear secret-sharing, general secret-sharing, secure message transmission, Perfectly-secure MPC tolerating general adversaries	K1
CO4	Applying concept of the Various optimizations of Yao's 2PC protocol, Mixed-world MPC protocols: The case of 2 PC in ABY Framework	K3

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3		3		3							3		
CO2		3	3		3	2						3		
CO3		2	3	3	3							3		
CO4		3	3	3	3	2						3		

Course Contents

Week 1: Secure Computation: motivation and real-world examples, various dimensions, recalling relevant topics from abstract algebra (groups, rings, fields) and cryptography

Week 2: Secret sharing (motivation, definition and applications), Shamir secret-sharing, additive secret-sharing, replicated secret-sharing

Week 3: Linear secret-sharing, general secret-sharing, secure message transmission (SMT)

Week 4: BenOr-Goldwasser-Wigderson (BGW) protocol: security proof and detailed analysis

Week 5: Degree-Reduction problem and various solutions, efficient protocols for evaluating multiplication gates, MPC in the pre-processing model

Week 6: Perfectly-secure MPC tolerating general adversaries, perfectly-secure MPC for small number of parties (the case of secure 3PC)

Week 7: Goldreich-Micali-Wigderson (GMW) protocol: security proof and detailed analysis, Oblivious transfer (OT)

Week 8: Construction of OT protocols from various cryptographic assumptions, GMW protocol in the pre-processing model

Week 9: OT extension, Yao's protocol for secure 2-party computation

Week 10: Various optimizations of Yao's 2PC protocol

Week 11: Mixed-world MPC protocols: The case of 2 PC

Week 12: Mixed-world 2PC protocols in the ABY framework

Course Provider : NPTEL

Professor : By Prof. Ashish Choudhury

University : IIIT Bangalore

Academic Year: 2024-2025
Semester : Winter

COURSE CODE	COURSE TITLE	W	H	C
10212CS436	Privacy and Security in Online Social Media	12	-	3

Course Category:

Independent Learning (Self - Learning Course)

Course Outcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Understand the fundamental principles of Privacy and Security in Online Social Media	K1
CO2	Analyzing data collection from social networks, challenges, opportunities, and pitfalls in online social networks, APIs	K2
CO3	Understand the concept of Trust, credibility, and reputations in social systems	K1
CO4	Applying concept of the Information privacy disclosure, revelation and its effects in OSM and online social networks	K3

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3		3		3							3		
CO2		3	3		3	2						3		
CO3		2	3	3	3							3		
CO4		3	3	3	3	2						3		

Course Contents

Week 1: What is Online Social Networks, data collection from social networks, challenges, opportunities, and pitfalls in online social networks, APIs

Week 2: Collecting data from Online Social Media.

Week 3: Trust, credibility, and reputations in social systems

Week 4: Trust, credibility, and reputations in social systems

Week 5: Online social Media and Policing

Week 6: Information privacy disclosure, revelation and its effects in OSM and online social networks

Week 7: Phishing in OSM & Identifying fraudulent entities in online social networks

Week 8: Refresher for all topics

Week 9 to 12: Research paper discussion

Course Provider : NPTEL

Professor : Prof. Ponnurangam Kumaraguru

University : IIT Hyderabad

Academic Year: 2024-2025

Semester : Winter

COURSE CODE	COURSE TITLE	W	H	C
10212CS437	Games and Information	12	-	3

CourseCategory:

Independent Learning (Self - Learning Course)

CourseOutcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Understand the fundamental principles of static games, dynamic games and mechanism designs in gaming.	K1
CO2	Analyze the concept of information structures, role of information structure on equilibria, Bayesian games and information asymmetry.	K2
CO3	Apply the pre-play communication – achievable payoffs and correlated equilibria.	K3
CO4	Implement applications on game theory in finance.	K3

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3			3	2						2	2	
CO2	3	3	2		3	2						2	2	
CO3	3	3	2		3					3			2	
CO4	3	3	2		3								2	

Course Contents

Week 1: Outline of the course, Definition of a game, Nash equilibrium, Examples of Nash equilibrium, Weakly dominated strategies.

Week 2: Strictly dominated strategies, Aumann model of incomplete information, Knowledge operator, Common knowledge, Structure theorem of common knowledge.

Week 3: Dynamic games, Information structures, Commitment, Mixed and Behavioral strategies, Kuhn's theorem, Bayesian games, Bayesian Nash equilibrium. Mechanism design(ContdProof of the structure theorem of common knowledge, Aumann model of incomplete information with belief, Aumann's agreement theorem, Zero-sum game definition, Security strategies, Saddle point strategies.

Week 4: Further properties of saddle point strategies, Mixed strategies, Existence of mixed saddle point strategies, Von-Neumann minmax theorem.

Week 5: Computation of mixed saddle point strategies for various matrix games, Existence of Nash equilibrium for non-zero-sum game via Kakutani fixed point theorem.

Week 6: Existence of Nash equilibrium for infinite strategy space via Brouwer's fixed point theorem, Quantal response: definition and examples, Dynamic game definition, solution concept, Standard normal form of a dynamic game, Threat equilibrium.

Week 7: Extensive Form Game, Single Acts Games, Informationally Inferior Games

Week 8: Information Structure in Single Act Games, Nested and Ladder Nested Extensive, Equilibrium Algorithm Lecture, Stagewise Multi-Act Game, Feedback Nash Equilibrium, Stagewise Multi-Act Game, Feedback Nash Equilibrium

Week 9: Mixed & Behavioral Strategies, Conditions for Equivalence of Mixed & Behavioral Strategies, Kuhn's Theorem, Equivalence of Mixed and Behavioral Strategies

Week 10: Games of Incomplete Information, Bayesian Nash Equilibrium, Self-enforcement of Nash Equilibrium, Stackelberg game

Week 11: Principal-Agent Models, Moral Hazard and Adverse Selection, Games with Contracts

Week 12: Correlated Equilibrium, Bayesian Game with Mediated Communication, Revelation Principle, Bayesian Nash Equilibrium

CourseProvider : NPTEL

Professor : Prof. Ankur A. Kulkarni

University : IIT Bombay

Academic Year: 2024-2025

Semester : Winter

COURSE CODE	COURSE TITLE	W	H	C
10212CS438	Human Computer Interaction	12	-	3

CourseCategory:

Independent Learning (Self - Learning Course)

CourseOutcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Understand the user-centered design thinking, usability principles, interface development, rapid prototyping methodologies, and evaluation techniques.	K1
CO2	Differentiate between human-computer interaction, user experience design, and design thinking concepts.	K2
CO3	Apply user-centered design strategies, data gathering techniques, rapid prototyping tools, and methods for effectively communicating design concepts.	K3
CO4	Implement a prototype and assess innovative design ideas using user-centric approaches, leveraging both manual techniques and automated tools such as advancements in Artificial Intelligence, including Large Language Models (LLMs).	K3

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	2				1					2		
CO2	3		2				1					2		
CO3	3	2	2		1		1			3		2		
CO4	3	2	2		1		1					2		

Course Contents**Week 1:** Introduction to HCI + Case study**Week 2:** What is Design + Case study**Week 3:** What is Interaction + Case study**Week 4:** What is the User Perspective + Case study**Week 5:** What is an Interface + Case study**Week 6:** What is Data Requirement, Gathering and Analysis**Week 7:** Prototyping and Smart UI + Case study**Week 8:** Iterative Design and Evaluation + Case study**Week 9:** HCI with IoT and Applications + Case study**Week 10:** HCI and AI (including LLMs) + Case study**Week 11:** Privacy, Security, and HCI + Case study**Week 12:** HCI and AI based Conversational Systems and Applications(e.g., Finance, Healthcare, Education, Software Engineering, Assessment, Information Retrieval) + Projects**Course Provider** : NPTEL**Professor** : Prof. Rajiv Ratn Shah**University** : IIIT Delhi

Academic Year: 2024-2025
Semester : Winter

COURSE CODE	COURSE TITLE	W	H	C
10212CS439	Introduction to Graphic Design	8	-	2

Course Category:

Independent Learning (Self - Learning Course)

Course Outcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Understand the foundational elements of Graphic Design, including Basic Design Principles, Typography, Publication Design and Branding & Identity.	K1
CO2	Analyze the concepts of abstracting visual information through pictograms, icons, symbols, and the integration of typography to craft compelling visual narratives.	K2
CO3	Design innovative solutions that make information engaging, informative, entertaining, and thought-provoking, fostering critical thinking.	K3

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	3									2		
CO2	3	3	3		1		2					2	1	
CO3	3	3	3		1		2			3		2	1	

Course Contents

Week 1: Introduction to Design (Course plan and content)
Week 2: Introduction to Design Elements
Week 3: Introduction to Design Principles
Week 4: Exploring gestalt Principles

Week 5: Introduction to typography**Week 6:** Introduction to Identity Design**Week 7:** Introduction to Print and publication**Week 8:** Review & Discussion**Course Provider** : NPTEL**Professor** : Prof. Mohammad Shahid, Prof. Saurav Khuttiya Deori**University** : IIT Hyderabad

Academic Year: 2024-2025

Semester : Winter

COURSE CODE	COURSE TITLE	W	H	C
10212CS440	Augmenting Design Thinking with Human-Computer Interaction	8	-	2

Course Category:

Independent Learning (Self - Learning Course)

Course Outcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Understand and define the problem to arrive at an empathetic solution by prioritizing the user needs.	K1
CO2	Design an intuitive interface that provide efficiency to the user.	K2
CO3	Understand various approaches to idea generation and selection with the help of design principles and methods to create meaningful interfaces.	K1

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3										2	1	
CO2	3	2	3		2		1					2	1	
CO3	3	2	1		2		1					2	1	

Course Contents**Week 1:** Introduction to concepts of Design Thinking and Human-Computer Interaction.

- What is design thinking?
- Need for design thinking.
- Working Model of Design Thinking
- Process and phases of design thinking

- Design thinking applications
- How to get started with Design Thinking

Week 2: Divergent and Convergent Thinking and its impact on innovation.

- Definition, Necessities, Differences
- Strategies of design thinking
- Convergent Thinking Process
- Application, Planning and Implementation

Week 3: Principles and Methods of Idea Generation and Selection.

- Necessities and importance
- Various methods of Idea generation
- Techniques and tools for idea selection
- Methods of effective idea communication

Week 4: Understanding human interactions with interfaces, user behaviours, and their preferences.

- Need Analysis
- Design for experience
- Design for storytelling
- Empathy mapping
- Context mapping
- Stakeholder mapping
- Customer journey mapping
- Trend analysis
- Universal design and design for accessibility

Week 5: Design Research Methods.

- Primary and secondary methods
- Areas of research (ethnography, anthropometry, ergonomics)
- Types of interview and different methods
- Questionnaire design and application challenges
- Data analysis
- Case study

Week 6: Design process and Prototyping.

- Card sorting & Thematic analysis
- Information Architecture
- Low fidelity Wireframe
- Paper-prototyping
- Design elements (colour, shape, text, etc.)
- High fidelity Wireframe
- User testing
- Case study

Week 7: Tools and Techniques.

- Figma design walkthrough
- Figma prototype walkthrough
- Evaluation techniques
- Task analysis

Week 8: Case studies, best practices, and guidelines.

- Product case study
- Tips and techniques on best practices
- Dialogues notation

CourseProvider : NPTEL

Professor : Prof. Sonal Atreya

University : IIT Roorkee

OPEN ELECTIVES

COURSE CODE	COURSE TITLE	L	T	P	C
10213CS101	OBJECT ORIENTED PROGRAMMING USING JAVA	3	0	0	3

A.Preamble

This course provides basic concepts about Object Oriented Programming, Java Packages, Exceptions, Database connectivity, Java Streams, JavaFX and Java Servlets, JSP, Spring and Hibernate. After successful completion of this course learners can able to develop software modules for real world problem.

B.Pre-requisite courses:

10210CS101 - Problem Solving using C

C.Course Objectives:

Learners are exposed to,

- Implement Basic OOP principles to solve problems
- Practice Java basic constructs
- Build efficient Programming practice in Java
- Web application development with DB servers
- Work on Frameworks such as Spring and Hibernate

D.Course Outcomes:

Upon the successful completion of the course, students will be able to:

CO Nos	Course Outcomes	K - Level
CO1	Explain the fundamentals of Java and Object Oriented Programming (OOP) concepts to solve problems.	K2
CO2	Apply the concepts of java inheritance and exception handling mechanisms for real world problem.	K3
CO3	Solve the problems using Java Streams, Collections, UI Design and Threads	K3
CO4	Develop simple web applications using JSP, Servlet with JDBC and Hibernate Framework	K3
CO5	Build quality standalone and web applications using Spring Framework.	K3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E.Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1		2									
CO2		2	2								
CO3		2	2								
CO4		2	3		2						
CO5	2	2	3		2						

3- High; 2-Medium; 1-Low

F.Course contents**UNIT 1 Basics of OOPS And Java****L-9 Hours**

Introduction to object-oriented programming - Features of Java - JVM- Keywords- Variables - Data types – Operators - Control statements -Command line arguments - Classes and Methods - Objects –Java Packages – Build-in packages - user-defined package -Access specifiers: private, protected, default, public - Constructors - Method Overloading - Type Casting - this keyword- Static – Arrays, Multi-dimensional Arrays.

UNIT 2 INHERITANCE, EXCEPTION HANDLING AND STRING**L-9 Hours**

Basics of Inheritance - Forms of Inheritance - Super keyword – Final - Method Overriding – Abstraction: Abstract Classes - Interfaces. Exception Handling: Java Exception Hierarchy - Exception Types - Throwing and Catching exceptions -Declaring New Exception Types – custom exceptions. String class – Immutable String - String handling methods.

UNIT 3 JAVA STREAMS, COLLECTIONS, UI DESIGN, THREAD**L-9 Hours**

Java Streams: Features, methods – Generic Classes and Methods - Collections Frameworks: List, Queue, Set and Map. JavaFX: Architecture, Effects, Transformation, Animation, Layouts, UI Controls, Event Handling. Threads: Life cycle - Thread states - Thread priority– Thread Synchronization.

UNIT 4 SERVER PROGRAMMING, HIBERNATE FRAMEWORK**L-9 Hours**

Introduction to Servlet and JSP, JDBC, Example Web applications with DB - Basics of Java Networking. Introduction to Java Frameworks - Basics of Hibernate – Hibernate Web application - Hibernate Mapping and types – HQL – Annotations – Caching. Integrating Hibernate and Spring.

UNIT 5 JAVA SPRING FRAMEWORK**L-9 Hours**

Basics of Spring – Spring in IDE– Dependency Injection – Constructor Injection – Spring ORM: Spring with Hibernate – SPEL -Spring MVC – Spring Web – Spring JDBC – Spring with Angular – Spring Boot.

TOTAL: 45Hours**G.Learning Resources (in IEEE Format)****i. Text Books:**

1. Herbert Schildt, “Java: the Complete Reference”, Eleventh edition, Tata Mc-Graw Hill, 2019.[unit 1, 2, 3]
2. Jim Keogh, “J2EE: The complete Reference”, First Edition, Tata Mc-Graw Hill, 2017. [unit 4]
3. K. Santosh Kumar, “Spring and Hibernate”, McGraw Hill Education, 2017. [unit 4,5]

ii. Reference Books:

1. H.M. Deitel and P.J. Deitel,” Java How to Program”, Pearson Prentice Hall Seventh edition, 2018.
2. E. Balaguruswamy, “Programming in java”, Sixth Edition, Tata McGraw Hill, 2019.
3. Jason Hunter, William Crawford, “Java Servlet Programming”, Second Edition, O'Reilly-2018.

iii. Online Resources

- 1.“Programming in Java”, Accessed on: Apr. 20, 2021 [Online], Available: <https://nptel.ac.in/courses/106/105/106105191/>
- 2.“Java Tutorial”, Accessed on: Apr. 20, 2021 [Online], Available: <https://www.javatpoint.com/java-tutorial>

COURSE CODE	COURSE TITLE	L	T	P	C
10213CS102	DATA STRUCTURES	3	0	0	3

A.Preamble

This course aims at molding the learner to understand the various data structures, their organization and operations. The course helps the learners to assess the applicability of different data structures and associated algorithms for solving real world problem.

B.Prerequisite Courses

10210CS101 - Problem Solving using C

C.Course Objectives

Learners are exposed to

- Understand basic elementary data structures.
- Illustrate the basic properties of Trees and Graphs.
- Implement various techniques of sorting and searching algorithms.
- Choose efficient data structure and apply them to solve problems.

D.Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Design modular algorithms to find solution for computational problem with time and space complexities using suitable linear data structure.	K3
CO2	Demonstrate the familiarity with tree data structure, rule to manipulate those, and their canonical applications	K3
CO3	Solve unstructured problem using balanced search tree algorithm, hashing function.	K3
CO4	Implement an appropriate algorithm using graph ADT for an application.	K3
CO5	Apply sorting and searching techniques for real world problems.	K3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E.Correlation of Cos with Program Outcomes and Programme Specific Outcomes:

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	3	2	2	2							
CO2	2	3	3	2							
CO3	2	3	2	2							
CO4	2	2	2	2							
CO5	2	2	3	2							

3- High; 2-Medium; 1-Low

F.Course Contents

Unit 1 Introduction to Data Structure

L- 9 Hours

Introduction: Dynamic aspects of operations on data, Characteristics of data, Creation, Manipulation and Operations on data, Data Structure and its types, Abstract Data Types (ADTs), Analysis of algorithms and its Types, Asymptotic Notation. Arrays: Allocation, Operations and Storage with one-dimensional arrays and multidimensional arrays. Stacks: Operations on Stacks, Applications of Stacks, Queues: Operations and its types. Linked lists: Operations and Types of linked list (singly, doubly and circularly linked list), Implementation of Stack and Queue using linked list, Applications of Linked List.

Case Study: Tower of Hanoi, Sparse matrix.

Unit 2 Trees

L-9 Hours

Introduction to Trees – Definitions and concepts, Representation of Binary Tree, Types of Binary tree, properties, structure and applications of binary tree, Binary Tree Traversal with Recursive and Non-Recursive - Expression Tree, Binary Search Trees – Definition, Properties and Operations of binary search tree. Priority Queue: Operations, Implementations- Heaps: properties and types of Heaps, Binary Heap: Representation, Declaration and Creation of Heap, Heapifying, Inserting and Deleting an element, Destroying Heap, Heap Sort. Case Study: Priority Queue in bandwidth management

Unit 3 Special Trees & Hashing

L-9 Hours

AVL Tree: Properties, Declarations, Rotations, Insertion, Deletion. B-Tree: Properties, Insertion, Deletion –Trie: Insertion, Deletion, Searching.Hashing - Separate Chaining – Open Addressing – Linear Probing – Quadratic Probing – Double Hashing –Rehashing. Case Study: DNA sequence matching

Unit 4 Graphs

L-9 Hours

Introduction to Graphs – Operations on graphs – Representation of Graph – Topological Sort - Graph Traversal - Depth first search – Breadth First Search – Minimum Spanning Tree – Prim's Algorithm- Kruskal's Algorithm – Disjoint subsets and Union-Find algorithms – Shortest Path Algorithms – Unweighted Shortest Paths –Dijkstra's Algorithm. Applications of Graphs. Case Study: Representing Google maps, Recommendations on e-commerce websites.

Unit 5 Sorting, Searching

L-9 Hours

Searching Techniques: Linear Search, Binary Search.Sorting techniques: Insertion Sort, Selection sort, Merge sort, Quick Sort, Shell sort, Bubble sort, Counting sort, Bucket sort, Radix sort. Case Study: Tim Sort.

Total: 45 Hours

G.Learning Resources

i. Text Books:

1. M. A. Weiss, "Data Structures and Algorithm Analysis in C", Second Edition, Pearson Education, Reprinted edition 2013. [Unit 1,2,3]
2. Ellis Horowitz, Sartaj Sahni, Susan Anderson Freed, "Fundamentals of Data Structure in C", Universities Press,2017. [Unit 4,5]

ii. Reference Books:

1. V. Aho, J. E. Hopcroft, and J. D. Ullman, “Data Structures and Algorithms”, Pearson Education, First Edition, Reprint 2017. [Unit 1,2]
2. R. F. Gilberg, B. A. Forouzan, “Data Structures”, Second Edition, Thomson India Edition, 2007. [Unit 3,4]
3. Seymour Lipschutz, “Data Structures with C”, Scham’s outlines, Tata McGraw Hill, 2017. [Unit 4,5]

iii. Online References:

1. “Introduction to Data Structures and Algorithms” 31.12.2009. Accessed on Apr. 20, [Online]. Available: <https://nptel.ac.in/courses/106/102/106102064/>.
2. “Data Structures and algorithms”. Accessed on: Apr. 15, 2021 [Online]. Available: <https://www.coursera.org/specializations/data-structures-algorithms>.
3. “Data structures Visualization” 2011. Accessed on: Apr. 15, 2021 [Online]. Available: <https://www.cs.usfca.edu/~galles/visualization/Algorithms.html>
4. “Data Structures Fundamentals” 2021, Accessed on: Apr. 20,2021 [Online]. Available: <https://www.edx.org/course/data-structures-fundamentals>

COURSE CODE	COURSE TITLE	L	T	P	C
10213CS103	OPERATING SYSTEMS	3	0	0	3

A.Preamble

This course provides discussion about Operating systems kernel architecture, address spaces, system call interface, processes & threads, inter process communication, CPU scheduling deadlock, scheduling, main memory, virtual memory and file systems. This course aims to understand how Operating system makes computer system convenient to use in an efficient manner.

B.Prerequisite Courses

10211CS102 - Data Structures

C.Course Objective

Learners are exposed to

- Recognize operating Systems basic concepts, structures and System Calls
- Apply process management concepts and Synchronization techniques
- Learn different CPU scheduling algorithms and Deadlock Handling methods
- Understand Various Memory and File management techniques

D.Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Describe the operating system structures ,operations and system calls	K2
CO2	Demonstrate process management concepts and process synchronization methods for real time problems.	K3
CO3	Illustrate CPU scheduling algorithms and deadlock handling methods for the given situation.	K2
CO4	Explain the concepts of various memory management techniques.	K2
CO5	Discussthe concepts of disk Management and File System Interface.	K2
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E.Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	2										
CO2	2	3	2	2							
CO3	2	3	2	2							
CO4	2	2	2	2							
CO5	2	2	2	2							

3- High; 2-Medium; 1-Low

F.Course Contents

UNIT 1 STRUCTURE AND OVERVIEW OF OPERATING SYSTEMS L- 8 Hours

Operating system overview: Objectives – functions - Computer System Organization - operating System Operations- System Calls, System Programs-Operating- System Structure: Traditional UNIX system structure-The Mac OS X structure - Architecture of Google's Android.

UNIT 2 PROCESS MANAGEMENT L-10 Hours

Processes: Process Concept – Threads - Process Scheduling - Operations on Processes – Inter process Communication - Communication in Client–Server Systems-Pipes (RPC, Pipes) - Process Synchronization: The Critical-Section Problem - Semaphores – Mutex Locks- Classic Problems of Synchronization – Monitors. Case Study: Windows Threads and Linux Threads

UNIT 3 CPU SCHEDULING AND DEADLOCK MANAGEMENT L-9 Hours

CPU Scheduling: Scheduling Criteria - Scheduling Algorithms. Deadlocks: Deadlock Characterization - Methods for Handling Deadlocks - Deadlock Prevention - Deadlock Avoidance - Deadlock Detection - Recovery from Deadlock. Case Study: Real Time CPU scheduling

UNIT 4 MEMORY MANAGEMENT L-9 Hours

Main Memory: Swapping - Contiguous Memory Allocation, Segmentation, Paging - Structure of the Page Table - Virtual Memory: Demand Paging - Page Replacement - Allocation of Frames – Thrashing. Case study: Virtual machine

UNIT 5 STORAGE STRUCTURE & FILE SYSTEMS L-9 Hours

Mass Storage Structure: Disk Structure - Disk Scheduling - Disk Management-Structure - File-System Interface: File Concepts -Directory Structure - File Sharing – Protection- File Allocation Methods-NFS. Case study: Recovery in Windows.

TOTAL: 45 Hours

G.Learning Resources (in IEEE Format)

i. Text Books:

1. Abraham Silberschatz, Peter Baer Galvin and Greg Gagne, “Operating System Concepts”, 9th Edition, John Wiley and Sons Inc., 2016.

ii .Reference Books:

1. Andrew S. Tanenbaum, “Modern Operating Systems”, 4th Edition, Prentice Hall, Wesley, 2014.
2. Operating Systems: Principles and Practice (2nd Edition), by Thomas Anderson and Michael Dahlin.

iii. Online Resources:

1. “Learn Operating System” Accessed on: April 21, 2021 [Online]. Available:https://www.tutorialspoint.com/operating_system/index.htm
2. Prof.Santanu Chattopadhyay, “Operating System Fundamentals” July 25,2019. Accessed on: April 21, 2021 [Online]. Available:<https://nptel.ac.in/courses/106/105/106105214/>

COURSE CODE	COURSE TITLE	L	T	P	C
10213CS104	DATABASE MANAGEMENT SYSTEMS	3	0	0	3

A.Preamble

Databases form the backbone of all major applications today – tightly or loosely coupled, intranet or internet based, financial, social, administrative, and so on. Structured Database Management Systems (DBMS) based on relational and other models have long formed the basis for such databases. While DBMS's differ in the details, they share a common set of models, design paradigms via a Structured Query Language (SQL) or NoSQL. More specifically, this course introduces relational data models; entity-relationship modeling, SQL, data normalization, database design and modern databases as well lightweight databases.

B.Prerequisite Courses

10211CS102 – Data Structures

C.Course Objectives

Learners are exposed to

- Design a database at basic level like Entity-Relationship modeling, Relational Modeling.
- Devise a database using Armstrong axioms, functional dependencies and views.
- Acquire knowledge of Query processing and concurrency transaction control.
- Learn an implementation issues with nontraditional databases.

D.Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Infer database system with basic database designing methodologies.	K2
CO2	Design a database schema for a given problem-domain.	K3
CO3	Apply normalization for the given database application.	K3
CO4	Relate the properties of transaction, and recovery management.	K2
CO5	Use of storage and CRUD operations in modern databases.	K3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E.Correlation of COs with Program Outcomes and Programme Specific Outcomes

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	2										
CO2		3	2	2							
CO3		3	2								
CO4		2	2								
CO5		2		2							

3- High; 2-Medium; 1-Low

F.Course Contents

Unit 1 Introduction to DBMS**L-9 Hours**

Purpose of Database System – Database Schema and Instances- Views of data – Database Languages - Database System Architecture – Database users and Administrator –Distributed Databases-DBD System Architecture-Database models-Entity–Relationship model – E-R Diagrams - Introduction to relational databases –Structure of relational databases- Relational model-Basics- From E/R diagrams to Database Design- Hierarchical model-Tree-structure diagrams- data retrieval-update facility-virtual records-mapping of hierarchies to files-Network Model- Data-structure diagrams-DBTG model- data retrieval-Find-Get-Set processing-update facility-Mapping network to files. Case study: Conference Management system.

Unit 2 Database Design and Query Optimization**L-9 Hours**

Keys and Integrity Constraints - Relational Algebra – Relational Calculus-Tuple –Structured Query language(SQL) Basic and additional Operations –Database Languages Revisited – create-alter- update-insert-rename-truncate-drop-select query commands-grant-revoke-rollback-commit control commands-Operators-aggregate functions-string-indent-date functions-set-union-union all-aliasing-Nested Queries & Join Queries–Join-Natural-inner-equi-cross/cartesian-outer-left-right-full outer join-Common Table Expressions-limit function in statistics-summarizing-Embedded SQL- Indexes and Triggers –compound-Views Definitions -Materialized view-Assertions-Query Processing and Optimization-Heuristics and cost estimates in Query Optimization-Tuning. Case study: Restaurant chain for network model.

Unit 3 Functional Dependencies and Normalization**L-9 Hours**

Introduction and problem of data Redundancy-Features of good Relational database design-Functional Dependencies - Normalization – First Normal Form-minimal cover-canonical Cover-Decomposition- Second Normal Form- Third Normal Form –controversies with SQL in Normalization-Advanced Normalization -Boyce/Codd Normal Form, Fourth Normal Form and Fifth Normal Form- Dependencies preservation-Denormalization-Case Studies of database system like Air-cargo services.

Unit 4 Transaction and Concurrency**L-9 Hours**

Transaction Concepts –Schedules- Transaction States - Concurrent Execution-Serializability- -Two Phase locking Protocol –Timestamp ordering Protocol-Validation Based Protocol-Multiple Granularity-Intention lock modes-Multiple granularity locking scheme-Replication in Distributed databases-Deadlock-Handling-Prevention-Wait-die scheme- wound-wait scheme- Detection-Wait-for graph-Phantom phenomenon-Distributeddeadlock detection-Index locking-weak levels of concurrency- Concurrency in index structures- B+ Tree crabbing- -Types of Failure-Recoverability -Recovery and Prevention- Log based recovery-undo-redo-checkpoint-Recovery Algorithm-Transaction rollback with Logical undo- Recovery Algorithm with Logical Undo-ARIES Recovery Algorithm.

Case study: Instances and Log recovery process in Oracle and Microsoft Access.

Unit 5 Storage and Modern Databases**L-9 Hours**

RAID storage -Modern databases-XML Databases- CAP Theorem-Datalog-Basic structure-Syntax-semantics-Relational operation in Datalog-Recursion in Datalog-Power of Recursion-Monotonicity-Non-monotonicity-NoSQL-Data models-Key value store, column families-Document Databases-MongoDB-Introduction to MongoDB- CRUD operations-

Create-Insert-Update-Query-Removing documents-Sharding-Introduction to Mongoose-core concepts-extending models-CRUD operations with mongoose. Case study: Designing local library in Amazon using mongoose-SQLite-Building and Installing SQLite-DDL- DML-TCL-Select-compound select- alternate Join- JOIN..ON, JOIN..USING-Database design-SQL functions and SQLite extensions-Virtual Tables and Modules.

Total: 45 Hours

G.Learning Resources

i.Text Books:

1. Abraham Silberschatz, Henry F. Korth and S. Sudharshan, “Database System Concepts”, Seventh Edition, Tata McGraw Hill, 2019.[Units-1,2,3,4,5]
2. Shannon Bradshaw, E Brazil, Kristina Chodorow, "MongoDB: The Definitive Guide - Powerful and Scalable Data Storage", Third Edition, Shroff/O'Reilly Inc., January 2020.[Unit-5]
3. AgusKurniawin, “Python and SQLite Development”, First Edition, PE Press, January 2021. [unit-5]

ii.Reference Books:

1. Raghu Ramakrishnan et al, “Database Management Systems”, Third Edition, McGraw Hill, 2014.
2. Elmasri Ramez, Navathe S, “Fundamentals of Database System”, Seventh Edition, Pearson, 2017.
3. J.D.Ullmann et al, "Database Systems: The Complete Book", Second Edition, Pearson Ed, Inc, 2009.

iii.Online References:

1. “Database Management System Part-1&11” Mar.25, 2019. Accessed on Feb.20, 2021[Online].Available:https://infyqt.onwingspan.com/en/viewer/web_module/lex_auth_0130542736573808641006_shared?collectionId=lex_auth_01275806667282022456_shared&collectionType=Course.
2. “Database Design”, Dec 31, 2009. Accessed on May05,2021[Online]. Available: <https://nptel.ac.in/courses/106/106/106106093/>.
3. “Designing local library models”, Accessed on: May 05, 2021[online]. Available: https://developer.mozilla.org/en-US/docs/Learn/Server-side/Express_Nodejs/mongoose.

COURSE CODE	COURSE TITLE	L	T	P	C
10213CS105	COMPUTER NETWORKS	3	0	0	3

A.Preamble

This course is to provide students with an overview of the concepts and fundamentals of computer networks. Topics to be covered include: data communication concepts and techniques in a layered network architecture, communications switching and routing, types of communication, network congestion, network topologies, network configuration and management, network model components, layered network models (OSI reference model, TCP/IP networking architecture) and their protocols, various types of networks and their protocols.

B.Prerequisite Courses

10211CS102 - Data Structures

C.Course Objectives

Learners are exposed to

- Build an understanding of the fundamental concepts of computer networking, protocols, architectures, and applications.
- Understand the division of network functionalities into layers.
- Learn the flow control and error control mechanisms.
- Obtain the skills of subnetting and routing mechanisms.
- Explain the transport layer and application layer protocols.

D.Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Discuss the basic fundamentals of networks for data communication and protocol Layering.	K2
CO2	Know the various techniques and standards of analog and digital data communication and its standards	K2
CO3	Apply various error detection, correction and flow control techniques for efficient transmission of data.	K3
CO4	Interpret the functions and protocols of network and transport layer.	K2
CO5	Outline the various protocols of application layer.	K2

Knowledge Level (Based on revised Bloom's Taxonomy)

K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create

E.Correlation of COs with Program Outcomes and Programme Specific Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3										
CO2	3	2									
CO3	3	3		2							
CO4	2	3		2							
CO5	3										

3- High; 2-Medium; 1-Low

F.Course Contents

Unit 1 Introduction**L-8 Hours**

Data Communication: system components (Passive and Active)-Topology-Network Models: OSI-TCP/IP-Physical and Logical Addressing-WLAN-Data and Signals - Transmission Impairment - Data rate and Channel capacity.

Unit 2 Physical Layer**L- 9 Hours**

Transmission Media: Guided Media - Unguided Media -Digital Transmission - Digital-To - Digital Conversion - Analog Transmission - Digital-To-Analog Conversion - Wireless - Wired LANs: Ethernet - Token ring - Connecting Devices – Switching techniques – Bandwidth Metrics.

Unit 3 Data Link Layer**L-10 Hours**

Link Layer: Types of errors –Error detection- VRC, LRC, CRC andChecksum techniques - Data Forward and backward error correction - Hamming code. Flow control: stop and wait- sliding window protocol, Error control: Stop and wait ARQ- Go-Back-N ARQ- Selective Repeat ARQ Protocols- Asynchronous and Synchronous Protocol- Logical Link Control protocols: HDLC, ALOHA and FDDI.

Unit 4 Network layer, Transport Layer**L-10 Hours**

Logical Addressing: Addressing Methods - IPV4 - IPV6 - Frame Format - IPV6 implementation scenarios and challenges – ARP– RARP - DHCP – ICMP - Unicast Routing Protocols – Intra and Interdomain Routing - Distance Vector Routing - Link State Routing-Transport Layer: Process-to-Process Delivery: UDP – TCP - Congestion Control Techniques - Quality of Service - Techniques to Improve QoS.

Unit 5 Application Layer and Network Security**L-8 Hours**

Session Layer: Functions-Protocols: Winsock, Socket-Remote Procedure Call-Presentation Layer: Functions- Compression Techniques: Lossy and Lossless - Application layer protocols: REMOTE LOGGING - TELNET -EMAIL– DNS – FTP -HTTP – SDN-Introduction-Security Fundamentals.

Total: 45 Hours**F.Learning Resources****i. Text Books:**

1. Behrouz Forouzan, “Data Communications and Networking”, Tata McGraw Hill, 5th Edition,2021. (Reprinted) (Unit 1 to 5)

ii. Reference Books:

- 1.Kurose and K.W. Ross, “Computer Networking: A Top-Down Approach Featuring the Internet”, Addison-Wesley,2017
2. Andrew S. Tanenbaum, David J Wetherall, Computer Networks, 5th Edition, Pearson Edu, 2013.
3. Foundations of Modern Networking: SDN, NFV, QoE, IoT, and Cloud,5th Edition,2017

iii. Online References:

1. “Computer Communication Specialization” Course Era [online] 2019.
Available: <https://www.coursera.org/specializations/computer-communications>
2. “Introduction to Networking” edx [online] 2020, Available:
<https://www.edx.org/course/introduction-to-networking>
3. “Computer Networking” Udacity [online] 2020,
[../WELCOME/AppData/Roaming/Microsoft/Word/](https://www.udacity.com/course/computer-networking--ud436) Available:
<https://www.udacity.com/course/computer-networking--ud436>
4. “Computer Networks and Internet Protocol” NPTEL [online] 2018, Available:
<https://nptel.ac.in/courses/106/105/106105183/>

COURSE CODE	COURSE TITLE	L	T	P	C
10213CS106	DATA WAREHOUSING AND DATA MINING	3	0	0	3

A.Preamble

The course introduces the concepts of data warehouses, modeling and design of data warehouse, techniques for preprocessing, data mining algorithms such as Association rule analysis, classification, clustering and applications of data mining.

B.Prerequisite Courses

10213CS104- Database Management Systems

C.Course Objectives

Students undergoing this course are expected to

- Be familiar with the concepts of Data warehouse, Multidimensional data model and OLAP.
- Characterize the kinds of patterns that can be discovered by association rule mining, classification and clustering.
- Apply data mining techniques in various applications like social, scientific and environmental context.
- Develop skills in selecting the appropriate data mining algorithm for solving practical problems.

D.Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Discuss the subject areas for which a data warehouse is to be built.	K2
CO2	Design Multidimensional data model for data warehouse and analyze the market needs by applying the suitable OLAP operations.	K3
CO3	Choose appropriate input data and preprocess it to provide suitable input for range of data mining algorithms	K3
CO4	Apply Association rule mining, classification techniques to discover various mining patterns.	K3
CO5	Identify the similar objects using clustering techniques and explore recent trends in data mining such as web mining, text mining and spatial	K3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E.Correlation of COs with Program outcomes and Programme Specific Outcomes:

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	2										
CO2		3	2	3							
CO3	2										

CO4	2	2		2							
CO5	2	2		2	2						

3- High; 2-Medium; 1-Low

F.Course Content

UNIT 1 Data Warehousing

L-9 Hours

Introduction to Data warehousing – Evolution of Decision Support systems – Modeling a Data Warehouse – Granularity in the Data Warehouse - Data Warehousing Components, Building a Data Warehouse, Warehouse Database, Mapping the Data Warehouse to a Multiprocessor Architecture, DBMS Schemas for Decision Support, Metadata Management in Data Warehouse: Introduction to Metadata, Categorizing Metadata, Metadata management in practice, Tools for Metadata management.

UNIT 2 Business Analysis

L-9 Hours

Extract, Transform and Load: ETL Overview, ETL Requirements and Steps, Data Transformation, Data Loading, ETL Tools. – Reporting and Query tools and Applications – Online Analytical Processing (OLAP) – Need – Multidimensional Data Model – OLAP Guidelines – Hypercube & Multicubes- Multidimensional versus Multirelational OLAP – Categorization of OLAP Tools- Tuning for data warehouse - Computation of Data Cubes, modeling - OLAP data, OLAP queries - Data Warehouse back end tools - Tuning and testing of Data Warehouse.

UNIT 3 Data Mining

L-9 Hours

Introduction to Data mining and Knowledge Discovery – Data – Databases – Data Mining Functionalities – Steps in Data Mining Process, Architecture of Typical Data Mining systems – Interesting Pattern - Classification of Data Mining Systems – Data Mining Task Primitives – Data Mining Query Languages, Designing Graphical User Interfaces Based on a Data Mining Query Language Architectures of Data Mining Systems. – Issues –Data Preprocessing: Needs Preprocessing the Data, Data Cleaning, Data Integration and Transformation, Data Reduction, Discretization and Concept Hierarchy Generation.

UNIT 4 Association Rule Mining and Classification

L-9 Hours

Association Rule Mining, Mining SingleDimensional Boolean Association Rules from Transactional Databases, Mining Multilevel Association Rules from Transaction Databases, Mining Multidimensional Association Rules from Relational Databases and DataWarehouses, From Association Mining to Correlation Analysis, – Constraint Based Association Mining – Classification and Prediction - Basic Concepts - Decision Tree Induction - Bayesian Classification –Support Vector Machines – Associative Classification – Lazy Learners – Other Classification Methods – Prediction: Linear Regression - Logistic Regression.

UNIT 5 Clustering, Applications And Trends In Data Mining

L-9 Hours

Cluster Analysis - Types of Data – Categorization of Major Clustering Methods– Partitioning Methods: K-Means – Hierarchical Methods: Agglomerative and Divisive Clustering –BRICH - ROCK- Chameleon- Density Based Method- DBSCAN-OPTICS-DENCLUE- Outlier Analysis

– Mining Complex Types of Data: Multi dimensional Analysis and Descriptive Mining of Complex, Data Objects, Mining Spatial Databases, Mining Multimedia Databases, Mining Time-Series and Sequence Data, Mining Text Databases, Mining the World Wide Web. Case Studies (Simulation Tool).

Total : 45 Hours

G.Learning Resources (in IEEE Format)

i.Text Books:

1. W.H. Inmon, “Building the Data Warehouse”, John Wiley & Sons, Inc, 4th Edition, 2005-(Online Publication : 2014). (Unit 1)
2. Jiawei Han and Micheline Kamber, “Data Mining Concepts and Techniques”, Third Edition, Elsevier, 2012. (Unit 2-5)

ii.Reference Books:

1. Hadley Wickham and Garrett Grolemund ”R for Data Science”- O’Reilly Media Inc.,2017.
2. Alex Berson and Stephen J. Smith, “Data Warehousing, Data Mining & OLAP”, Tata McGraw – Hill Edition, Tenth Reprint 2007.
3. Julia and David Robinson-“Text Mining with R” -O’Reilly Media Inc.,2017.

iii.Online References:

1. Data Mining, Stanford University, [Online]
<http://infolab.stanford.edu/~ullman/mmds/ch1.pdf>
2. Data Warehouse Design: Modern Principles and Methodologies, Golfarelli & Rizzi, [online]
https://cdn.ttgtmedia.com/searchDataManagement/downloads/Data_Warehouse_Design.pdf

COURSE CODE	COURSE TITLE	L	T	P	C
10213CS107	DESIGN AND ANALYSIS OF ALGORITHMS	3	0	0	3

A. Preamble

The course introduces the basics of computational complexity analysis and various algorithm design paradigms. The goal is to provide students with solid foundations to deal with a wide variety of computational problems, and to provide a thorough knowledge of the most common algorithms and data structures.

B. Prerequisite Course

10211CS102- Data Structures

C. Course Objectives

Students are able to

- Develop the problem-solving ability and algorithmic analytical skills
- Produce optimized code
- Understand the different techniques involved in solving a problem
- Analyze the logic in terms of Efficiency, Time & Space Complexity
- Break the large size algorithm into smaller size components

D. Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Analyze the time complexity of algorithms using various asymptotic notations.	K3
CO2	Implement solution for the given problems using Brute force and Divide and Conquer techniques.	K3
CO3	Apply Dynamic programming and Greedy techniques to solve the given problems.	K3
CO4	Solve the problems using Backtracking and Branch & Bound techniques.	K3
CO5	Interpret the computational efficiency using iterative approaches	K2
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	3	3				1	1		1	3		
CO2	3	3	3	3				1	1		1	3		
CO3	3	3	3	3				1	1		1	3		

CO4	3	3	3	3				1	1		1	3		
CO5	3	3	3	3				1	1		1	3		

High- 3; Medium-2; Low-1

F. Course Contents

Unit 1 Introduction

L-9 Hours

Notion of an Algorithm – Fundamentals of Algorithmic Problem Solving – Important Problem Types – Analysis of Algorithm Efficiency - Introduction to Asymptotic Notations– Solving Recurrence relations – Master’s theorem - Mathematical analysis for Recursive and Non-recursive algorithms.

(Case Study: Design a gaming application like Chess, Candy crush etc. and select the appropriate algorithm and solve it).

Unit 2 Brute Force and Divide & Conquer

L-9 Hours

Brute Force: Closest-Pair and convex-Hull Problems- Exhaustive Search - Traveling Salesman Problem - Knapsack Problem - Assignment problem. Divide and conquer methodology: Merge sort – Quick sort – Binary search- Heap sort- Closest-Pair and convex-Hull Problems

(Case Study: Brute Force: Solve the hide and seek game with a search time limit as 5 minutes and check its efficiency.)

(Case Study: Divide & Conquer: A king with a huge kingdom struggles to oversee from a single central city. Suggest how the king will handle the local problems of the smaller parts of the kingdom).

Unit 3 Dynamic Programming and Greedy Technique

L-9Hours

Dynamic Programming: Principle of optimality - Computing Binomial Coefficient - Warshall’s and Floyd’ algorithm – Knapsack Problem. Greedy Technique: Minimum spanning tree-Prim’s algorithm- Kruskal's Algorithm- Huffman Trees and Codes- Graph algorithms: Topological sort- Dijkstra's Algorithm.

Case Study: Dynamic Programming: A Taxi Operator firm needs to design a growth strategy plan that should focus on the revenue for the next 25 years considering various constraints like car turns old, reduction in mileages etc)

(Case Study: Greedy Technique: Assume the list of arrival and departure time; find the minimum number of platforms required for the railway as no train waits).

Unit 4 Backtracking and Branch & Bound

L-9Hours

Backtracking: n-Queens problem- Hamiltonian Circuit Problem- Subset sum problem-Branch & Bound: Assignment problem-Knapsack Problem- Traveling Salesman Problem

(Case Study: Backtracking: Solve a constraint network to find an assignment of values for each variable so that all constraints are satisfied.)

(Case Study: Branch & Bound: Optimize the tour to deliver the packages at randomly chosen pizza centres in the city of Chennai.)

Unit 5 Iterative Improvements and Limitations Of Algorithm Power

L-9Hours

The simplex method– The maximum-flow problem– The Maximum matching in bipartite graph- Introduction to unsolvable problem-Halting problem. Limitations of Algorithm Power--Decision Trees- Lower bound arguments- P, NP and NP-Complete Problems. The Classes NP and P.

(**Case Study:** Identify solution to the telecommunication networks to handle the crucial problems such as flow control and routing)

Total: 60 Hours

H. Learning Resources

i.Text Books:

1. Anany Levitin, “Introduction to the Design and Analysis of Algorithms”, Third Edition, Pearson Education, 2012. [Unit 1, 2 & 4]
2. Horowitz, S. Sahni, and S. Rajasekaran, “Computer Algorithms,” 2nd Edition, Galgotia Publications, 2007. [Unit 3 & 5]

ii.Reference Books:

1. Alfred V. Aho, John E. Hopcroft and Jeffrey D. Ullman, —Data Structures and Algorithms, Pearson Education, Reprint 2017.
2. Thomas H.Cormen, Charles E.Leiserson, Ronald L. Rivest and Clifford Stein, “Introduction to Algorithms”, Third Edition, PHI Learning Private Limited, 2012.
3. Donald E. Knuth, “The Art of Computer Programming”, Volumes 1&3 Pearson Education, 2009.

iii.Online References:

1. “Design and analysis of algorithms”, Jan 21 2015, Accessed on: Jan 18 2021, [Online]. Available: https://onlinecourses.nptel.ac.in/noc21_cs22/preview
2. "Algorithms Specialization", Accessed on: April 21 2021, [Online], Available: <https://www.coursera.org/specializations/algorithms>

COURSE CODE	COURSE TITLE	L	T	P	C
10213CS108	Big Data Analytics	3	0	0	3

A. Preamble

This course provides an understanding of Big Data concepts, distributed data processing, and organizational data management. It introduces Hadoop, MapReduce, NoSQL databases, Pig, Hive, and real-time data processing using Spark and Kafka. Students gain practical, industry-relevant skills to manage and analyze large-scale datasets effectively.

B. Prerequisite Course – Nil

C. Course Objectives

Learners are exposed to:

- Provide students with a strong foundation in Big Data concepts and its significance in modern organizations.
- Familiarize students with the tools, frameworks, and technologies used for storing and processing large-scale data.
- Develop the ability to design and plan scalable data management strategies for organizational data.
- Cultivate an understanding of real-time data processing, streaming analytics, and emerging trends in Big Data technologies.
- Prepare students for industry-relevant challenges by integrating practical exposure, case studies, and contemporary applications.

C. Course Outcomes

Upon the successful completion of the course, students will be able to:

COs	Course Outcomes	K - Level
CO1	Understand the fundamentals of Big Data, its characteristics, challenges, and applications in organizations.	K2
CO2	Illustrate the architecture, components, and data processing mechanisms of Hadoop and MapReduce.	K2
CO3	Implement NoSQL concepts using MongoDB and Cassandra to manage large-scale datasets.	K3
CO4	Utilize Pig and Hive for efficient querying and data processing of large datasets.	K3
CO5	Design real-time data processing workflows using Spark Streaming, Kafka, and streaming analytics.	K3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3		3	3							
CO2		2	2	2	3						
CO3	2		3		3	2					
CO4	2	3		3	3	3					
CO5	2	2		3		3					

D. Course Contents

UNIT I – Introduction To Big Data

L-9 Hours

Introduction – Classification of Digital Data - Distributed file system – Big Data and its importance - Four V's - Big data applications - Traditional Business Intelligence (BI) versus Big Data - Challenges of Big Data – **Role of Big Data in Modern Organizations – Data Governance and Organizational Data Management.**

UNIT II – Big Data Using Hadoop & Map Reduce

L-9 Hours

Introduction to Hadoop – RDBMS vs Hadoop – History and Overview of Hadoop – Use Cases – Hadoop Distributed File System (HDFS) – Processing Data with Hadoop – Resource Management with YARN (Yet Another Resource Negotiator) – Introduction to MapReduce Programming – Mapper, Reducer, Combiner, Partitioner – Data Searching, Sorting, and Compression – Organizational Data Processing Scenarios using Hadoop.

UNIT III – Nosql, Mongoddb & Cassandra

L-9 Hours

NoSQL Databases: Types and Advantages – SQL vs NoSQL vs NewSQL – Introduction to MongoDB – JSON Data Format – CRUD Operations – Data Modeling and Data Types in MongoDB – MapReduce Functions in MongoDB – Introduction to Cassandra: Features, CQL, CQLSH, CRUD Operations – Comparison of SQL, NoSQL, and NewSQL for Organizational Data Management.

UNIT IV – Pig And Hive

L-9Hours

Introduction to Hive: Architecture, Data Types, File Formats, Hive Query Language (HQL), RC File Implementation, User Defined Functions (UDF) – Introduction to Pig: Anatomy, Pig on Hadoop, Pig Philosophy, Pig Latin Overview, Data Types, Execution Modes, HDFS Commands, Relational Operators, Eval Functions, Complex Data Types, Piggy Bank, User Defined Functions – Comparative Analysis: Pig vs Hive – Applications in Organizational Data Analytics.

UNIT V - Spark, Kafka, and Streaming Analytics

L-9 Hours

Introduction to Apache Spark: Evolution, Features, Spark Built on Hadoop, Components, Resilient Distributed Datasets (RDDs), Iterative Operations, Data Sharing using Spark RDD, Interactive Operations – **Introduction to Kafka: Architecture, Producers, Consumers, Topics,**

Partitions, Brokers, and Zookeeper – Streaming Data with Spark
Streaming and Kafka Integration – Real-Time Data Processing – Applications of Streaming
Analytics in Organizations for Event Monitoring, Predictive Analytics, and Decision Making.

E. Learning Resources

i. Text Books:

1. Seema Acharya and Subhashini Chellappan “Big data and Analytics” Wiley India Publishers, 2nd Edition, 2019.
2. Rajkamal and Preeti Saxena, “Big Data Analytics, Introduction to Hadoop, Spark and Machine Learning”, McGraw Hill Publication, 2019.

ii. Reference Books:

1. Adam Shook and Donald Mine, “MapReduce Design Patterns: Building Effective Algorithms and Analytics For Hadoop and Other Systems” - O'Reilly 2012
2. Tom White, “Hadoop: The Definitive Guide” 4th Edition, O'reilly Media, 2015.
3. Thomas Erl, Wajid Khattak, and Paul Buhler, Big Data Fundamentals: Concepts, Drivers & Techniques, Pearson India Education Service Pvt. Ltd., 1st Edition, 2016
4. John D. Kelleher, Brian Mac Namee, Aoife D'Arcy -Fundamentals of Machine Learning for Predictive Data Analytics: Algorithms, Worked Examples, MIT Press 2020, 2nd Edition

iii. Web Resources:

1. “Big data Analytics”, Accessed on Nov 8, 2025 [online]. Available: <https://www.geeksforgeeks.org/data-engineering/what-is-big-data-analytics/>
2. “Apache Spark Tutorial”, Accessed on Nov 8, 2025 [online]. Available: https://www.tutorialspoint.com/apache_spark/apache_spark_deployment.htm
3. “MongoDB Tutorial”, Accessed on Nov 8, 2025 [online]. Available: <https://www.w3schools.com/mongodb/>
4. “Pig and Hive”, Accessed on Nov 8, 2025 [online]. Available: <https://www.geeksforgeeks.org/dbms/difference-between-pig-and-hive/>

COURSE CODE	COURSE TITLE	L	T	P	C
10213GE103	SOFTWARE DEVELOPMENT AND MAINTENANCE	3	0	0	3

(This course is offered under General Elective Category. All School students can opt the course.)

A. Preamble

This course provides basic concepts about SDLC, Unix commands and shell operations, Object Oriented Programming, JavaScript technologies to create the interactive client-side design of web applications. This course introduces DBMS concepts, to share a common set of models, design paradigms via a Structured Query Language (SQL). More specifically, this course also deals with UI/UX concepts to develop a pleasant user interface and deploy the application using Cloud AWS services.

B. Prerequisite Course

10210CS101 - Problem Solving using C

C. Course Objectives

Learners are exposed to:

- Make use of basic SDLC, Unix Commands.
- Practice Java basic constructs for real world problems.
- Build efficient client-side validation using javascript.
- Acquire knowledge on basic SQL queries and designing database schema
- Learn and build the workflow of UI/UX
- Deploy the application on AWS cloud

D. Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Interpret the Software Development Life Cycle (SDLC) stages, Unix basics commands and shell operations, and OOP.	K2
CO2	Articulate proficiency in exception handling, inheritance, and API fundamentals.	K3
CO3	Design the interactive and dynamic webpage using JavaScript.	K3
CO4	Model the Relation Database schema and perform CRUD operations.	K3
CO5	Chart UI/UX and deploy the applications using AWS cloud.	K3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	3	3	2									
CO2	3	3	3	3	2									
CO3	3	3	3	3	2									
CO4	3	3	3	3	2									
CO5	3	3	3	3	2									

High- 3; Medium-2; Low-1

F. Course Contents

Unit 1 Introduction to SDLC, UNIX/OS and OOP Concepts

L-9 Hours

SDLC: Requirements gathering - System design - Implementation - Testing - Various types of testing - Maintenance - Documentation - Project Management - Agile. UNIX/OS: Unix Basics - Unix shell - File Operations - Process Management - Networking and Communication - System Administration - Shell Customization. OOPs Concept: Benefits and Multiple Features of OOP - Object Oriented Programming Vs Procedural Programming - Class - Cohesion and Coupling in design process - Popular OOP Languages.

Unit 2 Exception Handling, Inheritance & APIs

L-9 Hours

Exception Handling - Categories of Exceptions - Hierarchy of Exception Class - Exception Types - Checked and Unchecked Exceptions. Inheritance and its Types - Creating Subclass - Method Overriding - Super keyword - Polymorphism - instanceof operator - Abstract Vs Non-Abstract classes - Interface. **API:** Introduction - API Basics - Types of API Protocols - Styles and Benefits of API Development.

Unit 3 Scripting Language - Javascript

L-9 Hours

Basic Web Development Concepts - **Javascript:** Introduction to Javascript - Syntax and Basic Concepts - Document Object Model (DOM) Manipulation - Arrays and Objects - Asynchronous JavaScript - Error Handling and Debugging - ECMAScript 6 (ES6)+ Features.

Unit 4 Database Fundamentals: RDBMS, SQL

L-9 Hours

DBMS: History of Data and its definition - Various formats to recognize databases - DBMS through MySQL Workbench - DBMS Types and its features - Requirements of DBMS - Roles, job-related tasks.

RDBMS: Introduction and its uses - popular RDBMS and their uses - Oracle RDBMS versions and features - SQL Server RDBMS, history and its features - MySQL RDBMS, history and its features - MySQL Workbench - Normalization - E.F. Codd.

SQL: Introduction - Connectivity - Reserved words - Data Types - SQL Commands: DDL - Alter tables - drop and truncate - table constraints - constraint violation.

Unit 5 Fundamentals of UI/UX, Generative AI and AWS Cloud Practitioner L-9 Hours

UI/UX: Introduction - importance - UI Vs UX - Future of UI/UX - User prevention on errors - flowcharts of UI/UX - journey maps of UX design - micro interactions - age-responsive design - interstitial anxiety - VR interfaces Vs normal interfaces - Modern concepts of UI/UX.

Generative AI: Definition - Working principle - Model Types - Applications.

AWS Cloud Practitioner: Introduction - AWS services - SDKs Vs APIs - AWS Management Console - Infrastructure as Code (IaC) - Deployment and Management - Serverless Computing - Data Storage and Analytics.

Total: 45 Hours

G.Learning Resources

i.Text Books:

1. "Software Engineering: A Practitioner's Approach" by Roger S. Pressman is published by McGraw-Hill Education. September 2021 , 9th edition. [Unit - 1]
2. "UNIX and Linux System Administration Handbook" is published by Pearson and the 5th edition was published in 2017. [Unit - 1]
3. "Introduction to Java Programming and Data Structures, Comprehensive Version" (12th Edition) By Y. Daniel Liang, Pearson in 2018. [Unit - 1 & 2]
5. "APIs: A Strategy Guide" by Daniel Jacobson, Greg Brail, and Dan Woods, O'Reilly Media, Inc. December 2011. [Unit - 2]
6. "JavaScript: The Definitive Guide, 7th Edition." by David Flanagan. Released May 2020. Publisher(s): O'Reilly Media, Inc. [Unit - 3]
7. "Database System Concepts" by Abraham Silberschatz, Henry F. Korth, and S. Sudarshan 7th edition McGraw-Hill Education. [Unit - 4]
8. "Don't Make Me Think: A Common Sense Approach to Web Usability" by Steve Krug 3rd edition by New Riders. [Unit - 5]
9. "AWS Certified Cloud Practitioner Study Guide" 2nd Edition by Sybex. [Unit - 5]

ii.Reference Books:

1. H.M. Deitel and P.J. Deitel, "Java How to Program", Pearson Prentice Hall Seventh edition, 2018.
2. E. Balaguruswamy, "Programming in java", Sixth Edition, Tata McGraw Hill, 2019.
3. Mary Delamater, "Murach's JavaScript and jQuery (4th Edition) Paperback – Illustrated", Mike Murach & Associates Inc, August 26, 2020.
4. Elmasri Ramez, Navathe S, "Fundamentals of Database System", Seventh Edition, Pearson, 2017.
5. "The Design of Everyday Things: Revised and Expanded Edition" by Don Norman.
6. "Learning Amazon Web Services (AWS): A Hands-On Guide to the Fundamentals of AWS Cloud" by Mark Wilkins.

iii. Online References:

1. “Java Tutorial”, June 20, 2023 [Online], Available: <https://www.javatpoint.com/java-tutorial>
2. “Programming in Java”, Accessed on: June 20, 2023 [Online], Available: <https://nptel.ac.in/courses/106/105/106105191/>
3. “UI / UX Design Tutorial – Wireframe, Mockup & Design in Figma “
https://www.youtube.com/watch?v=c9Wg6Cb_YIU
4. Get started with Amazon EC2 Linux instances
https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/EC2_GetStarted.html
5. “ AWS EC2 Instances | How to Create AWS EC2 Instance | AWS EC2 Tutorial For Beginner” <https://www.youtube.com/watch?v=LZXWIF5udYs>

COURSE CODE	COURSE TITLE	L	T	P	C
10213GE302	PROGRAMMING CHALLENGES	0	0	4	2

A. Preamble

This course will empower students with a strong foundation in problem-solving, algorithmic thinking, coding proficiency, and code optimization. This course helps, students to participate in live coding challenges on platforms like Code chef, hacker earth, code forces, enhancing their problem-solving skills and increasing their competitiveness in technical interviews and coding competitions.

B. Prerequisite Course:

10210CS101-Problem solving using C

C. Course Objectives

Learners are exposed to:

- Master algorithmic thinking and the design of efficient algorithms.
- Learn strategies for competitive coding and performing well under time constraints.
- Understand the importance of code optimization for real-world applications.

D.Course Outcomes

Upon the successful completion of the course, students will be able to:

CONo's	Course Outcomes	K - Level
CO1	Apply algorithms to solve real-world coding challenges.	K3
CO2	Develop effective coding and debugging skills in time-sensitive scenarios.	K3
CO3	Identify optimized algorithms to efficiently solve complex problems in coding challenges.	K3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of COs with POs:

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PS O1	PS O2	PS O3
CO1	3	3	3	3							3			
CO2	3	3	3	3							3			
CO3	3	3	3	3							3			

3 - High; 2 - Medium; 1 – Low

F. Course Contents

This programming challenges course facilitates practical, hands-on experiences for students, empowering them to apply their knowledge and enhance their problem-solving skills. One of the key components of this course involves participating in live coding challenges on platforms like CodeChef, hacker earth, leet code. These challenges are designed to test the ability of the student to solve real-world problems and to reinforce the concepts and skills learned by the student throughout the course.

Total: 60 Hours**G.Learning Resources****Online Resources:**

1. <https://www.hackerrank.com>
2. <https://leetcode.com>
3. <https://www.geeksforgeeks.org>
4. <https://www.codechef.com>

COURSE CODE	COURSE TITLE	L	T	P	C
10213GE204	APPLICATION DEVELOPMENT AND MAINTENANCE	3	0	2	4

A. Preamble

This course provides basic concepts about SDLC, Unix commands and shell operations, Object Oriented Programming, JavaScript technologies to create the interactive client-side design of web applications. This course introduces DBMS concepts, to share a common set of models, design paradigms via a Structured Query Language (SQL). More specifically, this course also deals with UI/UX concepts to develop a pleasant user interface and deploy the application using Cloud AWS services.

B. Prerequisite Course

10210CS101 - Problem Solving using C.

C. Course Objectives

Learners are exposed to:

- Make use of basic SDLC, Unix Commands.
- Practice Java basic constructs for real world problems.
- Build efficient client-side validation using javascript.
- Acquire knowledge on basic SQL queries and designing database schema
- Learn and build the workflow of UI/UX
- Deploy the application on AWS cloud

D. Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Interpret the Software Development Life Cycle (SDLC) stages, Unix basics commands and shell operations, and OOP.	K2
CO2	Articulate proficiency in exception handling, inheritance, and API fundamentals.	K3
CO3	Design the interactive and dynamic webpage using JavaScript.	K3
CO4	Model the Relation Database schema and perform CRUD operations.	K3
CO5	Chart UI/UX and deploy the applications using AWS cloud.	K3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of COs with POs:

COs	P O 1	P O 2	P O 3	P O 4	P O 5	P O 6	P O 7	P O 8	P O 9	P O 10	P O 11	P S O 1	P S O 2	P S O 3
CO1	3	3	3	3	2						3			
CO2	3	3	3	3	2						3			
CO3	3	3	3	3	2						3			
CO4	3	3	3	3	2						2			
CO5	3	3	3	3	2						2			

3 - High; 2 - Medium; 1 - Low

F. Course Contents**Unit 1 Introduction to SDLC, UNIX/OS and OOP Concepts****L-9 Hours**

SDLC: Requirements gathering - System design - Implementation - Testing - Various types of testing - Maintenance - Documentation - Project Management - Agile. UNIX/OS: Unix Basics - Unix shell - File Operations - Process Management - Networking and Communication - System Administration - Shell Customization. OOPs Concept: Benefits and Multiple Features of OOP - Object Oriented Programming Vs Procedural Programming - Class - Cohesion and Coupling in design process - Popular OOP Languages.

Unit 2 Exception Handling, Inheritance & APIs**L-9 Hours**

Exception Handling - Categories of Exceptions - Hierarchy of Exception Class - Exception Types - Checked and Unchecked Exceptions. Inheritance and its Types - Creating Subclass - Method Overriding - Super keyword - Polymorphism - instanceof operator - Abstract Vs Non-Abstract classes - Interface. **API:** Introduction - API Basics - Types of API Protocols - Styles and Benefits of API Development.

Unit 3 Scripting Language - Javascript**L-9 Hours**

Basic Web Development Concepts - **Javascript:** Introduction to Javascript - Syntax and Basic Concepts - Document Object Model (DOM) Manipulation - Arrays and Objects - Asynchronous JavaScript - Error Handling and Debugging - ECMAScript 6 (ES6)+ Features.

Unit 4 Database Fundamentals: RDBMS, SQL**L-9 Hours**

DBMS: History of Data and its definition - Various formats to recognize databases - DBMS through MySQL Workbench - DBMS Types and its features - Requirements of DBMS - Roles, job-related tasks.

RDBMS: Introduction and its uses - popular RDBMS and their uses - Oracle RDBMS versions and features - SQL Server RDBMS, history and its features - MySQL RDBMS, history and its features - MySQL Workbench - Normalization - E.F. Codd.

SQL: Introduction - Connectivity - Reserved words - Data Types - SQL Commands: DDL - Alter tables - drop and truncate - table constraints - constraint violation.

Unit 5 Fundamentals of UI/UX, Generative AI and AWS Cloud Practitioner**L-9 Hours**

UI/UX: Introduction - importance - UI Vs UX - Future of UI/UX - User prevention on errors - flowcharts of UI/UX - journey maps of UX design - micro interactions - age-responsive design - interstitial anxiety - VR interfaces Vs normal interfaces - Modern concepts of UI/UX.

Generative AI: Definition - Working principle - Model Types - Applications.

AWS Cloud Practitioner: Introduction - AWS services - SDKs Vs APIs - AWS Management Console - Infrastructure as Code (IaC) - Deployment and Management - Serverless Computing - Data Storage and Analytics.

Total: 45 Hours

G.Laboratory Experiments

Total: 30 Hours

Part – 1

Task 1: Working with Unix OS Commands.

[OS: UNIX, LINUX, IDE: Terminal]

Task2. Develop a Simple Java application with class (Object), Methods and Packages.

[Package: java.io, java.util, IDE: Eclipse]

Task 3: Inheritance & Interfaces

- Demonstrate Method overloading, constructor and Method Overriding.
- Develop a code for user-defined interface and pre-defined interface concept

[Package: java.io, java.util, IDE: Eclipse]

Task 4: Exception Handling Mechanism in Java

- Construct a program to handle user defined and predefined exceptions
- Design the code with performing Checked and Unchecked Exceptions

[Package: java.io, java.lang, IDE: Eclipse]

Task 5: Conceptual Design after FTR

Using basic database design methodology and ER modeler, design Entity

Relationship Diagram by satisfying the following sub tasks:

- Identifying the entities.
- Identifying the attributes.
- Identification of relationships, cardinality, type of relationship.
- Reframing the relations with keys and constraints.

[Database: MySQL]

Task6: Generating design of another traditional database model

Creating Hierarchical /Network model of the database by enhancing the sound abstract data by performing following tasks using forms of inheritance:

- Identify the specificity of each relationship, find, and form surplus relations.
- Check is-a hierarchy/ has-a hierarchy and performs generalization and/or specialization relationship.
- Find the domain of the attribute and perform a check constraint to the applicable.
- Rename the relations.
- Perform SQL Relations using DDL, DML, DCL commands

[Database: MySQL]

Task7. Construct a dynamic webpage using JavaScript.

[IDE: Visual Studio]

Task8. Manipulating the DOM with javascript

[IDE: Visual Studio]

Task 9: UI/UX Design

- Design a UI for the home page using figma tool.

[Tool: figma]

Task 10: AWS Cloud Practitioner

Deployment and Management

a) Choose a sample web application.

b) Deploy the application on an EC2 instance.

c) Outline the steps taken and any challenges faced during deployment.

[Services: AWS Cloud]

Part– 2**Use cases:****Use case-1: Develop Online learning platform**

An educational technology company is planning to develop an online learning platform to offer programming courses to students. The platform aims to provide a seamless learning experience.

Technologies Used: JavaScript, UI/UX principle, Generative AI, AWS Cloud services.

Use case-2: Tracking the student marks statement:

In the college, every month the assessment test will be scheduled and results will be updated in the portal. Create the web page with username and password of each student and with admin controls. The student can track the test marks by applying their login credentials in the server page and get the exact marks, when the test is attended and when the marks are updated in the portal and result status. Demonstrate the web page with frontend and backend control to track the mark statements and design the interface to make it user friendly.

Technologies Used: JavaScript, UI/UX principle, Generative AI, AWS Cloud services.

Total: 75 Hours**G.Learning Resources****i.Text Books:**

1. "Software Engineering: A Practitioner's Approach" by Roger S. Pressman is published by McGraw-Hill Education. September 2021 , 9th edition. [Unit - 1]
2. "UNIX and Linux System Administration Handbook" is published by Pearson and the 5th edition was published in 2017. [Unit - 1]

3. "Introduction to Java Programming and Data Structures, Comprehensive Version" (12th Edition) By Y. Daniel Liang, Pearson in 20
18. [Unit - 1 & 2]
5. APIs: A Strategy Guide" by Daniel Jacobson, Greg Brail, and Dan Woods, O'Reilly Media, Inc. December 2011. [Unit - 2]
6. JavaScript: The Definitive Guide, 7th Edition. by David Flanagan. Released May 2020. Publisher(s): O'Reilly Media, Inc. [Unit - 3]
7. Database System Concepts" by Abraham Silberschatz, Henry F. Korth, and S. Sudarshan 7th edition McGraw-Hill Education. [Unit - 4]
8. "Don't Make Me Think: A Common Sense Approach to Web Usability" by Steve Krug 3rd edition by New Riders. [Unit - 5]
9. AWS Certified Cloud Practitioner Study Guide" 2nd Edition by Sybex. [Unit - 5]

ii. Reference Books:

1. H.M. Deitel and P.J. Deitel, "Java How to Program", Pearson Prentice Hall Seventh edition, 2018.
2. E. Balaguruswamy, "Programming in java", Sixth Edition, Tata McGraw Hill, 2019.
3. Mary Delamater, "Murach's JavaScript and jQuery (4th Edition) Paperback – Illustrated", Mike Murach & Associates Inc, August 26, 2020.
4. Elmasri Ramez, Navathe S, "Fundamentals of Database System", Seventh Edition, Pearson, 2017.
5. The Design of Everyday Things: Revised and Expanded Edition by Don Norman.
6. Learning Amazon Web Services (AWS): A Hands-On Guide to the Fundamentals of AWS Cloud by Mark Wilkins.

iii. Online References:

1. "Java Tutorial", June 20, 2023 [Online], Available: <https://www.javatpoint.com/java-tutorial>
2. "Programming in Java", Accessed on: June 20, 2023 [Online], Available: <https://nptel.ac.in/courses/106/105/106105191/>
3. "UI / UX Design Tutorial – Wireframe, Mockup & Design in Figma " https://www.youtube.com/watch?v=c9Wg6Cb_YIU
4. Get started with Amazon EC2 Linux instances https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/EC2_GetStarted.html
5. "AWS EC2 Instances | How to Create AWS EC2 Instance | AWS EC2 Tutorial For Beginner" <https://www.youtube.com/watch?v=LZXWIF5udYs>

OPEN ELECTIVE

(Self Learning Courses)

COURSE CODE	COURSE TITLE	W	H	C
10213CS403	Fuzzy Sets, Logic and Systems & Applications	12	-	3

Course Category:

Open Elective (Self - Learning Course)

Course Contents:

Week 1: Introduction and Fuzzy Sets Theory

Week 2: Membership Functions

Week 3: Set Theoretic Operations

Week 4: Fuzzy Arithmetic

Week 5: Fuzzy Relations

Week 6: Fuzzy Inference Systems I

Week 7: Fuzzy Inference Systems II

Week 8: Wang and Mendel Model

Week 9: TSK Model

Week 10: Fuzzifiers and Defuzzifiers

Week 11: ANFIS Architecture

Week 12: Fuzzy Systems and Machine Learning

Course Provider : NPTEL
Professor : Prof. Nishchal Kumar Verma
University : IIT Kanpur

COURSE CODE	COURSE TITLE	W	H	C
10213CS431	Business Intelligence & Analytics	12	-	3

Course Contents

Week 1: Introduction to Business Intelligence & Analytics (BIA), drivers of BIA, types of analytics: descriptive to prescriptive, vocabulary of business analytics, course plan and resources

Week 2: Technical architecture of BIA, case analysis of AT&T Long distance, F management, OnLine Transaction Processing (OLTP), design process of databases

Week 3: Relational databases, normalisation, SQL queries, ShopSense case of management questions, data warehousing, OnLine Analytical Processing (OLAP), data cube

Week 4: Descriptive analytics, and visualization, customer analytics, survival analysis, customer lifetime value, case study

Week 5: Data mining process, introduction to statistical learning, data pre-processing, data quality, overview of data mining techniques, case study using regression analysis

Week 6: Introduction to classification, classification techniques, scoring models, classifier performance, ROC and PR curves

Week 7: Introduction to decision trees, tree induction, measures of purity, tree algorithms, pruning, and ensemble methods

Week 8: Tree implementation in Python: problem of targeted mailing

Week 9: Cluster analysis, measures of distance, clustering algorithms, K-means and other techniques, cluster quality

Week 10: A store segmentation case study using clustering, implementation in Python, profiling clusters, cluster interpretation and actionable insights, RFM sub-segmentation for customer loyalty

Week 11: Machine learning, Artificial Neural Networks (ANN), topology and training algorithms, back propagation, financial time series modelling using ANN, implementation in Python

Week 12: Text mining, process, key concepts, sentiment scoring, text mining using R-the case of a movie discussion forum, summary

Course Provider : NPTEL
Professor : Prof. Saji K Mathew
University : IIT Madras

Academic Year: 2024-2025

Semester : Summer

COURSE CODE	COURSE TITLE	W	H	C
10213CS442	SOCIAL NETWORKS	12	-	3

CourseCategory:

Independent Learning (Self - Learning Course)

Course Contents**Week 1:** Introduction**Week 2:** Handling Real-world Network Datasets**Week 3:** Strength of Weak Ties**Week 4:** Strong and Weak Relationships (Continued) & Homophily**Week 5:** Homophily Continued and +Ve / -Ve Relationships**Week 6:** Link Analysis**Week 7:** Cascading Behaviour in Networks**Week 8:** Link Analysis (Continued)**Week 9:** Power Laws and Rich-Get-Richer Phenomena**Week 10:** Power law (contd..) and Epidemics**Week 11:** Small World Phenomenon**Week 12:** Pseudocore (How to go viral on web)**CourseProvider** : NPTEL**Professor** : Prof. Sudarshan Iyengar

Academic Year: 2024-2025

Semester : Summer

COURSE CODE	COURSE TITLE	W	H	C
10213CS443	INTRODUCTION TO INDUSTRY 4.0 AND INDUSTRIAL INTERNET OF THINGS	12	-	3

CourseCategory:

Independent Learning (Self - Learning Course)

Course Contents

Week 1 : Introduction: Sensing & actuation, Communication-Part I, Part II, Networking-Part I, Part II

Week 2 : Industry 4.0: Globalization and Emerging Issues, The Fourth Revolution, LEAN Production Systems, Smart and Connected Business Perspective, Smart Factories

Week 3 : Industry 4.0: Cyber Physical Systems and Next Generation Sensors, Collaborative Platform and Product Lifecycle Management, Augmented Reality and Virtual Reality, Artificial Intelligence, Big Data and Advanced Analysis

Week 4 : Cybersecurity in Industry 4.0, Basics of Industrial IoT: Industrial Processes-Part I, Part II, Industrial Sensing & Actuation, Industrial Internet Systems.

Week 5 : IIoT-Introduction, Industrial IoT: Business Model and Reference Architecture: IIoT-Business Models-Part I, Part II, IIoT Reference Architecture-Part I, Part II.

Week 6 : Industrial IoT- Layers: IIoT Sensing-Part I, Part II, IIoT Processing-Part I, Part II, IIoT Communication-Part I.

Week 7 : Industrial IoT- Layers: IIoT Communication-Part II, Part III, IIoT Networking-Part I, Part II, Part III.

Week 8 : Industrial IoT: Big Data Analytics and Software Defined Networks: IIoT Analytics - Introduction, Machine Learning and Data Science - Part I, Part II, R and Julia Programming, Data Management with Hadoop.

Week 9 : Industrial IoT: Big Data Analytics and Software Defined Networks: SDN in IIoT-Part I, Part II, Data Center Networks, Industrial IoT: Security and Fog Computing: Cloud Computing in IIoT-Part I, Part II.

Week 10 : Industrial IoT: Security and Fog Computing - Fog Computing in IIoT, Security in IIoT-Part I, Part II, Industrial IoT- Application Domains: Factories and Assembly Line, Food Industry.

Week 11 : Industrial IoT- Application Domains: Healthcare, Power Plants, Inventory Management & Quality Control, Plant Safety and Security (Including AR and VR safety applications), Facility Management.

Week 12 : Industrial IoT- Application Domains: Oil, chemical and pharmaceutical industry, Applications of UAVs in Industries, Real case studies :

Case study - I : Milk Processing and Packaging Industries

Case study - II: Manufacturing Industries - Part I

Case study - III : Manufacturing Industries - Part II

Case study - IV : Student Projects - Part I

Case study - V : Student Projects - Part II

Case study - VI : Virtual Reality Lab

Case study - VII : Steel Technology Lab

Academic Year: 2024-2025

Semester : Summer

COURSE CODE	COURSE TITLE	W	H	C
10213CS445	PYTHON FOR DATA SCIENCE	4	-	1

Course Category:

Independent Learning (Self - Learning Course)

Course Contents**Week 1:****BASICS OF PYTHON SPYDER (TOOL)**

- Introduction Spyder
- Setting working Directory
- Creating and saving a script file
- File execution, clearing console, removing variables from environment, clearing environment
- Commenting script files
- Variable creation
- Arithmetic and logical operators
- Data types and associated operations

Week 2:**Sequence data types and associated operations**

- Strings
- Lists
- Arrays
- Tuples
- Dictionary
- Sets
- Range

NumPy

- ndArray

Week 3:**Pandas dataframe and dataframe related operations on Toyota Corolla dataset**

1. Reading files
2. Exploratory data analysis
3. Data preparation and preprocessing

Data visualization on Toyoto Corolla dataset using matplotlib and seaborn libraries

1. Scatter plot
2. Line plot
3. Bar plot
4. Histogram
5. Box plot
6. Pair plot

Control structures using Toyota Corolla dataset

1. if-else family
2. for loop
3. for loop with if break
4. while loop

Functions**Week 4: CASE STUDY****Regression**

1. Predicting price of pre-owned cars

Classification

1. Classifying personal income

CourseProvider	:	NPTEL
Professor	:	Prof Ragunathan Rengasamy
University	:	IIT Madras

Academic Year: 2024-2025

Semester : Summer

COURSE CODE	COURSE TITLE	W	H	C
10213CS446	Google Cloud Computing Foundations	8	-	2

CourseCategory: Independent Learning (Self - Learning Course)**Course Contents**

Week 0 : Introduction to the course

Week 1 : So, What's the Cloud anyway? Start with a Solid Platform

Week 2 : Use GCP to build your Apps

Week 3 : Where do I store this stuff?

Week 4 : There's an API for that! You can't secure the Cloud right?

Week 5 : It helps to network!

Week 6 : It helps to network (continued)

Week 7 : Let Google keep an eye on things. You have the data, but what are you doing with it?

Week 8 : Let machines do the work

CourseProvider : NPTEL**Professor** : Prof. Soumya Kanti Ghosh**University** : IIT Kharagpur

Academic Year: 2024-2025

Semester : Summer

COURSE CODE	COURSE TITLE	W	H	C
10213CS444	RESPONSIBLE & SAFE AI SYSTEMS	12	-	3

CourseCategory:

Independent Learning (Self - Learning Course)

Course Contents**Week 1 & 2:**

AI Capabilities Improvement in last 5-10 years

- Imminent risks from AI Models: Toxicity, bias, goal misspecification, adversarial examples etc.
- Long-term risks from AI Models: Misuse, Misgeneralization, Rogue AGI
- Principles of RAI - Transparency; Accountability; Safety, Robustness and Reliability; Privacy and Security; Fairness and non-discrimination; Human-Centred Values; Inclusive and Sustainable development, Interpretability
- Recap of Deep Learning Techniques, Language/Vision Models
- AI Risks for Gen models
- Adversarial Attacks – Vision, NLP, Superhuman Go agents

Week 3 & 4:

- ML Poisoning Attacks like Trojans
- Implications for current and future AI safety
- Explainability
- Imminent and Long-term potential for transparency techniques
- Mechanistic Interpretability
- Representation Engineering, model editing and probing
- Critiques of Transparency for AI Safety

Week 5 & 6:

Privacy & Fairness in AI

Week 7 & 8:

- Metrics and Tools for RAI - measuring bias/fairness, adversarial testing, explanations (Lime/SHAP/GradCam), audit mechanisms

- Regulation landscape - DPDP act (India), GDPR (EU), EU AI act, US presidential declaration, Ethical approvals, informed consent, participatory design, future of work, Indian context
- What is AGI? When could it be achieved?
- Instrumental Convergence: Power Seeking, Deception etc.

Week 9 & 10:

- RAI in Legal domain
- RAI in Health care domain
- RAI in Education domain
- A few other domains
- Policy issues in RAI

Week 11 & 12:

- Couple of panel discussion with industry practitioners, academic, government (possibly), and others.
- Fireside chat with eminent personalities
- Recorded Paper reading discussion

CourseProvider : NPTEL
Professor : Prof Ponnurangam Kumaraguru
 Prof Balaraman Ravindran
 Prof Arun Rajkumar
University : IIIT Hyderabad
 IIT Madras

Academic Year: 2024-2025

Semester : Winter

COURSE CODE	COURSE TITLE	W	H	C
10213CS448	Data Analytics with Python	12	-	3

Course Category:

Open Elective (Self- Learning Course)

Course Outcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Understand and apply data analytics fundamentals	K3
CO2	Analyze sampling and distributions	K3
CO3	Perform hypothesis testing for decision-making	K3
CO4	Model relationships through regression	K3
CO5	Explore advanced regression techniques	K2

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	2	3	2	2	1	2	1					2	3
CO2	2	3	3	2	1	2	1					2	3
CO3	2	3	3	2	1	2	1					2	2
CO4	3	3	2	3	2	2	1					3	3
CO5	2	3	2	3	3	1	1					2	3

Course Contents:

Week 1	:	Introduction to data analytics and Python fundamentals
Week 2	:	Introduction to probability
Week 3	:	Sampling and sampling distributions
Week 4	:	Hypothesis testing
Week 5	:	Two sample testing and introduction to ANOVA
Week 6	:	Two way ANOVA and linear regression
Week 7	:	Linear regression and multiple regression
Week 8	:	Concepts of MLE and Logistic regression
Week 9	:	ROC and Regression Analysis Model Building
Week 10	:	C ² Test and introduction to cluster analysis
Week 11	:	Clustering analysis
Week 12	:	Classification and Regression Trees (CART)

Professor : By Prof. A Ramesh

University : IIT Roorkee

COURSE CODE	COURSE TITLE	Course Provider	W	H	C
10213CS449	The Joy of Computing using Python	NPTEL	12	-	3

Course Category:

Open Elective (Self - Learning Course)

Course Outcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Motivation for Computing	K2
CO2	Variables and Expressions: Design your own calculator	K3
CO3	Loops and Conditionals: Hopscotch once again	K3
CO4	Lists, Tuples and Conditionals: Lets go on a trip	K3
CO5	Sorting: Arrange the books and Searching : Find in seconds	K3

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	2	3	1	1	1	-	-	-	-	2	2	
CO2	2	3	3	3	2	2	1	-	-	-	-	3	3	
CO3	2	3	3	3	2	2	1	-	-	-	-	2	3	
CO4	2	3	3	3	2	3	1	-	-	-	-	3	3	
CO5	3	3	2	3	2	1	1	-	-	-	-	3	2	

Course Contents:

- Motivation for Computing
- Welcome to Programming!!
- Variables and Expressions : Design your own calculator
- Loops and Conditionals : Hopscotch once again
- Lists, Tuples and Conditionals : Lets go on a trip
- Abstraction Everywhere : Apps in your phone
- Counting Candies : Crowd to the rescue
- Birthday Paradox : Find your twin
- Google Translate : Speak in any Language
- Currency Converter : Count your foreign trip expenses
- Monte Hall : 3 doors and a twist
- Sorting : Arrange the books

- Searching : Find in seconds
- Substitution Cipher : What's the secret !!
- Sentiment Analysis : Analyse your Facebook data
- 20 questions game : I can read your mind
- Permutations : Jumbled Words
- Spot the similarities : Dobble game
- Count the words : Hundreds, Thousands or Millions.
- Rock, Paper and Scissor : Cheating not allowed !!
- Lie detector : No lies, only TRUTH
- Calculation of the Area : Don't measure.
- Six degrees of separation : Meet your favourites
- Image Processing : Fun with images
- Tic tac toe : Let's play
- Snakes and Ladders : Down the memory lane.
- Recursion : Tower of Hanoi
- Page Rank : How Google Works !!

Professor : By Prof.Sudarshan
University : IIT Ropar

COURSE CODE	COURSE TITLE	Course Provider	W	H	C
10213CS450	Introduction to Large Language Models (LLMs)	NPTEL	12	-	3

CourseCategory:

Open Elective (Self- LearningCourse)

CourseOutcomes

CO Nos	Course Outcomes	Skill Level (Based on revisedBloom's Taxonomy)
CO1	Understand the Foundations of Natural Language Processing (NLP)	K2
CO2	Apply Deep Learning Techniques to NLP Tasks	K3
CO3	Build and Analyze Language Models	K3
CO4	Develop Models with PyTorch for NLP Applications	K3
CO5	Design and Train Advanced Sequence-to-Sequence Models	K3

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO 7	PO 8	PO 9	PO1 0	PO1 1	PSO1	PSO 2	PSO 3
CO1	3	3	2	3	1	1						2	3	
CO2	2	3	3	3	2	1						3	3	
CO3	3	3	3	3	2	1						3	3	
CO4	3	3	3	3	2	1						3	3	
CO5	3	3	3	3	3	2						3	3	

Course Contents:**Week 1**

1. Introduction to NLP – Part 1 (NLP pipeline, applications of NLP)
2. Introduction to NLP – Part 2 (distributional semantics)

Week 2

1. Introduction to Deep Learning (Perceptron, ANN, backpropagation, CNN)
2. Word vectors (Word2Vec, GloVe, fastText)

Week 3

1. Introduction to Statistical Language Models (N-gram LM, Perplexity, Smoothing)

2. Language Models with CNN and RNN

Week 4

1. Introduction to PyTorch
2. Implementation of RNNs and LSTMs using PyTorch

Week 5

1. Sequence-to-sequence models, Beam search
2. Attention and self-attention

Week 6

1. Introduction to Transformers
2. Positional Embedding, various tokenization strategies
3. Implementation of Transformers using PyTorch

Week 7

1. Transfer Learning: ELMo, BERT (Encoder-only Model)
2. Transfer Learning: GPT (Decoder-only Model), T5 (Encoder-decoder model)

Week 8

1. Prompting (hard and soft) and Instruction fine-tuning (FLAN)
2. Advanced prompting (Chain of Thoughts, Graph of Thoughts, Prompt Chaining, etc.)
3. Introduction to HuggingFace Library

Week 9

1. Alignment with Human Feedback: RLHF, RLAIIF
2. Parameter-efficient adaptation (Prompt tuning, Prefix tuning, LoRA)

Week 10

1. Knowledge graphs (KGs)
 - a. Representation, completion
 - b. Tasks: Alignment and isomorphism
 - c. Distinction between graph neural networks and neural KG inference

Week 11

1. Open-book question answering: The case for retrieving from structured and unstructured sources; retrieval-augmented inference and generation
2. Retrieval augmentation techniques
 - a. Key-value memory networks in QA for simple paths in KGs
 - b. Early HotPotQA solvers, pointer networks, reading comprehension

- c. REALM, RAG, FiD, Unlimiformer
- d. KGQA (e.g., EmbedKGQA, GrailQA)

Week 12

1. Overview of recently popular models such as GPT4, Llama 3, Claude 3, Mistral, and Gemini
2. Ethical NLP – Bias and Toxicity
3. Conclusion

Professor : **Tanmoy Chakraborty,**
Prof.
SoumenChakraborti
University : IIT Delhi, IIT Bombay
Academic Year : 2024-2025
Semester : Winter

COURSE CODE	COURSE TITLE	W	H	C
10213CS451	Data Analytics with Python	12	-	3

Course Category:

Open Elective (Self - Learning Course)

Course Contents

Week 1	:	Introduction to data analytics and Python fundamentals
Week 2	:	Introduction to probability
Week 3	:	Sampling and sampling distributions
Week 4	:	Hypothesis testing
Week 5	:	Two sample testing and introduction to ANOVA
Week 6	:	Two way ANOVA and linear regression
Week 7	:	Linear regression and multiple regression
Week 8	:	Concepts of MLE and Logistic regression
Week 9	:	ROC and Regression Analysis Model Building
Week 10	:	Test and introduction to cluster analysis
Week 11	:	Clustering analysis
Week 12	:	Classification and Regression Trees (CART)

Course Provider : NPTEL
Professor : Prof. A Ramesh
University : IIT Roorkee

Academic Year: 2024-2025

Semester : Summer

COURSE CODE	COURSE TITLE	W	H	C
10212CS410	Introduction to Internet of Things	12	-	3

Course Category:

Program Elective (Independent Learning - Self - Learning Course)

Course Outcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Understand the fundamental principles of Introduction to IoT, Basics of Networking	K1
CO2	Analyzing the Sensor Networks, Integration of Sensors and Actuators with Arduino	K2
CO3	Understand the concept of Introduction to Python programming, Introduction to Raspberry Pi, Implementation of IoT with Raspberry Pi	K1
CO4	Applying concept of the IoT with Cloud Computing, Fog Computing, Industrial IoT, Case Study: Agriculture, Healthcare, Activity Monitoring	K3

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3		3		3							3		
CO2	3	3	3		3	2						3		
CO3		2		3	3							3		
CO4		3		3	3	2						3		

Course Contents

Week 1: Introduction to IoT: Part I, Part II, Sensing, Actuation, Basics of Networking: Part-I

Week 2: Basics of Networking: Part-II, Part III, Part IV, Communication Protocols: Part I, Part II

Week 3: Communication Protocols: Part III, Part IV, Part V, Sensor Networks: Part I, Part II

Week 4: Sensor Networks: Part III, Part IV, Part V, Part VI, Machine-to-Machine Communications

Week 5: Interoperability in IoT, Introduction to Arduino Programming: Part I, Part II, Integration of Sensors and Actuators with Arduino: Part I, Part II

Week 6: Introduction to Python programming, Introduction to Raspberry Pi, Implementation of IoT with Raspberry Pi

Week 7: Implementation of IoT with Raspberry Pi (contd), Introduction to SDN, SDN for IoT

Week 8: SDN for IoT (contd), Data Handling and Analytics, Cloud Computing

Week 9: Cloud Computing(contd), Sensor-Cloud

Week 10: Fog Computing, Smart Cities and Smart Homes

Week 11: Connected Vehicles, Smart Grid, Industrial IoT

Week 12: Industrial IoT (contd), Case Study: Agriculture, Healthcare, Activity Monitoring

CourseProvider : NPTEL
Professor : Prof. SudipMisra
University : IIT Kharagpur

Academic Year: 2024-2025
Semester : Summer

COURSE CODE	COURSE TITLE	W	H	C
10213CS447	Advanced Computer Architecture	12	-	3

Course Category: Open Elective (Self - Learning Course)

Course Outcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Apply the concepts of in-order and out-of-order pipelines to analyze processor performance.	K3
CO2	Implement branch prediction and advanced speculation techniques to optimize execution.	K3
CO3	Demonstrate compiler optimization techniques to enhance instruction-level parallelism (ILP).	K3
CO4	Design cache memory systems with optimization strategies for reduced latency and improved performance.	K3
CO5	Apply low-power design techniques and hardware reliability principles to modern processors.	K3

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	3	3	2	2								2
CO2	3	3	3	3	3								2
CO3	3	2	3	3	3								2
CO4	3	3	3	3	3								2
CO5	3	3	3	3	2	2							2

Course Layout

Week-1: In-order pipelines overview
 Week-2: Out-of-order pipelines, Branch prediction
 Week-3: Advanced branch prediction techniques
 Week-4: Issue, select, and commit
 Week-5: Aggressive speculation
 Week-6: Compiler techniques for enhancing ILP
 Week-7: Caches: optimizations Design, modeling, and
 Week-8: On-chip networks
 Week-9: Theory of memory models
 Week-10: Coherence Protocols
 Week-11: Low power design
 Week-12: Reliability and Hardware Security

Course Provider : NPTEL
Professor : Prof. Smruti Ranjan Sarangi
University : IIT DELHI

Industry/Higher Institute Interaction Learning Courses

COURSE CODE	COURSE TITLE	L	T	P	C

10215CS945	Next Generation of Wired and Wireless Networks	1	-	-	1

Course Category

Industry / Institute Higher Learning

Course Outcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Understand the foundational principles of next-generation networks, including advanced wired and wireless networking technologies, and their role in modern communication systems.	K2
CO2	Analyze and apply network virtualization and cloud computing concepts to optimize network performance and scalability in diverse environments.	K3
CO3	Evaluate security challenges in next-generation networks and develop strategies to ensure secure communication and data integrity.	K3
CO4	Explore emerging networking technologies and their applications in industries, focusing on innovation and future trends.	K3
CO5	Assess the regulatory and ethical considerations associated with next-generation networks, ensuring compliance and responsible implementation.	K3

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO6	PO 7	PO 8	PO 9	PO1 0	PO1 1	PSO 1	PSO 2	PSO 3
CO1	3	3	3	2	2									2
CO2	3	3	3	3	3									2
CO3	3	2	3	3	3									2
CO4	3	3	3	3	3									2
CO5	3	3	3	3	2	2								2

Course Content

Module 1: Foundations of Next Generation Networks

Module 2: Advanced Wired Networking Technologies

Module 3: Wireless Networking Technologies

Module 4: Network Virtualization and Cloud Computing

Module 5: Security and Next Generation Networks

Module 6: Emerging Technologies and its Applications

Module 7: Regulatory and Ethical Considerations

COURSE CODE	COURSE TITLE	L	T	P	C
10215CS948	Innovation Management	1	-	-	1

Course Category

Industry / Institute Higher Learning

Course Outcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Develop an understanding of innovation concepts, models, and methodologies to address real-world challenges through creative problem-solving approaches.	K3
CO2	Apply tools and techniques for fostering innovation and building an innovative mindset in both individual and organizational contexts.	K3
CO3	Analyze case studies and examples to explore the relationship between innovation and entrepreneurship, and anticipate future trends in innovation.	K3

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1		3	3	2				2	2	2	2			2
CO2		3	3	3				2	2	2	2			2
CO3		2	3	3				2	2	2	2			2

Course Content

1. Introduction to Innovation
2. Creative Thinking and Problem-Solving
3. Innovation Models and Methodologies
4. Innovation in Action: Case Studies and Examples
5. Tools and Techniques for Sparking Innovation
6. Building an Innovative Mindset
7. Innovation and Entrepreneurship
8. The Future of Innovation

COURSE CODE	COURSE TITLE	L	T	P	C
10215CS946	Entrepreneurs Essentials	1	-	-	1

Course Category

Industry / Institute Higher Learning

Course Outcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Understand the fundamental principles of entrepreneurship, including the traits and characteristics of successful entrepreneurs, and their role in economic and social development.	K2
CO2	Analyze the opportunities and risks associated with entrepreneurship to develop informed strategies for starting and managing sustainable ventures.	K3
CO3	Explore sustainable practices in entrepreneurship by designing activities and products that address societal needs while being economically and politically viable.	K3

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1							2	2	2	2	2			2
CO2							2	2	2	2	2			2
CO3							2	2	2	2	2			2

Course Content

1. Becoming An Entrepreneur
2. Goals
3. Identify characteristics of successful entrepreneurs
4. Describe the opportunities and risks of entrepreneurship.
5. Entrepreneurs Essentials
6. To introduce activities and products that serve people's needs and are sustainable
7. Economically and politically

COURSE CODE	COURSE TITLE	L	T	P	C
10215CS921	Natural Language Processing	1	-	-	1

Course Category

Industry / Institute Higher Learning

Course Outcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Understand the foundational concepts of Natural Language Processing (NLP) and apply them to real-world problems, such as text classification and sentiment analysis, using Python.	K2
CO2	Implement language models, syntax parsing, and text processing techniques to analyze and extract meaningful insights from textual data.	K3
CO3	Develop hands-on expertise in applying NLP methods and tools to build solutions for tasks like sentiment analysis and text classification.	K3

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO6	PO 7	PO 8	PO 9	PO1 0	PO1 1	PSO 1	PSO 2	PSO 3
CO1	2	2	2	2	2									2
CO2	2	2	2	2	2									2
CO3	2	2	2	2	2									2
CO4	2	2	2	2	2									2
CO5	2	2	2	2	2									2

Course Content

- Introduction to Natural Language Processing
- Text Classification + Hands-on Python
- Language Models + Hands-on Python
- Syntax & Parsing + Hands-on Python
- Sentiment Analysis + Hands-on Python

COURSE CODE	COURSE TITLE	L	T	P	C
10215CS935	Introduction to Symbolic and Connectionist AI	1	-	-	1

Course Category

Industry / Institute Higher Learning

Course Outcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Understand the fundamentals of Artificial Intelligence (AI), including symbolic AI, classical expert systems, and fuzzy expert systems, and their applications in solving real-world problems.	K2
CO2	Develop hands-on skills in building and implementing AI systems, such as rule-based systems, classical expert systems, and fuzzy expert systems, to address complex decision-making tasks.	K3
CO3	Explore advanced AI concepts, including connectionist AI, neural networks, and hybrid systems, to design and apply innovative	K3

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	2	2	3	2	2									2
CO2	2	2	3	2	2									2
CO3	2	2	3	2	2									2

Course Content

- Introduction to AI and Symbolic AI
- Hands-on Session on Building a Simple Rule-Based System
- Classical Expert Systems
- Hands-on Session on Developing a Classical Expert System
- Fuzzy Expert Systems
- Hands-on Session on Implementing a Fuzzy Expert System
- Introduction to Connectionist AI and Neural Networks
- Advanced Neural Networks and Deep Learning
- Hybrid Systems and Application.

COURSE CODE	COURSE TITLE	L	T	P	C
10215CS949	Organizational Innovation	0	0	2	1

Course Category

Industry / Institute Higher Learning

Course Outcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Understand the principles of innovation and its role in organizations	K2
CO2	Analyze the relationship between organizational learning	K3
CO3	Evaluate new organizational forms and strategies for information management to enhance communication	K3

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1							2	2	2	2	2			2
CO2							2	2	2	2	2			2
CO3							2	2	2	2	2			2

Course Content

- Defining innovation
- Innovation in organizations
- Innovation culture
- Innovation ecosystems
- Organizational learning
- Innovation and design
- Communication and decision-making
- New organizational forms
- Information management
- Open business models
- Problem solving and innovation
- Risk taking
- Organizational change

COURSE CODE	COURSE TITLE	L	T	P	C
10215CS936	Intelligent Optimization and Machine Learning Applications	1	-	-	1

Course Category

Industry / Institute Higher Learning

Course Outcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Understand the fundamental principles of optimization and intelligent optimization techniques	K2
CO2	Develop hands-on skills in designing and implementing optimization algorithms	K3
CO3	Apply optimization methods and machine learning techniques to real-world scenarios	K3

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	2	2	2		2									2
CO2	2	2	2		2									2
CO3	2	2	2		2									2

Course Content

- Introduction to Optimization and Intelligent Optimization
- Simulated Annealing
- Tabu Search
- Introduction to Tabu Search
- Hands-on Session on Developing a Tabu Search Algorithm
- Evolutionary Algorithms
- Swarm Intelligence
- Machine Learning Applications

COURSE CODE	COURSE TITLE	L	T	P	C

10215CS944	Fieldbus Architecture	1	-	-	1

Course Category

Industry / Institute Higher Learning

Course Outcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Understand the fundamental principles of optimization and intelligent optimization techniques	K2
CO2	Develop hands-on skills in designing and implementing optimization algorithms	K3
CO3	Apply optimization methods and machine learning techniques to real-world scenarios	K3

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	2	2	2											2
CO2	2	2	2											2
CO3	2	2	2											2

Course Content

Module 1: Introduction to Industrial Automation and Fieldbus

Module 2: Fieldbus Standards and Protocols

Module 3: Profibus

Module 4: Foundation Fieldbus

Module 5: Modbus

Module 6: Device Net and Other Fieldbus Protocols

Module 7: Fieldbus Network Design and Troubleshooting

Module 8: Fieldbus Security and Integration

Module 9: Case Studies and Practical Applications

Module 10: Future Trends in Fieldbus Technology

COURSE CODE	COURSE TITLE	L	T	P	C
10215CS933	Industrial IoT	1	-	-	1

Course Category

Industry / Institute Higher Learning

Course Outcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Understand the fundamental concepts of IoT, including its ecosystem, market, enabling technologies, and communication protocols, to build a strong foundation in IoT systems.	K2
CO2	Analyze the role of data management, analytics, and wireless communication technologies in IoT, while addressing critical aspects of security and privacy.	K3
CO3	Explore Industrial IoT (IIoT) technologies, protocols, and real-world case studies to design and implement innovative IoT solutions for diverse applications.	K3

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1		2	2	2	2									2
CO2		2	2	2	2									2
CO3		2	2	2	2									2

Course Content

- Overview of IoT
- IoT Ecosystem and Market
- Enabling Technologies
- IoT Communication and Networking
- Wireless Communication Technologies
- Networking Protocols
- Data Management and Analytics in IoT
- Security and Privacy in IoT
- IIoT Technologies and Protocols
- Case Studies and Real-World Applications
- Future Trends and Conclusion

COURSE CODE	COURSE TITLE	L	T	P	C
10215CS932	Vehicular and Mobile Networks	1	-	-	1

Course Category

Industry / Institute Higher Learning

Course Outcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Understand the fundamentals of vehicular networks and their integration with cellular and 5G technologies to enable intelligent transportation systems.	K2
CO2	Analyze security, trust, and privacy challenges in vehicular networks and develop strategies to address them effectively.	K3
CO3	Explore advanced topics in vehicular and mobile networks, including emerging 5G and beyond technologies, to design innovative solutions for future mobility.	K3

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1		2	2	2	2									2
CO2		2	2	2	2									2
CO3		2	2	2	2									2

Course Content

- Fundamentals of Vehicular Networks
- Vehicular Network Security, Trust and Privacy
- Cellular Network Technologies
- 5G and Beyond Technologies
- Advanced Topics in Vehicular and Mobile Networks

COURSE CODE	COURSE TITLE	L	T	P	C
10215CS950	Entrepreneurial strategies	1	-	-	1

Course Category

Industry / Institute Higher Learning

Course Outcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Understand the fundamentals of entrepreneurial strategy, including opportunity identification, validation, and strategic planning for market entry.	K2
CO2	Develop financial and operational strategies to effectively manage resources and drive business growth.	K3
CO3	Explore approaches to scaling, ensuring sustainability, and crafting a long-term vision for entrepreneurial ventures.	K3

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1							2	2	2	2	2			2
CO2							2	2	2	2	2			2
CO3						2	2	2	2	2	2			2

Course Content

- Module 1: Introduction to Entrepreneurial Strategy
- Module 2: Identifying and Validating Opportunities
- Module 3: Strategic Planning and Go-to-Market (GTM) Strategy
- Module 4: Financial and Operational Strategies
- Module 5: Scaling, Sustainability, and Long-Term Vision

COURSE CODE	COURSE TITLE	L	T	P	C
10215CS940	Large Language Model	1	-	-	1

Course Category

Industry / Institute Higher Learning

Course Outcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Understand the fundamental concepts of Natural Language Processing (NLP)	K2
CO2	Develop hands-on skills in implementing deep learning techniques such as Perceptrons, ANN, CNN, and Transformers using tools like PyTorch for NLP tasks.	K3
CO3	Explore advanced topics in NLP including knowledge graphs and LLM Models, to analyze their impact on language understanding and generation.	K3

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	2	2	2	2								2		2
CO2	2	2	2	2								2		2
CO3	2	2	2	2								2	2	2

Course Content

- Introduction to NLP
- Introduction to Statistical Language Models
- Statistical Language Models: Advanced Smoothing and Evaluation
- Introduction to Deep Learning (Perceptron, ANN, Backpropagation, CNN)
- Introduction to PyTorch
- Neural Language Models
- Introduction to Transformers
- Knowledge graphs
- Overview of recently popular models such as GPT-4, Llama-3, Claude-3, Mistral, and Gemini

COURSE CODE	COURSE TITLE	L	T	P	C
10215CS947	Venture Development	1	-	-	1

Course Category

Industry / Institute Higher Learning

Course Outcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Develop an entrepreneurial mindset by understanding the process flow and essential skills required to identify and capitalize on business opportunities.	K2
CO2	Analyze various business models and design effective operational, logistical, and financial strategies for a successful venture.	K3
CO3	Critically evaluate and present new venture plans by assessing their feasibility, sustainability, and market potential.	K3

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1						2	2	2	2	2	2			2
CO2						2	2	2	2	2	2			2
CO3						2	2	2	2	2	2			2

Course Content

- Overview of The Entrepreneurial Mindset and Skills-Process Flow
- Understanding Various Business Models
- Outlining Operations and Logistics
- Comprehensive Financial Planning and Management Team
- Presenting and Evaluating-Critiquing New Venture Plans

Course code	Course Name	L	T	P	C
10215CS951	Green AI	1	0	2	2

Course Category

Industry / Institute Higher Learning

Resource persons

1. Sagar Yadav, Data Scientist, Edunet foundation
2. Mr. SarthakNarnor Curriculum Developer for AI, ML and Green Technology, Edunet foundation

Course Outcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Understand the principles of green skilling, sustainability, and the contributions of Artificial Intelligence (AI) to sustainable development.	K2
CO2	Gain foundational knowledge of Artificial Intelligence, Machine Learning, and Python programming to address environmental challenges.	K2
CO3	Apply data analytics and Machine Learning techniques using Python for environmental and sustainability-focused applications.	K3
CO4	Design and implement AI-driven solutions, including Generative AI and Large Language Models (LLMs), for sustainable innovations.	K3
CO5	Evaluate the ethical, fairness, and sustainability considerations in AI applications to ensure responsible and impactful deployment.	K3

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	2	3	1	2						2	3	
CO2	2	3	3	3	2	2						3	3	
CO3	3	3	3	3	2	2						3	3	
CO4	3	3	3	3	2	2						3	3	
CO5	3	3	3	3	3	2						3	3	

Course content

Module 1: Foundation of Green Skilling, Sustainability and AI Contributions

Module II: Fundamentals of Artificial Intelligence, Machine Learning and Python Programming

Module III: Applied Data Analytics with Python for Environmental Applications

Module IV: Implementation and Deployment of Machine Learning and Deep Learning Algorithms

Module V: AI for Green Applications

Module VI: Generative AI and Large Language Models (LLMs) for Sustainable Solutions

Module VII: AI Ethics, Fairness, and Sustainability

Course code	Course Name	L	T	P	C
10215CS954	Introduction to IoT& Applications	1	0	1	1

Course Category

Industry / Institute Higher Learning

Resource persons

Prof. Michael OpokuAgyeman, Professor of Computer Engineering, University of Northampton, UK

Course Outcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Understand the fundamental concepts of the Internet of Things (IoT) and its applications in creating smart, connected systems.	K2
CO2	Develop knowledge of embedded systems, including design principles, basic computer architecture, and edge computing integration.	K3
CO3	Analyze and apply the architecture and functionalities of microcontrollers (e.g. ATmega328P) for IoT-based edge computing applications.	K3
CO4	Explore the role of sensors and network connectivity in IoT, enabling the development of efficient and responsive smart systems.	K3
CO5	Design and implement IoT solutions by integrating embedded systems, sensors, and edge computing for real-world applications	K3

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3													
CO2	2													
CO3	3	3	3	3		2						3		
CO4	3	3	3	3		2						3	3	
CO5	3	3	3	3		2						3	3	

Course Content

Module1: Introduction to IoT

Module 2: Embedded Systems

- Embedded System Design and Edge Computing
- Basic Computer Architecture

Module 3: Processor Architecture and Edge Computing

- Microcontroller (ATmega328P) Architecture
- Edge Computing Principles in IoT

Module 4: Sensors and Network Connectivity

- Sensors, Smart Connected Systems

Course code	Course Name	L	T	P	C
10215CS953	Exploring Machine Learning Tools, Techniques, and Trends	1	0	2	2

Course Category

Industry / Institute Higher Learning

Resource Persons

PROF. TS. DR. AFIZAN BIN AZMAN, School of Computer Science, Faculty of Innovation & Technology, Taylor's University, Malaysia

Course Outcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Understand the fundamental concepts of data analytics, including the differences between supervised and unsupervised learning, and the importance of data preprocessing in machine learning workflows.	K2
CO2	Apply regression techniques and regularization methods to develop predictive models for data-driven decision-making.	K3
CO3	Analyze and implement clustering algorithms and address challenges such as class imbalance to enhance model accuracy.	K3
CO4	Design and evaluate classification models, including ensemble methods, to improve prediction performance in diverse datasets.	K3
CO5	Explore the architecture and applications of neural networks while addressing ethical considerations and biases in intelligent systems.	K3

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3													
CO2	2	2	2	2	2									
CO3	3	3	2	2	2	2						3		
CO4	3	3	2	3	2	2						3	3	2
CO5	3	3	3	3	2	2						3	3	2

Course Content

- Introduction to Data Analytics
- Supervised vs Unsupervised Learning & Data Preprocessing
- Regression| Regularisation
- Clustering |Class Imbalance
- Classification | Ensemble Models
- Neural Networks, Ethical from Intelligent Systems

Course Code	Course Name	L	T	P	C
10215CS919	Computer Vision with Deep Learning	1	0	1	1

Course Category

Industry / Institute Higher Learning

Resource Persons

Prof.Sandeep Singh Sengar, Lecturer, Cardiff Metropolitan University, Wales, UK

Course Outcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Understand the foundational concepts of deep learning and explore various architectures, including Convolutional Neural Networks (CNN), for computer vision tasks.	K2
CO2	Apply techniques such as data augmentation and transfer learning to optimize deep learning models for improved performance and generalization.	K3
CO3	Design and implement strategies for developing successful deep learning models in computer vision, addressing practical challenges in real-world applications.	K3
CO4	Explore the principles and applications of Generative Adversarial Networks (GANs) for creating synthetic data and enhancing model robustness.	K3
CO5	Investigate real-time computer vision applications in the medical domain, demonstrating the potential of deep learning for healthcare advancements.	K3

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO 7	PO 8	PO 9	PO1 0	PO1 1	PSO1	PSO 2	PSO 3
CO1	2													
CO2	2	2	2	2	2									
CO3	3	3	2	2	2	2						3		
CO4	3	3	2	3	2	2						3	3	2
CO5	3	3	3	3	2	2						3	3	2

Course Content

- Introduction to Deep Learning and Convolutional Neural Networks (CNN)
- Different deep learning architectures
- Data Augmentation
- Transfer Learning
- Ideas to make a successful deep learning model in computer vision
- Generative Adversarial
- Networks (GAN), Computer vision works in medical domain (a real time application)

Course Code	Course Name	L	T	P	C
10215CS952	Medical image analysis with AI and ML	1	0	1	1

Course Category

Industry / Institute Higher Learning

Resource persons

Prof.Sandeep Singh Sengar, Lecturer, Cardiff Metropolitan University, Wales, UK

Course Outcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Understand the fundamental concepts of machine learning and deep learning, and their relevance to solving complex problems in medical imaging.	K2
CO2	Analyze various data formats in medical imaging and apply deep learning techniques to process and interpret medical data effectively.	K3
CO3	Explore real-world case studies and implement practical applications of machine learning and deep learning in medical imaging for diagnostic and analytical purposes.	K3

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	2													
CO2	2	2	2	2	2								2	
CO3	3	3	2	2	2	2						3	2	2

Course Content

Module 1: Machine Learning Concepts

Module 2: Deep Learning Concepts

Module 3: Introduction to Medical Imaging

Module 4: Data Formats in Medical Imaging

Module 5: Case Studies and Practical Applications

Course Code	Course Name	L	T	P	C
10215CS929	Generative AI	1	0	1	1

Course Category

Industry / Institute Higher Learning

Resource Persons

Prof.Sandeep Singh Sengar, Lecturer, Cardiff Metropolitan University, Wales, UK

Course Outcomes

CO Nos	Course Outcomes	Skill Level (Based on revised Bloom's Taxonomy)
CO1	Understand the foundational concepts of deep learning, including Convolutional Neural Networks (CNN) and their applications in various domains.	K2
CO2	Explore the principles and applications of Generative AI, Transformers, and the BERT model for natural language processing and other advanced tasks.	K3
CO3	Design and implement solutions using Generative Pre-Trained Transformers (GPT) to address real-world problems requiring AI-driven automation and creativity.	K3

Correlation of COs with Program outcomes and Programme Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	2	2	2	2	2									
CO2	2	2	2	2	2	2						2	2	
CO3	3	3	2	2	2	2						3	2	2

Course Content

Introduction to Deep Learning and Convolutional Neural Networks (CNN)

- Generative AI
- Transformers
- BERT Mode
- Generative Pre- Trained Transformer

Course Code	Course Name	L	T	P	C
10215CS955	Data modeling and design factors	1	0	1	2

Course Content

- Introduction to Data Modeling and Database Design
- Understanding the Data Requirements

- Advanced Data Modeling Techniques
- Relational Data Modeling
- Database Design
- Performance and Scalability in Data Modeling
- Data Security and Privacy
- Implementing Data Models in Databases
- Real-World Data Modeling Applications
- Best Practices in Data Modeling and Design

Resource Person:

Dr Husna Sarirah Binti Husin, Senior Lecturer, Taylor's University, Malaysia.

Course Code	Course Name	L	T	P	C
10215CS957	Cyber Threat Intelligence using Splunk	1	0	0	1

Course Content

- Network Security and Linux, Networking Basics, Common Network Threats

• TCP Communications, UDP Communications, TCP Header Flags, Encryption, Authentication, and Integrity

- Social Engineering, Phishing, Security Awareness, Security Assessment and Audit
- HTTPS Everywhere, TLS Handshake, TLS Session Resumption, Chain of Trust and Certificate Authorities
- Social Engineering, Phishing, Security Awareness, Security Assessment and Audit
- Introduction to Splunk and its role in enterprise security, Splunk Enterprise vs. Splunk Cloud
- Ingesting data into Splunk (log files, network data, etc.), Collecting and analyzing log data.
- Cybersecurity threats: Phishing, ransomware, SQL injection.
- Incident handling process: Detection, response, recovery.
- Case study on using Splunk for a real-world cybersecurity incident response
- Monitoring and analyzing network traffic using Splunk to identify suspicious activities and threats.
- Techniques for wireless hacking, including sniffing and spoofing traffic
- Web application vulnerabilities like XSS, SQL injection, and cross-site request forgery
- Testing and securing web applications using tools. Conducting penetration testing/reporting

Resource person:

Mr. Sankarraj Subramanian, Founder CEO, Prompt infotech

Course Code	Course Name	L	T	P	C
10215CS946	Entrepreneurs Essentials	1	0	0	1

Course Content

- Becoming An Entrepreneur
- Identify characteristics of successful entrepreneurs
- Recognize the importance of entrepreneurship in the economy.
- Entrepreneurship occurs in all areas of life. In business, academy, government, and NGOs.
- Describe the opportunities and risks of entrepreneurship.
- Entrepreneurship
- Entrepreneurs Essentials

Resource Person:

Ms. Kalaiarasi, Founder & CEO, Thremear Pvt Ltd

Course Code	Course Name	L	T	P	C
10215CS958	Radar Signal Processing ML/AI Framework for Radars	1	0	1	2

Course Content

- Fundamentals of Radar
- Operating principle of radar detection and ranging
- Mathematical foundations of Machine Learning
- Review of linear algebra concepts
- Review of Machine Learning and Deep Learning Algorithms
- Machine learning process and methods
- Deep Learning for Single-Target Classification in Sar Imagery
- SAR open datasets
- Deep Learning for Multiple-Target Classification in Sar Imagery
- Challenges in multiple-target classification

Resource Person:

Dr. Mercy Latha, Principal, Scientist, CSIR – CEERI

Course Code	Course Name	L	T	P	C
10215CS960	Agile project management and methodologies	1	0	0	1

Course Content

- Foundational Concepts
- Agile Methodologies
- Team Roles and Responsibilities
- Agile Planning
- Agile Execution
- Risk Management
- Stakeholder Engagement
- Tools and Technologies
- Scaling Agile
- Continuous Improvement

Resource person:

Mr. Anil kumar, Associate Director- IT/Cloud consulting, Kyndryl

Course Code	Course Name	L	T	P	C
10215CS961	Innovation & Design Thinking	1	0	0	1

Course Content

- Introduction to Innovation & Design Thinking
- Human-Centered Design Approach
- Design Thinking Process - Empathize & Define
- Ideation Techniques for Innovation
- Prototyping & Rapid Experimentation
- Design Validation & User Testing
- Business Model Innovation
- Emerging Technologies & Responsible AI in Innovation
- Innovation Strategy & Scaling Ideas
- Capstone Project & Industry Insights

Resource Person:

B.Senthil Kumar, Managing Director, IyarKai Tech Lab. Pvt. Ltd.

Course Code	Course Name	L	T	P	C
10215CS962	Data scraping and data wrangling	1	0	0	1

Course Content:

- Data Scraping
- Web Scraping Overview
- Popular Tools & Libraries used for Web Scraping
- Components of Web Scraping
- Data Wrangling
- Get to know Python Important Libraries
- Sentiment Analysis
- Problem Statement
- Data Inspection
- Data Cleaning

Resource person:

Mr.Kolla Ravikanth Yadav, Data Scientist APAC Business Operations,
Cepheid Hyderabad

Course Code	Course Name	L	T	P	C
10215CS959	Cellular interception Technologies	1	0	0	1

Course Content:

- Localization/Location fixing of Mobile Users
- Localization/Location fixing of Mobile Base Station/Tower
- Explore precise Location of Mobile Users using GPS
- Estimate Mobile Base Station/Tower to Mobile User distance
- Target Following of Mobile User
- Denial of Service for Mobile Communications up to 20 Km
- Smart Jamming Techniques for Mobile Communications up to 20 Km
- Interception of 4G and 5G voice (including volte , voip)and data
- Passive A5/3 Decryption for 2G
- Metadata Extraction from 4G and 5G communication for users' identification and linking to permanent identities (IMSI, IMEI).
- Denial of service of 5G and 5G.

Resource person:

Mrs. Vijayalakshmi, Engineer, ISRO

Course Code	Course Name	L	T	P	C
10215CS963	AI ML support for command-and-control applications in Disaster Management	1	0	0	1

Course Content:

- Predictive Modelling and Early Warning
- Forecasting and Early Warning Systems
- Disaster Response
- Real-time Decision Support
- Optimized Logistics
- Communication and Information Sharing
- Recovery
- Damage Assessment
- Specific Applications of AI in Disaster Management
- Challenges and Considerations

Resource person:

Mr.Vivek Saxena, Senior Scientist, DGRE, DRDO

Course Code	Course Name	L	T	P	C
10215CS967	Predictive analytics and business forecasting	1	0	0	1

Course Content:

- Introduction to Predictive Analytics
- Data Preparation and Exploration
- Statistical Foundations for Predictive Analytics
- Time Series Forecasting Techniques
- Business Applications of Predictive Analytics
- Big Data and Cloud-Based Predictive Analytics
- Business Forecasting and Decision-Making
- Implementation and Case Studies
- Capstone Project

Resource person:

Mr. Sabarinathan Muthu, Senior Technical Manager.

Course Code	Course Name	L	T	P	C
10215CS966	Structured problem-solving skills	1	0	0	1

Course Content:

- Understanding Problem-Solving Frameworks

- Industrial Problem Assignment & Team Formation
- Developing the Solution Blueprint
- Prototyping & Validation
- Refinement & Industry Alignment
- Final Presentation & Evaluation

Resource person:

Deepak Chandrasekaran, Director & CEO, Modern Molecular Diagnostics

Course Code	Course Name	L	T	P	C
10215CS968	IT Project Delivery Essentials: From Concept to Completion	1	0	0	1

Course Content:

- Project Management Fundamentals

- Stakeholder Management
- Scope Management
- Risk Management
- Communication Management
- Project Budgeting and Scheduling
- Change Management
- Quality Management
- Project Closure

Resource Person:

Mr. Vijayarajan Ramanathan, Project Manager, Hogarth Worldwide

COURSE CODE	COURSE TITLE	L	T	P	C
10215CS995	Kinematics, Dynamics, and Intelligent Control of Mobile Robotic Systems	1	0	0	1

Course Category:

Higher Learning Institute Interaction

Resource Person:

Resource Person: Prof. Minoru Sasaki, PhD, Gifu University / Tokai National Higher Education and Research System, Japan

Title: Kinematics, Dynamics, and Intelligent Control of Mobile Robotic Systems**Course Objective:**

This course introduces foundational and advanced topics in mobile robotics, with an emphasis on motion models, control theory, and applications in real-world environments. The content progresses from fundamental kinematics to modern control strategies used in autonomous robotic systems. Also, this course focuses on key technologies enabling autonomous navigation: SLAM (Simultaneous Localization and Mapping), RTK-GNSS integration, sensor fusion, and terrain-aware mobility. Real-world challenges in logistics, aerospace, and terrain navigation are emphasized through practical examples and research from Gifu University.

Course Content:

- Introduction to Mobile Robots
 - Types and structures: unicycle, bicycle, differential drive, Ackermann
 - Applications in industry, aerospace, and society
- Kinematics of Mobile Robots
 - State variables and control inputs
 - Unicycle and bicycle models
 - Ackermann and differential drive formulations
- Dynamic Modelling and State-Space Representation
 - Dynamics of coaxial two-wheeled robots
 - Lagrangian and Newtonian formulations
 - Practical modelling challenges and sensor integration
- Control Strategies
 - PID, LQI and MPC control of mobile platforms
 - Robust control using LMIs (Linear Matrix Inequalities)
 - Posture and trajectory control on uneven surfaces
- Experimental Platforms and Case Studies
 - Segway-type and stable two-wheeled AGVs
 - Control hardware/software (RTK-GNSS, thermographic sensors, ultrasonic sensing)
- Advanced Applications
 - Road surface classification and environmental mapping
 - Model predictive control (MPC) for stabilized platforms
 - Human-tracking robots using stereo-thermal vision
- Introduction to Autonomous Mobile Robots
 - Industry demand and challenges
 - Logistics, exploration, and disaster site applications

- SLAM Fundamentals
 - LiDAR, Visual, and Depth SLAM
 - ROS and GMapping
 - Monte Carlo Localization and Particle Filter methods
- Sensor Fusion Techniques
 - IMU, GNSS, thermal and stereo vision
 - RTK-GNSS architecture and integration
 - Obstacle avoidance using ultrasonic sensors
- Advanced Perception and Mapping
 - Human recognition with stereo thermography
 - Road surface classification using deep learning (SegNet)
 - Real-time environmental map generation
- System Design and Experiments
 - Sensor integration architecture
 - Case study: Coaxial robot with RTK-GNSS and auto-balance platform
 - Verification of performance: flat vs. uneven terrain
- Toward Intelligent Robots
 - AI-based perception for service robots
 - Navigation in GPS-denied environments
 - Applications in smart factories and planetary exploration
- Conclusion and future work

Lecture Schedule: 1 lecture hours and one two-hour lab per day

Prerequisites: Programming in Python

Grading: Assignment and Final Assessment

Note: This course is both analytical and practical. Attendance and participation in class are required

COURSE CODE	COURSE TITLE	L	T	P	C
10215CS134	Azure Cloud Computing and Python for AI Services	1	0	0	1

Course Category: Industry/Higher Learning Institute Interaction

Resource Person: Mr.Jeralad Felix, IT-Analyst Bosch Global, Software Technologies.

Title: - Azure Cloud Computing and Python for AI Services

Course Objective:

This course aims to build strong foundational skills in Microsoft Azure cloud services and enable learners to develop AI-driven applications using Python. It focuses on deploying, managing, and integrating cloud-based AI solutions using Azure's cognitive and machine learning services. By the end of the course, learners will be able to create scalable, intelligent applications by combining Azure cloud capabilities with practical Python programming techniques..

Course Content:

- Azure Cloud Fundamentals & Python Basics
- Introduction to Cloud & Azure
- Core Azure Services
- Python Refresher for AI
- Python + Azure SDKs
- Quick Quiz & Day 1 Wrap-up
- Azure AI & Industry Application
- Azure AI Services Overview
- Python for AI Services
- Security, Governance & Cost Management
- Mini Project Simulation
- Final Wrap-up & Assessment

Lecture Schedule: 1 lecture hours and one two-hour lab per day

Prerequisites: Basic understanding of computers and the internet. Fundamental knowledge of programming.

Grading: Assignment and Final Assessment

Note: This course is both analytical and practical. Attendance and participation in class are required

COURSE CODE	COURSE TITLE	L	T	P	C
10215CS133	Power BI and Data Analytics with Python for Business Intelligence	1	0	0	1

Course Category: Industry/Higher Learning Institute Interaction

Resource Person: Dr.Gomathi S

Title: Power BI and Data Analytics with Python for Business Intelligence

Course Objective:

Students will gain an understand data visualization principles and techniques using Power BI. Gain hands-on experience in building interactive dashboards and reports. Develop proficiency in using Python for statistical analysis and machine learning.

Course Content:

Introduction to BI & Power BI

- What is Business Intelligence?
- Role of Power BI and Python in analytics.
- Real-world examples of BI dashboards.
- Activity: Group discussion – “How does Netflix/Amazon use BI?”

Getting Started with Power BI

- Power BI Desktop Interface.
- Importing data (Excel, CSV, Web).
- Creating first visuals: bar, pie, line, KPI cards.
- Exercise

Data Transformation (ETL)

- Power Query basics – cleaning & shaping data.
- Handling missing values, duplicates, formatting.
- Python integration with Power BI.
- Using Python (Pandas) for preprocessing.
- Exercise: Clean a messy dataset (student groups).

Data Modelling & DAX Basics

- Data relationships (star schema basics).
- Measures vs. calculated columns.
- DAX functions: SUM, AVERAGE, CALCULATE.
- Activity: Group challenge – create KPIs

- Mini-Project 1: Build departmental performance dashboard.
Custom Visuals in Power BI
- Importing visuals from AppSource.
- Common custom visuals: Word Cloud, Decomposition Tree, Funnel, KPI visual.
- When to use custom visuals in storytelling.
- Hands-on Exercise
Installing & Configuring Python in Power BI
- Installing Python
- Setting up Python scripting in Power BI
- Running first Python script in Power BI
- Exercise: Import dataset using Python script inside Power BI.
Python Basics for Data Analytics
- Pandas DataFrames (read CSV, head, describe).
- Simple statistics (mean, median, mode, correlation).
- Exercise: Calculate average sales, highest profit product using Python.
Python Visualization in Power BI
- Using Matplotlib/Seaborn inside Power BI.
- Scatter plot, histogram, box plot.
- Hands-on Exercise: Scatter plot of Sales vs. Profit.
Predictive Analytics Basics
- Intro to regression (simple linear regression).
- Running regression in Python and displaying results in Power BI.
- Exercise: Predict next month's sales from historical dataset.
Final Project (Group Work)
- End-to-End Workflow:
 - Import raw dataset
 - Clean data with Power Query
 - Build model with DAX
 - Run Python analytics
 - Add custom visuals
 Project Presentations & Wrap-Up
- Each group presents their dashboard & insights.
- Peer feedback and instructor review.
- Key takeaways from 2-day program

Lecture Schedule: 1 lecture hours and one two-hour lab per day

Prerequisites: Microsoft Excel and basic data analytics and business analytics concepts

Grading: Assignment and Final Assessment

COURSE CODE	COURSE TITLE	L	T	P	C
10215CS127	Introduction to Quantum Computing	1	0	0	1

Course Category: Industry/Higher Learning Institute Interaction

Resource Person: Jaya Kumar Tatipathi, Founder & CEO Quant Curious Technologies Pvt Ltd.

Title: - Introduction to Quantum Computing

This course aims to introduce learners to the fundamental principles of quantum computing, including qubits, superposition, and entanglement. It focuses on building a conceptual understanding of how quantum algorithms differ from classical approaches. By the end of the course, learners will be able to explain core quantum concepts and apply basic quantum programming techniques using beginner-friendly tools.

Course Content:

- Introduction to Quantum Principles
- Quantum Gates and Circuits
- Quantum Algorithms Overview
- Quantum Computing Platforms
- Quantum Programming Basics
- Mini Project Sprint
- QsA and Day-end Assessment

Lecture Schedule: 1 lecture hours and one two-hour lab per day

Prerequisites: Basic knowledge of mathematics, especially linear algebra and complex numbers. c Familiarity with fundamental computing concepts or programming basics.

Grading: Assignment and Final Assessment

Note: This course is both analytical and practical. Attendance and participation in class are required

COURSE CODE	COURSE TITLE	L	T	P	C
10215CS130	Human-Centered Design Thinking for Sustainable Development Goals	1	0	0	1

Course Category: Industry/Higher Learning Institute Interaction

Resource Person: Ms.Somasundaram Mira, Founder & CEO, Astralbeat Innovations Pvt. Ltd.

Title: Human-Centered Design Thinking for Sustainable Development Goals

Course Objective:

This course aims to introduce learners to the principles of human-centered design and its application in addressing real-world challenges aligned with the Sustainable Development Goals (SDGs). It enables students to develop innovative, user-focused, and socially impactful solutions through empathy, creativity, and iterative problem-solving.

Course Content:

- Introduction to sdgs and human-centered design thinking
- Problem framing and systems thinking
- Empathy and immersion
- Ideation and prototyping for impact
- Testing, validation, and sdg alignment

Lecture Schedule: 1 lecture hours and one two-hour lab per day

Prerequisites: Basic understanding of business concepts or interest in entrepreneurship.

Grading: Assignment and Final Assessment

Note: This course is both analytical and practical. Attendance and participation in class are required

COURSE CODE	COURSE TITLE	L	T	P	C
10215CS125	Practical Data Science for Beginners	1	0	0	1

Course Category: Industry/Higher Learning Institute Interaction

Resource Person: Mr. P. Mohith, designation, Data Science Consultant.

Title: - Practical Data Science for Beginners

Course Objective:

This course aims to introduce beginners to the essential concepts, tools, and workflows of Data Science through hands-on practice. It focuses on enabling students to work with real datasets, perform analysis, and derive meaningful insights using simple and practical techniques.

Course Content:

- Google Cloud setup
- Practical/hands-on demonstration of Image-based Classification
- Model using Google Cloud ML Algorithms, with a large dataset
- Practical session to conduct data-driven root-cause analysis
- Git and GitHub session

Lecture Schedule: 1 lecture hours and one two-hour lab per day

Prerequisites: Basic computer skills and willingness to learn Python.

Grading: Assignment and Final Assessment

Note: This course is both analytical and practical. Attendance and participation in class are required

COURSE CODE	COURSE TITLE	L	T	P	C
10215CS131	Entrepreneurship and Innovation	1	0	0	1

Course Category: Industry/Higher Learning Institute Interaction

Resource Person: Mr. Joyal Francis, Designation : executive Director, Industry : Smartedgeteq Pvt. Ltd.,

Title: Entrepreneurship and Innovation

Course Objective:

This course aims to provide learners with a deep understanding of how innovative thinking drives entrepreneurial success in today's dynamic business environment. It equips students with practical tools to identify opportunities, develop creative solutions, and build impactful ventures through innovation-led strategies.

Course Content:

- Introduction to Entrepreneurship
- Opportunity Identification and Market Gaps
- Innovation and Creativity
- Lean Startup and Design Thinking
- Business Model Design
- Market Research and Validation
- MVP Development
- Go-to-Market
- Startup Financials and Funding
- Team and Culture
- Scaling and Growth

Lecture Schedule: 1 lecture hours and one two-hour lab per day

Prerequisites: Basic understanding of general business concepts is helpful but not mandatory.

Grading: Assignment and Final Assessment

Note: This course is both analytical and practical. Attendance and participation in class are required

COURSE CODE	COURSE TITLE	L	T	P	C
10215CS998	Explainable AI	1	0	0	1

Course Category: Industry/Higher Learning Institute Interaction

Resource Person: Dr. Arivukkarasan Raja, Director – IT, AstraZeneca, Chennai.

Title: Explainable AI

Course Objective:

Students will have an insight to the recent techniques in Explainable AI.

Course Content:

Introduction to Explainable AI (XAI)

- What is XAI?
- Motivation and Importance
- Regulatory and ethical considerations

Fundamentals of Machine Learning Interpretability

- Global vs local interpretability
- Black-box vs white-box models
- Measuring explainability

Overview of XAI Techniques - Intrinsic Interpretability

- Linear models, Decision trees
- Rule-based systems
- Case studies

Post-hoc Model Explanation Methods

- Feature importance
- Sensitivity analysis
- Visualization techniques

Local Explanation Methods

- LIME (Local Interpretable Model-agnostic Explanations)
- Use-cases and limitations

Local Explanation Methods II

- SHAP (SHapley Additive exPlanations)
- Comparison with LIME
- Hands-on exercise

XAI for Deep Learning

- Visual explanations: saliency maps, Grad-CAM
- Layer-wise relevance propagation
- Examples in vision and NLP

XAI in Practice

- Interpreting XAI in real-world settings
- Challenges in deployment
- User-centric explanations

XAI Toolkits and Libraries

- Introduction to Alibi, Captum, ELI5, SHAP, LIME

- Practical hands-on labs
- **Ethical, Legal, and Social Implications (ELSI) of XAI**
- Transparency, fairness, accountability
- Case studies: Bias detection
- Policy and governance

Lecture Schedule: 1 lecture hours and one two-hour lab per day

Prerequisites: Lecture hall and Lab for demonstration.

Grading: Quiz by the Industry Expert

Note: This course is both analytical and practical. Attendance and participation in class are required

COURSE CODE	COURSE TITLE	L	T	P	C
10215CS996	Android App Development using Java and Kotlin	1	0	0	1

Course Category: Industry/Higher Learning Institute Interaction

Resource Person: Mr. M. Trishiraj, Professional Android Engineer.

Title: Android App Development using Java and Kotlin

Course Objective:

To provide learners with a strong foundation in Android development using Kotlin, covering core components, UI/UX principles, and modern Jetpack libraries, to enable learners to design and develop scalable, user-centered Android applications using MVVM architecture, Room database, Retrofit networking, and Dependency Injection and equip learners with essential product management skills—ideation, feature prioritization, MVP design, testing, and deployment—to support full product lifecycle development from concept to launch.

Course Content:

Android Fundamentals & UI/UX Basics

Introduction to Android Development

- Android Ecosystem Overview (OS versions, devices, Play Store).
- Setting up Android Studio and SDK.
- Requirements and getting things ready

Introduction to Kotlin:

- Kotlin Fundamentals (variables, data types, control flow).
- Key Kotlin Features for Android (null safety, extension functions, data classes).
- Basic Coroutines for asynchronous programming.

Core Android Components:

- Activities and their Lifecycle.
- Declarative UI with Jetpack Compose
- Basic UI Elements (TextView, Button, EditText, ImageView).

UI/UX Principles - Part 1

- Introduction to User-Centered Design.
- Material Design Guidelines for Android.
- Accessibility basics for Android apps.

Advanced UI & Navigation

- RecyclerView for efficient list display.
- Fragments and Fragment Manager.
- App Navigation (Navigation Component basics).
- Theming and Styling in Android.

Product Management - Part 1 (Ideation & Discovery):

- Understanding the Product Lifecycle.
- Identifying User Needs and Pain Points.
- Market Research and Competitor Analysis.
- Defining Product Vision and Goals.

Architecture, Data & Advanced UI/UX

Android Architecture Components - Part 1

- Introduction to Clean Architecture principles.
- MVVM (Model-View-ViewModel) pattern in Android.
- LiveData and ViewModel for UI state management.
- Lifecycle-aware components.

Data Persistence

- Introduction to Room Persistence Library (SQLite abstraction).
- Working with local databases.
- Preferences (SharedPreferences, DataStore).

UI/UX Principles - Part 2

- Information Architecture and User Flows.
- Wireframing and Prototyping tools.
- Usability Testing basics.

Networking & APIs:

- Making network requests with Retrofit.
- Handling JSON data (Serialization/Deserialization).
- Asynchronous operations with Coroutines.

Product Management - Part 2 (Strategy & Development)

- Feature Prioritization Techniques (e.g., MoSCoW, RICE).
- Creating User Stories and Acceptance Criteria.
- Agile Methodologies (Scrum, Kanban) in product development.
- Minimum Viable Product (MVP) concept.

Advanced Topics, Testing & Product Launch

Advanced Architecture & Dependency Injection

- Repository Pattern for data abstraction.
- Introduction to Hilt/Dagger for Dependency Injection.
- Testing Fundamentals (Unit testing with JUnit, Mockito).

Background Processing

- WorkManager for deferrable background tasks.
- Notifications in Android.

UI/UX Principles - Part 3

- Visual Design Best Practices (typography, color, imagery).
- Microinteractions and animations.
- Feedback and Error States.

Product Management - Part 3 (Launch & Growth)

- App Store Optimization (ASO) strategies.
- Analytics and User Feedback Collection.
- Measuring Product Success (KPIs).
- Iterative Development and Post-Launch Strategy.

Deployment & Best Practices

- Preparing your app for release (signing, ProGuard/R8).
- Publishing to Google Play Console.
- Security considerations and common pitfalls.
- Q&A and Project Showcase.

Lecture Schedule: 1 lecture hours and one two-hour lab per day

Prerequisites: Basic Programming Knowledge, Familiarity with Mobile or Software Development Concepts (Preferred, Not Mandatory)

Grading: Assignment and Final Assessment

Note: This course is both analytical and practical. Attendance and participation in class are required

COURSE CODE	COURSE TITLE	L	T	P	C

10215CS997	Time Series Analysis and Forecasting	1	0	0	1

Course Category: Industry/Higher Learning Institute Interaction

Resource Person: Mr. Kolla Ravikanth Yadav, Data Scientist, CEPHEID.

Title: **Time Series Analysis and Forecasting**

Course Objective:

Understand the fundamental concepts of time series data, including properties, patterns, and methods for preprocessing, visualization, and exploratory analysis. Apply classical and modern forecasting techniques, including ARIMA, SARIMA, Machine Learning regression models, and deep learning models like LSTM. Build, evaluate, and deploy accurate time series forecasting models using Python libraries, with a strong focus on practical, real-world datasets.

Course Content:

- **Fundamentals & Time Series Exploration**
 - **Introduction to Time Series**
- What is Time Series Data
- Key applications (finance, sales, demand forecasting)
- Types of time series (univariate vs multivariate, stationary vs non-stationary)
 - Python Setup & Libraries
- pandas, matplotlib, seaborn, statsmodels, scikit-learn
- Installing via pip/conda and Jupyter setup
 - Time Series Exploration & Visualization
- Time-based indexing
- Line plots, rolling statistics
- Seasonal decomposition (Additive vs Multiplicative)
- ACF & PACF plots
- Handling Missing Values, Outliers, and Noise
- Imputation techniques (forward fill, interpolation)
- Outlier detection (IQR, Z-score, visual inspection)
 - Stationarity & Transformation
- Importance of stationarity
- Augmented Dickey-Fuller (ADF) test
- Differencing, log transform, Box-Cox
- Practical hands-on: Airline or Sales dataset seaborn
 - **Forecasting Models – Traditional & ML**
 - Classical Models: AR, MA, ARMA, ARIMA
- Assumptions & parameters
- Model building using statsmodels
- Parameter tuning (p,d,q) using AIC/BIC
- Model diagnostics and residual analysis
- Practical: Build ARIMA for monthly sales data

- SARIMA & Seasonal Models
- SARIMA parameters (P,D,Q,s)
- Grid search for best model
- Seasonal decomposition + SARIMA practical
 - ML Models for Time Series
- Framing TS for ML – Lag features, rolling stats
- Train-test split for TS (walk-forward validation)
- Random Forest, XGBoost for regression
- Hands-on: Forecast electricity or weather data
 - Evaluation Metrics
- MAPE, MAE, RMSE, SMAPE
- Backtesting with cross-validation
- Use sklearn, tscv tools
- **Advanced Techniques & Deployment**
 - Facebook Prophet
- Overview, capabilities, and assumptions
- Modeling with trend, seasonality, holidays
- Hands-on: Forecast future sales using Prophet
 - LSTM (Deep Learning for Time Series)
- Why deep learning for TS?
- Preparing sequences for LSTM
- Model creation using Keras or PyTorch
 - Train & evaluate LSTM on stock/sales data
 - Multivariate Forecasting
- Add exogenous variables (ARIMAX, VAR)
- Feature selection & correlation
- Example: Predict demand with price, promo data

Lecture Schedule: 1 lecture hours and one two-hour lab per day

Prerequisites: Basic knowledge of Python programming

Grading: 3 Assessment

Note: This course is both analytical and practical. Attendance and participation in class are required

COURSE	COURSE TITLE	L	T	P	C

CODE					
10215CS121	Edge Computing and IoT-Transforming Smart Connectivity	1	0	0	1

Course Category: Industry/Higher Learning Institute Interaction

Resource Person: Mr. Manicka Raja Senior Manager, Novac Technology Solutions
Chennai

Title: Edge Computing and IoT-Transforming Smart Connectivity

Course Objective:

This course aims to provide learners with a strong foundation in edge computing concepts and their integration with IoT systems for real-time data processing. It focuses on understanding how distributed architectures improve speed, efficiency, and smart connectivity across devices. By the end of the course, learners will be able to design and implement edge-IoT solutions that enable intelligent, low-latency applications.

Course Content:

- IoT Foundations & Sensor-Cloud Pipeline
- Introduction to IoT and Ecosystem Overview
- Development Environment Setup
- Cloud Integration
- Mini Discussion
- Edge Computing & Real-time Decision Making
- Introduction to Edge Computing
- Software Patterns for Edge Intelligence
- Deploying Intelligence to the Edge
- Wrap-up & Real-world Scenarios

Lecture Schedule: 1 lecture hours and one two-hour lab per day

Prerequisites: Basic understanding of computer networks and digital devices. Familiarity with sensors, microcontrollers, or IoT fundamentals.

Grading: Assignment and Final Assessment

Note: This course is both analytical and practical. Attendance and participation in class are required

COURSE CODE	COURSE TITLE	L	T	P	C
10215CS123	Microsoft Azure Fundamentals for Cloud Services	1	0	0	1

Course Category: Industry/Higher Learning Institute Interaction

Resource Person: Bharath Aginati , Associate Consultant, Capgemini.

Title: -Microsoft Azure Fundamentals for Cloud Services

Course Objective:

The objective of the Microsoft Azure Fundamentals for Cloud Services course is to introduce learners to essential cloud computing concepts and Azure's global architecture. It aims to develop an understanding of core Azure services, including compute, storage, networking, and databases. The course equips learners with basic skills to create, manage, and monitor cloud resources using the Azure Portal and command-line tools. Overall, it prepares learners to apply Azure solutions responsibly and build a foundation for further cloud learning.

Course Outcomes:

Student will be able to

Demonstrate industry practices or research methodologies through interactions with industry professionals or higher learning institutions.

Demonstrate industry practices or research methodologies through interactions with industry professionals or higher learning institutions.

Analyse and reflect on advanced technologies, tools, or processes encountered during the interaction.

Course Content:

- Introduction to Cloud Computing & Microsoft Azure
- Overview of cloud computing models (IaaS, PaaS, SaaS)
- Benefits of cloud computing
- Introduction to Microsoft Azure and its global infrastructure
- Azure services overview
- Core Azure Services & Hands-on Demonstration
- Compute services: Virtual Machines, Azure App Services
- Storage services: Blob Storage, Azure Files, Azure SQL Database
- Networking: Virtual Networks, Load Balancers, VPN Gateway
- Hands-on demonstration: Deploying a Virtual Machine in Azure
- Security, Compliance, and Azure Pricing
- Azure Identity and Access Management (IAM)
- Azure pricing models and cost management
- Hands-on demonstration: Configuring security policies in Azure

Lecture Schedule: 1 lecture hours and one two-hour lab per day

Prerequisites: Lecture hall with internet access and computers for hands-on demonstrations.

Grading: Assignment and Final Assessment

Note: This course is both analytical and practical. Attendance and participation in class are required

COURSE CODE	COURSE TITLE	L	T	P	C
10215CS976	Entrepreneurship Essentials	1	0	0	1

Course Category: Industry/Higher Learning Institute Interaction

Resource Person: Mr. M. Selvam, Designation, Founder, SM Smartygen Media Networks Pvt Ltd,

Title: Entrepreneurship Essentials

Course Objective:

This course aims to introduce learners to the foundational principles of entrepreneurship, including opportunity identification, business model development, and value creation. It equips students with the skills and mindset needed to evaluate ideas, assess market potential, and build sustainable ventures.

Course Content:

- Introduction to Entrepreneurship
- Ideation and Opportunity Recognition
- Introduction to Entrepreneurship
- Ideation and Opportunity Recognition
- Business Planning
- Funding and Financial Management
- Marketing and Sales
- Operations and Management
- Scaling and Growth

Lecture Schedule: 1 lecture hours and one two-hour lab per day

Prerequisites: No prior business knowledge required. Interest in innovation, creativity, and entrepreneurial thinking.

Grading: Assignment and Final Assessment

Note: This course is both analytical and practical. Attendance and participation in class are required

COURSE CODE	COURSE TITLE	L	T	P	C
10215CS999	Modern AI Tools Fundamentals	1	0	0	1

Course Category: Industry/Higher Learning Institute Interaction

Resource Person: Ms.Sindhuja Nagarajan, Consultant AI Specialist, Stealth Startup, Founder, Quambase Innovations Pvt Ltd.

Title: Modern AI Tools Fundamentals

Course Objective:

To introduce students to the fundamentals of Artificial Intelligence and Generative AI, including key concepts, recent breakthroughs, and major AI categories such as NLP, Computer Vision, and generative models.

To develop students' skills in using modern AI tools and platforms, including ChatGPT, Claude, DALL-E, Midjourney, Copilot, and workflow automation tools like n8n.

To enable students to apply AI for productivity, automation, and business use cases, while understanding ethical considerations and career opportunities in the AI ecosystem.

Course Outcomes:

Student will be able to

Explain the core concepts of AI, Generative AI, and LLMs and describe recent advancements such as transformers and diffusion models.

Effectively use modern AI tools for text, image, video, and audio generation, and apply prompt engineering techniques (zero-shot, few-shot, chain-of-thought).

Apply AI tools for real-world tasks such as content creation, coding assistance, automation, and workflow integration, demonstrating responsible and ethical usage of AI technologies.

Course Content:

- Introduction to Modern AI and Generative AI Tools Session 1: AI & GenAI Fundamentals
- What is Artificial Intelligence? (Narrow AI, General AI, Generative AI)
- Recent breakthroughs (Transformers, Diffusion Models)
- Overview of major AI categories: NLP, Computer Vision, Generative AI
- Demo: Using ChatGPT for summarization and ideation Session 2: Large Language Models & Prompting (1.5 hrs)
- Understanding LLMs: GPT, LLaMA, Claude
- Prompt Engineering Basics: Zero-shot, Few-shot, Chain of Thought
- Demo: Writing effective prompts in ChatGPT and Claude
- Generative AI Tools Overview (2 hrs)
- Overview of leading tools:
 - Text: ChatGPT, Claude, Gemini
 - Image: Midjourney, DALL-E, SDXL
 - Video: Sora, RunwayML
 - Audio: ElevenLabs, Suno
- Hands-on Lab: Generate text and images using ChatGPT and DALL-E

•Productivity, Automation & Business Applications Session 1:

AI for Content, Productivity & Coding

- AI writing tools: Jasper, Copy.ai, Notion AI
- AI summarization and note-taking: Fireflies, Otter, Notion AI
- Using AI in coding: GitHub Copilot, Replit Ghostwriter
- Hands-on: Automating meeting notes with Fireflies & writing Python scripts with Copilot
- Session 2: AI Automation & Integration
- Introduction to AI Agents (AutoGPT, AgentGPT)
- AI + n8n for workflow automation
- Practical examples: Automate emails and reports
- Demo: Connecting ChatGPT with Google Sheets via n8n
- Ethics, Applications & Roadmap
- AI in business use cases (marketing, HR, customer support, operations)
- AI ethics: Bias, hallucinations, responsible usage
- Career roadmap in AI: Prompt Engineer, AI Product Manager, AI Researcher
- Certifications: OpenAI Prompt Engineering, DeepLearning.AI Generative AI
- Recap, Q&A, Feedback, and Quiz/Game session

Lecture Schedule: 1 lecture hours and one two-hour lab per day

Prerequisites: None

Grading: Assignment and Final Assessment

Note: This course is both analytical and practical. Attendance and participation in class are required

COURSE	COURSE TITLE	L	T	P	C

CODE					
10215CS947	Venture Development	1	0	0	1

Course Category: Industry/Higher Learning Institute Interaction

Resource Person: Mr. Arshath Mohammed, Co-Founder & CEO Industry: Showbay Technologies Pvt Ltd, Chennai

Title: Venture Development

Course Objective:

The course aims to equip students with the knowledge and skills required to develop, validate, and launch new business ventures. It focuses on nurturing entrepreneurial thinking through practical tools for opportunity identification, business modeling, and venture growth strategies.

Course Content:

- Overview of entrepreneurship and innovation
- Identifying and refining business ideas
- Entrepreneurial mindset frameworks
- Case discussion: Successful startups
- Building scalable business models
- Market research techniques
- Idea validation and feasibility
- Business Model Canvas exercise
- Go-to-market strategies
- Lean startup principles
- Pitch preparation and presentation practice
- Basics of venture funding

Lecture Schedule: 1 lecture hours and one two-hour lab per day

Prerequisites: None, but a keen interest in business and startups is recommended.

Grading: Students will complete a concise take-home assignment that includes a brief written reflection on a venture idea, a simple business model sketch, and a market validation plan.

Note: This course is both analytical and practical. Attendance and participation in class are required

COURSE CODE	COURSE TITLE	L	T	P	C
10215CS124	AWS Data Analytics Fundamentals	1	0	0	1

Course Category: Industry/Higher Learning Institute Interaction

Resource Person:

Title: AWS Data Analytics Fundamentals

Course Objective:

Students will gain a solid understanding of AWS services, cloud computing, and data analytics concepts, design and implement data pipelines using AWS services like S3, Glue, and Redshift and understand how to analyze data using Athena and Redshift, and visualize it using QuickSight.

Course Content:

- Introduction to AWS and Data Analytics
- **Cloud & AWS Fundamentals**
- What is Cloud Computing? (IaaS, PaaS, SaaS)
- Why AWS? Introduction to AWS Console
- Overview of global infrastructure (Regions, AZs, Edge Locations)
- Free Tier & account setup demo
- **AWS Storage Services**
- Amazon S3: Buckets, objects, versioning, lifecycle
- Demo: Uploading, accessing and securing files
- Concepts of Data Lake
- **Basics of Data Analytics**
- What is Data Analytics?
- ETL vs ELT, Batch vs Streaming
- Analytics workflow on AWS (Collect → Store → Process → Analyze → Visualize)
- Overview of relevant services: S3, Glue, Redshift, Athena, QuickSight, Kinesis
- Hands-on Lab: Run SQL queries on CSV data using Amazon Athena
- ETL & Data Warehousing
- **Data Processing with AWS Glue**
- Intro to Glue: Jobs, Crawlers, Data Catalogue
- ETL using Glue Studio (Visual + PySpark)
- Hands-on: Create a Glue crawler and ETL job to transform data from S3
- **Data Warehousing with Amazon Redshift**
- What is a Data Warehouse?
- Redshift basics: Cluster, Node Types, Redshift Spectrum
- Demo: Load data from S3 to Redshift using COPY command Query across S3 & Redshift using Redshift Spectrum
- When to use Athena vs Redshift
- **Real-time Data, Visualization & Certification Prep**
- Real-time Data Streaming

- Use cases of real-time analytics
- Introduction to Amazon Kinesis: Data Streams vs Firehose
- Demo: Stream data into S3 with Firehose
- **Data Visualization with Amazon QuickSight**
- What is BI? Why visualizations matter
- Create dashboards using QuickSight from S3 / Athena / Redshift
- Hands-on: Build a basic dashboard
- **Recap & Career Guidance**
- Recap of all services used
- Real-world analytics pipeline on AWS
- Career roadmap for Data Analytics on AWS
- Certification path: AWS Certified Data Analytics – Specialty
- Q&A, Feedback, and Quiz/Game session

Lecture Schedule: 1 lecture hours and one two-hour lab per day

Prerequisites: Basic AWS Knowledge, Data analytics and Sql.

Grading: Assignment and Final Assessment

Note: This course is both analytical and practical. Attendance and participation in class are required

COURSE CODE	COURSE TITLE	L	T	P	C
10215CS132	Introduction to Data Science with ChatGPT & LLMs	1	0	0	1

Course Category: Industry/Higher Learning Institute Interaction

Resource Person: Ms.Somasundaram Mira, Founder & CEO, Astralbeat Innovations Pvt. Ltd.

Title: Introduction to Data Science with ChatGPT & LLMs

Course Objective:

This course aims to introduce learners to the core concepts of Data Science while demonstrating how ChatGPT and Large Language Models (LLMs) can enhance data exploration, analysis, and decision-making. It equips students with practical skills to integrate AI-driven tools into real-world data workflows for improved productivity and insights.

Course Content:

- Introduction to Data Science: workflow, tools, and applications.
- Fundamentals of Large Language Models (LLMs) and how ChatGPT works.
- Using ChatGPT for data cleaning, preprocessing, and exploratory data analysis.
- Visualization and basic modeling with Python, supported by LLM-assisted insights.
- Building AI-enhanced data workflows and real-world case studies using LLMs.

Lecture Schedule: 1 lecture hours and one two-hour lab per day

Prerequisites: Basic knowledge of Python programming is helpful but not mandatory.

Grading: Assignment and Final Assessment

Note: This course is both analytical and practical. Attendance and participation in class are required

COURSE	COURSE TITLE	L	T	P	C

CODE					
10215CS129	AWS Migration and Cloud Adoption Expert	1	0	0	1

Course Category: Industry/Higher Learning Institute Interaction

Resource Person: Mr. Barath Kumar.D, Designation: Associate - Tech Lead Industry
Cognizant Technology Solutions

Title: AWS Migration and Cloud Adoption Expert

Course Objective:

This course aims to equip learners with advanced knowledge and hands-on expertise in planning, executing, and managing cloud migration strategies using AWS. It focuses on best practices for cloud adoption, modernization, security, and cost optimization to help organizations transition smoothly and efficiently to AWS cloud environments.

Course Content:

- On-Premises Storage Technologies
- Introduction to Cloud Computing
- AWS Cloud Computing
- On-Premises to AWS Migration
- Cloud Adoption Frameworks and Best Practices
- Hands-On AWS Projects for Beginners
- Career Paths and Opportunities in Cloud Computing

Lecture Schedule: 1 lecture hours and one two-hour lab per day

Prerequisites: Basic knowledge of Python programming is helpful but not mandatory.

Grading: Assignment and Final Assessment

Note: This course is both analytical and practical. Attendance and participation in class are required.