NATIONAL ENGINEERING COLLEGE

(An Autonomous Institution Affiliated to Anna University Chennai)

K.R.NAGAR, KOVILPATTI

www.nec.edu.in

Estd : 1984

REGULATIONS – 2023

CURRICULUM & SYLLABUS

B. E. COMPUTER SCIENCE AND ENGINEERING

(Outcome Based Education & Choice Based Credit System)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

I. VISION

To produce globally competent, innovative and socially responsible computing professionals

II. MISSION

- To provide world-class teaching-learning and research facilities
- To stimulate students' logical thinking, creativity, and communication skills
- To cultivate awareness about emerging trends through self-initiative
- To instill a sense of societal and ethical responsibilities
- To collaborate with industries and government organizations

III. PROGRAM EDUCATIONAL OBJECTIVES (PEOs)

PEO 1: Achieve their professional career in industry/academia by applying the acquired knowledge of computer science and engineering.

PEO 2: Engage in life-long learning and enhance their capabilities by embracing cutting edge technical advancements.

PEO 3: Excel in collaboration with interdisciplinary teams and diverse stake holders for persevering successful start-ups.

IV. PROGRAM SPECIFIC OUTCOMES (PSOs)

PSO1: Build domain specific expertise by showcasing deliverables in the field of Application development, Business Intelligence, Computational Intelligence and Cyber Security

PSO2: Build knowledge base for students to solve complex technical problems through participation in global contests and hackathons.

V. PROGRAM OUTCOMES (POs)

PO 1: Apply **knowledge** of mathematics, natural science, computing and engineering **fundamentals**, with **specializations** in computational intelligence, web development, business intelligence, and cyber security to develop solutions to **complex engineering problems**

PO 2: Identify, formulate, research literature and analyze **complex engineering problems** reaching substantiated conclusions using **first principles** of mathematics, natural sciences and engineering sciences with holistic considerations for sustainable development

PO 3: **Design creative solutions** for **complex engineering problems** and design systems, components or processes to meet identified needs with appropriate consideration for public health and safety, whole-life cost, net zero carbon as well as resource, cultural, societal, and environmental considerations as required

PO 4: Conduct **investigations** of **complex engineering problems** using research methods including **research-based knowledge, design of experiments, analysis and interpretation of data, and synthesis of information** to provide valid conclusions

PO 5: **Create, select and apply, and recognize** limitations of appropriate techniques, resources, and modern engineering and IT tools, including prediction and modeling, to **complex engineering problems**

PO 6: When solving complex engineering problems, analyze and evaluate sustainable development impacts to: society, the economy, sustainability, health and safety, legal frameworks, and the environment

PO 7: Apply ethical principles and commit to professional ethics and norms of engineering practice and adhere to relevant national and international laws. Demonstrate an understanding of the need for diversity and inclusion

PO 8: Function effectively as an individual, and as a member or leader in diverse and inclusive teams and in multi-disciplinary, face-to-face, remote and distributed settings

PO 9: Communicate effectively and inclusively on **complex engineering activities** with the engineering community and with society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations, taking into account cultural, language, and learning differences

PO10: 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: 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

REGULATIONS 2023

CURRICULUM AND SYLLABUS

SEMESTER - I

S. No	Course Code	Course Title	Category	Periods Per Week				Total Contact Periods	Credits
				L	T	P	E		
Induction Programme – 2 weeks									**0**
Theory Courses									
	23SH11C	தமிழர் மரபு / Heritage of Tamils	HSMC	1	0	0	0	1	1
	23SH12C	Mathematical Foundations for Engineers	BSC	3	1	0	0	4	4
	23SH13C	Introduction to Engineering	ESC	1	0	0	0	1	1
Integrated Courses									
	23SH14C	Technical English	HSMC	1	0	2	0	3	2
	23SH15C	Engineering Physics	BSC	2	0	2	0	4	3
	23SH16C	Engineering Chemistry	BSC	2	0	2	0	4	3
	23CS11C	Problem Solving Techniques	ESC	3	0	2	0	5	4
	23EE11C	Basic Electrical and Electronics Engineering	ESC	3	0	2	0	5	4
			TOTAL	**16**	**1**	**10**	**0**	**27**	**22**

SEMESTER – II

S. No	Course Code	Course Title	Category	Periods Per Week				Total Contact Periods	Credits
				L	T	P	E		
Theory Courses									
1.	23SH21C	தமிழரும் தொழில்நுட்பமும் / Tamils & Technology	HSMC	1	0	0	0	1	1
2.	23GN01C	Aptitude Essentials	EEC	1	0	0	0	1	1
3.	23CS21C	Digital Principles and System Design	ESC	3	0	0	0	3	3
4.	23CS22C	Discrete Mathematics	BSC	3	1	0	0	4	4
Integrated Courses									
5.	23SH22C	Professional English	HSMC	1	0	2	0	3	2
6.	23CS23C	Semiconductor and Quantum Physics	ESC	2	0	2	0	4	3
7.	23ME11C	Engineering Graphics	ESC	2	0	4	0	6	4
8.	23CS24C	Object Oriented Programming using C++	PCC	2	0	2	2	6	4
Practical Courses									
9.	23GN02C	Innovation through Design Thinking	EEC	0	0	0	4	4	2
			TOTAL	**15**	**1**	**12**	**4**	**32**	**24**

SEMESTER – III

S. No	Course Code	Course Title	Category	Periods Per Week				Total Contact Periods	Credits
				L	T	P	E		
Theory Courses									
1.	23CS31C	Computer Organization and Architecture	PCC	3	0	0	0	3	3
2.	23CS32C	Data Structures	PCC	3	0	0	0	3	3
3.	23CS33C	Operating Systems	PCC	2	1	0	0	3	3
4.	23GN05C	Professional Ethics and Human Values	HSMC	2	0	0	0	2	2
5.	23CS34C	Probability and Statistics	BSC	3	1	0	0	4	4
Integrated Courses									
6.	23CS35C	Computer Networks	PCC	3	0	2	0	5	4
Practical Courses									
7.	23CS36C	Data Structures Laboratory	PCC	0	0	4	0	4	2
8.	23CS37C	Linux System Administration	PCC	0	0	2	0	2	1
9.	23GN03C	Intellectual Property Rights Study	EEC	0	0	0	4	4	2
10.	23GN04C	Aptitude Excellence	EEC	0	0	2	0	2	1
			TOTAL	**16**	**2**	**10**	**4**	**32**	**25**

SEMESTER – IV

S. No	Course Code	Course Title	Category	Periods Per Week				Total Contact Periods	Credits
				L	T	P	E		
Theory Courses									
1.	23CS41C	Java for Developers	PCC	3	0	0	0	3	3
2.	23CS42C	Cryptography and Cyber Security	PCC	3	0	0	0	3	3
3.	23CS43C	Embedded Systems	ESC	3	0	0	0	3	3
4.	23SHXXE	Elective – Science Stream	BSC	3	0	0	0	3	3
5.	23MC02C	Environmental Science and Engineering	MC	2	0	0	0	2	0
Integrated Courses									
6.	23CS44C	Design and Analysis of Algorithm	PCC	3	0	2	0	5	4
7.	23CS45C	Database Management Systems	PCC	3	0	2	0	5	4
Practical Courses									
8.	23CS46C	Java Programming Laboratory	PCC	0	0	2	2	4	2
9.	23CS47C	System Modeling Projects	EEC	0	0	2	2	4	2
			TOTAL	**20**	**0**	**8**	**4**	**32**	**24**

SEMESTER – V

S. No	Course Code	Course Title	Category	Periods Per Week				Total Contact Periods	Credits
				L	T	P	E		
Theory Courses									
1.	23CS51C	Theory of Computation	PCC	3	1	0	0	4	4
2.	23CS52C	Object Oriented Analysis and Design	PCC	3	0	0	0	3	3
3.	23CSCSE	Program Elective Course - I	PEC	3	0	0	0	3	3
Integrated Courses									
4.	23CS53C	Devops and Agile Methodologies	PCC	1	0	2	0	3	2
5.	23CS54C	Modern Web Technologies	PCC	1	0	2	4	7	4
6.	23CS55C	Artificial Intelligence	PCC	3	0	2	0	5	4
7.	23CSCSE	Program Elective Course - II	PEC	2	0	2	0	4	3
Practical Courses									
8.	23CS56C	Simulation using Modern Tools	EEC	0	0	2	2	4	2
			TOTAL	**16**	**1**	**10**	**6**	**33**	**25**

SEMESTER – VI

S. No	Course Code	Course Title	Category	Periods Per Week				Total Contact Periods	Credits
				L	T	P	E		
Theory Courses									
1.	23GN06C	Project Management and Finance	HSMC	2	0	0	0	2	2
2.	23CS61C	Business Process Management	PCC	3	0	0	0	3	3
3.	23CS62C	Compiler Design	PCC	3	1	0	0	4	4
4.	23CSCSE	Program Elective Course - III	PEC	3	0	0	0	3	3
5.	23YYXXN	Open Elective Course – I	OEC	3	0	0	0	3	3
6.	23MC01C	Constitution of India	MC	2	0	0	0	2	0
Integrated Courses									
7.	23CS63C	Data Science	PCC	2	0	2	2	6	4
8.	23CSXXE	Program Elective Course - IV	PEC	2	0	2	0	4	3
Practical Courses									
9.	23CS64C	Product Development Practice	EEC	0	0	0	4	4	2
			TOTAL	**20**	**1**	**4**	**6**	**31**	**24**

SEMESTER – VII

S. No	Course Code	Course Title	Category	Periods Per Week				Total Contact Periods	Credits
				L	T	P	E		
Theory Courses									
1.	23CSCSE	Program Elective Course - V	PEC	3	0	0	0	3	3
2.	23YYXXE	Open Elective Course – II	OEC	3	0	0	0	3	3
3.	23YYXXE	Open Elective Course - III	OEC	3	0	0	0	3	3
Practical Courses									
4.	23CSXXE	Program Elective Course - VI	PEC	0	0	0	6	6	3
5.	23CS71C	Mini Project	EEC	0	0	0	6	6	3
6.	23CS72C	Internship / In-Plant Training	EEC	0	0	0	4	-	2
			TOTAL	**9**	**0**	**0**	**12**	**21**	**17**

SEMESTER – VIII

S. No	Course Code	Course Title	Category	Periods Per Week				Total Contact Periods	Credits
				L	T	P	E		
Practical Courses									
1.	23CS81C	Capstone Project / Industry Practices	EEC	0	0	0	12	12	6
			TOTAL	**0**	**0**	**0**	**12**	**12**	**6**

Total No. of Credits: 167

PROGRAMME ELECTIVE COURSES (PEC)

S. No	Course Category	Course Code	Course Name	L	T	P	E	C
BLOCKCHAIN TECHNOLOGY								
1.	PEC	23CS01E	Blockchain Architecture and Design	3	0	0	0	3
2.	PEC	23CS02E	Smart Contract and DApp	2	0	2	0	3
3.	PEC	23CS03E	Advanced Ethereum	3	0	0	0	3
4.	PEC	23CS04E	Blockchain Security	2	0	2	0	3
5.	PEC	23CS05E	Permissioned Blockchain	2	0	2	0	3
6.	PEC	23CS06E	Blockchain Use Cases and applications	3	0	0	0	3
7.	PEC	23CS07E	Blockchain in FinTech	3	0	0	0	3
8.	PEC	23CS33E	Blockchain Gaming	2	0	2	0	3
9.	PEC	23CS01L	Zero-Knowledge Proofs in Blockchain	1	0	0	0	1

S. No	Course Category	Course Code	Course Name	L	T	P	E	C
10.	PEC	23CS0ME	Mini-Capstone Project	0	0	0	6	3
FULLSTACK DEVELOPMENT								
11.	PEC	23CS11E	Web Frameworks using Python	2	0	2	0	3
12.	PEC	23CS12E	No SQL Databases	3	0	0	0	3
13.	PEC	23CS13E	Open Source Technologies for Web Development	3	0	0	0	3
14.	PEC	23CS14E	PostgreSQL Database Management	2	0	2	0	3
15.	PEC	23CS15E	Mobile Application Development	2	0	2	0	3
16.	PEC	23CS17E	Web Application Testing	2	0	2	0	3
17.	PEC	23CS18E	Fullstack App with Flutter and Firebase	2	0	2	0	3
18.	PEC	23CS32E	UI/UX Design	2	0	0	2	3
19.	PEC	23CS1ME	Mini-Capstone Project	0	0	0	6	3
COMPUTATIONAL INTELLIGENCE								
20.	PEC	23CS21E	Computer Vision and Image Processing	3	0	0	0	3
21.	PEC	23CS22E	Machine Learning Techniques	2	0	2	0	3
22.	PEC	23CS23E	AI tools for Natural Language Processing	2	0	2	0	3
23.	PEC	23CS24E	Soft Computing	3	0	0	0	3
24.	PEC	23CS25E	Fundamentals of Deep Learning	2	0	2	0	3
25.	PEC	23CS26E	Time Series Analysis	2	0	2	0	3
26.	PEC	23CS27E	Nature and Bio-inspired Computing	3	0	0	0	3
27.	PEC	23CS43E	Reinforcement Learning	2	0	2	0	3
28.	PEC	23CS47E	Responsible AI	3	0	0	0	3
29.	PEC	23CS67E	Feature Engineering	2	0	2	0	3
30.	PEC	23CS21L	Generative AI Technologies	0	0	2	0	1
31.	PEC	23CS22L	Agentic AI Tools and Frameworks	0	0	2	0	1
32.	PEC	23CS23L	Accelerated AI Development using CUDA	0	0	2	0	1
33.	PEC	23CS24L	Fundamentals of Remote Sensing and GIS	1	0	0	0	1
34.	PEC	23CS2ME	Mini-Capstone Project	0	0	0	6	3
AUGMENTED REALITY/ VIRTUAL REALITY								
35.	PEC	23CS31E	Computer Graphics	3	0	0	0	3
36.	PEC	23CS32E	UI/UX Design	2	0	0	2	3
37.	PEC	23CS33E	Blockchain Gaming	2	0	2	0	3
38.	PEC	23CS34E	3D Modeling and Design	2	0	2	0	3
39.	PEC	23CS35E	Augmented Reality and Virtual Reality	3	0	0	0	3
40.	PEC	23CS36E	AI For Human Computer Interaction	3	0	0	0	3
41.	PEC	23CS37E	Design of Virtual Reality Systems	1	0	0	4	3
42.	PEC	23CS38E	Metaverse Development	3	0	0	0	3
43.	PEC	23CS01E	Blockchain Architecture and Design	3	0	0	0	3
44.	PEC	23CS31L	Creative Designing with FIGMA	0	0	0	2	1
45.	PEC	23CS3ME	Mini-Capstone Project	0	0	0	6	3
INDUSTRIAL AI								
46.	PEC	23CS41E	Generative AI	3	0	0	0	3
47.	PEC	23CS42E	AI for Industrial applications	3	0	0	0	3
48.	PEC	23CS43E	Reinforcement Learning	2	0	2	0	3
49.	PEC	23CS44E	AI for Robotics	3	0	0	0	3
50.	PEC	23CS45E	AI in Supply Chain	3	0	0	0	3
51.	PEC	23CS46E	Ethical AI	3	0	0	0	3

S. No	Course Category	Course Code	Course Name	L	T	P	E	C
52.	PEC	23CS47E	Responsible AI	3	0	0	0	3
53.	PEC	23CS48E	Artificial General Intelligence	3	0	0	0	3
54.	PEC	23CS49E	AI Model Deployment using MLOPs	2	0	2	0	3
55.	PEC	23CS4AE	Bio Informatics	3	0	0	0	3
56.	PEC	23CS41L	Intelligent Dashboard Development using Modern Tools	0	0	2	0	1
57.	PEC	23CS4ME	Mini-Capstone Project	0	0	0	6	3
CYBER SECURITY								
58.	PEC	23CS51E	Network Security	3	0	0	0	3
59.	PEC	23CS52E	Information Security	2	0	2	0	3
60.	PEC	23CS53E	Cloud Security	3	0	0	0	3
61.	PEC	23CS54E	Security and Privacy in IoT	2	0	2	0	3
62.	PEC	23CS55E	Social Network Security	3	0	0	0	3
63.	PEC	23CS56E	Cyber Security and Ethical Hacking	2	0	2	0	3
64.	PEC	23CS57E	Firewalls and Intrusion Detection Systems	3	0	0	0	3
65.	PEC	23CS58E	Threat Intelligence and Risk Management	3	0	0	0	3
66.	PEC	23CS59E	Digital Forensics	3	0	0	0	3
67.	PEC	23CS5AE	Malware Analysis	3	0	0	0	3
68.	PEC	23CS51L	Tools for Network Protection	0	0	2	0	1
69.	PEC	23CS5ME	Mini-Capstone Project	0	0	0	6	3
BUSINESS ANALYTICS								
70.	PEC	23CS61E	Principles of Business Analytics	3	0	0	0	3
71.	PEC	23CS62E	Data Mining	3	0	0	0	3
72.	PEC	23CS63E	Data Engineering	2	0	2	0	3
73.	PEC	23CS64E	Image and Video Analytics	3	0	0	0	3
74.	PEC	23CS65E	Healthcare Analytics	3	0	0	0	3
75.	PEC	23CS66E	Prescriptive Analytics and optimization	3	0	0	0	3
76.	PEC	23CS67E	Feature Engineering	2	0	0	2	3
77.	PEC	23CS68E	Predictive Analytics	3	0	0	0	3
78.	PEC	23CS61L	Data Analytics Using Snowflake	0	0	2	0	1
79.	PEC	23CS6ME	Mini-Capstone Project	0	0	0	6	3
THEORETICAL COMPUTER SCIENCE								
80.	PEC	23CS71E	Algorithmic Game Theory	3	0	0	0	3
81.	PEC	23CS72E	Approximation Algorithms	3	0	0	0	3
82.	PEC	23CS73E	Parallel and Randomized Algorithms	3	0	0	0	3
83.	PEC	23CS74E	Computational Graph Theory	2	0	2	0	3
84.	PEC	23CS75E	Computational Number Theory	2	0	2	0	3
85.	PEC	23CS76E	Quantum Algorithms and Qiskit	2	0	2	0	3
86.	PEC	23CS77E	Computational Complexity Theory	3	0	0	0	3
87.	PEC	23CS7ME	Mini-Capstone Project	0	0	0	6	3

ELECTIVE COURSES (SCIENCE STREAM)

S. No	Course Category	Course Code	Course Name	L	T	P	E	C
MATHEMATICS								
1.	BSC	23SH01E	Linear Algebra, Mathematical Logic and Set Theory	2	1	0	0	3
2.	BSC	23SH02E	Linear Structures and Transformations	2	1	0	0	3
3.	BSC	23SH03E	Number Theory	2	1	0	0	3
4.	BSC	23SH04E	Numerical Analysis	2	1	0	0	3
5.	BSC	23SH05E	Optimization Techniques	2	1	0	0	3
6.	BSC	23SH06E	Principles of Discrete Mathematics	2	1	0	0	3
7.	BSC	23SH07E	Random Processes and Queuing Theory	2	1	0	0	3
8.	BSC	23SH08E	Statistical Techniques and Numerical Methods	2	1	0	0	3
9.	BSC	23SH09E	Transforms, Mathematical Logic and Set Theory	2	1	0	0	3
PHYSICS								
10.	BSC	23SH10E	Fundamentals of Laser Technology	3	0	0	0	3
11.	BSC	23SH11E	Nanomaterials for Engineers	3	0	0	0	3
12.	BSC	23SH12E	Photonics	3	0	0	0	3
CHEMISTRY								
13.	BSC	23SH13E	Biology for Computing	3	0	0	0	3
14.	BSC	23SH14E	Biology for Engineers	3	0	0	0	3
15.	BSC	23SH15E	Polymer Science and Technology	3	0	0	0	3
16.	BSC	23SH16E	Sensors for Engineering Applications	3	0	0	0	3

OPEN ELECTIVE COURSES (Diversified Technologies)

S. No	Course Category	Course Code	Course Name	L	T	P	E	C
1	OEC	23CS01N	Python Programming	2	0	2	0	3
2	OEC	23CS02N	Foundations of Object Oriented Programming	2	0	2	0	3
3	OEC	23CS03N	Computer Graphics and Virtual Reality	3	0	0	0	3
4	OEC	23CS04N	Data Structures and Algorithms	2	0	2	0	3
5	OEC	23CS05N	SQL Programming	2	0	2	0	3
6	OEC	23CS06N	Fullstack Development	2	0	2	0	3

Category	I	II	III	IV	V	VI	VII	VIII	Credits	Percentage of credits
HSMC	3	3	2			2			**10**	**6.0%**
BSC	10	4	4	3					**21**	**12.6%**
ESC	9	10		3					**22**	**13.17%**
PCC		4	16	16	14	14			**64**	**38.32%**
PEC					9	3	6		**18**	**10.8%**
OEC						3	6		**9**	**5.4%**
EEC		3	3	2	2	2	5	6	**23**	**13.8%**
Total	22	24	25	24	25	24	17	6	**167**	

Course Code	**தமிழர் மரபு (HERITAGE OF TAMILS)**	**L**	**T**	**P**	**E**	**C**
23SH11C	(Common to all B.E. / B.Tech. Degree Programmes)	**1**	**0**	**0**	**0**	**1**

COURSE OUTCOMES

இப்பாடம் முடிந்ததும் மாணவர்களிடம் வளரும் திறன்

CO1: தமிழ் மொழியின் இலக்கிய வளம், ஓவிய, சிற்பக் கலையின் பரிணாம வளர்ச்சி நாட்டுப்புறக் கலை மற்றும் வீர விளையாட்டுக்கள் பற்றிய அறிவு மற்றும் விளக்கும் திறன்

CO2: தமிழர்களின் திணை சார் கோட்பாடுகள் மற்றும் இந்திய பண்பாட்டில் தமிழர்களின் பங்கு பற்றிய அறிவு மற்றும் விளக்கும் திறன்

Upon the successful completion of the course, the student will be able to

Theory Component

CO1:know andexplain about Tamil literary resources, Dimensional growth of painting and sculpture arts, folk art and martial arts.

CO2: know andexplain about Tamils Thinai concepts, contribution of Tamils in Indian National Movements and Indian Culture

CO1: தமிழ் மொழியின் இலக்கிய வளம், ஓவிய, சிற்பக் கலையின் பரிணாம வளர்ச்சி நாட்டுப்புறக்கலை மற்றும் வீர விளையாட்டுக்கள் பற்றிய அறிவு மற்றும் விளக்கும் திறன் **L:9**

இந்திய மொழிக்குடும்பங்கள் - திராவிட மொழிகள் - தமிழ் ஒரு செம்மொழி – தமிழ் செவ்விலக்கியங்கள் - சங்க இலக்கியத்தின் சமயச் சார்பற்ற தன்மை – சங்க இலக்கியத்தில் பகிர்தல் அறம் - திருக்குறளில் மேலாண்மைக் கருத்துக்கள் - தமிழ்க் காப்பியங்கள், தமிழகத்தில் சமண பௌத்த சமயங்களின் தாக்கம் - பக்தி இலக்கியம், ஆழ்வார்கள் மற்றும் நாயன்மார்கள் - சிற்றிலக்கியங்கள் - தமிழில் நவீன இலக்கியத்தின் வளர்ச்சி - தமிழ் இலக்கிய வளர்ச்சியில் பாரதியார் மற்றும் பாரதிதாசன் ஆகியோரின் பங்களிப்பு - நடுகல் முதல் நவீன சிற்பங்கள் வரை – ஐம்பொன் சிலைகள் - பழங்குடியினர் மற்றும் அவர்கள் தயாரிக்கும் கைவினைப் பொருட்கள், பொம்மைகள் - தேர் செய்யும் கலை - சுடுமண் சிற்பங்கள் - நாட்டுப்புறத் தெய்வங்கள் - குமரிமுனையில் திருவள்ளுவர் சிலை - இசைக் கருவிகள் - மிருதங்கம், பறை, வீணை, யாழ், நாதஸ்வரம் - தமிழர்களின் சமூக பொருளாதார வாழ்வில் கோவில்களின் பங்கு - தெருக்கூத்து, கரகாட்டம், வில்லுப்பாட்டு, கணியான் கூத்து, ஒயிலாட்டம், தோல்பாவைக் கூத்து, சிலம்பாட்டம், வளரி, புலியாட்டம், தமிழர்களின் விளையாட்டுகள்.

CO1:know andexplain about Tamil literary resources, Dimensional growth of painting and sculpture arts, folk art and martial arts.

Language Families in India - Dravidian Languages – Tamil as a Classical Language – ClassicalLiterature in Tamil – Secular Nature of Sangam Literature – Distributive Justice in Sangam Literature- Management Principles in Thirukural - Tamil Epics and Impact of Buddhism & Jainism in Tamil Land- Bakthi Literature Azhwars and Nayanmars - Forms of minor Poetry - Development of Modernliterature in Tamil - Contribution of Bharathiyar and Bharathidhasan - Hero stone to modern sculpture - Bronze icons - Tribes and their handicrafts - Art of temple carmaking - - Massive Terracotta sculptures, Village deities, Thiruvalluvar Statue at Kanyakumari,Making of musical instruments - Mridhangam, Parai, Veenai, Yazh and Nadhaswaram - Role ofTemples in Social and Economic Life of Tamils - Therukoothu, Karagattam, Villu Pattu, Kaniyan Koothu, Oyillattam, Leather puppetry, Silambattam,Valari, Tiger dance - Sports and Games of Tamils.

CO2:தமிழர்களின் திணை சார் கோட்பாடுகள் மற்றும் இந்திய பண்பாட்டில் தமிழர்களின் பங்கு பற்றிய அறிவு மற்றும் விளக்கும் திறன் **L:6**

தமிழகத்தின் தாவரங்களும், விலங்குகளும் - தொல்காப்பியம் மற்றும் சங்க இலக்கியத்தில் அகம் மற்றும் புறக் கோட்பாடுகள் - தமிழர்கள் போற்றிய அறக்கோட்பாடு – சங்ககாலத்தில் தமிழகத்தில் எழுத்தறிவும், கல்வியும் - சங்ககால நகரங்களும் துறை முகங்களும் - சங்ககாலத்தில் ஏற்றுமதி மற்றும் இறக்குமதி கடல்கடந்த நாடுகளில் சோழர்களின் வெற்றி - இந்திய விடுதலைப் போரில் தமிழர்களின் பங்கு - இந்தியாவின் பிறப்பகுதிகளில் தமிழ்ப் பண்பாட்டின் தாக்கம் - சுயமரியாதை இயக்கம் - இந்திய மருத்துவத்தில், சித்த மருத்துவத்தின் பங்கு – கல்வெட்டுகள், கையெழுத்துப் படிகள் - தமிழ்ப் புத்தகங்களின் அச்சு வரலாறு.

CO2: know andexplain about Tamils Thinai concepts, contribution of Tamils in Indian National Movements and Indian Culture

Flora and Fauna of Tamils & Aham and Puram Concept from Tholkappiyam and Sangam Literature -Aram Concept of Tamils - Education and Literacy during Sangam Age - Ancient Cities and Ports ofSangam Age - Export and Import during Sangam Age - Overseas Conquest of Cholas - Contribution of Tamils to Indian Freedom Struggle - The Cultural Influence of Tamils over the otherparts of India – Self-Respect Movement - Role of Siddha Medicine in Indigenous Systems ofMedicine – Inscriptions & Manuscripts – Print History of Tamil Books.

REFERENCES:

1. தமிழக வரலாறு –மக்களும் பண்பாடும் - கே.கே.பிள்ளை (வெளியீடு:தமிழ்நாடு பாடநூல் மற்றும் கல்வியியல் பணிகள் கழகம்)
2. கணினித் தமிழ் - முனைவர்.இல.சுந்தரம் (விகடன் பிரசுரம்)
3. கீழடி – வைகை நதிக்கரையில் சங்ககால நகர நாகரீகம் (தொல்லியல் துறை வெளியீடு)
4. பொருநை –ஆற்றங்கரை நாகரீகம். (தொல்லியல் துறை வெளியீடு)
5. Social Life of Tamils (Dr.K.K.Pillay) A joint publication of TNTB & ESC and RMRL (in print)
6. Social Life of the Tamils - The Classical Period (Dr.S.Singaravelu) (Published by: International Institute of Tamil Studies.
7. Historical Heritage of the Tamils (Dr.S.V.Subatamanian, Dr.K.D. Thirunavukkarasu) (Published by: International Institute of Tamil Studies).
8. The Contributions of the Tamils to Indian Culture (Dr.M.Valarmathi) (Published by: International Institute of Tamil Studies.)
9. Keeladi - 'Sangam City C ivilization on the banks of river Vaigai' (Jointly Published by: Department of Archaeology & Tamil Nadu Text Book and Educational Services Corporation, Tamil Nadu)
10. Studies in the History of India with Special Reference to Tamil Nadu (Dr.K.K.Pillay) (Published by: The Author)
11. Porunai Civilization (Jointly Published by: Department of Archaeology & Tamil Nadu Text Book and Educational Services Corporation, Tamil Nadu)
12. Journey of Civilization Indus to Vaigai (R.Balakrishnan) (Published by: RMRL) – Reference Book.

L: 15; TOTAL: 15 PERIODS

Course Code	MATHEMATICAL FOUNDATIONS FOR ENGINEERS	L	T	P	E	C
23SH12C	(Common to all B.E. / B.Tech. Degree Programmes)	**3**	**1**	**0**	**0**	**4**

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Component

CO1: interpretthe nature of quadratic form by orthogonal transformation.
CO2: identify the maxima and minima of functions.
CO3: solve ordinary differential equations.
CO4: find the solution of partial differential equations.
CO5: evaluate integrals of multivariate calculus.

Soft skill Component

CO6 :develop communication, problem solving and interpersonal skills

CO1: Interpret the nature of quadratic form by orthogonal transformation. **L:9; T:3**

Characteristic equation – Eigen values and Eigen vectors of a real matrix – Properties of Eigen values and Eigen vectors – Diagonalisation of a matrix by orthogonal transformation – Quadratic forms – Reduction of quadratic form to canonical form by orthogonal transformation and its nature; Cayley – Hamilton theorem (excluding proof) -Application: Stretching of a elastic membrane.

CO2: Identify the maxima and minima of functions. **L:9; T:3**

Functions of two variables: Limit, continuity and partial derivatives; Total derivative, Jacobian, Taylor series- Application :Linearization of Non Linear systems using Taylor Series - Maxima and minima - Method of Lagrange multipliers.

CO3: Solve ordinary differential equations. **L:9; T:3**

Solutions of first order ordinary differential equations - Equations solvable for ‘p’, equations solvable for ‘y’, equations solvable for ‘x’ - Solutions of higher order linear differential equations with constant coefficients – Cauchy’s and Legendre’s linear equations - Method of variation of parameters – Solution of simultaneous linear differential equation. Application RCL – circuit and Mass Spring System.

CO4: Find the solution of partial differential equations. **L:9; T:3**

Formation of partial differential equations – Solutions of standard types of first order partial differential equations - Lagrange’s linear equations - Solutions of homogeneous and Non homogeneous linear partial differential equations of second and higher order with constant coefficient – Application - Shallow wave equations of first order PDE.

CO5 : Evaluate integrals of multivariate calculus **L:9; T:3**

Double integration – Cartesian and polar coordinates - Change of order of integration - Change of variables - Cartesian to polar coordinates - Area as double integral - Triple integration - Cartesian and polar coordinates – Change of Variables- Cartesian to spherical and cylindrical coordinates. Application – Moments and centers of mass.

TEXT BOOKS:

1. Grewal. B.S., Higher Engineering Mathematics, Khanna Publications, 44th Edition, 2021.
2. James E. Gentle, Matrix Algebra, Springer International Publishing, 2nd Edition, 2017
3. Shanker Rao.G., Linear Albebra, WileyIndia, 1st Edition , 2017

REFERENCES:

1. Bali.N.P. and Manish Goyal, A Textbook of Engineering Mathematics, Laxmi Publications Private Limited, 10th Edition, 2016.
2. Erwin Kreyszig, Advanced Engineering Mathematics, Wiley India, 10th Edition, 2017.
3. Kenneth B. Howell, Ordinary Differential Equations, CRC Press, 2020.
4. James Strewart, Daniel Clegg, Saleem Watson, Essential Calculus Early Trancendentals, Cengage Learning, 9th Edition, 2021.
5. Nanda Kumar A.K, P.S.Datti: Raju .K.George , Ordinary Differential Equations, Cambridge University press, 2017.

L: 45; T: 15; TOTAL: 60 PERIODS

Course Code	INTRODUCTION TO ENGINEERING	L	T	P	E	C
23SH13C	(Common to all B.E. / B.Tech. Degree Programmes)	**1**	**0**	**0**	**0**	**1**

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Component

CO1: articulate the importance of Engineering and its role in society through OBE framework
CO2: identify and describe academic pathways towards career settlement

CO1: Articulate the importance of Engineering and its role in society through OBE framework

Engineering – An introduction, Classification of different Engineering Disciplines, Role of Engineers in Society. Graduate Attributes (GA), Program Specific Criteria (PSC)- Program Educational Objectives (PEO), Program Outcomes (PO), Course Outcomes (CO), Choice Based Credit System (CBCS), course categories, teaching and learning process, active and passive learning, project / problem based learning, different assessments process. **L:9**

CO2: Identify and describe academic pathways towards career settlement

Curriculum, cafeteria curriculum and self learning big picture of the Program and the significance of each course in the undergraduate Engineering Program, Discuss the different career paths for an engineering graduate. Career objective, competency requirement. **L:6**

Case study: Each student has to interact with alumni mentors/seniors/faculty members/surf the internet and present a career path that inspires him/her at the end of the course

REFERENCES:

1. Quamrul H. MazumderIntroduction to Engineering, An Assessment and Problem Solving Approach, CRC Press, 1st Edition, 2016.
2. Saeed Moaveni, "Engineering Fundamentals an Introduction to Engineering", Cengage Learning, USA, 4th Edition, 2011.

L: 15; TOTAL: 15 PERIODS

Course Code	TECHNICAL ENGLISH	L	T	P	E	C
23SH14C	(Common to all B.E. / B.Tech. Degree Programmes)	**1**	**0**	**2**	**0**	**2**

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Component

CO1: apply the fundamental grammar rules in writing

CO2: utilizing phonetic transcription for pronunciation

Practical Component

CO3: apply the basic language skills in various aspects of communication

CO4: utilize technical terms and phrases in specific contexts

CO5: develop the pronunciation skill through various language components

CO6: distinguish different writing forms and interpret text through divergent thinking

CO7: develop effective reports with grammatical and language components

Soft skill Component

CO8: develop communication, team spirit, creativity and time management

CO1: Apply the fundamental grammar rules in writing **L:13; P:26**

Parts of Speech - Word Formation using Prefix and Suffix - Sentence formation (Kinds of Sentences) - Tenses (Present, Past & Future tense) – Concord

CO3: Apply the basic language skills in various aspects of communication

Diary Writing - Greeting and Self Introduction

CO4: Utilize technical terms and phrases in specific contexts

Technical terms and extended definition - Essay Writing (Argumentative Essay and Analytical Essay) - Situational phrases & Conversation - Formal Letter Writing (Permission & Requisition letters)

CO6: Distinguish different writing forms and interpret text through divergent thinking

Picture Description, Introduction to Reading Techniques (Skimming, scanning, inferring, predicting, Reading and Reviewing a book (Sci – Fi), E Mail Writing

CO7: Develop effective reports with grammatical and language components
Listening and responding to general information (Business context) - Report Writing (Types, Structure, and Stages of report writing) - Checklist

CO2: Utilizing phonetic transcription for pronunciation **L:2;**

Phonetics (Vowels & Consonants) **P:4**

CO5: Develop the pronunciation skill through various language components

Word Transformation from one form to another - Letter Writing (Informal) - Listening and responding to general information (General context)

TEXT BOOKS:

1. Paul V. Anderson, Technical Communication: A Reader - Centered Approach, Cengage Learning, 9th Edition, 2017.
2. Ravindra Nath Tiwari, Technical English-II, Shashwat Publication, 1st Edition, 2020.
3. Stephen D. Krashen, Principles and Practice in Second LanguageAcquisition. Pergamon, 1987.
4. Lester Kaufman and Jane Straus, The Blue Book of Grammar and Punctuation: An Easy-to Use Guide with Clear Rules, Real-World Examples, and Reproducible Quizzes, Wiley, 2021.
5. Wells H. G., The Time Machine, Penguin Classics, 2012.

REFERENCES:

1. Michael McCarthy, English Grammar: The Basics, Taylor & Francis, 2021.
2. Peter Lucantoni and Lydia Kellas, Cambridge IGCSE(TM) English as a Second Language Workbook, Cambridge University Press, 6th Edition, 2022.

L: 15; P: 30; TOTAL: 45 PERIODS

Course Code	ENGINEERING PHYSICS	L	T	P	E	C
23SH15C	(Common to all B.E. / B.Tech. Degree Programmes)	**2**	**0**	**2**	**0**	**3**

COURSE OUTCOMES:

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

Theory Component:

CO1: identify the structural properties of crystalline materials
CO2: comprehend and apply the concepts of centre of mass and elasticity
CO3: explain thermodynamic parameters and fundamental laws and their application in various processes
CO4: illustrate the applications of different lasers and optical fibers
CO5:interpret the quantum concepts, to illustrate the quantization of energy, and computation

Practical Component:

CO6: compare the mechanical properties of the materials due to bending and torsion
CO7: analyze thermal conductivity of different bad conducting materials
CO8: explore the light-matter interaction by the phenomenon of interference and diffraction and photoelectric effect

Soft skill Component:

CO9: develop the team spirit and communication skill through group activities

CO1: Identify the structural properties of crystalline materials L:10

Crystalline and amorphous materials - unit cell - primitive cell - crystal systems, Bravais lattices - Miller indices – interplanar distance – Characteristics of SC, BCC, FCC, HCP structures - Bragg's law - X-ray diffraction and its applications - Synthesis of crystalline materials

CO2: Comprehend and apply the concepts of centre of mass and elasticity **L:6;**
CO6: Compare the mechanical properties of the materials due to bendingand torsion **P:10**

Multi-particle dynamics - Introduction - Center of mass (CM) – CM of continuous bodies - Introduction to rigid bodies - translation - rotation – moment of inertia – theorems of moment of inertia – Torsional pendulum.

Elasticity – Stress - strain diagram and its applications - Moduli of elasticity and its relation - bending of beams - Bending moment – cantilever - theory and experiment - Uniform bending - theory and experiment – Non Uniform bending - I-shaped girders

CO3: Explain thermodynamic parameters and fundamental laws and their application in various processes **L:6; P:8**
CO7: Analyse thermal conductivity of different bad conducting materials.

Laws of thermodynamics –Thermo dynamical processes – Introduction to heat transfer – conduction - convection and radiation – thermal conductivity of good conductor –Radial flow of heat - Spherical shell method and cylindrical shell method – Thermal conductivity of poor conductor- Lee's disc method– Applications - heat exchangers - refrigerators and ovens

CO4: Illustrate the applications of different lasers and optical fibers **L:6;**
CO8: Explore the light-matter interaction by the phenomenon ofInterference and diffraction and photoelectric effect **P:6**

Lasers: Interaction of light with matter - Einstein coefficients and their relations – characteristics of laser - components of laser – Lasing action – Pumping methods – Types of Laser - Nd-YAG laser -semiconductor laser- Applications

Fiber optics: principle and classification of optical fibers – propagation of light in optical fiber - Numerical aperture and Acceptance angle – losses associated with optical fibers (Qualitative) – Fiber optic communication system - Applications - Displacement and pressure sensors – Endoscopy

CO5: Interpret the quantum concepts, to illustrate the quantization of energy, and computation **L:6; P:2**
CO8: Explore the light-matter interaction by the phenomenon of interference and diffraction and photoelectric effect

Planck's radiation law - de-Broglie hypothesis – Matter waves - Heisenberg's uncertainty principle – elementary proof – applications – Schrödinger's time-dependent and time-independent wave equation – physical significance of wave function – Introduction to quantum tunneling - applications - particle in a one-dimensional box – tunneling microscope – quantum confinement in 0D, 1D, 2D systems - quantum computation

TEXT BOOKS:

1. AvadhanuluM. N., KshirsagarP.G and Arun MurthyT.V.S, A Text book of Engineering Physics, S.Chand & Co, 11th Edition, 2018.
2. Kleppner D and Kolenkow R. An Introduction to Mechanics. McGraw Hill Education (Indian Edition), 2017.
3. Kenneth S Krane, Modern Physics, Wiley, 4th Edition, 2021.

REFERENCES:

1. Wolfson R., Essential University Physics, Volume 1 & 2, Pearson Education, 2ndIndian

Edition, 2009.
2. Hitendra K. Malik, A.K.Singh, Engineering Physics, McGraw Hill Education, 2nd Edition, 2017.
3. Kyungwon An, Fundamentals of Laser Physics, World Scientific Publishing Company, 2023
4. Halliday D, Resnick R and Walker J, Principles of Physics, Wiley, 12th Edition, 2021.

L: 30; P: 30; TOTAL: 60 PERIODS

Course Code	ENGINEERING CHEMISTRY	L	T	P	E	C
23SH16C	(Common to all B.E. / B.Tech. Degree Programmes)	**2**	**0**	**2**	**0**	**3**

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Component

CO1: explain the suitable water treatment technologies for domestic and industrial applications

CO2: apply the knowledge of corrosion to solve the industrial problems

CO3: describe the preparation, properties and their applications of smart materials in various sectors

CO4: describe the basic components and performance analysis of batteries

CO5: predict the mechanical, electrical and electronics properties of materials using various instrumentation techniques

Practical Component

CO6: estimate the amount of Ca^{2+}/ Mg^{2+}, alkalinity and Chloride ion present in the water sample.

CO7: quantify the amount of acid and metal ion in the given samples by different analytical techniques

Soft skill Component

CO8: develop interpersonal, work ethics and communications skills for career settlement

CO1: Explain the suitable water treatment technologies for domestic and industrial applications

CO6: Estimate the amount of Ca^{2+}/ Mg^{2+}, alkalinity and Chloride ion present in the water sample.

Introduction, sources and impurities in water, potable water specifications (as per WHO and BIS) - hardness-types-estimation of $Ca^{2+}$and Mg^{2+} ion in water by EDTA method. Alkalinity-types-determination of alkalinity of water -chronic daily intake - incremental life time risk - hazard quotient, hazard index, contamination factor - determination of chloride ion in water using Argentometric method-municipal water treatment- physical methods and chemical methods. Disinfection-internal conditioning - calgon and carbonate conditioning. Desalination-types-Reverse Osmosis (RO) process- Forward osmosis (FO) - electro dialysis - demineralization. **L:6; P:12**

CO2: Apply the knowledge of corrosion to solve the industrial problems.

CO7: Quantify the amount of acid and metal ion in the given samples by different analytical techniques

Corrosion – mechanism of dry and wet corrosion-forms of corrosion– galvanic corrosion and differential aeration corrosion, crevice corrosion, pitting corrosion, **L:6; P:6**

microbial corrosion-stress corrosion, intergranular corrosion - determination of rate of corrosion by weight loss method.

Protection: cathodic protection, surface coatings, corrosion inhibitors. Corrosion of industrial components: corrosion and its control in power industries, automotive industries, chemical processing industries and marine industries.

CO3: Describe the preparation,properties and their applications of smart materials in various sectors

Polymers: introduction - classification - functional polymers: electroluminescence polymer, biodegradable polymers, fire retardant polymer, thermo responsive polymer - piezo, ferro and pyroelectric polymer - nanocomposites: introduction, synthesis, properties & applications- synthesis of nanocomposites using sol -gel process **L:6**

CO4: Describe the basic components and performance analysis of batteries

Introduction - components - operation principle - Lead acid – Nickel metal hydride batteries- Lithium ions batteries: Lithium polymer battery, Lithium sulphur battery - fabrication and performance evaluation- safety issues - battery management system - recycling of lithium batteries. **L:6**

CO5: Predict the mechanical, electrical and electronics properties of materials using various instrumentation techniques

CO7: Quantify the amount of acid and metal ion in the given samples by different analytical techniques.

Spectroscopy methods: Beer-Lambert's law and its limitations– UV-visible spectroscopy and IR spectroscopy – principle - instrumentation– applications. Estimation of copper. Electro analytical methods: potentiometric titration - Estimation of Fe^{2+} ion by potentiotometric method. Conductometric method-estimation of HCl by conductometric titration- pH metric method-Estimation of HCl by pH metric titration- applications. Thermal analytical methods: Thermal Gravimetric Analysis (TGA) and Differential Thermal Analysis (DTA)- Thermo Mechanical Analysis (TMA) –principle - instrumentation - Thermo gravimetric analysis of $CuSO_4.5H_2O$- applications. **L:6; P:12**

TEXT BOOKS:

1. Jain P.C. and Jain M, Engineering Chemistry, Dhanpat Rai Publishing Company, New Delhi, 17th Edition, 2021.
2. Dara S.S and Umare S.S, A Text Book of Engineering Chemistry, S.Chand & Company Limited, 20th Edition, 2018.
3. Agarwal S, Engineering Chemistry, Cambridge Publishing Company, 2ndEdition, 2019

REFERENCES:

1. Benjamin M. M, Water Chemistry, Waveland Press, 2nd Edition, 2019.
2. Cicek V, Corrosion Engineering, Springer Publishing, 1st Edition, 2021.
3. Shahinpoor. M, Fundamentals of Smart Materials, Publisher: Royal Society of Chemistry, 1st Edition, 2020.
4. Berg H, Bernhardsson S, and Johansson P, Electric Vehicle Batteries: Moving from Research towards Innovation, Publisher: Springer, 1st Edition, 2019.

5. Crouch S, Skoog D, Holler F, Principles of Instrumental Analysis, 2017.

L: 30; P: 30; TOTAL: 60 PERIODS

Course Code	PROBLEM SOLVING TECHNIQUES	L	T	P	E	C
23CS11C	(Common to all B.E. / B.Tech. Degree Programmes)	**3**	**0**	**2**	**0**	**4**

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Component

CO1: apply fundamentals of problem solving techniques to develop simple algorithms for arithmetic and logical problems (CDL1)

CO2: apply fundamental, sequential, conditional logic statements and arrays for solving basic problems (CDL1)

CO3: implement modular programming concept using user defined functions (CDL2)

CO4: inscribe programs using pointers and to allocate memory for user defined data types using dynamic memory management functions (CDL2)

CO5: develop file processing application programs (CDL1)

Practical Component

CO6: develop programs for simple algorithms using sequential and Control structures (PDL1)

CO7: inscribe programs using arrays, functions and pointers to work with multiple data items.(PDL1)

CO8: develop application programs using structures and files concept. (PDL1)

CO1: Apply fundamentals of problem solving techniques to develop simple algorithms for arithmetic and logical problems **L:6**

Overview of programming: Problem Solving in Everyday Life, Types of Problem, Computer-based problem solving, Algorithms - Building blocks of algorithms (statements, control flow, functions) - Notation (pseudo code, flow chart) – Problem solving aspect – Top down design – Implementation of algorithms – Program Verification – Efficiency of algorithms – Analysis of algorithm.

CO2: Apply fundamental, sequential, conditional logic statements and arrays for solving basic problems **L:12; P:10**

Data Types - Constants – Variables - Keywords – Operators– Problem Solving using fundamental algorithms. Control Statements: Branching and Looping - Algorithms Using Selection and Repetition - Summation of a set of numbers, Reversing Digits of an Integer - Implementation of fundamental algorithms and factoring methods - Array Techniques - Array order reversal, Array Counting, Finding maximum and the minimum value in a set

CO6: Develop programs for simple algorithms using sequential and Control structures

Solve problems using control statements (Decision making and Looping)

CO7: Inscribe programs using arrays, functions and pointers to work with multiple data items.

Problem solving based on Array Handling(1D and 2D, Multi-dimensional arrays, traversal, rotation) - Solve problems to handle strings

CO3: Implement modular programming concept using user defined functions **L:10; P:8**

Modular Programming approach: Modularization and recursion - Bubble Sort, Selection Sort, Linear Search, Binary Search, Implementation of sorting and searching

CO7: Inscribe programs using arrays, functions and pointers to work with multiple data items.

Solve problems by using modular approach (Functions and Recursion)

CO4: Inscribe programs using pointers and to allocate memory for user defined data types using dynamic memory management functions **L:12; P:10**

Pointer Concept – add numbers using call by reference – finding maximum number from list of numbers - permutations of a given string using pointers – Implementation of function returns a pointer;

Structures &Union - finding the largest element of an array using Dynamic Memory Allocation – Implementation of Student database in structure using Dynamic Memory Allocation;

CO7: Inscribe programs using arrays, functions and pointers to work with multiple data items.

Build efficient solutions to manage memory efficiently through Pointers.

CO8: Develop application programs using structures and files concept.

Develop applications using Structures

CO5: Develop file processing application programs **L:5; P:2**

File Handling: Files - Introduction, Types of file processing: Sequential access, Random access – Implementation of word count, copy file, Voter's age validation, Marks range validation

CO8: Develop application programs using structures and files concept.

Develop appliactions using Files

TEXT BOOKS:

1. Maureen Sprankle and Jim Hubbard, Problem Solving and Programming Concepts, Prentice Hall, 9th Edition, 2012.
2. R.G Dromey, How to solve it by Compute, Pearson education, Delhi, 2nd Edition, 2021.

REFERENCES:

1. Behrouz A. Forouzan, Richard F.Gilberg, P.Golda Jeyasheeli, G.Priyanka, S.T.Veena, Problem solving Using C A Structured Programming Approach, Volume I & II, 1st Edition, Cengage Publication, 2022
2. Karl Beecher, Computational Thinking: A Beginner's Guide to Problem Solving and Programming, BCS Learning & Development Limited, 1st Edition, 2017.
3. Byron S. Gottfried, Jitendar Kumar Chhabra, Programming with C, Tata McGraw Hill Publishing Company, New Delhi, 4th Edition, 2018.
4. Kernighan B.W., Ritchie D.M., C Programming Language (ANSI C), Prentice Hall of India Private Limited., New Delhi, 2nd Edition, 2010.
5. Pradip Dey and Manas Ghosh, Programming in C, Oxford University Press, New Delhi, 2018.
6. Yashavant P. Kanetkar, Let Us C, BPB Publications,16th Edition, 2020

7. H. M.Deitel, P. J. Deitel, C How to Program, Pearson Education., New Delhi, 7th Edition, 2016.

L: 45; P: 30; TOTAL: 75 PERIODS

Course Code	BASIC ELECTRICAL AND ELECTRONICS	L	T	P	E	C
23EE11C	ENGINEERING	3	0	2	0	4

COURSE OUTCOMES:

Upon the successful completion of the course, the student will be able to

Theory Component

CO1: Demonstrate and explain the characteristic parameters of DC and AC circuits.
CO2: Explain the working of AC and DC machines.
CO3: Describe analog and digital instruments for monitoring and control.
CO4: Demonstrate the operation of electronic devices for applications.
CO5: Describe the purpose of safety standards and equipment.

Practical Component

CO6: Analyze the basic electric circuits and characteristics of electrical machines.
CO7: Demonstrate the functionality of instruments and characteristics of electronics devices.
CO8: Perform residential wiring and measure earth resistance.

CO1: Demonstrate and explain the characteristic parameters of DC and AC circuits **L:9; P:6**

Sources - Passive Elements – Electrical Quantities: Voltage, Current, Power and Energy. DC circuits: Ohms Law – Kirchhoff's Laws – Mesh analysis. AC Circuits: Waveforms, RMS, Peak, real power, reactive power and apparent power, power factor.

CO6: Analyze the basic electric circuits and characteristics of electrical machines

1. Verification of Ohms law & Kirchhoff law.
2. Measurement of power and power factor for R, L load

CO2: Explain the working of AC and DC machines **L:9; P:6**

Machines: Construction, Types of DC motors – Working Principles – Need for Starters - AC Motors: Construction and Working of Single Phase and Three Phase Induction Motor– Servomotor -Stepper motor.

CO6: Analyze the basic electric circuits and characteristics of electrical machines

1. Analyze the characteristics of DC Shunt Motor and DC series motor
2. Distinguish the operation of single phase and three phase induction motor

CO3: Describe analog and digital instruments for monitoring and control **L:9; P:6**

Analog instruments: Functional Elements, Principles: PMMC, MI, Electrodynamometer wattmeter – Digital voltmeter - multimeter – DSO – Digital Energy meter - Multifunction meter.

CO7: Demonstrate the functionality of instruments and characteristics of electronics devices.

1. Calibration of single phase energy meter using wattmeter
2. Measurement of AC signal parameter (Peak-Peak, RMS, Period and Frequency) using DSO

CO4: Demonstrate the operation of electronic devices for applications **L:9; P:6**

Characteristics and applications: Diode – Rectifiers, Zener Diode – Regulators, BJT - Configuration, Amplifier – LEDs – Photo Diodes, Opto-Isolators.

CO7: Demonstrate the functionality of instruments and characteristics of electronics devices.

1. Experimental Verification of PN Junction diode as rectifier.
2. Experimental Verification of Zener Diode as Voltage Regulators.
3. Input and Output Characteristics of BJT in CE Configuration.

CO5: Describe the purpose of safety standards and equipment **L:9; P:6**

Electric shock -Protection: PPE, Switches, Plug and Socket, Fuse, MCB, ELCB, MCCB and Earthing - Types of wires and cables - Energy storage devices - Inverters – UPS - Energy Consumptions and Battery Charging system – Electrical safety standards in IT industry – Schematic Electrical Layout of Computer Lab with battery backup.

CO8: Perform residential wiring and measure earth resistance

1. Measurement of Earth Resistance using Electrical Equipment.
2. Harness residential house wiring, staircase wiring and fuse connections

TEXT BOOKS:

1. D.P. Kothari and I J Nagrath, "Basic Electrical and Electronics Engineering", Tata Mc-Graw Hill, 4th Edition, 2019.
2. R.K.Rajput, "Basic Electrical and Electronics Engineering", University Science Press, 2017.

REFERENCES:

1. Lionel Warnes, "Electrical and electronics engineering: Principles and practice, Palgrave Macmillan publication, 3rd Edition, 2003.
2. D.C. Kulshreshtha, "Basic Electrical Engineering", Tata McGraw Hill, Revision 1stEdition, 2011.
3. David Bell, "Electronic Devices and Circuits", Oxford University Press, 5th Edition, 2008.
4. Mohamed A. El-Sharkawi, "Electric Safety Practice and Standards", Taylor & Francis, 2013.

L: 45; P: 30; TOTAL: 75 PERIODS

Course Code	தமிழரும் தொழில்நுட்பமும் (TAMILS AND TECHNOLOGY)	L	T	P	E	C
23SH21C	(Common to all B.E. / B.Tech. Degree Programmes)	1	0	0	0	1

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

CO1: தமிழர்களின் நெசவு மற்றும் பானைத் தொழில் நுட்பம், வடிவமைப்பு மற்றும் கட்டிடத் தொழில்நுட்பம், உற்பத்தித் தொழில்நுட்பம் பற்றியஅறிவுமற்றும் விளக்கும் திறன்.

CO2: தமிழர்களின் வேளாண்மைமற்றும் நீர்ப்பாசனத் தொழில்நுட்பம்,அறிவியல் தமிழ் மற்றும் கணினித் தொழில்நுட்பம் பற்றிய அறிவு மற்றும் விளக்கும் திறன்.

Upon the successful completion of the course, the student will be able to

CO1: Know and explain about Tamils weaving and Pottery technology, Designand construction

Technology and Manufacturing Technology.

CO2:Know and explain about Tamils Agriculture and irrigation technology, Scientific Tamil and Tamil computing

CO1: தமிழர்களின் நெசவு மற்றும் பானைத் தொழில்நுட்பம், வடிவமைப்பு மற்றும் கட்டிடத் தொழில்நுட்பம் மற்றும் உற்பத்தித் தொழில்நுட்பம் பற்றிய அறிவு மற்றும் விளக்கும் திறன் **L:9**

சங்க காலத்தில் நெசவுத் தொழில் - பானைத் தொழில்நுட்பம் - கருப்பு சிகப்பு பாண்டங்கள் - பாண்டங்களில் கீறல் குறியீடுகள்- சங்க காலத்தில் வடிவமைப்பு மற்றும் கட்டுமானங்கள் & சங்க காலத்தில் வீட்டுப் பொருட்களில் வடிவமைப்பு - சங்க காலத்தில் கட்டுமான பொருட்களும் நடுகல்லும் - சிலப்பதிகாரத்தில் மேடை அமைப்பு பற்றிய விவரங்கள் - மாமல்லபுரச் சிற்பங்களும் கோவில்களும் - மாதிரி கட்டமைப்புகள் பற்றி அறிதல், மீனாட்சி அம்மன் ஆலயம் மற்றும் திருமலை நாயக்கர் மஹால் - செட்டி நாடு வீடுகள் - பிரிட்டிஷ் காலத்தில் சென்னையில் இந்தோ-சாரோசெனிக் கட்டிடக்கலை- கப்பல் கட்டும் கலை - உலோகவியல் - இரும்புத் தொழிற்சாலை - இரும்பை உருக்குதல், எஃகு - வரலாற்றுச் சான்றுகளாக செம்பு மற்றும் தங்க நாணயங்கள் - நாணயங்கள் அச்சடித்தல் - மணி உருவாக்கும் தொழிற்சாலைகள் - கல் மணிகள், கண்ணாடி மணிகள் - சுடுமண் மணிகள் - சங்கு மணிகள் - எலும்புத் துண்டுகள் - தொல்லியல் சான்றுகள் - சிலப்பதிகாரத்தில் மணிகளின் வகைகள்.

CO1: KNOW AND EXPLAIN ABOUT WEAVING AND CERAMIC TECHNOLOGY, DESIGN AND CONSTRUCTION TECHNOLOGY, MANUFACTURING TECHNOLOGY

Weaving Industry during Sangam Age – Ceramic technology – Black and Red Ware Potteries (BRW)– Graffiti on Potteries- Designing and Structural construction House & Designs in household materials duringSangam Age - Building materials and Hero stones of Sangam age — Details of Stage Constructions in Silappathikaram- Sculptures and Temples of Mamallapuram- Great Temples of Cholas and other worship places - Temples of Nayaka Period - Type study (Madurai Meenakshi Temple)- ThirumalaiNayakarMahal - Chetti Nadu Houses, Indo –Saracenic architecture at Madras during British Period- Art of Ship Building - Metallurgical studies - Iron industry - Iron smelting, steel -Copper and gold Coins as source of history - Minting of Coins — Beads making-industries Stone beads -Glass beads - Terracotta beads -Shell beads/ bone beats - Archeological evidences – Gemstone types described in Silappathikaram.

CO2: : தமிழர்களின் வேளாண்மை, நீர்ப்பாசனத் தொழில்நுட்பம், அறிவியல் தமிழ் மற்றும் கணினித் தமிழ் பற்றிய அறிவு மற்றும் விளக்கும் திறன். **L:6**

அணை, ஏரி, குளங்கள், மதகு - சோழர்காலக் குமுழித் தூம்பின் முக்கியத்துவம் - கால்நடை பராமரிப்பு - கால்நடைகளுக்காக வடிவமைக்கப்பட்ட கிணறுகள் - வேளாண்மை மற்றும் வேளாண்மைச் சார்ந்த செயல்பாடுகள் - கடல்சார் அறிவு - மீன்வளம் - முத்து மற்றும் முத்துக்குளித்தல் - பெருங்கடல் குறித்த பண்டைய அறிவு - அறிவுசார் சமூகம்- அறிவியல் தமிழின் வளர்ச்சி - கணினித்தமிழ் வளர்ச்சி - தமிழ் நூல்களை மின்பதிப்பு செய்தல் - தமிழ் மென்பொருட்கள் உருவாக்கம் - தமிழ் இணையக் கல்விக் கழகம் - தமிழ் மின் நூலகம் - இணையத்தில் தமிழ் அகராதிகள் - சொற்குவைத் திட்டம்.

CO2: KNOW AND EXPLAIN ABOUT AGRICULTURE TECHNOLOGY, IRRIGATION TECHNOLOGY, SCIENTIFIC TAMIL & TAMIL COMPUTING

Dam, Tank, ponds, Sluice, Significance of KumizhiThoompu of Chola Period, Animal Husbandry - Wells designed for cattle use - Agriculture and Agro Processing – Knowledge of Sea - Fisheries — Pearl - Conche diving - Ancient Knowledge of Ocean – Knowledge Specific Society- Development of Scientific Tamil - Tamil computing – Digitalization of Tamil Books –Development of Tamil Software – Tamil Virtual Academy – Tamil Digital Library – Online Tamil Dictionaries– Sorkuvai Project.

REFERENCE BOOKS:

1. தமிழக வரலாறு – மக்களும் பண்பாடும் - கே.கே.பிள்ளை (வெளியீடு: தமிழ்நாடு பாடநூல் மற்றும் கல்வியியல் பணிகள் கழகம்)
2. கணினித் தமிழ் - முனைவர் இல. சுந்தரம் (விகடன் பிரசுரம்)
3. கீழடி – வைகைநதிக்கரையில் சங்ககாலநகரநாகரிகம் (தொல்லியல் துறைவெளியீடு)
4. பொருநை – ஆற்றங்கரைநாகரிகம் (தொல்லியல் துறைவெளியீடு)
5. Social Life of Tamils(Dr.K.K.Pillay)A joint publication of TNTB & ESC and RMRL
6. Social Life of the Tamils - The Classical Period (Dr.S.Singaravelu) (Published by: International Institute of Tamil Studies.
7. Historical Heritage of the Tamils (Dr.S.V.Subatamanian, Dr.K.D. Thirunavukkarasu) (Published by: International Institute of Tamil Studies).
8. The Contributions of the Tamils to Indian Culture (Dr.M.Valarmathi) (Published by: International Institute of Tamil Studies.)
9. Keeladi-Sangam City Civilization on the banks of river Vaigai (Jointly Published by:Department of Archaeology &Tamil Nadu Text Book and Educational Services Corporation, Tamil Nadu)
10. Studies in the History of India with Special Reference to TamilNadu (Dr.K.K.Pillay) (Published by: The Author)

L: 15; TOTAL: 15 PERIODS

Course Code	APTITUDE ESSENTIALS	L	T	P	E	C
23GN01C	(Common to all B.E. / B.Tech. Degree Programmes)	**1**	**0**	**0**	**0**	**1**

COURSE OUTCOMES:

Upon the completion of the course the students will be able to

CO1: recall the fundamentals in quantitative techniques and solve Number series problems quickly
CO2: develop problem solving skills on Numbers and enhance arithmetic ability
CO3: infer appropriate comparison and distribution methods using ratio and to form equations
CO4: improve quantitative skills and solve problems on percentages and profit loss
CO5: calculate data interpretation and data sufficiency in quantitative aptitude

CO1: Recall the fundamentals in quantitative techniques and solve Number series problems quickly **L : 3**

Numeric series – Finding missing numbers – Odd number out series - Letter series – Symbol series - Alphanumeric series

CO2: Develop problem solving skills on Numbers and enhance arithmetic ability **L : 3**

Number Types - HCF & LCM – Square root- Cubic root - divisibility criteria- Unit digit calculation- Prime factors

CO3: Infer appropriate comparison and distribution methods using ratio and to form equations **L : 3**

Ratio & Proportion: Comparison of Ratios - Variations: Direct and indirect proportion
Ages: Present Age, Past Age & Future calculation

CO4: Improve quantitative skills and solve problems on percentage and profit loss **L : 3**

Concept of Percentage – Percentage calculation - Calculation of Percentage on Population Results on Depreciation. Profit and Loss –Percentage of Profit and Loss – Discount

CO5: Calculate data interpretation and data sufficiency in quantitative aptitude **L : 3**

Data Interpretation – Pie Chart – Bar Chart – Table Chart .Data Sufficiency in Logical Reasoning : Numbers, Ratio, Ages, Percentage and Profit Loss

REFERENCES:

1. Dr.R.Aggarwal, " Quantitative Aptitude", S Chand Publishing, Revised Edition 2017
2. R.V.Praveen, "Quantitative Aptitude and Reasoning" , 3rd Edition , Eastern Economy Edition, PHI Learning 2016

Video Materials

Profit Loss

https://youtu.be/PpVO7I8dx6U

https://youtu.be/_cW7_BUDYcw

Number series

https://youtu.be/_4ZJFkFE2XU

https://youtu.be/83nJmniFmNk

Numbers

https://youtu.be/81pwuMJ8OIU

https://youtu.be/VT_N9cacgl4

Square root and Cube root

https://youtu.be/nJSqsaT0AgU

https://youtu.be/_HyhwS8P9KY

Problems on Ages

https://youtu.be/6PCTRVmu-ek

https://youtu.be/eAl3BvO_Ipw

Data Interpretation

https://youtu.be/s99rda8e0vc

Course Code	DIGITAL PRINCIPLES AND SYSTEM DESIGN	L	T	P	E	C
23CS21C		3	0	0	0	3

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Component

CO1: apply the principles of number systems and logic gates to design digital circuits (CDL1)
CO2: design of combinational and sequential logic circuits for basic processing units (CDL2)
CO3: analyze the design of Asynchronous Sequential circuits (CDL2)
CO4: apply the concept of Programmable Logic Devices for the design of digital circuits (CDL1)
CO5: implement the Digital Logic circuits using VHDL and functions (CDL2)

CO1: Apply the principles of number systems and logic gates to design digital circuits **L:9**

Number Systems and Codes - Binary Number system - Binary to decimal - decimal to binary – hexadecimal - ASCII code - Excess-3 Code - Gray code - Digital Logic - Basic Gates-Universal Logic Gates - Combinatorial Logic Circuits - Boolean Laws and Theorems-Sum of Products method - Truth table to Karnaugh Map - Don't Care Conditions

CO2: Design of combinational and sequential logic circuits for basic processing units **L:9**

Arithmetic Circuits - Binary Addition and Subtraction - 2's Complement Representation–

Data Processing Circuits - Multiplexers - Demultiplexers - Decoders - Encoders - Sequential Circuits - Flip-Flops -operation and excitation tables - Triggering of FF - Analysis and design of clocked sequential circuits - Moore/Mealy models - state minimization and state assignment- Design and analysis of sequential circuits

CO3: Analyze the design of Asynchronous Sequential circuits **L:9**

Asynchronous sequential logic circuits - Stable and Unstable states - output specifications - cycles and races - state reduction - race free assignments - Hazards - Design and analysis of asynchronous sequential circuits

CO4: Apply the concept of Programmable Logic Devices for the design of digital circuits **L:9**

Classification of memories - ROM organization – PROM – EPROM – EEPROM - RAM organization - Write operation - Read operation - Static RAM - Programmable Logic Devices - Programmable Logic Array (PLA) - Programmable Array Logic

CO5: Implement the Digital Logic circuits using VHDL and functions **L:9**

Introduction to HDLs Library – Entity – Architecture - Modeling styles -Data flow modeling –Behavioral modeling - Structural modeling - Data objects - Concurrent and sequential statements - Design examples using VHDL for basic combinational and sequential circuits

TEXT BOOKS

1. M. Morris Mano, “Digital Logic and Computer Design”, 6th Edition, Pearson Education, 2018.
2. M. Morris Mano, Michael D. Ciletti, “Digital Design: With an Introduction to the Verilog HDL, VHDL, and System Verilog”, 6th Edition, Pearson Education, 2018.

REFERENCES

1. Thomas Floyd, “Digital fundamentals”, 11th Edition, Pearson Education, 2021.
2. Stephen Brown, “Fundamentals of Digital Logic with Verilog”, 2nd Edition, McGraw Hill, 2017.
3. John F.Wakerly, “Digital Design Principles and Practices”, 5th Edition, Pearson Education, 2017.
4. James W. Bignel, Digital Electronics, 5th Edition, Cengage learning, 2013.
5. G. K. Kharate, “Digital Electronics”, 2nd Edition, Oxford University Press, 2010.
6. R.P. Jain, “Modern digital electronics”, 3rd Edition, 12th reprint TMH Publication, 2007.
7. Samir Palnitkar, “Verilog HDL”, 2nd Edition, Pearson Education, 2003

L: 45; TOTAL: 45 PERIODS

Course Code	DISCRETE MATHEMATICS	L	T	P	E	C
23CS22C		3	1	0	0	4

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Component

CO1: illustrate the validity of the arguments.

CO2: analyze the concepts of Sets, Relations and Functions.

CO3: perform the principles of counting and solve recurrence relations.
CO4: interpret the basic concepts of graphs.
CO5: compute minimum Spanning trees and shortest routefor the graph.

CO1: Illustrate the validity of the arguments.

Propositional Logic – Equivalences and Implications – Normal forms –Predicate Calculus and Quantifiers - Rules of inference – Proof methods and Strategies - Forward Proof, Proof by Contradiction, Proof by Contraposition, Proof of Necessity and Sufficiency. **L:9; T:3**

CO2: Analyze the concepts of Sets, Relations and Functions

Basic Definitions - Set operations – Laws of set theory – Relations – Properties of relations - Partial Ordering Relation - Equivalence Relation - Matrices of relations - Closure of relations – Functions – Bijective functions - Inverse and Compositions of functions. **L:9; T:3**

CO3: Perform the principles of counting and solve recurrence relations.

Mathematical induction - Strong induction and well ordering -The basics of counting –The pigeonhole principle - Permutations and combinations – Recurrence relations – Solving linear recurrence relations – Generating functions - Inclusion and exclusion principle. **L:9; T:3**

CO4: Interpret the basic concepts of graphs

Graphs and their properties - Special types of graphs – Matrix representation of graphs and graph isomorphism- connectivity, Cut vertex and cut edge - Euler and Hamiltonian graphs. **L:9; T:3**

CO5: Compute minimum Spanning Trees and shortest route for the graph

Trees – Some properties of Trees – Pendant vertices in a Tree – Distance and centers in a Tree – Rooted and Binary Trees - Spanning Trees- minimum spanning tree– Prim's algorithm- shortest route - Dijkstra's algorithm **L:9; T:3**

TEXT BOOKS:

1. Kenneth H.Rosen, Discrete Mathematics and its Applications (with Combinatory and Graph Theory), Special Indian Edition, Tata McGraw-Hill Publishing Company Limited, New Delhi, 7th Edition, 2017.
2. Trembly J.P and Manohar.R. Discrete Mathematical Structures with Applications to Computer Science, 1st Edition, Tata McGraw-Hill Pub. Company Limited, New Delhi, 2017.
3. Narsingh Deo, Graph Theory with Applications to Engineering and Computer Science, 1st Edition, Dover Publications Inc., 2016.

REFERENCES:

1. Ralph.P. Grimaldi, Discrete and Combinatorial Mathematics: An Applied Introduction, 5th Edition, Pearson Education Asia, Delhi, 2019.
2. Bondy,J.A., Murty. U.S.R., Graph Theory with applications, North Holland publication, 2008.
3. V.K.Balakrishnan, Schaum's Outline of Graph Theory, Tata Mc Graw-Hill Pub, 2020.

L: 45; T: 15; TOTAL: 60 PERIODS

Course Code	PROFESSIONAL ENGLISH	L	T	P	E	C
23SH22C	(Common to all B.E. / B.Tech. Degree Programmes)	**2**	**0**	**2**	**0**	**2**

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Component

CO1: extend the primary language skills to develop critical thinking

CO2: build the secondary language skillsfor professional competence

Practical Component

CO3: apply the vital sub-functions of listening in particular context

CO4: take part in propagating ideas through effective oral communication

CO5: inferring information using various reading techniques

CO6: construct professional content via distinct methods of writing

Soft skill Component

CO7: develop interpersonal, communicational and behavioral attributes

CO1: Extend the primary language skills to develop critical thinking
CO3: Apply the vital sub-functions of listening in particular context
CO4: Take part in propagating ideas through effective oral communication

If Conditionals – Standard Abbreviations –Types of Listening (Comprehensive, Informational, Critical Listening) –One Word Substitution, Components of Speaking Listening for Specific Information –Listening to Speech (Oxford Union Society) – Listening to Science Talks or Theories **L:6; P:16**

Product Description – Chart Description – Process Description – Group Discussion (Uses – Structure – Strategies – Team Work – Positive & Negative Body Languages – Samples – Demo)

CO2:build the secondary language skills for professional competence
CO5: inferring information using various reading techniques **L:5; P:18**
CO6: construct professional content via distinct methods of writing

Synonyms – Intensive and Extensive Reading –Error Spotting (Based on Concord, Pronoun, Articles & Adverb Placement)– Writing Style (Persuasive, Expository & Descriptive)

Newspaper Reading – Reading Comprehension (Fiction & NonFiction)

Business Letters for Quotations and Clarification, Placing Orders and Making Complaints – Proposal Writing – Job Application Letter & Resume Preparation –Paragraph Writing – Content Writing

TEXT BOOKS:

1. Lucantoni, Peter & Lydia Kellas. "English as a Second Language Workbook", 6th Edition, Cambridge University Press, 2022.
2. Twain, Mark. "The Adventures of Tom Sawyer", 1st Edition, Pegasus, 2012.
3. Clear, James. "Atomic Habits", 1st Edition, Dreamliners, 2022.
4. Garcia, Hector & Francesc Miralles. Ikigai: The Japanese Secret to a long and Happy Life. 1st Edition, Tuttle Publishing, 2021.

5. Elbow, Peter, "Writing with Power" 2nd Edition, Oxford University Press, 1998.

REFERENCES:

1. Butterfield, Jeff. "Soft Skills for Everyone". 2nd Edition, Cengage, 2020
2. Raman, Meenashi & Sangeetha Sharma. Professional English. 1st Edition, Oxford University Press, 2018

L: 11; P: 34; TOTAL: 45 PERIODS

Course Code	SEMICONDUCTOR AND QUANTUM PHYSICS	L	T	P	E	C
23CS23C		2	0	2	0	3

COURSE OUTCOMES:

Upon the successful completion of the course, the student will be able to

Theory Component

CO1: explain the conductivity in metals using free electron theories
CO2: describe the fundamental properties of semiconductors
CO3: illustrate the optical properties and their applications to optical devices
CO4: apply the basics of sensing and imaging techniques in engineering and medical applications
CO5: apply the concepts of quantum mechanics in quantum computing

Practical Component:

CO6:determine the bandgap and hall coefficient of semiconductors
CO7: demonstrate the I-V characteristics of PN junction diodes
CO8: analyze the characteristics of light sensor
CO9:demonstrate the basic quantum computing using simulation

Soft skill Component:

CO10: develop a sense of teamwork and enhance communication abilities through collaborative group activities

CO1: Explain the conductivity in metals using free electron theories

Conduction in metals - Classical free electron theory of metals – Mobility and electrical conductivity - Thermal conductivity of metals – Wiedemann-Franz law – Quantum free electron theory - Merits and limitations of free electron theory - Fermi-Dirac Statistics - Density of States. **L:6**

CO2: Describe the fundamental properties of semiconductors
CO6: Determine the bandgap and hall coefficient of semiconductors

Energy band diagram - Direct and indirect band gap - Intrinsic semiconductors - Carrier concentration and Fermi level in an intrinsic semiconductor- Extrinsic semiconductors - Carrier concentration and Fermi level in N-type and P-type semiconductors - Hall effect – Determination of band gap and hall coefficient **L:6; P:8**

CO3: Illustrate the optical properties and their applications to optical devices
CO7: Demonstrate the I-V characteristics of PN junction diodes **L:6;**

Classification of optical materials – Absorption emission and scattering of light in metals, insulators and semiconductors (quantitative) – Carrier generation and **P:12**

recombination -photocurrent in a P-N diode – Principle and working of solar cell - LED – Organic LED – Laser diodes - Photo diode – Determination of V-I Characteristics -Photoconductors - Optical data storage techniques.

CO4: Apply the basics of sensing and imaging techniques in engineering and medical applications

CO8: Analyze the characteristics of light sensor

L:6; P:4

Sensors - principle – working – bio sensors -LDR– Determination of V-I-Characteristics-wearable sensors in health care- Medical imaging techniques – ultrasound- Non destructive testing (NDT) - X ray –CT scan – MRI scan-FMRI Scan

CO5: Apply the concepts of quantum mechanics in quantum computing

CO9: Demonstrate the basic quantum computing using simulation

L:6; P:6

Introduction - Quantum confinement - Band gap of nanomaterials – Quantum tunneling – Quantum cellular automata - Quantum system for information processing - Quantum states – Classical bits – Quantum bits –CNOT gate - Multiple qubits – quantum gates – Fundamental quantum computing experiment using simulation software-Advantage of quantum computing

TEXTBOOKS:

1. S.O Pillai, Solid State Physics, 10th Edition, NEW AGE International Publishers, 2022
2. Progress in Nanoscale and Low-Dimensional Materials and Devices, Hilmi Unlu and Norman J M.Horing, Springer Link, 2022.
3. Parag K. Lala, Quantum Computing: A Beginner's Introduction, McGraw-Hill Education (Indian Edition), 2020.

REFERENCES:

1. Principles of Electronic Materials and Devices, S.O.Kasap, McGraw Hill Education, 2017.
2. Physics for Computer Science Students, N.Garcia, A.Damask and S.Schwarz, Springer-Verlag, 2012
3. Fundamentals of Nanoelectronics, G.W. Hanson, Pearson Education, 2009.
4. Optoelectronics. Pearson Education, J. Wilson and J.F.B. Hawkes, 2018.

L: 30; P: 30; TOTAL: 60 PERIODS

Course Code	ENGINEERING GRAPHICS	L	T	P	E	C
23ME11C	(Common to MECH, CIVIL, CSE, AI&DS, EEE, IT degree Programmes)	2	0	4	0	4

COURSE OUTCOMES:

Upon the successful completion of the course, the student will be able to

CO1: construct the Engineering Curves and Perform Freehand Sketching.

CO2: construct the Orthographic Projections of Points, Straight Lines and Lamina

CO3: draw the Projections of Simple Solids in Different Positions.

CO4: visualize the Sectional Views and Surface of Various Solids.

CO5: draw the Isometric and Perspective Projections of Various Solids.

CO1: Construct the Engineering Curves and Perform Freehand Sketching. **L:6; P:12**

Principles of Engineering Graphics – significance. Usage of Drawing Instruments. Lettering and dimensioning exercise Construction of ellipse, parabola and hyperbola using eccentricity method– Construction of cycloids, Epi and Hypo-cycloids. Orthographic views of simple components by Free hand drawing - Transferring measurement from the given object to the free hand sketches.

CO2: Construct the Orthographic Projections of Points, Straight Lines and Lamina **L:6; P:12**

Principle of orthographic projections – Conventions - First angle and third angle projections. Projections of points placed in all quadrants – projections of straight lines – inclined to both reference planes - determination of true length and inclinations. Projections of regular polygonal surfaces and circular lamina inclined to both reference planes.

CO3: Draw the Projections of Simple Solids in Different Positions. **L:6; P:12**

Projections of simple solids like prisms, pyramids, cylinder and cone - axis inclined to one reference plane - change of position method.

CO4: Visualize the Sectional Views and Surface of Various Solids. **L:6; P:12**

Sectioning of simple solids – Axis perpendicular to horizontal plane- Drawing sectional views with true shape of the section. Development of lateral surfaces of truncated solids – Prisms, pyramids, cylinder and cone.

CO5: Draw the Isometric and Perspective Projections of Various Solids. **L:6; P:12**

Principles of isometric projection – Isometric scale – Isometric projections of simple solids like prism, pyramid, cone and cylinder – Combination of solids. Perspective projections of simple solids by visual-ray method

TEXT BOOKS:

1. Bhatt N.D, "Engineering Drawing", 54th Edition, Charotar Publishing House, 2023.
2. Shah M.B and Rana B.C, "Engineering Drawing", Pearson Education, 2nd Edition, 2009.

REFERENCES:

1. Natrajan K.V., "A text book of Engineering Graphics", Dhanalakshmi Publishers, Chennai, 2009.
2. Agrawal B. & Agrawal C.M., "Engineering Graphics", TMH Publication,2nd Edition, 2013
3. Narayana K.L. &Kannaiah P, "Text book on Engineering Drawing", Scitech Publishers, 2011.
4. Gopalakrishna K.R, "Engineering Drawing", Subhas Publications, 32nd Edition, 2017.

L: 30; P: 60; TOTAL: 90 PERIODS

Course Code	OBJECT ORIENTED PROGRAMMING USING C++	L	T	P	E	C
23CS24C	(Common to CSE, IT & ECE degree Programmes)	**2**	**0**	**2**	**2**	**4**

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Component

CO1: apply the object oriented programming constructs to solve known applications (CDL1)

CO2: design effective application with inheritance, compile time and run time polymorphism (CDL1)

CO3: develop real-world applications by using files, streams, and exceptions (CDL2)

CO4: construct well-defined, efficient data handling strategies using templates and STL (CDL2)

Practical Component

CO5: demonstrate the basic OO principles such as class, objects, and constructors (PDL1)

CO6: implement code reusability through overloading, inheritance and polymorphism (PDL2)

CO7: solve problems using files and exception handling (PDL2)

CO8: employ problem solving skill using templates and STL (PDL2)

Experiential Component

CO9: create efficient solutions for solving real-world OOP applications (PDL3)

Soft Skill Component

CO10: demonstrate diversity and inclusive attitude while practicing project component as a team

CO1: Apply the object oriented programming constructs to solve known applications **L:8;**
CO5: Demonstrate the basic OO principles such as class, objects, and constructors **P:8;**
Introduction- Comparison between procedural programming paradigm and object-oriented programming paradigm. Features of object-oriented programming. Functions - Inline functions- Friend functions. Arrays -Array of objects. Pointer - Function pointer. Memory management: New and Delete. Classes and Objects - Access specifiers, Types of classes- Constructor and destructor - Types of constructor - Static members **E:8**

CO2: Design effective application with inheritance, compile time and run time polymorphism **L:8;**
P:8;
CO6:Implement code reusability through overloading, inheritance and polymorphism **E:10**
Function Overloading, Overloading Constructors, Ambiguity in Overloading. Operator overloading - Overloading Using Friend Function- Overloading New and Delete- Overloading Special Operators. Inheritance – Types of Inheritance - Typing Conversions and Visibility – Code Reuse- Aggregation. Polymorphism- Virtual Functions – Pure Virtual Functions – Early vs. Late Binding. Run-Time Type ID and Casting Operators: RTTI – Casting Operators – Dynamic Cast.

CO3: Develop real-world applications by using files, streams, and exceptions **L:7;**
CO7: Solve problems using files and exception handling **P:6;**
Streams and Files: Streams classes - Sequential Input and Output operations – Random Access - File pointers - Error handling in file I/O with member function - command line arguments. Exception handling – expected and unexpected exceptions - uncaught exception - resource captures and release. Case study with real time applications. **E:4**

CO4: Construct well-defined, efficient data handling strategies using templates and **L:7;**

STL **P:8;**

CO8: Employ problem solving skill using templates and STL **E:8**

Templates- Generic programming - variadic templates – template compilation model – Generic Classes. Standard Template Library: Iterators – Auxiliary Iterator function – Algorithms – Non-modifying sequence operations – mutating sequence operations – Containers: Sequence and associative containers - Algorithms, string class – explicit, mutable and operator keywords. Namespaces: user defined namespaces, namespaces provided by library

TEXT BOOKS:

1. Bjarne Stroustrup, "A Tour of C++", 3rdEdition, Pearson Education, April 2023.
2. Herbert Schildt, "C++: The Complete Reference", 4thEdition, Tata Mc-Graw Hill Publishers, 2017.

REFERENCE BOOKS:

1. Reema Thareja, "Object oriented programming with C++", Revised 1stEdition, Oxford University Press, 2018.
2. E.Balagurusamy, "Object oriented programming with C++", 8thEdition, McGraw Hill Education (India) Private Limited, September 2020.
3. Ivor Horton, Peter van, "Beginning C++ 20 from novice to professional", 6thEdition, APRESS media, 2020.
4. Bjorin Andrist, Viktor Sehr, "C++ High Performance: Master the art of optimizing the functioning of your C++ code", 2ndEdition, Packt Publishing Limited, December 2020.
5. Nicoloai.MJosuttis and Doug Gregor, "C++Templates: The complete guide", 1stEdition, Addison Wesley, 2020

ONLINE COURSES:

1. https://onlinecourses.nptel.ac.in/noc23_cs78/preview
2. https://www.udemy.com/course/oops-and-c-from-basic-to-advanced
3. https://www.udemy.com/course/crash-course-on-cpp-stl/
4. https://www.coursera.org/lecture/c-plus-plus-b/1-3-standard-template-library-o3v9K
5. https://www.coursera.org/learn/object-oriented-cpp

L:30; P:30; E:30; TOTAL: 90 PERIODS

Course Code	**INNOVATION THROUGH DESIGN THINKING**	**L**	**T**	**P**	**E**	**C**
23GN02C	(Common to all B.E. / B.Tech. Degree Programmes)	**0**	**0**	**0**	**4**	**2**

COURSE OUTCOMES

Upon completion of this course, the students will be able to

Experiential Component

CO1: analyse the impact of design thinking process.

CO2: practice design thinking process through real world problems.

Soft skill Component

CO3: present survey conclusions on selected real-world problems.

CO1: Analyse the impact of design thinking process **E:30**

Design thinking process: history and phases -Ideation tools: brainstorming, mind mapping, scrambler method, six thinking hats -case studies.

CO2: Practice design thinking process through real world problems **E:30**

Real world problem selection-Practicing the preliminary stages of design thinking process - work presentation.

TEXT BOOKS:

1. Falk Uebernickel, Li Jiang, Walter Brenner, Britta Pukall, Therese Naef, "Design Thinking: The Handbook", WS Professional, 2020
2. PavanSoni, "Design Your Thinking: The Mindsets, Toolsets and Skill Sets for Creative Problem solving", Penguin Random House, 2020

REFERENCES:

1. Michael Lewrick, "The Design Thinking Playbook", Wiley, 2019
2. Kathryn Christopher, "Design Thinking in Engineering", Kendall Hunt Publishing Company,2019
3. Robert Curedale, "Design Thinking Process & Methods" Design Community College Inc, 5th Edition, 2019
4. David Lee, "Design Thinking in the Classroom", Ulysses Press, 2018
5. Jimmy Jain, "Design Thinking for Startups", Notion Press, 2018
6. Monika Hestad Silvia Rigoni Anders Grnli, "The Little Booklet on Design Thinking: An Introduction", Zaccheus Entertainment, 2nd Edition, 2017
7. Scott Swan, Michael G.Luchs and Abbie Griffin, "Design Thinking: New Product Development Essentials", Wiley-Blackwell, 2016
8. D.M. Arvind Mallik, "Design Thinking for Educators", Notion Press, 2019

E:60; TOTAL:60 PERIODS

Course Code	COMPUTER ORGANIZATION AND ARCHITECTURE	L	T	P	E	C
23CS31C	(Common to CSE, IT and AI&DS degree Programmes)	**3**	**0**	**0**	**0**	**3**

COURSE OUTCOMES

Upon completion of this course, the student will be able to

Theory Component

CO1: apply the fundamental design techniques of a digital computer to execute simple arithmetic operations (CDL1)

CO2: relate the execution sequence of an instruction to design the datapath and control unit for a processor (CDL1)

CO3: analyze the hierarchical structure of various memory systems and assess the performance (CDL1)

CO4: relate various types of I/O interfaces and their functionalities (CDL1)

CO5: analyze the impact of hazards in pipeline performance by utilizing the different stages of instruction execution in a pipelined processor (CDL1)

CO1: Apply the fundamental design techniques of a digital computer to execute simple arithmetic operations **L:10**

Functional units – Basic operational concepts – Bus Structures – Performance – Memory locations and addresses – Instructions and instruction sequencing – Addressing modes.Arithmetic: Addition and Subtraction of Signed Numbers – Design of fast adders – Multiplication of unsigned and signed numbers – Fast Multiplication – Integer division.

CO2: Relate the execution sequence of an instruction to design the datapath and control unit for a processor **L:8**

Fundamental concepts – Execution of a complete instruction – Multiple bus organization – Instruction Fetch and Execution Steps – Control Signals – Hardwired control – Micro programmed control.

CO3: Analyze the hierarchical structure of various memory systems and assess the performance **L:8**

Basic concepts – RAM – ROM – Cache memories – Improving cache performance – Virtual memory – Memory management requirements – Secondary storage devices.

CO4: Relate various types of I/O interfaces and their functionalities **L:8**

Accessing I/O devices – Interrupts – Direct Memory Access – Buses – Interface circuits – Standard I/O Interfaces – PCI, SCSI, SATA, USB – Advanced IO interfaces.

CO5: Analyze the impact of hazards in pipeline performance by utilizing the different stages of instruction execution in a pipelined processor **L:11**

Basic concepts – Parallel processing – Instruction pipeline – Data hazards – Instruction hazards – Influence on instruction sets – datapath and control consideration – Super scalar operation – RISC vs CISC processors – Instruction Level Parallelism – Parallel IO Organization – IO in Multicore Computers.

TEXT BOOKS:

1. Carl Hamacher, Zvonko Vranesic, Safwat Zaky, and Naraig Manjikian, "Computer Organization and Embedded Systems", McGraw Hill Higher Education, 6th Edition, 2022.
2. William Stallings, "Computer Organization and Architecture - Designing for Performance", Pearson Education, 6th Edition, 2021.

REFERENCES:

1. David A. Patterson, John L.Hennessy, "Computer Organization and Design, The Hardware/Software Interface", 6th Edition, Morgan Kaufmann/Elsevier, 2020.
2. M. Morris Mano, "Computer System Architecture", 3rd Edition, Pearson Education, 2017.
3. John P. Hayes, "Computer Architecture and Organization", 3rd Edition, Tata Mc-GrawHill, 2017.
4. Yan Solihin, "Fundamentals of Parallel Multicore Architecture", CRC Press, 2015.
5. V.P. Heuring, H.F. Jordan, T.G.Venkatesh, "Computer Systems Design and Architecture", 2nd Edition, Pearson Education, 2008.

ONLINE SOURCES:

1. https://onlinecourses.nptel.ac.in/noc20_cs64/preview
2. https://www.udemy.com/course/computer-organization-and-architecture-j/

L: 45; TOTAL: 45 PERIODS

Course Code	DATA STRUCTURES	L	T	P	E	C
23CS32C	(Common to CSE & IT Degree Programmes)	3	0	0	0	3

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Component

CO1: apply appropriate sorting and searching algorithms using list for solving real world problems.(CDL1)
CO2: apply appropriate linear data structures for different applications. (CDL2)
CO3: implement various tree operations that handle non linear data organization. (CDL2)
CO4: employ suitable indexing and hashing techniques for optimal data accessing. (CDL2)
CO5: implement various graph and its traversal techniques for solving network problems. (CDL2)

CO1: Apply appropriate sorting and searching algorithms using list for solving real world problems. **L:10**

List ADT – Array based Implementation - Linked List Implementation – Singly linked list - Doubly linked list - Circular linked list – Sorting –Insertion Sort – Merge Sort – Quick Sort – Searching – Linear Search – Binary Search – Fibonacci Search.

CO2: Apply appropriate linear data structures for different applications. **L:9**

Stack ADT – Operations (using Array and Linked List) – Applications of Stack – Balancing symbols – Infix to postfix conversion – Evaluating postfix expression.

Queue ADT – Operations (using Array and Linked List) – Circular Queue – De-Queue – Application of Queues.

CO3: Implement various tree operations that handle nonlinear data organization. **L:10**

Tree ADT – Basic Tree Terminologies – Types of Trees – Binary tree ADT – Operations – Tree Traversals – Expression Trees - Binary search Tree ADT – Threaded Binary Tree - AVL Trees - Priority Queue(Heaps)

CO4: Employ suitable indexing and hashing techniques for optimal data accessing. **L:7**

Indexing: B Tree and B+ Tree: Definitions – algorithms - Hashing: Hash Function – Separate chaining – Open Addressing – Rehashing – Extendible hashing - Dictionary: Dictionary ADT - Implementation

CO5: Implement various graph and its traversal techniques for solving network problems. **L:9**

Graphs: Basic Terminologies and Representation – Types of graphs – Breadth first search – Depth first search – Bi-connectivity – Topological sort – Dijkstra's algorithm- Minimum Spanning Tree – Prim's algorithm – Kruskal's algorithm –Applications of Graphs

TEXT BOOKS

1. Mark Allen Weiss, "Data Structures and Algorithm Analysis in C++", 4th edition, Pearson Education, 2017.
2. R.G. Dromey, "How to Solve it by Computer", 1st edition, Prentice Hall International, 2011.

REFERENCES

1. Langsam, Augenstein and Tanenbaum, "Data Structures Using C and C++", 2nd Edition, Pearson Education, 2015.
2. Alfred V. Aho, Jeffrey D. Ullman,John E. Hopcroft, "Data Structures and Algorithms", 5th Edition, Pearson, 2008.
3. Ellis Horowitz, Sartaj Sahni and Susan Anderson-Freed, "Fundamentals of Data Structures in C", 2nd Edition, Universities Press, 2022.

4. Kamthane, "Introduction to Data Structures in C", 1st Edition, Pearson Education, 2007.
5. Kruse, "Data Structures and Program Design in C", 2nd Edition, Pearson Education, 2006.

ONLINE SOURCES:

1. https://people.ok.ubc.ca/ylucet/DS/Algorithms.html - Data Structure Visualization tool
2. https://nptel.ac.in/courses/106/102/106102064/ - NPTEL Lecture series

L: 45; TOTAL: 45 PERIODS

Course Code	OPERATING SYSTEMS	L	T	P	E	C
23CS33C	(Common to CSE, IT and AI&DS Degree Programmes)	**2**	**1**	**0**	**0**	**3**

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Component

CO1: conceive the basic components and working principles of operating systems.(CDL1)

CO2: analyze CPU scheduling mechanisms for diverse scheduling criteria .(CDL1)

CO3: devise the solutions for process synchronization issues. .(CDL2)

CO4: relate various techniques for handling memory management. (CDL2)

CO5: apply file management and I/O management techniques. (CDL2)

CO1: Conceive the basic components and working principles of operating systems. **L:6; T:1**

Introduction to operating systems - Types of operating system - structure of OS – system calls and its types – system programs – Processes : Concept – Process scheduling – operations on processes – Threads : concept – multithreading models – Inter process communication – Processes and threads in Linux os-Free and Open source-***Overview: Windows, Linux, Mobile, Real-Time, and Robotic Operating Systems.***

CO2: Analyze CPU scheduling mechanisms for diverse scheduling criteria **L:6; T:3**

CPU Scheduling – basic concepts – scheduling criteria – Preemptive and non preemptive scheduling algorithms: FCFS, SJF, Priority, Round Robin – scheduling in Real Time Operating System (RTOS) – Rate Monotonic (RM) scheduling algorithm – Least Laxity First (LLF) scheduling algorithm **– *Simulation of process scheduling.***

CO3: Devise the solutions for process synchronization issues. **L:6; T:3**

Process Synchronization – The Critical – Section problem – Peterson's solution – Mutex locks – concurrency - Semaphores – Classic problems of synchronization – monitors. Deadlock: System model – deadlock characterization – Methods for handling deadlock – deadlock prevention – deadlock avoidance – banker's algorithm – deadlock detection – recovery from deadlock.

CO4: Relate various techniques for handling memory management. **L:6; T:4**

Memory management – Background – Swapping – Contiguous memory allocation – Segmentation – Paging – Segmentation with paging – Intel 32 and 64 bit Architectures-Virtual memory: Background – Demand paging – page replacement – algorithms: FIFO, LRU, Optimal - allocation of frames – thrashing-***Case Study on Redis page replacement approach.***

CO5: Apply file management and I/O management techniques. **L:6; T:4**

File concept – Access methods – directory and disk structure – file system mounting – protection – File system implementation: Directory implementation – Allocation methods – Free space management. Disk scheduling – Algorithms: FCFS, SSTF, SCAN, CSCAN, LOOK, CLOOK – disk management.

TEXT BOOK:

1. Abraham Silberschatz, Peter Galvin, Greg Gagne, "Operating System Concepts", Enhanced eText, 10th Edition , Wiley Asia Student Edition, 2018.

REFERENCES:

1. William Stallings, "Operating Systems: Internals and Design Principles", 9thEdition, Prentice Hall of India, 2018.
2. Andrew S. Tanenbaum, Modern Operating Systems, Pearson, 4thEdition (2016).
3. Anderson, Thomas, and Dahlin, Michael. Operating Systems: Principles and Practice", 2nd Edition, United Kingdom, Recursive Books, 2014.

https://onlinecourses.nptel.ac.in/noc22_cs104/preview

L: 30; T: 15; TOTAL: 45 PERIODS

Course Code	PROFESSIONAL ETHICS AND HUMAN VALUES	L	T	P	E	C
23GN05C	(Common to all B.E. / B.Tech. Degree Programmes)	2	0	0	0	2

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

CO1: recognize and practice the core human values and theories related to ethical behavior. (CDL1)

CO2: analyze the engineering ethical breach from past study. (CDL1)

CO3: distinguish and apply safety, responsibility and rights in workplaces. (CDL2)

CO1: Recognize and practice the core human values and theories related to ethical behavior **L: 10**

Moral dilemmas and moral autonomy - Kohlberg's theory - Gilligan's theory - Consensus and controversy –Case studies: Vigil mechanism, Whistle blowing - Protected disclosures - Personal ethics, work ethics and human values - Governing Regulation.

CO2 : Analyze the engineering ethical breach from past study **L: 10**

Engineering as experimentation - engineers as responsible experimenters - codes of ethics - a balanced outlook on law - Case study: The challenger disaster

CO3 : Distinguish and apply safety, responsibility and rights in workplaces **L: 10**

Safety and risk - assessment of safety and risk - risk benefit analysis and reducing risk - Collegiality and loyalty - respect for authority – confidentiality; Collective bargaining, Conflicts of interest - Case study; Occupational crime - professional rights - employee rights - Intellectual Property Rights (IPR) - discrimination. Case studies: The Three mile island and Chernobyl disaster

TEXT BOOK

1. Mike W Martin and Roland Schinzinger, "Ethics in Engineering", McGraw Hill, New York,5th

Edition, 2022

REFERENCES

1. BehnamTaebi, "Ethics and Engineering: An Introduction", Cambridge University Press, 2021
2. AjeshFaizal, Aswathy S U, Roy V I, "Professional Ethics in Engineering: an Industry Perspective", Noor Publishing, 2021
3. R.S.Naagarazan, "A Textbook on Professional Ethics and Human Values", New age International Pvt. Ltd; 3rdEdition, 2022
4. Dr. P. Elamurugan, "Professional Ethics in Engineering", Notion Press, 2021

L:30; TOTAL:30 PERIODS

Course Code	**PROBABILITY AND STATISTICS**	**L**	**T**	**P**	**E**	**C**
23CS34C		**3**	**1**	**0**	**0**	**4**

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Component

CO1: perform basic probability concepts and standard distributions.
CO2: find the correlation and regression of two-dimensional random variables.
CO3: calculate the various measures of dispersion.
CO4: apply the concept of testing of hypothesis for small and large samples.
CO5: apply the basic concepts of classifications of design of experiments.

CO1: Perform basic probability concepts and standard distributions

Discrete and continuous random variables - Moments - Moment generating functions and their properties. Binomial, Poisson, Geometric, Uniform, Exponential, Gamma, and Normal distributions. **L:9; T:3**

CO2: Find the correlation and regression of two-dimensional random variables

Joint distributions - Marginal and conditional distributions – Covariance - Correlation and regression - Transformation of random variables – Central Limit Theorem. **L:9; T:3**

CO3: Calculate the various measures of dispersion

Central tendencies - Mean, median, mode - Measures of Dispersion–Mean deviation, and Quartile deviation -Moments – Skewness –Kurtosis - Correlation and Regression. **L:9; T:3**

CO4: Apply the concept of testing of hypothesis for small and large samples

Sampling distributions - Estimation of parameters - Statistical hypothesis - Large sample tests based on Normal distribution for single mean and difference of means -Tests based on t and Chi-square distribution - Contingency table for independent of attributes – Goodness of fit. **L:9; T:3**

CO5: Apply the basic concepts of classifications of design of experiments

Tests based on F distributions for mean, variance and proportion- One way and two way classifications - Completely randomized design – Randomized block design– Latin square design – 2^2 factorial design. **L:9; T:3**

TEXT BOOKS:

1. Richard A. Johnson, Irwin Miller, John Freund, Miller & Freund's, Probability and Statisticsfor Engineers, 9th Edition, Pearson Education Limited, Global Edition, 2017.
2. Grewal.B.S., Higher Engineering Mathematics, 44th Edition, Khanna Publications, Delhi, 2017.
3. Erwin Kreyszig, Advanced Engineering Mathematics, 10th Edition, Wiley India, 2017.

REFERENCES:

1. R.E. Walpole, R.H. Myers, S.L. Myers, and K Ye, Probability and Statistics for Engineers and Scientists, Pearson Education, Asia, 9th Edition, 2016.
2. M.R. Spiegel, J. Schiller and R.A. Srinivasan, Schaum Outlines, Probabilityand Statistics, Tata McGraw Hill Edition, 2017.
3. Chapra, S.C and Canale, R. P. Numerical Methods for Engineers, 7th Edition, Tata McGraw Hill, New Delhi, 2016.
4. Jain R.K. and Iyengar S.R.K., Advanced Engineering Mathematics, 5thEdition, Narosa Publishing House Private Limited, 2016.

L: 45; T: 15; TOTAL: 60 PERIODS

Course Code	**COMPUTER NETWORKS**	**L**	**T**	**P**	**E**	**C**
23CS35C	(Common to CSE and IT Degree Programmes)	**3**	**0**	**2**	**0**	**4**

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Component

CO1: analyze the functionalities of various layers and network components (CDL1)

CO2: evaluate various access control mechanisms for error free data communication. (CDL1)

CO3: apply various flow, congestion control and routing algorithms for optimal path detection (CDL2)

CO4: analyze the performance of various application layer protocols (CDL1)

Practical Component

CO5: design different network topologies using network simulation tool. (PDL1)

CO6: implement the various services of the data link and network layer.(PDL1)

CO7: demonstrate the protocols in application layer (PDL2)

CO1: Analyze the functionalities of various layers and network components **L:9;**

Introduction– Networks Types – connection-transfer modes-Network Components: Devices and medium - Topology – Protocol Layering – TCP/IP protocol suite –OSI Model - Switching Networks – Network operating system: modes of operation **P:6**

CO5: Design different network topologies using network simulation tool

Representing a network- configure a network switch – Network troubleshooting Commands- implement basic connectivity

CO2: Evaluate various access control mechanisms for error free data communication. **L:12; P:8**

Data Link Layer – Framing – Flow control – Error control – Data Link Layer Protocols

and standards – HDLC –PPP - Ethernet Basics-Media Access Control: Addresses-CSMA/CA-CSMA/CD – Virtual LAN – Wireless LAN (IEEE802.11).

CO6: Implement the various services of the data link and network layer.

Implementation of Error Detection Techniques and framing methods-simulation of VLAN

CO3: Apply various flow, congestion control and routing algorithms for optimal path detection **L:15; P:10**

Introduction - IPV4 Addresses –CIDR - Address Mapping - ARP, RARP, and DHCP-ICMP-NAT- IPv6 Addresses -Transition from IPV4 to IPV6 - Distance Vector Routing: RIP - Link State Routing: OSPF-TCP- Congestion control - Congestion avoidance-UDP-Datagram -Services- Applications.

CO6: Implement the various services of the data link and network layer

Configuration of router interfaces -Demonstrate the ARP & RARP process in local and remote network-Implement routing protocols

CO4: Analyze the performance of various application layer protocols **L:9; P:6**

Introduction to Sockets - Application Layer protocols: HTTP – FTP – Email protocols (SMTP - POP3 - MIME) – DNS – SNMP.

CO7: Demonstrate the protocols in application layer

analyze the network traffic - Implement Chat application

TEXT BOOKS

1. Behrouz A. Forouzan, Data Communication and Networking with TCP/IP suite, 6th Edition, McGraw Hill, 2022.
2. Larry L. Peterson, Bruce S. Davie, Computer Networks: A Systems Approach, 6th Edition, Morgan Kaufmann Publishers Inc., 2021.

REFERENCES

1. Andrew S. Tenenbaum, Nick feamster, David J. wetherall, Computer Networks, 6th Edition, Pearson Education, New Delhi, 2022
2. James F. Kurose, Keith W.Ross, Computer Networking A Top-down Approach, 8th Edition, Pearson Education., New Delhi, 2022
3. William Stallings, "Data and Computer Communications", 10th Edition, Pearson Education, 2017.

ONLINE SOURCES

1. https://nptel.ac.in/courses/106/105/106105081/www.nptel.ac.in

https://www.udemy.com/course/computer-networks-for-beginners-it-networking-fundamentals/

L: 45; P: 30; TOTAL: 75 PERIODS

Course Code	DATA STRUCTURES LABORATORY	L	T	P	E	C
23CS36C	(Common to CSE and IT Degree Programmes)	0	0	4	0	2

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Practical Component

CO1: develop suitable sorting, searching, linear and nonlinear data structures for efficient data handling. (PDL1)

CO2: perform hierarchical searching problems using hash, heaps and graph algorithms. (PDL2)

CO1: Develop suitable sorting, searching, linear and nonlinear data structures for efficient data handling. **P:40**

1. Implementation of Sorting and searching for various applications.
2. Implementation of Stack and Queue using Array and Linked List.
3. Applications of stack, Queue and Linked List.
4. Construction of Expression Tree.
5. Implementation of Binary Search Tree.
6. Implementation of Dictionary operations using AVL Tree.
7. Mini project

CO2: Perform hierarchical searching problems using hash, heaps and graph algorithms. **P:20**

1. Implementation of Priority Queue using Heap structures.
2. Applications of Hashing techniques.
3. Applications of Graph algorithms.
4. Mini project

Software Requirements

- GNU C/C++ Compiler
- OS - Linux

P: 60; TOTAL: 60 PERIODS

Course Code	LINUX SYSTEM ADMINISTRATION	L	T	P	E	C
23CS37C	(Common to CSE, IT and AI&DS Degree Programmes)	**0**	**0**	**2**	**0**	**1**

COURSE OUTCOMES

Upon completion of this course, the students will be able to

Practical Component

CO1: demonstrate the basic knowledge of Linux commands and file handling utilities by using Linux shell environment (PDL1)

CO2: implement basic shell Scripting (PDL1)

CO3: develop shell scripts to address real-world problems (PDL1)

LIST OF EXPERIMENTS

CO1: Demonstrate the basic knowledge of Linux commands and file handling utilities by using Linux shell environment **P:6**

Explore the LINUX Commands

a. Directory
b. File Manipulation
c. General-purpose
d. Network utilities
e. Disk utilities

f. Backup utilities and Filters

P:10

CO2: Implement basic shell Scripting

Shell Programming - Develop Shell script programs for the following:

a. Interactive shell script
b. Positional parameters
c. Arithmetic
d. If-then-fi, if-then-else-fi, & nested if-else
e. Logical operators
f. Else + if equals elif, case structure
g. While & for loop
h. Meta characters

P:14

CO3: Develop shell scripts to address real-world problems

Shell scripting for Real world problem solving

a. File Backup
b. Text File Search
c. Password Generator
d. Disk Cleanup
e. Memory Leak Detection
f. Cache Management
g. Swap Space Optimization

SOFTWARE REQUIREMENTS

Operating System: Linux (Ubuntu).

REFERENCES

1. Venkateshmurthy, "Introduction To Unix And Shell Programming", 1st Edition, Pearson Publisher India, 2016.
2. Behrouz A. Forouzan, Richard F. Gilberg, Unix and shell Programming, 1st Edition, Thomson Publisher, 2013.
3. Andrew S. Tanenbaum, ―Modern Operating Systems, 4th Edition, Pearson Education, 2014.
4. Robert Love, Linux System Programming - Talking Directly to the Kernel and C Library, O'Reilly Media, 2013.

P: 30 TOTAL: 30 PERIODS

Course Code	INTELLECTUAL PROPERTY RIGHTS STUDY	L	T	P	E	C
23GN03C	(Common to all B.E. / B.Tech. Degree Programmes)	0	0	0	4	2

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Experiential Component

CO1: Survey and practice the basic elements of existing patents.

CO2: Investigate and present the state of art technologies through effectual IP search.

Soft Skill Component

CO3: Present patent survey conclusions

CO1 Survey and practice basic elements of existing patents **E:30**

Basic elements of IPR – claims – infringements – Patent examination and Report - Case studies: patent survey.

CO2 Investigate and present the state of art technologies through effectual IP search **E:30**

Importance of IP search-factors to be considered for effective IP search-Hands-on Practice

REFERENCES

1. D.P. Mittal, "Indian Patents Law and Procedure", Taxman Publication, 2002
2. B.L. Wadera, "Patents, trademarks, copyright, Designs and Geographical Judications", 2010
3. P. Narayanan, "Intellectual Property Law", Eastern Law House, 2022
4. N.S.Gopalakrishnan& T.G.Agitha, "Principles of Intellectual Property", Eastern Book Company, Lucknow, 2009.

E:60 TOTAL:60 PERIODS

Course Code		L	T	P	E	C
23GN04C	**APTITUDE EXCELLENCE**	0	0	2	0	1

COURSE OUTCOMES

Upon the completion of the course the students will be able to

CO1: Infer appropriate methods to simplify computation
CO2: Develop problem solving skills on Time and Work and enhance arithmetic ability
CO3: Interpret fundamentals in quantitative techniques and solve problems quickly
CO4: Improve quantitative skills and solve problems on permutation and Combination
CO5: Acquire the knowledge of Cognitive ability and solve puzzles effectively

CO1: Infer appropriate methods to simplify computation

Simplification: Nested Series simplification(fraction) -BODMAS rule –Viraculum (or Bar) rule – Modulus of a real number –Multiplication shortcuts-Comparing Fractions-Data Sufficiency **P:6**

CO2: Develop problem solving skills on Time and Work and enhance arithmetic ability

Time and Work: Work Done - Days from Work:- Ratio – Efficiency –Work and wages - Data Sufficiency **P:6**

CO3: Interpret fundamentals in quantitative techniques and solve problems quickly

Time Speed Distance: Body moving in the same direction- Body moving in the opposite direction-Average speed- Meeting point- Data Sufficiency **P:6**

CO4: Improve quantitative skills and solve problems on permutation and Combination

Probability Permutation Combination: Fundamental Counting Principle – Computing Permutation – Circular Permutation – ComputingCombinations- Data Sufficiency **P:6**

CO5: Acquire the knowledge of Cognitive ability and solve puzzles effectively

Abstract reasoning: Mirror and water image – Figure Matrix –Pattern Completion – **P:6**

Logical puzzles –Dot situation - Ranking ordering.**Cognitive ability:** Blood Relation - Direction Sense Test-Data Sufficiency

REFERENCE BOOKS

1. R.V.Praveen, "Quantitative Aptitude and Reasoning", 3rd Edition , Eastern Economy Edition, PHI Learning 2016
2. Arun Sharma," Quantitative Aptitude for CAT", McGraw Hill Edge, 10th Edition 2022
3. Dr.R.Aggarwal, " Quantitative Aptitude", S Chand Publishing, Revised Edition 2017

P:30; TOTAL: 30 PERIODS

Course Code	JAVAFOR DEVELOPERS	L	T	P	E	C
23CS41C/ 23IT41C	(Common to CSE and IT Degree Programmes)	3	0	0	0	3

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Components:

CO1: employ object-oriented language principles to solve known applications (CDL1)
CO2: apply string handling, I/O streams and exception handling methods for developing text based applications (CDL1)
CO3: apply the principles of multithreading and collections to solve complex problems (CDL2)
CO4: explore Maven, Gradle and REST API for project management (CDL3)
CO5: build robust and efficient enterprise and web applications using Spring Boot.(CDL3)

CO1: Employ object-oriented language principles to solve known applications. L:10

Introduction to Java – OOPS Concepts: Classes and Objects – Methods – Inheritance – Polymorphism – Constructors– Abstract class and Interface - Packages and Access Modifiers.

CO2: Apply string handling, I/O streams and exception handling methods for developing text based applications L:8

Arrays - String Handling: Immutable – Comparison – StringBuffer and StringBuilder - String Tokenizer - File Handling: I/O Stream – Buffered Reader/Writer – Exception Handling – Built in and Custom Exception

CO3: Apply the principles of multithreading and collections to solve complex problems L:9

Multithreading: Thread Life Cycle –Thread Creation – Thread Priorities – Synchronization - Collections Framework: Collection Class – List – Queue – Map – Set – SortedSet – Deque – Comparator – Iterator in Java

CO4: Explore Maven, Gradle and REST API for project management L: 7

Maven: POM.XML, Maven Repositories- Gradle : Repositories and plugins – REST API – HTTP Requests and Responses – Annotations – Lombok

CO5: Build robust and efficient enterprise and web applications using Spring Boot L:11

Spring Core Framework – Features –Dependency Management – Spring Data –Spring Boot – JDBC Overview & Architecture –JDBC Driver Types–CRUD Operations – JPA – RESTful web services with Spring Boot

TEXT BOOKS:

1. Kathy Sierra, Bert Bates, Trisha Gee, "Head First Java – A Learner's Guide to Real World Programming", 3rd Edition, O'Reilly Media, Inc., May 2022.
2. Craig Walls, "Spring in Action", 6th Edition, Manning Publications, January 2022

REFERENCE BOOKS:

1. Herbert Schildt, "Java: The Complete Reference", 13th Edition, McGraw Hill, 2024.
2. Maurice Naftalin, Philip Wadler, "Java Generics and Collections", 2nd Edition, O'Reilly Media, Inc., June 2024.
3. Cay S.Horstmann, "Core Java, Volume I: Fundamentals", 12th Edition, Oracle Press, December 2021.

WEB REFERENCES:

1. Swayam NPTEL course on Programming in Java
2. https://www.udemy.com/course/neutrino-java-foundations
3. https://www.educative.io/path/spring-boot-development-for-java-programmers
4. https://www.udemy.com/course/spring-learnit
5. https://www.scaler.com/topics/course/java-beginners/

L: 45; TOTAL: 45 PERIODS

Course Code	CRYPTOGRAPHY AND CYBER SECURITY	L	T	P	E	C
23CS42C		3	0	0	0	3

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Component

CO1: recognize the basic mathematical concepts required for symmetric and asymmetric fields. (CDL1)

CO2: apply classical and modern cryptographic basic for secure communication in data transformation. (CDL1)

CO3: apply various symmetric and asymmetric algorithm for data encryption. (CDL2)

CO4: employ various hashing and authentication schemes for different applications. (CDL2)

CO5: analyze various attacks and its effects in cyber security.(CDL1)

CO1: Recognize the basic mathematical concepts required for symmetric and a symmetric fields **L:9**

Number theory – Modular Arithmetic - Euclid's algorithm – Congruence – Group, Rings, Fields, Finite Fields - Primes – Primality Testing – Euler's totient function, Fermat's and Euler's Theorem – Chinese Remainder Theorem – Exponentiation and logarithm.

CO2: Apply classical and modern cryptographic basic for secure communication in data transformation **L:9**

Computer Security Concepts – Security Attacks – Security Services and Mechanisms – Classical encryption techniques: Substitution techniques, Transposition techniques – Foundations of modern cryptography: Perfect security – Information Theory - Shannon's theorem – Product Cryptosystem – Cryptanalysis.

CO3: Apply various symmetric and asymmetric algorithm for data encryption **L:9**

Symmetric Key Algorithms: Stream and Block Ciphers - Data Encryption Standards (DES) Advanced Encryption Standard (AES) - Linear Cryptanalysis - Differential Cryptanalysis - Asymmetric Key Algorithms: Rivest Shamir Adleman (RSA) algorithm – Elliptic Curve Cryptography (ECC).

CO4: Employ various hashing and authentication schemes for different applications L:9

Cryptographic Hash functions: Message Digest 5(MD5) – Secure Hash Algorithm (SHA512) – Message Authentication Code (MAC): HMAC – CMAC - Digital signatures - Diffie Hellman key Exchange - Authentication Applications: Kerberos-X.509 Authentication Service - Public key infrastructure.

CO5: Analyze various attacks and its effects in cyber security L:9

Cyber Security – Classification – Threads and Attacks: Password Cracking – Keyloggers Spywares - SQL Injection – Session Hijacking – Spoofing – Sniffing - Backdoor Trojan – Cross Site Scripting.

TEXT BOOKS

1. William Stallings, "Cryptography and Network Security - Principles and Practice", 8th Edition, Pearson Education, 2022.
2. Johann Rehberger, "Cybersecurity Attacks - Red Team Strategies", 1st Edition,Packt Publishing, 2020.
3. Jim S.Kraft, Larry C.Washington, "An Introduction to Number Theory with Cryptography", 2nd Edition, CRC Press, Taylor & Francis Group, 2018.

REFERENCE BOOKS

1. B.A. Forouzan, "Cryptography & Network Security", 3rd Edition, Tata Mc Graw Hill, 2016.
2. Charles Pfleeger, Shari Pfleeger, Jonathan Margulies, "Security in Computing", 5th Edition, Prentice Hall, 2015.
3. Jeffrey Hoffstein, Jill Pipher, Joseph H. Silverman, "An Introduction to Mathematical Cryptography", 2nd Edition, Springer, 2014.
4. Nina Godbole, Sunit Belapure, "Cyber Security: Understanding Cyber crimes, Computer Forensics and Legal Perspectives", 1st Edition, Wiley India, 2011.

L: 45; TOTAL: 45 PERIODS

Course code	EMBEDDED SYSTEMS	L	T	P	E	C
23CS43C		3	0	0	0	3

COURSE OUTCOMES

Upon completion of this course, the students will be able to

Theory Components:

CO1: relate the components of an Embedded System. (CDL 1)
CO2: utilize the peripherals of LPC2148 microcontroller. (CDL 1)
CO3: implement a basic interface program for the LPC2148 peripherals. (CDL 2)
CO4: analyze the programming fundamentals of Arduino UNO. (CDL 2)
CO5: implement a basic interface program in Python using the Raspberry Pi. (CDL 2)

Softskill Component:

CO6: Proficient to work effectively as an individual and in multidisciplinary teams.

CO1: Relate the components of an Embedded System(CDL 1) **L:9**

Definition: Embedded system, Intelligent System, Expert system – software architecture, hardware Architecture; Embedded System design process–types and comparison-Embedded System design examples: Modern train controller design, Elevator controller design.

CO2:Utilize the peripherals of LPC2148 microcontroller (CDL 1) **L:9**

Functional Block diagram - internal peripherals: GPIO, UART, PWM and ADC- On chip organization of I2C and SPI.

CO3: Implement a basic interface program for the LPC2148 peripherals(CDL 2) **L:9**

Introduction to ARM programming environment- Interfacing program for LEDs and Switches -Serial IO interfacing (UART) – LCD interface -Motor Interfacing

CO4: Analyze the programming fundamentals of Arduino UNO(CDL 2) **L:9**

Arduino IDE Overview-Operators in Arduino – Function Libraries – Control Statement – Loops – Arrays – String - Math library - Random Number – Interrupts -Sensor Interface examples with Arduino. Case Study: Traffic Control System.

CO5: Implement a basic interface program in Python using the Raspberry Pi(CDL 2) **L:9**

Installing Python - Python Data Types - Variables – Working with loops - Control flow statements - Functions-classes - Controlling LED using Raspberry pi - Interfacing LED and Switch with Raspberry pi - IR Sensor and Buzzer Interfacing-Ultrasonic Sensor Interfacing - Relay Interfacing.

TEXT BOOKS

1. Danny Staple, “Learn Robotics Programming” 2nd Edition, Packt Publishing, 2021.
2. Gary Smart, “Practical Python Programming for IoT”, Packt Publishing 2020.
3. Rajkamal, “Embedded system- Architecture, programming and design”, Mcgraw Hill, 2017
4. Shibu K.V, “Introduction to Embedded systems”, Mcgraw Hill, 2017. LPC 2148 User manual (www.arm.com)
5. Richard Blum Christine Bresnahan “Python Programming for Raspberry Pi”, Pearson Education, 2016

L: 45; TOTAL: 45 PERIODS

Course code	ENVIRONMENTAL SCIENCE AND ENGINEERING	L	T	P	E	C
23MC02C		2	0	0	0	0

COURSE OUTCOMES:

Upon the successful completion of the course, the student will be able to

CO1: explain the structure and functions of an ecosystem and the importance of biodiversity.

CO2: interpret the causes, effects of air and water pollution.

CO3: comprehend the causes, impacts and management of e-waste and municipal waste.

CO4: apply the knowledge of sustainability practices in the environment.

CO1: Explain the structure and functions of an ecosystem and the importance of biodiversity. **L:6**

Introduction to Environment**,** scope and importance of environment – need for public awareness. Eco-system: structure and function. Biodiversity: Introduction - types – values of biodiversity – threats to biodiversity: habitat loss, poaching of wildlife, man-wildlife conflicts – endangered and endemic species of India. Conservation of biodiversity: In-situ and ex-situ - Biodiversity index calculation (Simpson and Shannon diversity Index, Sorenson coefficient)

CO2: Interpret the causes, effects of air and water pollution. **L:6**

Air pollution - Classification of air pollutants – sources – Effects - Measurements: dust monitor – gas analyzer, particle size analyzer. Water pollution – Classification – health hazards – sampling and analysis of water. Waste water treatment – different industrial effluents and their treatment – Measurement: BOD and COD – atomic absorption spectrometer. Case study (Okhla sewage water treatment plant)

CO3: Comprehend the causes, impacts and management of e-waste and municipal waste. **L:12**

Integrated Waste Management: Introduction – Generation and types of solid waste – Swachh Bharat Mission – Solid waste management: collection, transportation, segregation and processing – Disposal: landfill – biochemical processes and energy recovery - Municipal solid waste management rules 2016.

e-Waste Management: Introduction – Composition - Types – Generation – Environmental and health hazards of e-waste – Recycling - Recovery of metals: pyrometallurgical, hydrometallurgical, and biometallurgical process – e-waste management and handling rules 2016 – e-waste management companies in India.

CO4: Apply the knowledge of sustainability practices in the environment. **L:6**

Sustainability and Management**:** Introduction - concept, needs and challenges –economic and social aspects of sustainability – unsustainability to sustainability –millennium development goals and protocols – Sustainable Development Goals-targets, indicators and intervention areas – Climate change – Global, Regional and local environmental issues and possible solutions – case studies. Concept of Carbon Credit – Carbon Footprint – Environmental management in industry – A case study – Zero waste and R concept – Circular economy – ISO 14000 Series – Material Life cycle assessment.

TEXT BOOKS:

1. Miller. G.T and Spoolman. S, ‘Environmental Science’, 16th Edition, Brooks/Cole Publishing Co., 2018.
2. Peavy. H.S, Rowe. D.R and Tchobanoglous. G, “Environmental Engineering”, 2nd Edition, McGraw Hill Education, 2020.
3. Benny Joseph, ‘Environmental Engineering’, Tata-Mc-Graw Hill, New Delhi, 2016.
4. Gilbert M. Masters, ‘Introduction to Environmental Science and Engineering’, 2nd Edition, Pearson Education, 2016.

REFERENCES:

1. Kaushik. A and Kaushik. C.P, ‘Environmental Science and Engineering’, 6th Edition, New Age International Publishers, 2018.
2. Weller. K, ‘Environmental Science and Biological Engineering’, 1st Edition, WIT Press, 2015.

L:30; TOTAL : 30 PERIODS

Course Code	DESIGN AND ANALYSIS OF ALGORITHMS	L	T	P	E	C
23CS44C / 23IT44C	(Common to CSE and IT Degree Programmes)	3	0	2	0	4

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Components

CO1: apply algorithmic analysis techniques to evaluate the efficiency of algorithms in terms of time and space complexities. (CDL2)

CO2: analyze algorithmic design using divide and conquer/greedy approach to solve complex problems. (CDL2)

CO3: derive optimal solutions for complex problems using dynamic programming. (CDL2)

CO4: use branch and bound and backtracking strategies along with the state space tree method to solve complex problems. (CDL2)

CO5: apply polynomial time algorithms to solve problems efficiently within reasonable time bounds. (CDL1)

Practical Component

CO6: design and implement efficient algorithms to produce optimal solutions for solving real world use cases. (PDL2)

CO1: Apply algorithmic analysis techniques to evaluate the efficiency of algorithms in terms of time and space complexities **L:10; P:3**

Notion of an Algorithm - Time and Space Complexity - Asymptotic notation and its properties - Recurrence Relations - The substitution method for solving recurrences - The master method for solving recurrences - Algorithm Analysis: Sorting Techniques - Towers of Hanoi, Sieve of Eratosthenes

CO6: Design and implement efficient algorithms to produce optimal solutions for solving real world use cases.

Implement sorting algorithms and recursive algorithms.

CO2: Analyze algorithmic design using divide and conquer/greedy approach to solve complex problems **L:9 P: 8**

Divide-and-Conquer: Introduction - Algorithm Analysis: Binary Search, Merge sort, Quick sort - Strassen's algorithm for matrix multiplication - Closest pair of points - Randomized Quick Sort.

Greedy Method: Introduction - Activity Selection Problem - Huffman Trees - Minimum spanning tree: Prim's and Kruskal's - Knapsack problem

CO6: Design and implement efficient algorithms to produce optimal solutions for solving real world use cases.

Solve problems using divide and conquer / greedy approaches.

CO3: Derive optimal solutions for complex problems using dynamic programming **L:10 P: 8**

Approaches of dynamic programming - Principle of Optimality - Computing Binomial Coefficient - Multi-stage graph - Floyd-Warshall algorithm - Matrix chain multiplication - 0/1 Knapsack problem - Optimal binary search tree - Longest common subsequence.

CO6: design and implement efficient algorithms to produce optimal solutions for

solving real world use cases.

Solve problems using dynamic programming approach.

CO4: Use branch and bound and backtracking strategies along with the state space tree method to solve complex problems **L:9 P: 8**

Back tracking: Introduction to backtracking - N-Queens Problem - Hamiltonian Circuit Problem - Graph coloring problem - Knight's tour Problem.

Branch and Bound: Introduction - Classification of Branch and Bound Problems: FIFO, LIFO, LC - 0/1 Knapsack problem - Solving 8-Puzzle problem

CO6: Design and implement efficient algorithms to produce optimal solutions for solving real world use cases.

Solve problems using branch & bound and backtracking approaches.

CO5: Apply polynomial time algorithms to solve problems efficiently within reasonable time bounds **L:7 P: 3**

Tractable and Intractable Problems - Complexity Classes - NP hardness: Clique decision problem - NP Completeness - Concept of Cooks Theorem - Proof of NP Completeness: CNF-SAT, Vertex Cover, Subset-Sum Problem

CO6: Design and implement efficient algorithms to produce optimal solutions for solving real world use cases.

Solve NP hard and NP complete problems using suitable approaches.

TEXT BOOKS:

1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein, "Introduction to Algorithms", 4th Edition, Prentice Hall of India, 2022.
2. Anany Levitin, "Introduction to the Design and Analysis of Algorithms", 3rd Edition, Pearson Education, 2019.

REFERENCE BOOKS:

1. Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran "Computer Algorithms" Orient Blackswan, 2nd Edition, 2019.
2. Sandeep Sen, Amit Kumar, "Design and Analysis of Algorithms A Contemporary Perspective, 1st Edition, Cambridge University Press, 2019.
3. Jon Kleinberg and Eva Tardos, "Algorithm Design", 1st Edition, Pearson/Addison-Wesley, 2014.
4. Dave, P. H, "Design and Analysis of Algorithms", 1st Edition, Pearson Education Canada, 2007.

L: 45; P:30; TOTAL: 75 PERIODS

Course Code	DATABASE MANAGEMENT SYSTEMS	L	T	P	E	C
23CS45C / 23IT45C	(Common to CSE, IT and AI&DS Degree Programmes)	**3**	**0**	**2**	**0**	**4**

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Components:

CO1: design ER model and apply the SQL concepts to perform various operations in a relational

database (CDL2)

CO2: apply normalization concept to remodel the database and maintain consistency with suitable transaction management. (CDL2)

CO3: apply the intermediate query concept to retrieve data from multiple relations. (CDL2)

CO4: apply PL/SQL constructs to enhance the database activities and develop application using database connectivity drivers. (CDL2)

CO5: apply the query evaluation plan and optimize the query to reduce complexity. (CDL1)

Practical Components:

CO6: demonstrate database normalization using appropriate tool for enhancing data integrity (PDL2**)**

CO7: design and implement SQL queries for data manipulation. (PDL2)

CO8: build applications using database connectivity drivers and implement programming concepts using PL/SQL constructs. (PDL2)

CO9: demonstrate indexing and partitioning concepts using Common Table Expression. (PDL1)

CO1: Design ER model and apply the SQL concepts to perform various operations in a relational database

CO2: Apply normalization concept to remodel the database and maintain consistency with suitable transaction management.

CO6: Demonstrate database normalization using appropriate tool for enhancing data integrity

Introduction to Database System-Views of data - Database System Architecture – ER Models – Enhanced-ER Model - Relational Model- ER-to-Relational Mapping. SQL: Keys - DDL Statements - DML Statements – DCL & TCL Statements - SQL Aggregate Functions. **L:19; P:8**

Functional Dependencies –Non loss Decomposition–Normalization: First, Second, Third Normal Forms, Dependency Preservation –Boyce/Codd Normal Form. Denormalization Techniques: Horizontal, Vertical and Mixed. Transaction: Basic Concepts -Transaction Recovery –ACID Properties - Concurrency – Deadlock

Develop Database Design using ER Diagram and Perform Database Manipulations operations using DDL, DML, TCL and DCL commands.Perform normalization for the given schema using appropriate tool.

CO3: Apply the intermediate query concept to retrieve data from multiple relations **L:8;**
CO7: Design and implement SQL queries for data manipulation **P:7**

SQL Clauses - Group By – Having - Sub Query – Nested Sub Query -Null Functions – Indexes - Sequences – SQL Joins - Types – Views - SQL Injections - SQL Window functions -Types: Aggregate- Value- Ranking

Writing SQL Queries to retrieve data from multiple relations.

CO4: Implement PL/SQL constructs to enhance the database activities and develop application using database connectivity drivers. **L:9; P:10**

CO8: Build applications using database connectivity drivers and implement programming concepts using PL/SQL constructs.

PL/SQL Introduction - Control statements –Procedures - Functions – Cursors – Triggers – Exceptions – Embedded SQL - ODBC Connectivity- Case Study: CRUD operations in Industry specific application.

Develop PL/SQL Programs to demonstrate the concept of Control statements –Procedures - Functions – Cursors – Triggers – Exceptions.
Develop application to perform CRUD operations using ODBC connectivity.

CO5: Apply the query evaluation plan and optimize the query to reduce complexity. L:9; CO9: Demonstrate indexing and partitioning concepts using Common Table Expression. P:5

Query Processing: Parsing and Translation, Optimization, Evaluation; Measures of Query Cost. Query Optimization: Query Evaluation Plan (QEP), cost based query optimization, Estimation of QEP cost. Indexing concepts– CTE– Partitioning.

Create indexing and implement partitioning concepts using Common Table Expression (CTE) for the given schema.

TEXT BOOKS

1. Silberschatz, A, Henry F. Korth, and S. Sudharshan, "Database System Concepts", 7th Edition, Tata McGraw Hill, 2019.
2. David M. Kroenke, David J. Auer, Scott L. Vandenberg, Robert C. Yoder, "Database Concepts", 9thEdition, Pearson Education, 2020.

REFERENCE BOOKS

1. Ramez Elmasri and Shamkant B. Navathe, "Fundamentals of Database Systems", 7th Edition, Pearson/Addisionwesley, 2019.
2. Wilfried Lemahieu, KU Leuven, BelgiumSeppe vanden Broucke, KU Leuven, BelgiumBart Baesens, KU Leuven, Belgium, "Principles of Database Management: The Practical Guide to Storing, Managing and Analyzing Big and Small Data",August 2018
3. C. J. Date, A. Kannan and S. Swamynathan, "An Introduction to Database Systems", 8th Edition, Pearson Education, 2006.

WEB REFERENCE

"Introduction to Database Systems" -NPTEL Course.

L:45; P:30; TOTAL: 75 PERIODS

Course Code	JAVA PROGRAMMING LABORATORY	L	T	P	E	C
23CS46C / 23IT46C	(Common to CSE and IT Degree Programmes)	0	0	2	2	2

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Practical Components:

CO1: develop modular and scalable programs by applying object-oriented programming concepts for different scenarios. (PDL2)

CO2: build simple and robust applications using Spring Boot (PDL2)

Experiential Component:

CO3: synthesize and articulate the ideas and experiences for solving known applications(PDL3)

Soft Skill Component:

CO4: function effectively in heterogeneous teams through the knowledge of team work, Inter-

personal relationships, conflict management and leadership quality

List of Experiments

CO1: Develop modular and scalable programs by applying object-oriented programming concepts for different scenarios. (PDL2) **P:20; E:10**

1. Implementation of OOPS concepts.
2. Exercises on abstract class, interfaces and package concepts.
3. Exercises on array and string manipulation
4. Exercises on file and exception handling.
5. Implementation of Multithreading concept
6. Implementation of collection classes.

CO2: Build simple and robust applications using Spring Boot (PDL2) **P:10; E:20**

1. Implementation of Spring Boot with JPA.
2. Mini Project

SOFTWARE REQUIREMENTS

- Front end: JAVA, IDE: Net beans / Eclipse
- Back end: Oracle

TEXT BOOKS:

1. Kathy Sierra, Bert Bates, Trisha Gee, "Head First Java – A Learner's Guide to Real World Programming", 3rd Edition, O'Reilly Media, Inc., May 2022.
2. Craig Walls, "Spring in Action", 6th Edition, Manning Publications, January 2022

REFERENCE BOOKS:

1. Herbert Schildt, "Java: The Complete Reference", 13th Edition, McGraw Hill, 2024.
2. Kishori Sharan, Peter Spath, "Learn JavaFX 17: Building User Experience and Interfaces with Java", 2nd Edition, APress, 2022.
3. Maurice Naftalin, Philip Wadler, "Java Generics and Collections", 2nd Edition, O'Reilly Media, Inc., June 2024.
4. Cay S.Horstmann, "Core Java, Volume I: Fundamentals", 12th Edition, Oracle Press, December 2021.

P:30; E:30; TOTAL:60 PERIODS

Course Code	SYSTEM MODELING PROJECTS	L	T	P	E	C
23CS47C/ 23IT47C	(Common to CSE & IT Degree Programmes)	**0**	**0**	**2**	**2**	**2**

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Practical component:

CO1: apply basic mathematics, science, and engineering concepts to develop models for systems or products across various domains.(PDL1)

Experiential component:

CO2: develop and analyze models using fundamental principles to solve real-life problems.(PDL2)

Soft skill component:

CO3: develop the ability to work collaboratively in teams to model and simulate systems while effectively communicating technical results using suitable tools and techniques.

COURSE OVERVIEW

This course is designed to equip students with the skills to develop and analyze mathematical models for various systems across different domains. It focuses on applying fundamental principles of mathematics, science, and engineering to represent and predict the behaviour of these systems. Through hands-on experience, students will learn to create robust models that address real-world challenges, ranging from system optimization to problem-solving in practical applications. The course emphasizes integrating theoretical concepts with practical solutions, preparing students to design efficient and effective models while promoting technical excellence and professional growth.

P: 30; E: 30; TOTAL: 60 PERIODS

Course Code	THEORY OF COMPUTATION	L	T	P	E	C
23CS51C		3	1	0	0	4

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Components

CO1: show proficiency in converting finite automata to regular expressions and vice versa. (CDL2)
CO2: exhibit the conversion of context free grammars into normal forms. (CDL2)
CO3: apply the relationship between pushdown automata and context free grammars. (CDL2)
CO4: design Turing machines for recognizing computable languages. (CDL2)
CO5: apply the concept of reduction to show the problem is undecidable. (CDL1)

CO1: Show proficiency in converting finite automata to regular expressions and vice versa **L:9; T:3**

Introduction to Finite Automata (FA) – Basic notation and convention – Deterministic Finite Automata (DFA) – Nondeterministic Finite Automata (NFA) – Equivalence of NFA and DFA – NFA with ε moves – Equivalence of NFA's with and without ε-moves – DFA minimization-Regular Expression (RE) – Equivalence of FA and Res

CO2: Exhibit the conversion of context free grammars into normal forms **L:9; T:3**

Regular Language (RL) – Pumping Lemma for Regular Languages – Proving languages not to be regular. Context Free Grammar (CFG) – Derivations and Languages – Parse tree – Ambiguity in CFG – Simplification of CFG – Normal forms – Chomsky Normal Form (CNF) – Greibach Normal Form (GNF).

CO3: Apply the relationship between pushdown automata and context free grammars **L:9; T:3**

Context Free Language (CFL) – Pumping lemma for CFL- non-CFLs – Closure properties of CFL – Pushdown Automata (PDA) – Moves – Instantaneous descriptions – Acceptance by empty stack – Construction of PDA-Equivalence of PDA and CFG – Deterministic CFL

CO4: Design Turing machines for recognizing computable languages. **L:9; T:3**

Turing Machines ™ – Configuration – Models – Computable languages and functions – Construction of Turing Machine- Programming techniques for Turing machine construction – Non deterministic Turing machine- Configuration graphs.

CO5: Apply the concept of reduction to show the problem is undecidable. **L:9; T:3**

Undecidability - Recursive and Recursively enumerable languages – Properties of Recursive and Recursively enumerable languages - Encoding of Turing machine - Halting problem – Universal Turing machine - Reduction - Application of Reduction - Rice's Theorem.

TEXT BOOKS

1. J.E. Hopcroft, R. Motwani and J.D. Ullman, "Introduction to Automata Theory, Languages and Computations", 3rd Edition, Pearson Education, 2008.
2. Michael Sipser, "Introduction to the Theory of Computation", 4th Edition, Cengage Publisher, 2019.

REFERENCE BOOKS

1. J. Martin, "Introduction to Languages and the Theory of computation", 4th Edition, Tata McGraw Hill, 2011.
2. K.Anuradha, Y.Vijayalata, "Formal Languages and Automata Theory", CBS Publishers and Distributors, 2017.
3. Kamala Krithivasan and R.Rama, "Introduction to Formal Languages, Automata Theory and Computation" Pearson, 2nd Edition, 2019.
3. Peter Linz, "Introduction to Formal Languages and Automata", 7th Edition, Jones & Bartlett Publisher, 2021.
4. Pallavi Vijay Chavan and Ashish Jadhav, "Automata Theory and Formal Languages", Academic Press, 2023.

L: 45 T: 15 TOTAL: 60 PERIODS

Course code		L	T	P	E	C
23CS52C	OBJECT ORIENTED ANALYSIS AND DESIGN	3	0	0	0	3

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Components

CO1: realize OOAD principles and Unified Modeling Language to model the systems. (CDL1)
CO2: apply GRASP and GoF design patterns to develop modular, reusable, and maintainable designs. (CDL1)
CO3: design static UML diagrams by identifying the conceptual classes and its relationships for the given scenario. (CDL2)
CO4: create appropriate dynamic UML diagrams for different context. (CDL2)
CO5: apply Object-Oriented Analysis and Design (OOAD) using CASE tools. (CDL2)

CO1: Realize OOAD principles and Unified Modeling Language to model the systems. **L:9**

Introduction to OOAD with OO Basics - Unified Process – UML diagrams – Use Case – Case study –the Next Gen POS system, Inception -Use case Modeling – Relating Use cases – include, extend and generalization.

CO2: Apply GRASP and GoF design patterns to develop modular, reusable, and maintainable designs. **L:10**

GRASP: Designing objects with responsibilities – Creator – Information expert – Low Coupling – High Cohesion – Controller Design Patterns – creational – factory method – structural – Bridge – Adapter – behavioural –Strategy – observer –Applying GoF design patterns – Iterator Pattern-Mapping design to code.

CO3: Design Static UML diagrams by identifying the conceptual classes and its relationships for the given scenario. **L:9**

Class Diagram— Elaboration – Domain Model – Finding conceptual classes and description classes –Associations – Attributes – Domain model refinement – Finding conceptual class Hierarchies – Aggregation and Composition

CO4: Create appropriate dynamic UML diagrams based for different context. **L:9**

Dynamic Diagrams: Interaction diagrams: Sequence diagram – Collaboration diagram – Communication Diagrams - State machine diagram and Activity diagram

CO5: Apply Object-Oriented Analysis and Design (OOAD) using CASE tools. **L:8**

Implementation Diagrams: Package diagram- Component and Deployment Diagrams- Case Studies: Analyzing Autonomous drone management system, Smart home automation Systems. CASE Tools: Star UML/ UML Graph for the above Case studies.

TEXT BOOKS

1. Ali Bahrami,"Object Oriented System Development", Tata McGraw Hill, 2nd Edition, 2017.
2. Ralph Johnson, Erich Gamma, Richard Helm, John Vlissides, "Design Patterns: Elements of Reusable Object Oriented Software", Pearson Publication, 1st Edition, 2015.

REFERENCES:

1. Grady Booch, James Rumbaugh, Ivar Jacobson," The Unified Modeling Language User Guide", Addison Wesley, 2nd Edition, 2015.
2. Craig Larman, "Applying UML and Patterns: An Introduction to object-oriented Analysis and Design and Iterative Development", Pearson Education, 3rd Edition, 2008
3. Ambler SW. Bubbles and Lines—Useful Diagrams for Object-Oriented Analysis and Design. In: Building Object Applications That Work: Your Step-by-Step Handbook for Developing Robust Systems with Object Technology. SIGS: Managing Object Technology. Cambridge University Press; 1997:49-84.

L: 45; TOTAL: 45 PERIODS

Course code	DEVOPS AND AGILE METHODOLOGIES	L	T	P	E	C
23CS53C / 23AD51C	(Common to CSE and AI & DS degree Programmes)	1	0	2	0	2

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Components

CO1: understand the foundational knowledge of software process models, AWS, and core DevOps practices (CDL1)

Practical Components

CO2: develop hands-on expertise in deploying and managing core AWS services (PDL1)

CO3: acquire practical expertise in Git workflows and remote repository management on GitHub (PDL1)

CO4: demonstrate hands-on expertise in Jenkins job creation, pipeline setup, and plugin management (PDL1)

CO5: develop and configure Docker environments to automate the build and deployment of application (PDL1)

CO1: Understand the foundational knowledge of software process models, AWS, and core DevOps practices **L:9**

Software Engineering - Traditional Process models: Waterfall, Incremental, Spiral models - Agility principles - Agile process models: XP, SCRUM.

CO2: Develop hands-on expertise in deploying and managing core AWS services

AWS Cloud Services - Elastic Compute Cloud EC2 - Simple Storage Services (S3) - Virtual Private Cloud (VPC) - Identity and Access Management (IAM) - AWS Monitoring Service: Cloud watch. Design and implement a secure and scalable web application on AWS. **L:1; P:10**

CO1: Understand the foundational knowledge of software process models, AWS, and core DevOps practices

CO3: Acquire practical expertise in Git workflows and remote repository management on GitHub **L:3; P:6**

DevOps Essentials - DevOps Lifecycle - Concepts of CI/CD- Overview of Version Control System: Git workflows, Working with Remote Repositories - Github. Implement CI/CD concepts using Git workflows and collaborate on remote repositories through GitHub.

CO4: Demonstrate hands-on expertise in Jenkins job creation, pipeline setup, and plugin management **L:1; P:8**

Jenkins Overview & Architecture - Installation & Basic Configuration - Creating & Configuring Jenkins Jobs - Pipeline - Plugins. Implement Jenkins pipeline script to automate code building and deployment.

CO5: Develop and configure Docker environments to automate the build and deployment of applications.

Docker overview - Installing docker - Pulling images - Running images - Docker build and deployment- Exposing volumes and ports-Deploying a Multi-tier Application Using Docker Network. Design and implement a multi-tier web application using Docker Compose **L:1; P:6**

TEXT BOOKS

1. Rama Bedarkar, "Agile Scrum: Improving Practices for Business Gains", Wiley publications, 2020.
2. K K Aggarwal , Yogesh Singh, "Software Engineering", New Age International Private Limited publications, 4th Edition, 2022
3. Deepak Gaikwad, Viral Thakkar, "DevOps Tools from Practitioner's Viewpoint", Wiley publications, 2020.

REFERENCES:

1. Vikas B. Maral, Gajanan Parshuram Arsalwad, Deepali N. Bhaturkar, Prashant L. Mandale, Nagaraju Bogiri, Mr. Abhijeet D. Cholke , "Agile Methodologies and Devops", 2024.
2. Toby Velte, Anthony Velte, Robert Elsenpeter, "Cloud Computing A Practical Approach", 2017.

L: 15; P: 30 TOTAL: 45 PERIODS

Course code	MODERN WEB TECHNOLOGIES	L	T	P	E	C
23CS54C/ 23IT53C	(Common to CSE and IT degree Programmes)	1	0	2	4	4

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Component

CO1: apply basic design principles to design an interactive frontend web applications(CDL2)

CO2: recognize the importance of building scalable backend web solutions (CDL2)

Practical Component

CO3: design and implement a dynamic webpage using HTML, CSS, and JavaScript (PDL2).

CO4: build interactive and responsive frontend applications using React (PDL2).

CO5: develop, deploy and manage robust web applications with Node.js (PDL2).

Experiential Component

CO6: design and develop a MERN Stack application for real world scenarios (PDL3)

Soft skill Component

CO7: showcase effective communication, collaboration, and team spirit to thrive in diverse professional settings.

CO1: Apply basic design principles to design an interactive frontend web applications **L:9; P:18; E:30**

MVC Architecture — Objects and Arrays- Document Object Model (DOM) – Arrow function - Promises – Async/Await – Modules - Introduction to React – React DOM and Virtual DOM – JSX –Components and Life cycle - React Hooks– Redux– Store

CO3: Design and implement a dynamic webpage using HTML, CSS, and JavaScript

Build a simple computation application – Programs using objects and arrays with static and instance methods – Develop forms with validation and event handling – Implement API Call and handle responses - Design a portfolio website.

CO4: Build interactive and responsive frontend applications using React

Design a front-end application with Components, Properties, Lists, and Keys – Implement State management using hooks – Apply Conditional Rendering, Event Handling, and

Routing with React Router - Manage State with Reducers and Context API – Create a blog using API calls and Error handling

CO2: Recognize the importance of building scalable backend web solutions **L:6 ; P:12; E:30**

Introduction to Nodejs - Event driven Architecture - Node Package Manager (NPM) – Express framework – Server side rendering with Template engines – Mongo db – Container: Docker.

CO5: Develop, deploy and manage robust web applications with Node.js

Implement Node.js Event Loop – Create Modules and handle asynchronous operations using callbacks - Setup an Express.js application – Implement RESTful methods– Fetch and handle JSON data from Express.js - Implement CRUD Operation using Mongo DB – Deploy the application.

TEXT BOOKS

1. Alex Banks, Eve Porcello, "Learning React –Modern Patterns for developing React apps", 2nd Edition, O'Reilly, 2020.
2. Mario Casciaro and Luciano Mammino, "Node.js Design Patterns", 3rd Edition, O'Reilly Media, 2022.

REFERENCE BOOKS

1. Stoyan Stefanov," React: Up and Running: Building Web Applications", 2nd Edition, O'Reilly Media, 2021.
2. Bethany Griggs," Node Cookbook: Discover solutions, techniques, and best practices for server-side web development with Node.js 14", 4th Edition, Packt Publishing Ltd., 2020.
3. Shama Hoque ,"Full-Stack React Projects: Modern web development using React, Node, Express, and MongoDB", 2nd Edition, Packt Publishing Ltd., 2020

L: 15; P: 30; E: 60; TOTAL: 105 PERIODS

Course code	ARTIFICIAL INTELLIGENCE	L	T	P	E	C
23CS55C/ 23IT52C	(Common to CSE and IT degree Programmes)	3	0	2	0	4

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Components

CO1: apply the suitable agent strategy to solve a combinatorial problem (CDL1)
CO2: analyze the appropriate search algorithms for solving goal-oriented problems (CDL2)
CO3: build logical reasoning models using propositional and first order logic (CDL1)
CO4: analyze planning and acting principles with uncertainty in robot environments (CDL2)
CO5: apply emerging capabilities of AI based tools to automate tasks (CDL2)

Practical Components

CO6: develop solutions for AI problems using problem solving agents (PDL1)
CO7: apply various AI search strategies to solve goal-oriented problems (PDL1)
CO8: implement real time applications using probabilistic reasoning (PDL2)
CO9: design and develop a mini project using GPT-based AI tool for real world applications (PDL2)

CO1: Apply the suitable agent strategy to solve a combinatorial problem

CO6: Develop solutions for AI problems using problem solving agents

Introduction to Intelligent Agents: Definition – Foundations of Artificial Intelligence – Future of Artificial Intelligence – Characteristics of Intelligent Agents. Typical Intelligent Agents – Problem Solving Approach to Typical AI problems - Search algorithms - Uninformed Search Strategies – Informed Search Strategies **L: 11; P: 10**

Solve basic search strategies – 8-Puzzle, N-Queens problem. Implement BFS, DFS, A* algorithms

CO2: Analyze the appropriate search algorithms for solving goal-oriented problems

CO7: Apply various AI search strategies to solve goal-oriented problems

Problem solving: Local Search Algorithms and Optimization Problems – Searching with Partial Observations – Constraint Satisfaction Problems – Constraint Propagation – Backtracking Search – Game Playing – Optimal Decisions in Games – Alpha – Beta Pruning – Stochastic Games. **L: 9; P: 6**

Solve constraint satisfaction problems. Implement Minmax algorithm for game playing (Alpha-Beta pruning).

CO3: Build logical reasoning models using propositional and first order logic

Knowledge based agents – Propositional Logic– Propositional theorem proving-First order Logic: Representation – Syntax and Semantics – Inference in First order Logic: Unification-Forward Chaining – Backward Chaining – Resolution. **L: 7**

CO4: Analyze planning and acting principles with uncertainty in robot environments

CO8: Implement real time applications using probabilistic reasoning

Preferences and Utility – One-off decisions – Sequential decisions – Value of Information and Control – Decision processes: Markov Decision Process (MDP) - Fully observable MDP – partially observable MDP **L: 9; P: 6**

Implement decision tree for a delivery robot. Solve uncertainty using belief network.

CO5: Apply emerging capabilities of AI based tools to automate tasks

CO9: Design and develop a mini project using GPT-based AI tool for real world applications

AI Concerns - AI Bias, Regulations, and Trustworthy AI – Ethical considerations of AI. Prompt Engineering: Five Principles of Prompting - GPT-based AI tool - Naive Prompting Approach – Persona pattern - Chain-of-Thought – Free-of-Thought. **L: 9; P: 8**

Develop real time applications using GPT – based AI tools.

SOFTWARE REQUIREMENTS:

1. Python 3.7x/4.0, Jupiter NoteBook / Google Colab

TEXT BOOK

1. S.Russel, P.Norvig, "Artificial Intelligence - A Modern Approach", Pearson Education, New Delhi, 4th Edition, 2022.

REFERENCES:

1. David L. Poole, Alan K. Mackworth, "Artificial Intelligence - Foundations of Computational Agents", 3rd Edition, Cambridge University Press, 2023.

2. James Phoenix, Mike Taylor, "Prompt Engineering for Generative AI", 1st Edition, O'Reilly Media, 2024.
3. Paula Boddington, "AI Ethics: A Textbook (Artificial Intelligence: Foundations, Theory, and Algorithms)", 1st Edition, Springer Verlag, Singapore, 2023

L: 45; P: 30; TOTAL: 75 PERIODS

Course Code	SIMULATION USING MODERN TOOLS	L	T	P	E	C
23CS56C/ 23IT54C/ 23AD56C	(Common to CSE and IT Degree Programmes)	**0**	**0**	**2**	**2**	**2**

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Practical Component

CO1: apply appropriate simulation tools to address complex system dilemmas. (PDL1)

Experiential Component

CO2: simulate and validate models to solve real world problems using modern tools (PDL2)

Soft skill Component

CO3: collaborate effectively with peers, demonstrating strong communication, teamwork, and problem-solving skills to enhance employability.

CO1: Appropriate simulation tools to address complex system dilemmas. (PDL1) P:20

Task 1: Simulation Tool Familiarization and Practice

- Explore fundamental concepts & features of various simulation tools in their respective domain.
- Simulate various system models using built-in templates/modules.

CO2: Simulate and validate models to solve real world problems using modern tools (PDL2) E:10

Task 2: Feasibility Study and Preliminary Review

- Conduct a feasibility study to assess the suitability of simulation and modeling tools and techniques for the chosen problem or project idea.
- Review –I (Feasibility study, identify suitable tools and challenges within the project scope.)

Task 3: Model framework with SOTA Analysis E:30

- Design a preliminary model structure tailored to address the identified problem
- Review II (Tool Selection and Model Framework for Chosen Project – with SOTA Analysis)

Task 4: Model Refinement, Optimization and Final Validation

- Model Refinement, Optimization and Final Validation
- Review III (Present the final model, validate the outcomes & Report Submission)

P: 20; E: 40; TOTAL: 60 PERIODS

Software Requirements (As per the domain of verticals)

S. No	Verticals	Required Simulation Tools (Open Source)
1.	Fullstack Development	**Frontend:** React, Vue.js, Angular, SASS/LESS, Bootstrap
		Backend: Node.js, Express.js, Django, Flask, Spring Boot
		Database: MySQL, PostgreSQL, MongoDB, SQLite
2.	Computational Intelligence	DEAP, NetLogo
3.	AR/VR	Unity, Unreal Engine
4.	Blockchain technology	Remix IDE, Ganache, Metamask, Solidity
5.	Business Analytics	R, Python, Octave, KNIME, SciLab, Tableau, Apache Spark, ExtendSim
6.	IoT and Robotics Design	TinkerCAD, IoTIFY, OpenRemote, Node-RED, Autosim, ThingSpeak
7.	Industrial AI	ACUMOS AI, CLEARML, H20.AI MYCROFT.AI, OPENCV, OPENN, PYTORCH, RASA OPEN SOURCE, TENSORFLOW, TESSERACT OCR
8.	Network and Cyber Security	NS-2 / NS-3, Mininet, Wireshark, Snort, Kali Linux, OpenVAS, OpenVAS

Course Code	PROJECT MANAGEMENT AND FINANCE	L	T	P	E	C
23GN06C	(Common to all Degree Programmes)	2	0	0	0	2

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Component

CO1: select and formulate projects (CDL1)

CO2: estimate the project cost and make an investment decision (CDL1)

CO3: apply the scheduling and resource allocation techniques to control and monitor the project (CDL2)

CO1: Select and formulate projects

Project – Concepts –Project Life Cycle – Project constraints- Generation and Screening of Project Ideas - Project identification – Preliminary Analysis, Market, Technical, Financial, Economic and Ecological - Pre- Feasibility Report and its Clearance, Project Estimates and Techno-Economic Feasibility Report, Detailed Project Report – Different Project Clearances required. **L:10**

CO2: Estimate the project cost and make an investment decision

Project Evaluation under certainty - Net Present Value (Problems - Case Study), Benefit Cost Ratio, Payback Period– Methodology for project evaluation – Social Cost Benefit Analysis, Commercial or National Profitability, social or national profitability **L:10**

CO3: Apply the scheduling and resource allocation techniques to control and monitor the project

Developing a Project Plan - Developing the Project Network – Constructing a Project Network (Problems) – PERT – CPM – Crashing of Project Network (Problems - Case Study) – Resource Leveling and Resource Allocation – Steps in Project Appraisal Process – Project Control Process – Control Issues – Project Audits – the Project Audit Process – project closure – team, team member and project manager evaluations. **L:10**

TEXT BOOKS:

1. Clifford Gray, Erik Larson and Gautam Desai, Project Management: The Managerial Process, Tata McGraw Hill, 8th Edition, 2021
2. Prasanna Chandra, Projects, Planning, Analysis, Selection, Financing, Implementation and Review, 1st Edition, Tata McGrawHill, 2023

REFERENCE BOOKS:

1. M Y Khan, P K Jain , Management Accounting, 8th Edition, McGraw Hill, 2021
2. K. Swarup, P. K. Gupta, and M. Mohan, *Operations Research: An Introduction to Management Science*, 20th Edition, New Delhi, India: Sultan Chand & Sons, 2022
3. Sudhakar, G P, Project management: the managerial aspects, 5th Edition, New Century Pub, 2020.
4. P. Gopalakrishnan and V. E. Ramamoorthy, *Textbook of Project Management*, 1st Edition, New Delhi, India: Laxmi Publications, 2022
5. Dr. K.L. Gupta, Management Accounting, Sahitya Bhawan Publications, 2022
6. Prem Kumar Gupta, Dr.D.S.Hira, Problems in Operation Research (Principles & Solutions), Kindle Edition, 2018

L: 30; TOTAL: 30 PERIODS

Course Code		L	T	P	E	C
23CS61C	BUSINESS PROCESS MANAGEMENT	3	0	0	0	3

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Component

CO1: understand the fundamentals of Business Process Management (CDL-1)
CO2: recognize and apply the fundamentals of BPMN to model basic business processes (CDL-1)
CO3: analyze the process discovery techniques and modeling methods (CDL-1)
CO4: apply the heuristic techniques and product-based design principles to redesign business processes, using real-world case studies such as healthcare systems (CDL-2)
CO5: analyze and implement business process automation through BPMS, addressing challenges (CDL-1)

CO1: Understand the fundamentals of Business Process Management **L:9**

Introduction - Ingredients of a Business Process - Origins and History - BPM Lifecycle - Process Identification - Key Processes - Designation phase - Evaluation Phase - Designing a

Process Architecture - Process selection - Essential Process Modeling

CO2: Recognize and apply the fundamentals of BPMN to model basic business processes **L:9**

Essential Process Modeling – First Steps with BPMN - Branching and merging - Exclusive Decisions - Parallel Execution - Inclusive Decisions - Rework and Repetition - Advanced Process Modeling - Process Decomposition - Process Reuse - Parallel Repetition - Uncontrolled Repetition - Handling Events - Handling Exceptions - Processes and Business Rules

CO3: Analyze the process discovery techniques and modeling methods **L:9**

Process Discovery - Process Analyst Versus Domain Expert - Three Process Discovery Challenges - Discovery Methods - Process Modeling Method - Process Model Quality Assurance - Syntactic Quality and Verification - Semantic Quality and Validation - Pragmatic Quality and Certification - Modeling Guidelines and Conventions - Qualitative Process Analysis - Value-Added Analysis - Root Cause Analysis - Issue Documentation and Impact Assessment

CO4: Apply the heuristic techniques and product-based design principles to redesign business processes, using real-world case studies such as healthcare systems **L:9**

Quantitative Process Analysis - Performance Measures - Flow Analysis - Queues - Simulation - Process Redesign - Heuristic Process Redesign - The Case of a Health Care Institution - Sending Medical Files by Post - Periodic Meetings - Requesting Medical Files - Product-Based Design

CO5: Analyze and implement business process automation through BPMS, addressing challenges **L:9**

Process Automation - Automating Business Processes - Architecture of a BPMS - The Case of ACNS - Advantages of Introducing a BPMS - Challenges of Introducing a BPMS - Turning Process Models Executable - Process Intelligence - Process Execution and Event Logs - Automatic Process Discovery - Performance Analysis - Conformance Checking - Robotic Process Automation - Case study

TEXT BOOKS

1. Marlon Dumas, Marcello La Rosa, Jan Mendling, Hajo A. Reijers, "Fundamentals of Business Process Management", 2nd Edition, Springer, 2018.
2. Akhil Kumar, "Business Process Management", 1st Edition, Routledge, 2018

REFERENCE BOOKS

1. Lehmann, Carl F. Strategy and Business Process Management: Techniques for Improving Execution, Adaptability, and Consistency. 1st Edition, CRC Press, 2016.
2. John Jeston, Johan Nelis, "Business Process Management", 3rd Edition, Routledge, 2014.
3. Van der Aalst, Wil, "Process Mining: Data Science in Action", 2nd Edition, Springer 2016.

L: 45; TOTAL: 45 PERIODS

Course Code		L	T	P	E	C
23CS62C	**COMPILER DESIGN**	**3**	**1**	**0**	**0**	**4**

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Component

CO1: recognize the role of lexical analysis in the compilation process.(CDL1)
CO2: apply the suitable parsing technique for checking the correctness of the Syntax in source code. (CDL1)
CO3: apply syntax-directed translation schemes to generate intermediate representations. (CDL2)
CO4: transfer intermediate code into basic block and flow graphs to generate efficient target code. (CDL2)
CO5: apply code optimization techniques and data-flow analysis to improve performance of the code.(CDL1)

CO1:Recognize the role of lexical analysis in the compilation process **L:9; T:3**

Introduction to Compiler – Analysis of the source program - Phases of Compiler – The analysis-synthesis model – Compiler Construction tools – Cousins – The role of the lexical analyzer - Input buffering -Specification of tokens - Recognition of tokens.

CO2:Apply the suitable parsing technique for checking the correctness of the Syntax in source code **L:9; T:3**

Role of Parser – Grammars – Error Handling – Review of CFGs– Top Down Parsing - Recursive Descent Parser Predictive Parser - transformation on the grammars, predictive parsing, bottom-up parsing, LRparsers (SLR, CLR, LALR).

CO3: apply syntax-directed translation schemes to generate intermediate representations **L:9; T:3**

Syntax-directed definitions: inherited and synthesized attributes, dependency graph, evaluation order,bottom-up and top-down evaluation of attributes, L- and S-attributed definitions, Circularity disambiguation - Intermediate languages – Declarative Statement - Assignment statements – Booleanexpressions - Case statements - Backpatching - Procedure calls.

CO4: Transfer intermediate code into basic block and flow graphs to generate efficient target code **L:9; T:3**

Issues in the design of a code generator - The target machine - Run-time storage management – Basicblocks and flow graphs - Next-use information - A simple code generator - Register allocation and assignment - The Directed Acyclic Graph (DAG) representation of basic blocks - Generating code from DAGs.

CO5: Apply code optimization techniques and data-flow analysis to improve performance of the code. **L:9; T:3**

Introduction - The principle sources of optimization - Peephole optimization - Optimization of basicblocks - Loops in flow graphs - Introduction to global data-flow analysis - Code improving transformations - Design a Simple compiler using mini java compiler.

TEXT BOOK

1. A.V.Aho, M.S.Lam, R.Sethi, and J.D.Ullman, Compilers: Principles, Techniques, and Tools, 2nd Edition, Pearson, April 2024.

REFERENCE BOOKS

1. Seth D. Bergmann, “Compiler Design: Theory, Tools, and Examples”, 2nd Edition, Rowan University, 2017.
2. C.N.Fisher, R.J.LeBlanc “Crafting a Compiler with C”, 3rd Edition, Pearson Education, 2010.

3. AW Appel, M Ginsburg, Modern Compiler Implementation in C, 1st Edition, Cambridge University Press, 2004.
4. Steven S. Muchnick, "Advanced Compiler Design & Implementation", Morgan Kaufmann Publishers, 1st Edition, Elsevier India, 2008.

L: 45; T: 15 TOTAL: 60 PERIODS

Course Code	CONSTITUTION OF INDIA	L	T	P	E	C
23MC01C	(Common to All Degree Programmes)	2	0	0	0	0

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Component

CO1: describe the salient features of the Indian Constitution.
CO2: discuss the structure and functions of parliament.
CO3: elaborate the structure and functions of state legislature.
CO4: explain the fundamentals of organization and working of the Judiciary.
CO5: discuss the foreign policy of India.

CO1: Describe the salient features of the Indian Constitution **L:6**

Salient Features – Preamble-Pillars of constitution - Fundamental Rights – Directive Principles of State Policy - Fundamental Duties.

CO2: Discuss the structure and functions of parliament **L:6**

Powers and Functions of President and Prime Minister - Council of Ministers – The Legislature Structure and Functions of Lok Sabha and Rajya Sabha – Speaker.

CO3: Elaborate the structure and functions of state legislature **L:6**

Features of Federal System - Administrative Relationship between Union and States – Powers and Functions of Governor and Chief Minister – Council of Ministers –State Legislature.

CO4: Explain the fundamentals of organization and working of the Judiciary **L:6**

Organization and Composition of Judiciary - Powers and Functions of the Supreme Court -

CO5: Discuss the foreign policy of India **L:6**

Foreign Policy of India – VISA Application Process- International Institutions like UNO, WTO, SAARC and Environmentalism.

TEXT BOOKS:

1. D.D.Basu, Introduction to the Constitution of India, 22nd Edition, New Delhi, India: Lexis Nexis, 2015.
2. D.C. Gupta, Indian Government and Politics, 8th Edition, New Delhi, India: Vikas Publishing House, 2010.

REFERENCES:

1. M.V. Pylee, An Introduction to the Constitution of India, 5th Edition, New Delhi, India: Vikas

Publishing House, 2009.
2. S. C. Kashyap, Our Constitution: An Introduction to India's Constitution and Constitutional Law, New Delhi, India: National Book Trust, 2010.
3. V. N. Shukla, Constitution of India, 11th Edition, Lucknow, India: Eastern Book Company, 2011.

L: 30; TOTAL: 30 PERIODS

Course Code		L	T	P	E	C
23CS63C	**DATA SCIENCE**	**2**	**0**	**2**	**2**	**4**

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Component

CO1: apply linear algebra and statistical methods to explore data relationships and distributions (CDL1)

CO2: formulate hypotheses, apply statistical tests and evaluate results to draw valid conclusions from data .(CDL2)

CO3: apply data manipulation, regression, and time series models to analyze data patterns and relationships. (CDL2)

CO4: Design data visualization techniques and data science models to interpret patterns in real-world applications. (CDL3)

Practical Component

CO5: implement linear algebraic concepts to analyze and interpret real-world data. (PDL2)

CO6: Apply and assess statistical methods for interpreting relationships in data. (PDL2)

CO7: Implement and evaluate data handling, regression, and forecasting approaches to identify and predict trends. (PDL2)

CO8: Build interactive dashboards and predictive models for real-world applications. (PDL3)

Experiential Component

CO9: Design and develop data science solutions through hands-on data exploration, visualization, and modeling for real-world applications. (PDL3)

Soft Skill Component

CO10: demonstrate diversity and inclusive attitude while practicing project component as a team

CO1: Apply linear algebra and statistical methods to explore data relationships and distributions. **L:6; P:6; E:6**

Linear Algebra: vectors and matrices - descriptive statistics, probability, distributions, mean, variance, covariance, covariance matrix, understanding univariate and multivariate normal distributions - SQL query optimization module: EXPLAIN/EXPLAIN ANALYZE, indexing, performance tuning

CO5: Implement linear algebraic concepts to analyze and interpret real-world data.

Implement linear algebra and statistics to reduce the dimensionality of a dataset - Implement statistical techniques to summarize real-world datasets - Implement covariance and correlation analysis to explore relationships between multiple variables

CO2: Formulate hypotheses, apply statistical tests and evaluate results to draw valid conclusions from data **L:8; P:6; E:6**

Statistical hypothesis testing: Null vs Alternative Hypothesis – Rejection Region and Significance Level –Type I Error and Type II Error – t-Test, z-Test, ANOVA, Chi-Square – Correlation and covariance – Hypothesis formulation and testing - A/B testing fundamentals - statistical significance interpretation

CO6: Apply and assess statistical methods for interpreting relationships in data.

Implement One-Way ANOVA to Compare Group Means - Implement a Chi-Square Test for Independence - Implement Correlation Analysis

CO3: Apply data manipulation, regression, and time series models to analyze data patterns and relationships. **L:8; P:8; E:8**

Getting the data from files - Working with Data: Exploring 1D and 2D data – Manipulating the data – Predictive Models: Linear Regression – multiple regression – nonlinear relationships – logistic regression – Time series analysis – moving averages – missing values – serial correlation – autocorrelation.

CO7: Implement and evaluate data handling, regression, and forecasting approaches to identify and predict trends.

Implement data exploration and data manipulation - Implement regression algorithms – Implement Moving Average and Exponential Smoothing for Forecasting

CO4: Design data visualization techniques and data science models to interpret patterns in real-world applications. **L:8; P:10; E:10**

Need for visualization - Principles of effective visualization – Connections and correlations – Scatter plot maps - Trees, Hierarchies and Recursion - Interactive visualization- - Advanced tools for visualization – ETL - data warehousing basics - introduction to Apache Airflow. Data Science Tools: Working with data set, data manipulation, Data analysis models - Pre-processing, predictive models. Case studies: social media analysis - sentimental analysis, health care applications- E-commerce applications

CO8: Build interactive dashboards and predictive models for real-world applications.

Interactive scatter plots for correlation analysis - Heatmap Visualization - Implement Data Distribution Visualization Techniques - Dashboard creation using advanced tools - Data manipulation & preprocessing using data science tools (Python / R / Matlab) - Develop and analyze case studies using predictive modeling techniques.

TEXT BOOKS

1. Peter Bruce, Andrew Bruce, and Peter Gedeck, Practical Statistics for Data Scientists, , O'Reilly Media publishers, 3rd Edition ,2024
2. Sanjeev Wagh, Manisha Bhende, Anuradha Thakare, 'Fundamentals of Data Science, CRC Press, 1st Edition, 2022.
3. Sam Lau, Joey Gonzalez, and Deb Nolan, Principles and Techniques of Data Science, by, 2019.
4. Joel Grus, 'Data Science from Scratch', O'Reilly, 2nd Edition, 2019.

REFERENCE BOOKS

1. Avrim Blum, John Hopcroft, Ravindran Kannan, Foundations of Data Science, Cambridge

University Press, First Edition, 2020.

2. VanderPlas, Jake. *Python Data Science Handbook: Essential Tools for Working with Data*. O'Reilly Media, 2016.
3. Joel Grus, Data Science from Scratch: First Principles with Python, O'Reilly Media, 1 st Edition, 2015.

L: 30 P:30 E:30 TOTAL: 90 PERIODS

Course Code	PRODUCT DEVELOPMENT PRACTICE	L	T	P	E	C
23CS64C	(Common to CSE, IT and AI & DS Degree Programmes)	0	0	0	4	2

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Experiential Component

CO1: identify and analyze real-world problems using empathy techniques and reverse engineering approaches. (PDL2)

CO2: apply forward engineering to develop innovative solutions, focusing on technical feasibility, patentability, and market potential. (PDL2)

Soft skill component

CO3: demonstrate the functionality of the developed product through prototypes and validate its commercial and patenting potential. (PDL2)

CO1: Identify and analyze real-world problems using empathy techniques and reverse engineering approaches. **25**

- Empathy-driven customer need identification
- Problem definition and market analysis
- Study of existing solutions and reverse engineering analysis
- Deriving specifications and functional gaps
- Conceptual design based on gaps and feasibility

CO2: Apply forward engineering to develop innovative solutions, focusing on technical feasibility, patentability, and market potential. **35**

- Concept refinement and solution detailing
- Rough model and Working model prototype development (hardware/software)
- Technical, financial, and IP (intellectual property) feasibility study
- Cost estimation and business model canvas
- Prototype demonstration and product documentation

E: 60; TOTAL: 60 PERIODS

Course Code	MINI PROJECT	L	T	P	E	C
23CS71C		0	0	0	6	3

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Experiential component:

CO1: identify and define an engineering problem through need analysis, systematic literature review

and feasibility study.

CO2: develop an appropriate methodology, preliminary design/model, and project plan with required resources and timelines.

Soft skill component:

CO3: communicate project ideas effectively through structured documentation, teamwork, and technical presentations.

Course content:

Selection of topic/problem based on relevance to industry / society / research, literature review – survey of existing solutions – research papers, identification of gaps, defining the problem clearly. Objectives of the project & scope.

Project Planning, Methodology, Block diagrams, Flow diagrams, Algorithm design, Resource identification, Basic simulation and prototype (if applicable), Life cycle costing. Structured documentation and Presentation

E: 90; TOTAL: 90 PERIODS

Course code	INTERNSHIP / IN-PLANT TRAINING	L	T	P	E	C
23CS72C		0	0	0	4	2

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Experiential Component:

CO1: apply emerging technologies to solve real-world problems as per industry/ organizational workflows industrial training/internship/live-in lab (PDL2)

Soft skill Component:

CO2: demonstrate effectively as an individual, in diverse, inclusive and multi-disciplinary teams with face-to- face, remote and distributed settings.

CO1: Apply emerging technologies to solve real-world problems as per industry/ organizational workflows through industrial training/internship/live-in lab. (PDL2)

Students should obtain departmental approval and join the organization physically or virtually to observe the workflows-apply technical knowledge to assigned tasks/ projects and maintain a weekly log of activities and progress - Document all work, results, and insights gained during the internship for final submission and presentation.

a) **Industrial Training / Internship Guidelines**

- Students must undergo Industrial Training or Internship during summer or winter vacation at MNCs, Startups, Public sector units, societal setups, research organizations, or globally ranked universities as per the guidelines of Head of the Department and Focus Group Heads/Domain Experts.
- In-house Internship/Training may be permitted within the college if approved projects or facilities are available.
- Student must get prior approval from the Head of the Institution.
- The training/internship **shall be continuous for at least two weeks** to four weeks and students can acquire during semester holidays from third semester. However, evaluation will be carried out during seventh semester.

- On completion, students must obtain an Attendance Certificate (in the format provided by the Dean – Academic) duly signed by the competent authority of the host organization.
- The Attendance Certificate must be submitted to the Head of the Institution, who will forward it to the Controller of Examinations (COE) through the Dean (Academic) for result processing

Live-in-Lab Programme

- The Live-in-Lab is an experiential learning programme aimed at solving real-world rural problems.
- Students may stay in the adopted village for a minimum of two continuous weeks.
- Students shall interact with villagers, identify issues, and develop practical, multidisciplinary solutions.
- After completion, students must present their findings and outcomes, which will be considered equivalent to an internship.
- Two review evaluations will be conducted to assess progress
- **Review 1:** Mid-term progress and activity log evaluation.
- **Review 2:** Final presentation and report evaluation.
- The final evaluation will be based on internship performance, report quality, and presentation skills.

The internship certificate from the organization must be submitted along with the final report.

E:60;TOTAL:60 PERIODS

.

Course code	**CAPSTONE PROJECT/ INDUSTRY PRACTICE**	**L**	**T**	**P**	**E**	**C**
23CS81C		**0**	**0**	**0**	**12**	**6**

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Experiential component:

CO1: identify, analyze, and define a significant engineering problem using based on need assessment, literature survey and domain knowledge.

CO2: apply appropriate engineering methods, design principles, tools, and modern software to develop and implement a solution or prototype with sustainability considerations.

CO3: conduct experiments, analyze data, evaluate results, and interpret findings with integrity and ethics.

Soft skill component:

CO4: prepare comprehensive project documentation and effectively present technical results in reputed conferences / journals

Course Content

Domain Analysis, Problem Identification, Literature Review / Market survey, Methodology, Design and Implementation, Experimental Work / Simulation / Analytical studies / Optimization / Testing, Project Management - Life cycle costing, Documentation & Presentation.

E: 180; TOTAL: 180 PERIODS

R-2023 B.E. CSE

PROGRAMME ELECTIVE COURSES

Course code 23CS01E/ 23IT01E/ 23AD01E	BLOCKCHAIN ARCHITECTURE AND DESIGN (Common to CSE, IT and AI & DS degree Programmes)	L	T	P	E	C
		3	0	0	0	3

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Components

CO1: analyze cryptographic primitives and their roles in ensuring blockchain security. (CDL2)

CO2: evaluate Bitcoin's transaction lifecycle and the Proof of Work consensus mechanism for securing the network. (CDL2)

CO3: develop Ethereum smart contracts and token standards, including their role in the mining process.(CDL2)

CO4: analyze various consensus mechanisms in permissioned blockchains for their effectiveness and applicability. (CDL2)

CO5: develop blockchain solutions using Hyperledger frameworks, focusing on enterprise applications. (CDL2)

CO1: analyze cryptographic primitives and their roles in ensuring blockchain security.

Cryptographic Hash Function - Properties of Hash Function – Digital Signature – Blockchain: Introduction – Types of Blockchain - Block in a Blockchain: Structure of a Block – Block Header – Transactions in a Block - Hash Pointer – Merkle Tree - Blockchain as a Hash chain – Accepting the longest chain – Orphaned Blocks – Blockchain Forks – UTXO and Account/Balance Models **L:9**

CO2: evaluate Bitcoin's transaction lifecycle and the Proof of Work consensus mechanism for securing the network.

Basics - History – Technology behind Bitcoin – Bitcoin Transaction Life Cycle - Creation of Coins – Sending Payments - Transaction in Bitcoin Network – Double Spending - Bitcoin Scripts - Hashcash PoW - Consensus in Bitcoin: Proof of Work (PoW) – Mining in a Bitcoin network - Mining difficulty - Bitcoin Wallets. **L:9**

CO3: develop Ethereum smart contracts and token standards, including their role in the mining process.

Ethereum Blockchain - Ethereum Virtual Machine - Ethereum Accounts - Transactions in Ethereum - Ethereum Wallets - Ethereum Gas - Gas Price - Gas Limit - Mining in Ethereum - Ether Tokens – ERC 20 - Ethereum Improvement Proposals (EIP) **L:9**

CO4: analyze various consensus mechanisms in permissioned blockchains for their effectiveness and applicability.

Basics - Consensus - State Machine Replication - Distributed Consensus - Consensus Algorithms - PAXOS - RAFT - Byzantine General Problem - Practical Byzantine Fault Tolerance. **L:9**

CO5: develop blockchain solutions using Hyperledger frameworks, focusing on enterprise applications.

Introduction to Hyperledger - Hyperledger frameworks - Hyperledger Fabric Architecture - Hyperledger fabric transaction flow - Chaincode in Hyperledger Fabric - Hyperledger Composer - Hyperledger Indy **L:9**

TEXT BOOKS:

1. Kumar Saurabh, Ashutosh Saxena, "Blockchain Technology concepts and applications", Wiley. 2020.
2. Caitlin Long, Antony Lewis, "The Basics of Bitcoins and Blockchains: An Introduction to Cryptocurrencies and the Technology that Powers Them", Mango Media, 2020.

REFERENCE BOOKS:

1. Andreas M. Antonopoulos, "Mastering Bitcoin: Programming the open blockchain", 2nd Release, 2017.
2. Andreas M. Antonopoulos and Wood M., "Mastering Ethereum: Building Smart Contracts and DApps", O'Reilly Media, 2018.
3. Xun (Brian) Wu, Chuanfeng Zhang, and Andrew Zhang, "Hyperledger Cookbook: Over 40 recipes implementing the latest Hyperledger blockchain frameworks and tools", Packt Publishing, 2019.
4. Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller, Steven Goldfeder, "Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction" , Princeton University Press , Kindle Edition, 2016.
5. Imran. Bashir. Mastering block chain: Distributed Ledger Technology, Decentralization, and Smart Contracts Explained. Packt Publishing, 2nd Edition, 2018.
6. Roger Wattenhofer, "The Science of the Blockchain, 1st Edition, Inverted Forest Publishing, 2016.

L: 45; TOTAL: 45 PERIODS

Course Code	SMART CONTRACT AND DAPP	L	T	P	E	C
23CS02E/ 23IT02E/ 23AD02E	(Common to CSE, IT and AI & DS degree Programmes)	2	0	2	0	3

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Components:

CO1: apply blockchain terminologies and concepts to effectively design and implement smart contract solutions. (CDL1)

CO2: design and develop efficient smart contracts for decentralized application through utilizing Ethereum's higher-level programming languages. (CDL2)

CO3:implement and deploy Solidity smart contracts, using key language constructs and leveraging development tools like IDEs and compilers.(CDL2)

CO4:write, deploy, and evaluate smart contracts, considering factors like cost, performance, and appropriate blockchain platform selection.(CDL2)

CO5:design, develop, and deploy decentralized applications (DApps) within a private blockchain network environment.(CDL2)

Practical Components:

CO6: develop, deploy, and interact with Ethereum smart contracts using Solidity (PDL2)

CO7: write, deploy, and manage smart contracts, and assess the cost and efficiency of running smart contracts on various blockchain platforms. (PDL2)

CO8: configure and deploy DApps in a private blockchain environment. (PDL2)

CO1: apply blockchain terminologies and concepts to effectively design and implement smart contract solutions. **L:6 P:3**

Introduction to Smart Contracts- Definitions – Features - Life Cycle - Smart Contract execution- Virtual Machine of a Block chain- Source code of a Smart Contract- Wallets – Types – Wallets interaction with Blockchains - Useful Terminologies

CO6: develop, deploy, and interact with Ethereum smart contracts using Solidity.

Write a simple smart contract - Deploy the contract on a testnet - Interact with the deployed smart contract using a wallet (e.g., MetaMask)

CO2: design and develop efficient smart contracts for decentralized application through utilizing Ethereum's higher-level programming languages. **L:6 P:6**

Ethereum Smart Contracts - Ethereum Higher-Level Languages -Developing Smart Contracts - The Solidity language: An object-oriented, high-level language for implementing smart contract - Designing a Solidity Smart Contract- Ether and Gas – Mining Ethereum.

CO6: develop, deploy, and interact with Ethereum smart contracts using Solidity.

Smart contract for transferring Ether between two accounts - Solidity contract that implements a simple auction system - Identify the gas feesinvolved in the contract execution.

CO3: implement and deploy Solidity smart contracts, using key language constructs and leveraging development tools like IDEs and compilers. **L:6 P:6**

Solidity – Contracts- Constructors – Functions- Variables-Getters – Setters- Arrays - Memory vs Storage - Mappings in Solidity – Structs - Error Handling and Restrictions – Libraries - Global Variables - Abstract Contracts – Inheritance- Interfaces-Events Installing Solidity - Solc Compiler - Ethereum Contract Application Binary Interface - Remix-IDE for Smart Contract Development.

CO6: develop, deploy, and interact with Ethereum smart contracts using Solidity.

Write a Solidity contract that implements constructor, arrays, mappings, structs, libraries and error handling

CO4: write, deploy, and evaluate smart contracts, considering factors like cost, performance, and appropriate blockchain platform selection. **L:6 P:6**

Smart Contract programming languages - Execution Environments - Environment Setup for Truffle and Ganache - Truffle Project Creation - Writing Smart Contracts - Developing Smart Contract - Programmer's responsibility - Cost of running smart Contracts - Smart Contract development platforms

CO7: write, deploy, and manage smart contracts, and assess the cost and efficiency of running smart contracts on various blockchain platforms.

Deploying the smart contract with Truffle and Ganache- Interacting with the deployed contract–Estimate the cost of running smart contracts

CO5:design, develop, and deploy decentralized applications (DApps) within a private blockchain network environment. **L:6 P:9**

DApp – Metamask - Setting up a Private Ethereum Network: Install go-etherum – Create geth Data directory – Create a geth account – create genesis.json configuration File – Run the first Node of the Private Network – Run the second node of the Network – Creating

Smart Contract – Deploying the Smart Contract – Client Application.

CO8: configure and deploy DApps in a private blockchain environment.

Setting up a private Ethereum network using geth - Running and connecting two nodes - Interacting with the network using Metamask - Creating a DApp to interact with the private Ethereum network

TEXT BOOKS:

1. Ritesh Modi, "Solidity Programming Essentials", 2nd Edition, Packt Publishing, 2022.
2. Gavin Zheng, Longxiang Gao, Liqun Huang, Jian Guan, "Ethereum Smart Contract Development in Solidity", Springer Verlag Publisher, 2020.
3. Andreas Antonopoulos and Gavin Wood, "Mastering Ethereum: Building Smart Contracts and Dapps", Shroff /O′Reilly Publisher, 2018
4. Michael J. Casey and Paul Vigna, "The Truth Machine: The Blockchain and the Future of Everything", Kindle Edition,St. Martin's Press, 2018.

REFERENCE BOOKS:

1. Kevin Solorio, Randall Kanna, David Hoover, "Hands-On Smart Contract Development with Solidity and Ethereum: From Fundamentals to Deployment", 1st Edition, Shroff / O'Reilly Publisher, 2019.
2. Wei-Meng Lee, "Beginning Ethereum Smart Contracts Programming", 2nd Edition, Apress, 2023

L: 30; P: 30; TOTAL: 60 PERIODS

Course Code	ADVANCED ETHEREUM	L	T	P	E	C
23CS03E/ 23IT03E/ 23AD03E	(Common to CSE, IT and AI & DS degree Programmes)	3	0	0	0	3

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Component

CO1: demonstrate understanding of Ethereum's upgrade history and architecture by explaining the Ethereum 2.0 roadmap, finality mechanisms, and Layer 2 scalability solutions. (CDL1)

CO2: analyze the execution layer of Ethereum's Proof of Stake by explaining the functioning of Engine API, DevP2P networking and Ethereum standards. (CDL2)

CO3: analyze the consensus and validation layer of Ethereum's Proof-of-Stake by analyzing the fork choice rules, validator roles, and the functions of consensus clients. (CDL2)

CO4: demonstrate understanding of smart contract security vulnerabilities by referencing real-world incidents and applying best practices. (CDL1)

CO1: Demonstrate understanding of Ethereum's upgrade history and architecture by explaining the Ethereum 2.0 roadmap, finality mechanisms, and Layer 2 scalability solutions. **L:12**

Ethereum Hard Forks and Upgrade History - Ethereum 2.0 Roadmap: Phases and Goals - Finalization: Checkpoints, Supermajority Link, Justification, Finality - Epochs and Slots: Concepts and Purpose - Maximal Extractable Value (MEV): Risks and Implications -

Types of Ethereum Nodes: Archive, Full Node, Snap Node - Beacon Chain Overview – Rollups

CO2: Analyze the execution layer of Ethereum's Proof of Stake by explaining the functioning of Engine API, DevP2P networking and Ethereum standards **L:11**

Ethereum Virtual Machine (EVM): Architecture and Functioning - Gossip Protocols - DevP2P Networking - Engine API: Execution - Consensus Layer Communication - Ethereum Transaction Lifecycle - Ethereum Standards: EIPs (Ethereum Improvement Proposals) - ERC (Ethereum Request for Comments) Standards

CO3: Analyze the consensus and validation layer of Ethereum's Proof-of-Stake by analyzing the fork choice rules, validator roles, and the functions of consensus clients. **L:12**

GASPER Fork Choice Rule: Overview and Importance - Casper FFG - LMD GHOST - Consensus Clients: Prysm, Lighthouse - LibP2P Networking - Validator Clients: Roles and Responsibilities – RANDAO Algorithm - Attestations: Purpose and Process - Slashing Mechanisms: Conditions and Implications

CO4: Demonstrate understanding of smart contract security vulnerabilities by referencing real-world incidents and applying best practices. **L:10**

Security Vulnerabilities in Smart Contracts : Reentrancy Attacks, Integer Overflow/Underflow, Front-running - Mainnet Incidents and Case Studies - Security Best Practices in Solidity - Bug Bounty Platforms and Responsible Disclosure

REFERENCE BOOKS

1. Ben Edgington, "Upgrading Ethereum A technical handbook on Ethereum's move to proof of stake and beyond", 3rd Edition, Capella Publisher, 2025.
2. Vitalik Buterin. "Proof of Stake: The Making of Ethereum and the Philosophy of Blockchains", 1st Edition, Seven Stories Press, 2022.
3. Andreas M. Antonopoulos, Gavin Wood, "Mastering Ethereum: Building Smart Contracts and DApps", 1st Edition, O'Reilly Media, 2018.
4. Bruno Skvorc, Mateja Kendel, David Attard, Mislav Javor, Tonino Jankov, Chris Ward, "A Developer's Guide to Ethereum", 1st Edition, SitePoint Publisher, 2018.

L: 45; TOTAL: 45 PERIODS

Course Code	BLOCKCHAIN SECURITY	L	T	P	E	C
23CS04E / 23IT04E / 23AD04E	(Common to CSE, IT and AI & DS degree Programmes)	**2**	**0**	**2**	**0**	**3**

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory component:

CO1: apply threat modeling, cryptographic hash functions, and governance frameworks to design and maintain secure blockchain systems. (CDL1)

CO2: analyze security challenges and attack vectors in consensus algorithms such as PoW and PoS to evaluate blockchain network robustness. (CDL1)

CO3: apply strategies to identify and mitigate various attacks to ensure secure and resilient blockchain systems. (CDL1)
CO4: examine common smart contract vulnerabilities, such as arithmetic errors, reentrancy, front-running, and access control issues, and evaluate methods to secure blockchain contracts. (CDL2)
Practical component:
CO5: assess and address blockchain consensus, mining, network, and identity-based vulnerabilities using threat modeling and simulations (PDL1)
CO6: analyze and mitigate smart contract vulnerabilities through security assessment and secure coding practices (PDL2)

CO1: Apply threat modeling, cryptographic hash functions, and governance frameworks to design and maintain secure blockchain systems. (CDL1) **L:6**

Fundamentals of blockchain security - Threat modeling for blockchain- Benefits of threat modeling - STRIDE threat modeling framework - OWASP modeling for blockchain security - Hardware security modules - Blockchain security hash key functions - Security policies and governance.

CO2: Analyze security challenges and attack vectors in consensus algorithms such as PoW and PoS to evaluate blockchain network robustness. (CDL1) **L:8; P: 10**

Consensus algorithm security – Attacking PoW consensus - The 51% attack - Attacking PoS consensus - Fake stake attacks - Long-Range attacks - Nothing at stake problem - Sour milk attack - Attacking block creation - Attacking blockchain nodes - Attacking the blockchain network - Maximum Extractable Value.

CO5: Assess and address blockchain consensus, mining, network, and identity-based vulnerabilities using threat modeling and simulations (PDL1).

Consensus Vulnerabilities - Mining Vulnerabilities - Network Vulnerabilities

CO3: Apply strategies to identify and mitigate various attacks to ensure secure and resilient blockchain systems. (CDL1) **L:8; P:10**

Distributed Denial-of-service attacks – Frontrunning - Simplified Payment Verification (SPV) mining - Delay Attacks - Eclipse Attacks - P2P and DHT Attacks - Partition Attacks - Sybil Attacks - Time Jacking - Routing attack - Transaction Attacks - Selfish Mining Attacks

CO5: Assess and address blockchain consensus, mining, network, and identity-based vulnerabilities using threat modeling and simulations (PDL1)

Identity-based attacks - DDoS Threat Modeling on a Blockchain Node - Selfish Mining Strategy Simulation

TEXT BOOK

1. Howard E. Poston, "Blockchain Security from the Bottom Up: Securing and Preventing Attacks on Cryptocurrencies, Decentralized Applications, NFTs, and Smart Contracts" 1st Edition, Wiley, 2022

REFERENCE BOOKS

1. Taha Sajid, "Ultimate Blockchain Security Handbook", 1st Edition, Orange Education Pvt. Ltd., 2023.

2. Yassine Maleh, Mohammad Shojafar, Mamoun Alazab, Imed Romdhani, "Blockchain for Cyber security and Privacy", 1st Edition, CRC Press, 2020.
3. Udai Pratap Rao, Piyush Kumar Shukla, Chandan Trivedi, "Blockchain for Information Security and Privacy", 1st Edition, Auerbach Publications, 2021.

L: 30; P: 30;TOTAL: 60 PERIODS

Course Code	PERMISSIONED BLOCKCHAIN	L	T	P	E	C
23CS05E/ 23IT05E/ 23AD05E	(Common to CSE, IT and AI & DS degree Programmes)	**2**	**0**	**2**	**0**	**3**

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Component

CO1: demonstrate an understanding of Hyperledger Fabric architecture and components, and execute transaction flows across single and multi-channel networks (CDL2)

CO2: execute Hyperledger Fabric Chaincode using Go programming constructs and apply it to real-world use cases (CDL2)

CO3: demonstrate a Hyperledger Fabric business network by defining various components and generating required certificates and artifacts (CDL2)

CO4: demonstrate the decentralized identity management concepts using Hyperledger Aries and Indy (CDL2)

Practical Component

CO5: develop a Hyperledger Fabric network, deploy Chaincode, and connect the DApp frontend to the Fabric network for transaction processing (PDL2)

CO6: develop a local Aries and Indy network and register a Decentralized Identifiers on the Aries and Indy network (PDL2)

CO1: **Demonstrate an understanding of Hyperledger Fabric architecture and components, and execute transaction flows across single and multi-channel networks** **L:8; P:6**

Introduction to Hyperledger - Hyperledger Fabric: Architecture – Peers – Ledger – Assets – Ordering Service - Channels - Kafka Consensus - Certificate Authority - Events - Certificate Authority - Membership Service Provider – Transaction Flow – Single Channel Network – Multi-Channel Network – Membership and Identity Management – Endorsement Policies – New User Registration and Enrollment

CO5: Develop a Hyperledger Fabric network, deploy Chaincode, and connect the DApp frontend to the Fabric network for transaction processing

Setup and build a Hyperledger Fabric network using Docker Compose, CLI and SDK.

CO2: Execute Hyperledger Fabric Chaincode using Go programming constructs and apply it to real-world use cases **L:8; P:9**

Go Programming Language: Introduction – Fundamentals – Control Statements - Functions & Methods - Structure - Arrays - Slices – Strings - Pointers - Concurrency - Chaincode: Introduction - Lifecycle - Chaincode vs Smart Contract - Chaincode API - System Chaincode - Collections - RICH queries – Chaincode usecases: Supply Chain

CO5: Develop a Hyperledger Fabric network, deploy Chaincode, and connect the DApp frontend to the Fabric network for transaction processing

Develop and deploy simple Chaincode - Implement access control policies using collections - Perform RICH queries on ledger data stored in CouchDB.

CO3: Demonstrate a Hyperledger Fabric business network by defining various components and generating required certificates and artifacts **L:8; P:9**

Designing a business network structure - Define organizations - Identifying participants - Channel name identification - Rule definition – Generate certificates using cryptogen and Fabric CA - Certificate expiry - Generate channel artifacts - Creating genesis block - Creating channel - Creating anchor peers - Setting up ordering node - Create peer nodes - Deploying Chaincodes on Fabric network – DApp development usecases: Supply chain, Enterprise workflows

CO5: Develop a Hyperledger Fabric network, deploy Chaincode, and connect the DApp frontend to the Fabric network for transaction processing

Build frontend of DApps using React - **C**onnect the frontend of DApp to the Hyperledger Fabric blockchain - Communication for data retrieval and transaction processing

CO4: Demonstrate the decentralized identity management concepts using Hyperledger Aries and Indy **L:6; P:6**

Decentralized Identity Management - Self-Sovereignty Identity - Hyperledger Aries: Introduction - Architecture - Hyperledger Indy: Introduction - Architecture – Decentralized Identifiers (DIDs) - Verifiable Credentials: Issuing and verifying credentials, ZERO-KNOWLEDGE PROOFS IN BLOCKCHAIN s - Creating and managing DIDs - Amazon managed blockchain service

CO6: Develop a local Aries and Indy network and register a Decentralized Identifiers on the Aries and Indy network

Build a local Aries and Indy network - Create and register a DID and Credentials on a local Aries and Indy network.

REFERENCE BOOKS

1. Nitin Gaur, Luc Desrosiers, Venkatraman Ramakrishna, Petr Novotny, Salman A. Baset, Anthony O'Dowd, "Blockchain with Hyperledger Fabric", 2nd Edition, Packt. Publishers, 2020.
2. Nakul Shah, "Blockchain for Business with Hyperledger Fabric: A complete guide to enterprise Blockchain implementation using Hyperledger Fabric", 1st Edition, BPB Publications, 2019
3. Bhagvan Kommadi, "Go Lang in Depth: Guide to program microservices, networking, database and APIs using Go Lang",1st Edition, BPB Publications, 2024

L: 30;P: 30;TOTAL: 60 PERIODS

Course Code	ZERO-KNOWLEDGE PROOFS IN BLOCKCHAIN	L	T	P	E	C
23CS01L/ 23IT01L/ 23AD01L	(Common to CSE, IT and AI & DS Degree Programmes)	**1**	**0**	**0**	**0**	**1**

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory components

CO1: utilize the principles and mechanisms of zero-knowledge proofs to evaluate cryptographic proof systems effectively. (CDL1)

CO2: apply zero-knowledge proof concepts to enhance privacy and improve scalability in blockchain applications. (CDL2)

CO1: **Utilize the principles and mechanisms of zero-knowledge proofs to evaluate cryptographic proof systems effectively. (CDL1)**

Cryptographic Proofs - Fundamentals of Zero-Knowledge Proofs - Two Faces of Zero-Knowledge (ZK) - ZK for Scaling - ZK for Privacy - Types of ZK Proofs - ZK-SNARKs (Zero-Knowledge Succinct Non-Interactive Arguments of Knowledge) - ZK-STARKs (Zero-Knowledge Scalable Transparent Arguments of Knowledge) - ZK-Rollups vs. Optimistic Rollups **L:**

CO2: Apply zero-knowledge proof concepts to enhance privacy and improve scalability in blockchain applications. (CDL2)

Benefits of ZK Proofs in Blockchain - ZK Proofs in Blockchain Privacy - ZK Proofs in Blockchain Scalability - ZK-Based Blockchain Projects - Use of ZK Proofs in Web3 - Implementation Tools for ZK Proofs - ZK Rollup Projects : KYC, Voting System - Case Study : Polygon Hermez (Polygon's Zero Knowledge Strategy) **L:**

REFERENCES

1. Amit Dua, Gaurav Kumar, "Mastering Zero-knowledge Proofs: Practical study of security, s and privacy in blockchain and modern systems", 1st Edition, BPB Publications, 2024
2. Mastering Blockchain, "Mastering Blockchain", 3rd Edition, Packt> Publishing Limited, 202
3. Vitalik Buterin, “An Incomplete Guide to Rollups”, 2021.

L:15; TOTAL: 15 PERIODS

Course Code	MINI - CAPSTONE PROJECT	L	T	P	E	C
23CS0ME/ 23IT0ME/ 23AD0ME	(Common to CSE, IT and AI & DS degree Programmes)	**0**	**0**	**0**	**6**	**3**

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Experiential Component

CO1: analyze real-world problems that can be addressed using blockchain technology, and formulate an appropriate project proposal. (PDL2)

CO2: design blockchain system architecture by selecting suitable platforms and defining smart contract logic aligned with project objectives. (PDL2)

CO3: develop and deploy smart contracts on blockchain test networks using industry-standard tools and frameworks. (PDL2)

CO4: integrate blockchain-based smart contracts with decentralized applications (DApps) using

Web3 technologies and demonstrate end-to-end functionality. (PDL2)

Softskill Component

CO5: demonstrate and defend the completed blockchain project through technical discussions, Comprehensive reporting, and viva-voce.

CO1: analyze real-world problems that can be addressed using blockchain technology, and formulate an appropriate project proposal.

Module 1: Introduction & Project Ideation — **E;15**

1. Overview of Blockchain project types:
   - Cryptocurrency
   - NFT platforms
   - Supply chain tracking
   - Healthcare records
   - Voting systems
2. Brainstorming real-world problems
3. Finalizing project title & scope
4. Submission of Project Proposal and Problem Statement

Deliverable: Guide approved project proposal document.

CO2: design blockchain system architecture by selecting suitable platforms and defining smart contract logic aligned with project objectives.

Module 2: Blockchain Design & Architecture — **E:20**

1. Public vs private blockchains
2. Choosing a blockchain platform:
   - Ethereum
   - Hyperledger Fabric
   - Corda
3. Smart contract design basics
4. Data models: on-chain vs off-chain

Deliverable: Project Architecture & Technology Stack document.

Module 3: Smart Contract Development — **E:20**

1. Writing smart contracts using Solidity or Chaincode
2. Deployment on testnets (e.g., Ropsten, Mumbai, Sepolia)
3. Security best practices (reentrancy, overflows, access control)
4. Tools:
   - Remix, Truffle, Hardhat, Ganache
   - Block explorers: Etherscan, Polygonscan

Deliverables:

- Smart contract code
- Deployed contract addresses

CO3: develop and deploy smart contracts on blockchain test networks using industry-standard tools and frameworks.

Module 4: DApp Frontend & Integration — **E:20**

- Building user interface (React, Next.js, or simple HTML/JS)
- Integrating with blockchain via Web3.js or Ethers.js
- Using wallets (Metamask)
- Backend API (if off-chain data involved)
- IPFS/Pinata for file storage (for NFTs or documents)

Deliverable: Working DApp connected to deployed smart contract.

Module 5: Final Demo & Report Submission

- Live demonstration (functional DApp)
- Project Report
- Code repository (GitHub)
- Optional: Publish smart contract on Etherscan

E:15

Deliverables:

- Live demo session
- Project report
- Code submission

E: 90; TOTAL: 90 PERIODS

Course code 23CS11E/ 23IT11E/ 23AD11E	**WEB FRAMEWORKS USING PYTHON**	**L**	**T**	**P**	**E**	**C**
	(Common to CSE, IT and AI & DS degree Programmes)	**2**	**0**	**2**	**0**	**3**

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Components

CO1: demonstrate the use of Django's core programming constructs to design and implement structured solutions. (CDL1)

CO2: apply MVC architecture and CRUD opeartions of Django to develop simple applications. (CDL2)

CO3: design and develop UI for dynamic web applications using Flask. (CDL2)

CO4: analyze and implement data management strategies in Flask applications. (CDL2)

Practical Components

CO5: develop applications using Django for real-world scenarios. (PDL2)

CO6: develop applications using Flask for real-world scenarios. (PDL2)

CO1: demonstrate the use of Django's core programming constructs to design and implement structured solutions.

Working with APIs: RESTful architecture - Request library. Developer Tools and SQL: Assert statements **L: 7;**

Web development using Django: Models – Basic data access – Forms; Getting data from the Request object - Custom Views- GET and POST methods -URL shortener -User model - Logic in templates – Static files **P: 6**

CO2: apply MVC architecture and CRUD opeartions of Django to develop simple applications.

CO5: develop applications using Django for real-world scenarios.

MVC and MTV architecture in Django – Core files of Django -Object-Relational Mapping - CRUD operations using Django's ORM - JSON Building- Filtering Models -Working with Images- Authentication with tokens- Postman- Content types app **L: 8;**

Develop applications using CRUD operations and JSON objects using Django. **P: 8**

CO3: design and develop UI for dynamic web applications using Flask **L: 8;**

CO6: develop applications using Flask for real-world scenarios. **P: 8**

Flask Fundamentals – Back-End Web Development using Flask – URL Routing – Rendering Templates – Static Files: Document Object Model – CSS and DOM – HTTP Objects: Request and Response Object – Web Forms – Redirects and User session – Cookies – Sessions.

Build simple web page using Flask Core programming constructs.

CO4: analyze and implement data management strategies in Flask applications.

CO6: develop applications using Flask for real-world scenarios.

Databases – Python Database frameworks – Object Relational Model – Flask-SQLAlchemy – CRUD operations – NoSQL Databases – Flask-PyMongo – Integrating Email with application – Web API with Flask – Deploying Flask App. **L: 7; P: 8**

Develop applications using CRUD operations and JSON objects using Flask.

TEXT BOOKS:

1. W. S. Vincent, *Django for Professionals*, 1st Edition. Still River Press, 2022.
2. M. Lathkar, *Building Web Apps with Python and Flask, 1st edition paperback, BPB Publications, 2021.

REFERENCES:

1. Nigel George, "Mastering Django: Core", 1st Edition Packt, 2016
2. Miguel Grinberg, "Flask Web Development", 2nd Edition, O'Reilly Media, Inc., 2018

L:30; P:30; TOTAL: 60 PERIODS

Course code		L	T	P	E	C
23CS12E/ 23IT12E/ 23AD12E	**NO SQL DATABASES** (Common to CSE, IT and AI & DS degree Programmes)	3	0	0	0	3

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Components

CO1: explore NoSQL databases, focusing on scalability, efficiency, and data challenges compared to relational databases. (CDL1)

CO2: apply replication, sharding, and distribution models to balance consistency, durability, and performance in databases. (CDL1)

CO3: apply document-oriented databases to address real-world challenges in e-commerce, content management, and real-time analytics. (CDL2)

CO4: develop solutions using column-oriented NoSQL databases, focusing on scalability, consistency, and real-world applications. (CDL2)

CO5: apply graph databases to develop scalable solutions for recommendation engines and location-based services. (CDL1)

CO1: explore NoSQL databases, focusing on scalability, efficiency, and data challenges compared to relational databases. **L: 10**

NoSQL Databases -Value of Relational Databases- Getting at Persistent Data – Concurrency – Integration - Impedance Mismatch- Application and Integration Databases

- Attack of the Clusters -Emergence of NoSQL- Aggregate Data Models: Consequences of Aggregate Orientation- Key-Value and Document Data Models, Column-Family Stores- Aggregate-Oriented Databases.

CO2: apply replication, sharding, and distribution models to balance consistency, durability, and performance in databases.

Replication and sharding- MapReduce on databases - Distribution Models - Single Server – Sharding- Master-Slave Replication- Peer-to-Peer Replication- Combining Sharding and Replication- Consistency - The CAP Theorem- Relaxing Durability- Quorums. **L: 9**

CO3: apply document-oriented databases to address real-world challenges in e-commerce, content management, and real-time analytics.

NoSQL Key/Value databases using MongoDB - Document Databases - Document oriented Database Features: Consistency, Transactions, Availability, Query Features – Scaling - Suitable Use Cases: Event Logging, Content Management Systems, Blogging Platforms, Real-Time Analytics, E-Commerce Applications, Complex Transactions Spanning Different Operations, Queries against Varying Aggregate Structure **L: 9**

CO4: develop solutions using column-oriented NoSQL databases, focusing on scalability, consistency, and real-world applications.

Column- oriented NoSQL databases using Apache HBASE, Column-oriented NoSQL databases using Apache Cassandra, Architecture of HBASE, Column-Family Data Store Features, Consistency, Transactions, Availability, Query Features, Scaling, Suitable Use Cases, Event Logging, Content Management Systems, Blogging Platforms, Counters, Expiring Usage. **L: 9**

CO5: apply graph databases to develop scalable solutions for recommendation engines and location-based services.

Graph Databases: Features, Consistency, Transactions, Availability, Query Features, Scaling, Suitable Use Cases, Connected Data, Routing, Dispatch, and Location-Based Services, Recommendation Engines. **L: 8**

TEXT BOOKS:

1. Sadalage, P. & Fowler, "NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence", Wiley Publications, 1st Edition, 2019.
2. Adam Fowler, "NoSQL for Dummies", Wiley Publications, 2nd Edition, 2020.

REFERENCE BOOKS:

1. Dan Sullivan, "NoSQL For Mere Mortals", 2nd Edition, Addison-Wesley, 2021.
2. Dan McCreary and Ann Kelly, "Making Sense of NoSQL: A Guide for Managers and the Rest of Us", 2nd Edition, Manning Publications, 2020.
3. Kristina Chodorow, "MongoDB: The Definitive Guide - Powerful and Scalable Data Storage", 4th Edition, O'Reilly Media, 2020.

L: 45; TOTAL: 45 PERIODS

Course code 23CS13E/ 23IT13E/	**OPEN SOURCE TECHNOLOGIES FOR WEB DEVELOPMENT** (Common to CSE, IT and AI & DS degree Programmes)	**L**	**T**	**P**	**E**	**C**
		3	**0**	**0**	**0**	**3**

23AD13E

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Components

CO1: analyze the principles, benefits, and global impact of open source software. (CDL1)

CO2: apply version control principles using Git and Demonstrate repository management and collaboration (CDL2)

CO3: develop dynamic web applications that provide various functionalities (CDL2)

CO4: apply MySQL database management and PHP integration to create dynamic applications (CDL3)

CO5: analyze different open source ecosystems and adopt the most suitable solution for specific scenario (CDL2)

CO1: analyze the principles, benefits, and global impact of open source software.

Need of Open Sources – Advantages of Open Sources – Open Source Applications - FOSS usage - Free Software Movement - Commercial aspects of Open Source movement - Open Source Software Development Model - Comparison with close source / Proprietary software - open source software licenses - Copy right - Copy left – Patent - Zero Marginal Technologies - Income generation opportunities – Internalization. **L: 8**

CO2: apply version control principles using Git and Demonstrate repository management and collaboration

Git and version control - Getting oriented with Git - Making and using a Git repository – Command line interface - Using Git with a GUI - Tracking and updating files in Git - Committing parts of changes – Cloning - Collaborating with remotes - Pushing changes - Working with GitHub **L: 8**

CO3: develop dynamic web applications that provide various functionalities of PHP

Open-Source Programming: PHP - Introduction - LAMP/WAMP installation - PHP Script Elements: operators - Statements - Loops; Working with Arrays – Functions - Global variables - Exception Handling - Session Handling – Form Processing and Validation. **L: 10**

CO4: apply MySQL database management and PHP integration to create dynamic applications.

Open-Source Database: MySQL - Introduction - Features – Account Setup and Commands – String Handling - Date and Time – Metadata. PHP and MySQL: database connection through PHP - Data Objects (PDO) - Handling Errors in MySQL - Transaction management. **L: 10**

CO5: analyze different open-source ecosystems and adopt the most suitable solution for specific scenario

Open-Source Ecosystem: Open-Source Operating Systems: GNU/Linux – Android – Free BSD – Open Solaris. Open-Source Hardware: Virtualization Technologies. Containerization Technologies: Docker – Development tools: IDEs – Debuggers. **L: 9**

TEXT BOOKS:

1. Chris DiBona, Sam Ockman, and Mark Stone, "Open Sources 2.0: The Continuing Evolution", 1st Edition, O'Reilly Media, May 2021.
2. Scott Chacon & Ben Straub, "Pro Git", 2nd Edition, Apress, March 2020.
3. Tom Butler, "PHP & MySQL: Novice to Ninja", 6th Edition, SitePoint, July 2020.
4. Todd McLeod, "Docker: Up & Running", 2nd Edition, O'Reilly Media, January 2022.

REFERENCES:

1. Joseph Feller and Brian Fitzgerald, "Understanding Open Source Software: A Guide to Open Source Development, Business, and Law", 1st Edition, Addison-Wesley, January 2021.
2. Mark L. Shapiro, "Open Source Licensing: Software Freedom and Intellectual Property Law", 2nd Edition, Prentice Hall, June 2021
3. Chris Shiflett, "PHP Solutions: Dynamic Web Design Made Easy", 2nd Edition, Friends of ED, June 2020.
4. Ben S. Collins-Sussman, Brian W. Fitzpatrick, & Jason J. Merkey, "Version Control with Git", 1st Edition, O'Reilly Media, October 2020.
5. David Sklar, "PHP & MySQL: The Missing Manual", 3rd Edition, O'Reilly Media, September 2022.Ethan Brown, "Docker Deep Dive", 3rd Edition, Leanpub, February 2021.

L: 45; TOTAL: 45 PERIODS

Course Code	POSTGRESQL DATABASE MANAGEMENT	L	T	P	E	C
23CS14E/ 23IT14E/ 23AD14E	(Common to CSE, IT, AI&DS Degree Programmes)	**2**	**0**	**2**	**0**	**3**

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Component

CO1: utilize PostgreSQL for database design, data manipulation, and management using basic data types, JSON/JSONB, and GUI tools. (CDL2)

CO2: apply advanced querying, normalization, indexing, and transaction techniques for database optimization. (CDL2)

CO3: build PostgreSQL-based applications using stored procedures, functions, and cloud integration with Java. (CDL3)

Practical Component

CO4: implement data manipulation, advanced querying, and optimization using postgreSQL. (PDL2)

CO5: design PostgreSQL solutions with data aggregation, automation, cloud deployment, and Java integration for efficient database management.(PDL2)

CO1: Utilize PostgreSQL for database design, data manipulation, and management using basic data types, JSON/JSONB, and GUI tools.

Introduction to PostgreSQL - Database Models and Query Languages - PostgreSQL Architecture - Data Access and Manipulation with PostgreSQL - Storing and Accessing Data in a Database - Working with Multiple Tables and Join Operations - Designing Tables and Working with Basic Data Types - GUI Tools (pgAdmin, DBeaver) - Data Types and Creating User Records - JSON and JSONB Data Types

CO4: Implement data manipulation, advanced querying, and optimization using postgreSQL.

L: 10; P: 10

- Create Databases and Tables using PostgreSQL
- Perform Basic Data Manipulation operations.
- Perform Data Retrieval and Querying: using WHERE, ORDER BY, and GROUP BY also implement filtering with JSONB fields.
- Connect to the database using pgAdmin/DBeaver and Perform CRUD operations through the graphical interface.

CO2: Apply advanced querying, normalization, indexing, and transaction techniques for database optimization.

Advanced Data Selection and Aggregate Functions - Complex Query Techniques: Subquery, Union Join, Self Join, Outer Join - Views- Normalization -Transaction and Locking Mechanisms: ACID Rules, Logs, and Multi-User Transactions - Locking Techniques: Avoiding Deadlocks - Indexing Techniques for Performance Optimization: B-Tree, Hash, GIN, and BRIN Indexes

CO4: Implement data manipulation, advanced querying, and optimization using postgreSQL.

L: 10; P: 10

- Implement advanced query techniques using SUBQUERY, UNION, SELF JOIN, and OUTER JOIN.
- Implement aggregate functions like SUM, AVG, MIN, and MAX.
- Normalize a database to the 3NF level and Implement FOREIGN KEY constraints to ensure data integrity.
- Implement transaction handling using BEGIN, COMMIT, and ROLLBACK and resolve deadlocks using explicit locking mechanisms.
- Implement indexing techniques using (B-Tree/ GIN/ BRIN / Hash).

CO3: Build PostgreSQL-based applications using stored procedures, functions, and cloud integration with Java.

Operators and Built-in Functions - Procedural Languages in PostgreSQL (PL/pgSQL) - Creating and Using Stored Procedures - SQL Functions and Triggers - Accessing PostgreSQL from Java using JDBC - Database Connections, JDBC Result Sets, and Creating JDBC Statements - Cloud Deployment and Hosting: Heroku and AWS RDS

CO5: Design PostgreSQL solutions with data aggregation, automation, cloud deployment, and Java integration for efficient database management.

L: 10; P: 10

- Create stored procedures for data aggregation and develop SQL functions for data transformation.
- Establish database connectivity using the PostgreSQL JDBC driver and perform CRUD operations from a Java application.
- Implement triggers to log data changes and automate data update tasks using

triggers.

- Deploy PostgreSQL on Heroku or AWS RDS and connect a Java application to the cloud database and perform operations.

TEXT BOOKS

1. H.-J. Schönig, Mastering PostgreSQL 15: Advanced Techniques to Build and Manage Scalable, Reliable, and Fault-Tolerant Database Applications, 5th Edition, Birmingham, UK: Packt Publishing, 2022
2. R. Obe and L. Hsu, *PostgreSQL: Up and Running*, 3rd Edition, Sebastopol, CA, USA: O'Reilly Media, 2020.

REFERENCE BOOKS

1. Vadlamani, Venkateswara. *PostgreSQL Skills Development on Cloud: A Practical Guide to Database Management with AWS and Azure*, 1st Edition, Apress, 2024.
2. L. Ferrari and E. Pirozzi, Learn PostgreSQL: Use, Manage, and Build Secure and Scalable Databases with PostgreSQL 16, 2nd Edition. Birmingham, UK: Packt Publishing, 2023.

L: 30; P: 30; TOTAL: 60 PERIODS

Course Code	MOBILE APPLICATION DEVELOPMENT	L	T	P	E	C
23CS15E/ 23IT15E/ 23AD15E	(Common to CSE, IT, AI&DS Degree Programmes)	**2**	**0**	**2**	**0**	**3**

COURSE OUTCOMES

Upon completion of this course, the student will be able to

Theory Component

CO1: examine mobile application architecture and android development tools for building customized applications (CDL1)

CO2: design intuitive and responsive user interfaces using material design principles (CDL2)

CO3: integrate mobile apps with REST APIs and various backend data services for dynamic content management (CDL2)

CO4: utilize device features like camera, location, notifications, and implement asynchronous tasks (CDL2)

Practical Component

CO5: design and develop functional Android user interfaces and layouts using Android development tools (PDL2)

CO6: build and test robust cross-platform mobile apps using React Native and native Android APIs (PDL2)

CO1: Explore mobile application architecture and Android development tools for building customized applications **L:6; P:6**

Mobile App Development Overview – Native Cross-platform - Android Architecture, Lifecycle, and Tools - Introduction to Android Studio and Kotlin Basics - Kotlin: Variables, Control Flow, OOP Concepts- Kotlin (KMP) - Android Project Structure,

Gradle, and Build Process – APK Versioning - Design and implement a single-screen Android app using Kotlin and Android UI components.

CO5: Design and develop functional Android user interfaces and layouts using Android development tools

Use Kotlin control flow statements (if-else, when, loops) to create interactive app logic responding to user actions.

CO2: Design intuitive and responsive user interfaces using Material Design principles **L:8; P:7**

UI/UX Principles and Material Design Standards - Jetpack Compose: Composable, Modifiers, State - Layouts: Column, Row, Scaffold, LazyColumn-Forms: TextField, Button, CheckBox, RadioButton, Dropdown- Input Components, Navigation, and Theming - Error Handling -Event Handling and Recomposition- Material Design Colors, Typography, Shapes.

CO5: Design and develop functional Android user interfaces and layouts using Android development tools

Create a dynamic UI for a product catalog screen using Compose, with filtering options and responsive layouts.

CO3: Integrate mobile apps with REST APIs and various backend data services for dynamic content management **L:8; P:8**

Local Storage: SharedPreferences, Room DB (with DAO & ViewModel)- JSON Parsing, API Consumption using Retrofit,Coroutines- Integration with backend services: Flask (Python), Express (Node.js)- Introduction to SQLite access via APIs - Build an app that fetches and stores data via REST API.

CO6: Build and test robust cross-platform mobile apps using React Native and native Android APIs

Asynchronous data handling with coroutines – Node.js Express as backend and stores data in SQLite

CO4: Utilize device features like camera, location, notifications, and implement asynchronous tasks **L:8; P:9**

Accessing Device Features: Camera, Location (Fused API), Sensors - Permissions (runtime), Notifications, File I/O- Multithreading with Coroutines - Debugging & Performance Optimization Tools- Unit Testing with JUnit, UI Testing with Espresso - App Signing, Versioning, ProGuard, Obfuscation - Publishing to Google Play Store.

CO6: Build and test robust cross-platform mobile apps using React Native and native Android APIs

API with push notifications – real world mobile applications development – test application.

TEXT BOOKS

1. Neil Smyth, "Android Studio Development Essentials – Kotlin", 1st Edition, Techotopia, 2023.
2. Mark L. Murphy, "The Busy Coder's Guide to Android Development", 3rd Edition, CommonsWare, 2019.

REFERENCES

1. Valentino Lee, Heather Schneider, Robbie Schell, "Mobile Application Development:

Android and iOS", 2nd Edition, McGraw-Hill Education, 2023.
2. Reto Meier & Ian Lake, "Professional Android", Wiley, 4th Edition, 2022.
3. Bonnie Eisenman, "Learning React Native", 2nd Edition, O'Reilly Media, 2020.
4. Bill Phillips, Chris Stewart, Kristin Marsicano, "Android Programming: The Big Nerd Ranch Guide", 4th Edition, Pearson, 2019.

L: 30; P: 30; TOTAL: 60 PERIODS

Course Code	WEB APPLICATION TESTING	L	T	P	E	C
23CS17E/ 23IT17E/ 23AD17E	(Common to CSE, IT, AI&DS Degree Programmes)	**2**	**0**	**2**	**0**	**3**

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Components:

CO1: demonstrate test cases for fullstack applications using Jira and Zephyr Scale (CDL1)
CO2: apply functional and non-functional testing techniques across frontend and backend layers (CDL1)
CO3 analyze API, database and performance testing using Postman, SQL and JMeter (CDL2)
CO4: employ automated full-stack testing pipelines using Selenium Playwright and Cucumber. (CDL2)

Practical Components:

CO5: perform manual and API testing for full-stack applications using Jira, Zephyr, Postman, SQL, and JMeter. (PDL1)
CO6: implement automated end-to-end testing using Selenium, Playwright, Cypress, and Cucumber with CI/CD pipelines. (PDL2)

CO1: Demonstrate test cases for fullstack applications using Jira and Zephyr Scale (CDL1) **L:7; P:6**

Software Testing Life Cycle – Functional Testing – Non Functional Testing – Test Plan and Strategy – Test Case Design and Management and version control using Jira and Zephyr Scale — Defect Logging and Reporting.

CO5: Perform manual and API testing for full-stack applications using Jira, Zephyr, Postman, SQL, and JMeter. (PDL2)

Test Case Creation in Zephyr Scale - Defect Logging in Jira - End-to-End Manual Testing Project

CO2: Apply functional and non-functional testing techniques across frontend and backend layers (CDL1) **L:7; P:8**

Functional and UI testing – Responsive and cross-browser testing – Component testing with Jest and React Testing Library –Authentication and response validation – Usability and Accessibility Testing - Compatibility Testing - Error Handling and Validation Testing- Security testing: OWASP basics, Auth, Vulnerability scanning.

CO5: Perform manual and API testing for full-stack applications using Jira, Zephyr, Postman, SQL, and JMeter (PDL2)

Visual consistency testing – validate layout responsiveness – unit testing of react components – validate backend messages

CO3:Analyze API, database, and performance testing using Postman, SQL and JMeter (CDL2) **L:6; P:6**

API testing using Postman - Database Testing – Validating CRUD Operations – Writing SQL Queries for Data Verification – Data Migration and Schema Testing – Backend–Frontend Integration Testing – Performance testing - Apache JMeter – Test Scripts in JMeter – Performance Analysis and reporting.

CO5: P**erform manual and API testing for full-stack applications using Jira, Zephyr, Postman, SQL, and JMeter (PDL2)**

API request and response validation - Data Integrity and Constraint Testing in Database - Backend–Frontend Integration Testing with Live Data Mapping - Load and Stress Testing using JMeter - Performance Metrics Analysis using JMeter Listeners

CO4: Employ automated full-stack testing pipelines using Selenium Playwright and Cucumber (CDL2) **L:10; P:10**

Automation Framework Design - Hybrid, Keyword, Data-Driven, BDD with Cucumber – Setting up End-to-End (E2E) Tests – Test Execution Management – Continuous Testing with Selenium, Cypress, Playwright and cucumber – Playwright: Test scripts - locators selectors - Shadow DOM – POM – Cucumber:Gherin Language – Hooks – Test Reporting using Allure or Extent Reports – Maintaining Test Suites for Full Stack Applications.

CO6: Implement automated end-to-end testing using Selenium, Playwright, Cypress, and Cucumber with CI/CD pipelines. (PDL2)

Design Page Object Model (POM) for Full-Stack Applications - Test Case Execution Management using Test Suites and Test Runners - Playwright Test Scripts for UI Automation - Parallel and Cross-Browser Testing using Playwright - Generate Automation Test Reports using Allure Reports - Continuous Automation Execution using Git and Jenkins/GitHub Actions

TEXT BOOKS

1. Rahul Shetty, Full Stack Automation Testing Architect, Leanpub, 2023.
2. Gavin Lon and Andrew MacKenzie, “Playwright for Web Automation”, Packt Publishing, Birmingham, 2022.

REFERENCES

1. Rex Black and Dorothy Graham, “Foundations of Software Testing”, Cengage Learning, New Delhi, 2020.
2. Mark Winteringham, “Testing Web APIs”, Manning Publications, New York, 2020.
3. Erik Fentz, “API Testing and Development with Postman”, Packt Publishing, Birmingham, 2020.
4. Avinash Kumar and Srinivasan Desikan, “Mastering JMeter for Performance Testing”, Packt Publishing, Birmingham, 2019.

L: 30; P: 30; TOTAL: 60 PERIODS

Course Code	FULLSTACK APP WITH FLUTTER AND FIREBASE	L	T	P	E	C
23CS18E/ 23IT18E/ 23AD18E	(Common to CSE, IT, AI&DS Degree Programmes)	2	0	2	0	3

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Components:

CO1: demonstrate the foundational concepts of Flutter to build user interface. (CDL1)

CO2: apply Dart and UI functionalities to build mobile applications. (CDL1)

CO3: implement responsive applications with navigation, animation and state management. (CDL2)

CO4: apply Firebase framework to build real time applications. (CDL2)

Practical Components:

CO5: design UI layouts with widgets, validation, and multi-screen navigation for interactive applications. (PDL1)

CO6: develop and deploy real-time mobile applications (PDL2)

CO1: Demonstrate the foundational concepts of Flutter to build user interface. (CDL1) **L:7; P:6**

Introduction to Flutter – Widgets and Elements – Widget Lifecycle Events – Stateful and Stateless Widget Tree and Element Tree. Building a flutter project – Hot Reload – Themes – Global and parts of App – Stateful and Stateless Widgets using External Packages - Widget Rebuild Basics – Lazy Loading.

CO5: Design UI layouts with widgets, validation, and multi-screen navigation for interactive applications. (PDL1)

Creating starter projects – Designing UI screens and layouts – Applying themes and styling – Implementing multi-screen navigation using built-in widgets

CO2: Apply Dart and UI functionalities to build mobile applications. (CDL1) **L:8; P:8**

Basics of Dart – Main function – Variables – Data types – Operators – Flow Statements – Functions – Packages. Classes – Inheritance – Mixins. Asynchronous programming. Widgets – Images and Icons – Decorators – Orientation-Dart Performance Optimization

CO5: Design UI layouts with widgets, validation, and multi-screen navigation for interactive applications. (PDL1)

Creating structured widgets – Creating Form Widget to validate Text Fields – Develop a To-do list – widget that add and delete job lists

CO3: Implement responsive applications with navigation, animation and state management. (CDL2) **L:9; P:8**

Animation Container – Navigators –Scrolling Lists and Effects – Building Layouts – Interactivity. State management using Riverpod and API Integration.

CO6: Develop and deploy real-time mobile applications. (PDL2)

Create a Quiz/Gaming widget with animations – Create an interactive Cafeteria application that includes complex layouts with header, main section and footer section.

CO4: Apply Firebase framework to build real time applications.(CDL2) **L:6;**
Firebase and Cloud Firestore – Structure and Data modelling – Authentication capabilities – Push notifications using Firebase Cloud Messaging and Advanced FCM –Security Rules. Building Firebase Project with CRUD operations. Case Study: Building Journal App in Firebase **P:8**

CO6: Develop and deploy real-time mobile applications (PDL2)
Develop a real time application with Firebase cloud services. Mobile CI/CD – Fastlane Automation, GitHub Actions Pipelines, and App Store Submission.

TEXT BOOKS

1. Marco L.Napoli, "Beginning Flutter – A Hands-on Guide to App Development", 1st Edition, John Wiley & Sons, 2020.

REFERENCES

1. Priyanka Tyagi, "Pragmatic Flutter - Building Cross-Platform Mobile Apps for Android, iOS, Web & Desktop", 1st Edition, CRC Press, 2024.
2. Alberto Miola, "Flutter Complete Reference 2.0", 2nd Edition, Independently Published, 2020.

L:30; P: 30; TOTAL: 60 PERIODS

Course Code	MINI – CAPSTONE PROJECT	L	T	P	E	C
23CS1ME/ 23IT1ME/ 23AD1ME	(Common to CSE, IT and AI & DS degree Programmes)	0	0	0	6	3

COURSE OUTCOMES
Upon the successful completion of the course, the student will be able to

Experiential Component

CO1: apply client-side (HTML, CSS, JavaScript) and server-side (Node.js, Express) technologies to design and implement responsive, scalable full-stack web applications. (PDL2)
CO2: design and integrate RESTful APIs with relational and NoSQL databases (e.g., MySQL, MongoDB) to enable efficient data exchange and persistence in full-stack systems. (PDL2)
CO3: develop and deploy secure, performance-optimized full-stack applications using CI/CD pipelines, containerization (Docker), and cloud platforms (Heroku, AWS). (PDL2)
CO4: create and evaluate real-world full-stack applications that incorporate user authentication, role-based access control, and real-time data updates using technologies like WebSockets. (PDL3)

Soft Skill Component

CO5: demonstrate and present a full stack web application through effective team collaboration, technical and oral communication.

Module 1: Problem identification, technology selection and project ideation:

1. Overview of full stack architecture **E:15**
   - Role of frontend, backend, and databases
   - RESTful services and APIs
   - Tools and technologies stack
2. Project Ideation
   - Introduction to user-centered design principles

- Wireframing and prototyping tools
- UI/UX Design

Deliverable: A design prototype ready for development.

Module 2: Frontend Development **E: 20**

For Web Applications (MERN Stack / Python Web Frameworks):

- HTML5, CSS3, JavaScript
- Frontend frameworks: React.js or Django Templates
- State management: Context API / Redux
- Routing, component-based UI, API integration

For Mobile Applications (React Native):

- Layout using Flexbox
- Navigation and routing
- Form inputs, lists, custom widgets

Deliverable: Fully developed frontend with interactive UI and data binding.

Module 3: **E: 15**

Backend Development & API Integration

- REST API fundamentals and Frameworks
- Routing, middleware, and controllers
- CRUD operations, API error handling
- Authentication with JWT / OAuth or Firebase Auth
- Database Management
    - MongoDB (NoSQL) for MERN, PostgreSQL/MySQL for Django.
    - Database normalization, indexing and optimization

Deliverable: RESTful APIs to support frontend interactions.

Module 4: **E: 20**

Authentication & Security

- Firebase Authentication or manual setup using Auth modules
- Role-based access control
- Input validation and secure API design
- Data encryption and session/token management

Testing & Debugging

- Unit testing (frontend and backend)
    - React Testing Library / Jest / React Native Testing Library
    - Backend: PyTest, Jest, Mocha
- Integration testing using Postman or Swagger
- Debugging tools and console tracing

Deliverable: Secure login, registration and protected routes, Tested and verified components ensuring reliability.

Module 5: **E:20**

Deployment & Version Control

- Version control with Git and GitHub
- CI/CD pipeline basics : GitHub Actions / Firebase CI
- Deployment platforms:

- Web: Vercel, Netlify, Render, Firebase Hosting
- Mobile: APK build, Play Store Publishing

- Hosting backend servers and databases

Deliverable: Deployed application accessible via web or mobile platform.

E: 90; TOTAL: 90 PERIODS

Course code	COMPUTER VISION AND IMAGE PROCESSING	L	T	P	E	C
23CS21E/ 23IT21E	(Common to CSE and IT degree Programmes)	**3**	**0**	**0**	**0**	**3**

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Components

CO1: apply principles of image formation, Geometric transformations, and color image processing techniques to augment the visual of the image. (CDL1)

CO2: apply image transforms, filtering, and morphological operations for image enhancement. (CDL1)

CO3: implement feature extraction, edge detection, and shape representation methods for image processing. (CDL1)

CO4: analyze classification, clustering, and neural network methods for pattern recognition. (CDL2)

CO5: analyze and apply segmentation, and tracking techniques in image processing. CDL2)

CO1: Apply principles of image formation, Geometric transformations, and color image processing techniques to enhance the visual of the image.

Introduction to Computer Vision - Image Formation and Radiometry-Geometric Transformation-2D transformations-3D transformations-Geometric Camera Models-Image formation in a stereo vision setup-Image Reconstruction from a Series of Projections-Inverse Radon transform - back-projection -Fourier transform – Color models -Color constancy- Color Image Processing **L: 9**

CO2: Apply image transforms, filtering, and morphological operations for image enhancement.

Fundamentals of Image Processing-Image Transforms-Discrete Fourier transform - Discrete cosine transform-K-L transform -Wavelet transform-Image Filtering-Spatial domain filtering-Frequency domain filtering- Homomorphic filtering-Wiener filter for image restoration- -Mathematical Morphology **L: 9**

CO3: Implement feature extraction, edge detection, and shape representation methods for image processing.

Image Descriptors and Features-Texture Descriptors- Color Features-Edge Detection-Object Boundary and Shape Representations-SUSAN edge and corner point detector-Moravec corner detector-Histogram of Oriented Gradients-Scale Invariant Feature Transform-Speeded up Robust Features. **L: 9**

CO4: Analyze classification, clustering, and neural network methods for pattern recognition **L: 9**

Introduction to Pattern Recognition-Linear Regression-Basic Concepts of Decision

Functions-Elementary Statistical Decision Theory-Gaussian Classifier-Parameter Estimation-Clustering for Knowledge Representation-Dimension Reduction-Template Matching-Artificial Neural Network for Pattern Classification- Convolutional Neural Networks-Autoencoder.

CO5: analyze and apply segmentation, and tracking techniques in image processing.

Clustering for image segmentation-Graph partitioning methods-Image segmentation by neural networks-Probabilistic models for image segmentation-Motion Estimation and Object Tracking-Face and Facial Expression Recognition-Gesture Recognition-Image Fusion. **L: 9**

TEXT BOOKS:

1. Manas Kamal Bhuyan, "Computer Vision and Image Processing Fundamentals and Applications", CRC Press Taylor & Francis Group, Paperback, 2020.
2. Richard Szeliski, "Computer Vision: Algorithms and Applications", Springer- Texts in Computer Science, 2nd Edition, 2022.
3. Computer Vision: A Modern Approach, D. A. Forsyth, J. Ponce, Pearson Education, 2nd Edition, 2015.

REFERENCE BOOKS:

1. Yu-Jin Zhang, "3-D computer vision principles and algorithms", Springer Singapore,1st Edition, 2023
2. Sonka, Hlavac, and Boyle. Thomson, "Image Processing, Analysis, and Machine Vision", Cengage Learning, 4th Edition, 2017.
3. D. Forsyth and J. Ponce, "Computer Vision - A modern approach", Pearson Education, Inc., 2nd Edition, 2015.
4. B. K. P. Horn, "Prentice Hall Robot Vision", McGraw-Hill, 1st Edition, 2015.

L: 45; TOTAL: 45 PERIODS

Course code	**MACHINE LEARNING TECHNIQUES**	**L**	**T**	**P**	**E**	**C**
23CS22E/ 23IT22E	(Common to CSE and IT degree Programmes)	**2**	**0**	**2**	**0**	**3**

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Components

CO1: understand the foundations and mathematics of machine learning. (CDL1)

CO2: apply supervised learning algorithms to solve classification and regression problems. (CDL1)

CO3: apply the core concepts of deep neural networks and optimization techniques to enhance model performance. (CDL1)

CO4: apply unsupervised learning and dimensionality reduction to analyze and interpret data. (CDL1)

Practical Components

CO5: implement and evaluate supervised learning models to solve classification and regression problems. (PDL1)

CO6: design, implement, and optimize deep learning models for solving real-world problems. (PDL1)

CO7: implement and evaluate unsupervised models to enhance model performance and accuracy. (PDL1)

CO1: Understand the foundations and mathematics of machine learning.

Review of Linear Algebra for machine learning; Introduction and motivation for machine learning; Examples of machine learning applications, Vapnik-Chervonenkis (VC) dimension, Probably Approximately Correct (PAC) learning, Hypothesis spaces, Inductive bias, Generalization, Bias variance trade-off. **L: 8**

CO2: Apply supervised learning algorithms to solve classification and regression problems.

CO5: Implement and evaluate supervised learning models to solve classification and regression problems.

Linear Regression Models: Least squares, single & multiple variables, Bayesian linear regression, gradient descent, Linear Classification Models: Discriminant function – Perceptron algorithm, Probabilistic discriminative model - Logistic regression, Probabilistic generative model – Naive Bayes, Maximum margin classifier – Support vector machine, Decision Tree, Random Forests **L: 11; P: 9**

Implement linear regression to predict house prices based on a dataset (e.g., Boston Housing).

Implement an SVM classifier on a dataset (e.g., Iris or MNIST) and evaluate the model's accuracy and performance

Apply K-fold cross-validation and bootstrapping to evaluate Logistic Regression and SVM on a dataset.

CO3: Apply the core concepts of deep neural networks and optimization techniques to enhance model performance.

CO6: D**esign, implement, and optimize deep learning models for solving real-world problems.**

Multilayer perceptron, activation functions, network training – gradient descent optimization – stochastic gradient descent, error backpropagation, from shallow networks to deep networks –Unit saturation – ReLU, hyperparameter tuning, batch normalization, regularization, dropout. **L: 7; P: 12**

Implement an MLP for classification on a dataset (e.g., MNIST) and experiment with various activation functions (sigmoid, tanh, ReLU).

Train a deep network using gradient descent and stochastic gradient descent (SGD). Tune hyperparameters such as learning rate, batch size, and number of epochs to optimize the model's performance on a dataset.

Apply advanced techniques like dropout, batch normalization and regularization to a deep neural network trained on a dataset to prevent overfitting and improve generalization.

CO4: Apply unsupervised learning and dimensionality reduction to analyze and interpret data.

CO7: Implement and evaluate unsupervised models to enhance model performance and accuracy. **L: 4; P: 9**

Unsupervised learning: K-means, Instance Based Learning: KNN, Gaussian mixture

models and Expectation maximization, Principal Component Analysis
Implement K-means clustering to group data points into clusters and apply PCA for dimensionality reduction.

Implement K-Nearest Neighbors (KNN) for classification on a dataset (e.g., Wine Quality). Compare its performance with Gaussian Mixture Models (GMM) for the same dataset.

TEXT BOOKS

1. Ethem Alpaydin, "Introduction to Machine Learning", MIT Press, 4th Edition, 2020.
2. Stephen Marsland, "Machine Learning: An Algorithmic Perspective", 2nd Edition, CRC Press, 2014.

REFERENCE BOOKS

1. Christopher M. Bishop, "Pattern Recognition and Machine Learning", Springer, 2006.
2. Tom Mitchell, "Machine Learning", McGraw Hill, 3rd Edition, 1997.
3. Mehryar Mohri, Afshin Rostamizadeh, Ameet Talwalkar, "Foundations of Machine Learning", 2nd Edition, MIT Press, 2012, 2018.
4. Ian Goodfellow, Yoshua Bengio, Aaron Courville, "Deep Learning", MIT Press, 2016
5. Sebastain Raschka, Vahid Mirjalili, "Python Machine Learning", Packt publishing, 3rd Edition, 2019.

L: 30; P: 30; TOTAL: 60 PERIODS

Course Code	AI TOOLS FOR NATURAL LANGUAGE PROCESSING	L	T	P	E	C
23CS23E/ 23IT23E	(Common to CSE and IT Degree Programmes)	**2**	**0**	**2**	**0**	**3**

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Component

CO1: comprehend the fundamental concepts, architectures, and challenges in NLP(CDL1)
CO2: apply AI-based NLP tools for tasks such as text processing, sentiment analysis, and language modeling. (CDL1)
CO3: apply ML and deep learning models for NLP, fine-tune transformers, and evaluate model performance (CDL1)
CO4: integrate AI-driven NLP systems into real-world tasks, measure their impact, and address ethical concerns. (CDL2)

Practical Component

CO5: implement text preprocessing, POS tagging, and text similarity analysis through word embeddings (PDL1)
CO6: develop and deploy NLP and AI-based applications for sentiment analysis, chatbots, and APIs. (PDL2)

CO1: Comprehend the fundamental concepts, architectures, and challenges in NLP

Basics of Natural Language Processing: Components of NLP, Morphology, Syntax, Semantics, Pragmatics - AI in NLP: Machine Learning vs. Deep Learning Approaches - **L:5**

Challenges in NLP (Ambiguity, Context Understanding, etc.) - Overview of AI-based NLP Tools (NLTK, SpaCy)

CO2: Apply AI-based NLP tools for tasks such as text processing, sentiment analysis, and language modeling.

Text Preprocessing: Stopword Removal, Lemmatization, And Stemming - Named Entity Recognition (NER) and Part-of-Speech (POS) Tagging Word Embedding's: Word2Vec, GloVe, FastText - Text Classification and Sentiment Analysis - Implementing NLP pipelines using AI tools **L:7; P:10**

CO5: Implement text preprocessing, POS tagging, and text similarity analysis through word embeddings.

Install and set up NLP libraries: (NLTK, SpaCy), Implement basic text preprocessing (tokenization, stemming, and lemmatization), Perform Part-of-Speech (POS) tagging, Apply word embeddings (Word2Vec, GloVe) to analyze text similarity.

CO3: Apply ML and deep learning models for NLP, fine-tune transformers, and evaluate model performance

Traditional ML Models for NLP (Naïve Bayes, SVM, etc.) - Deep Learning for NLP: RNN, LSTM, GRU- Transformer Models: BERT, GPT, T5, etc. - Fine-tuning Pretrained Models for NLP Tasks - Comparing AI-based NLP models using performance metrics. **L: 9**

CO4: Integrate AI-driven NLP systems into real-world tasks, measure their impact, and address ethical concerns.

AI-powered Chatbots and Virtual Assistants - Machine Translation (Google Translate, DeepL, etc.) - Speech-to-Text and Text-to-Speech Applications - Ethical Considerations in AI-based NLP (Bias, Fairness, Privacy) - Case Studies of AI-driven NLP Systems- Customizing AI Models - Deploying AI-based NLP Systems (API, Cloud, etc.) **L:9**

CO6: Develop and deploy NLP and AI-based applications for sentiment analysis, chatbots, and APIs.

Implement a text and spam classifier using Naïve Bayes, Scikit-Learn and TF-IDF, Build a rule-based chatbot using NLTK, Implement a text-to-speech system using Google, Text-to-Speech (gTTS), **P:20**

Develop a social media sentiment analysis tool using BERT, Build and deploy a chatbot using Rasa or Dialog flow, Train a custom NLP model and deploy it as an API using FastAPI.

TEXT BOOKS:

1. Daniel Jurafsky and James H. Martin, "Speech and Language Processing", 3rd Edition, Pearson Education, 2024.
2. Goyal, Palash, et al. *Deep Learning for Natural Language Processing: Creating Neural Networks with Python*. 1st Edition, Apress, 2018.

REFERENCES:

1. Steven Bird, Ewan Klein and Edward Loper, Natural Language Processing with Python, 2nd Edition, O'Reilly Media publications, 2023.

2. Géron, Aurélien. *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*. 2nd Edition, O'Reilly Media, 2019.

L: 30; P: 30; TOTAL: 60PERIODS

Course Code	SOFT COMPUTING	L	T	P	E	C
23CS24E/ 23IT24E/	(Common to CSE, IT and AI & DS Degree Programmes)	**3**	**0**	**0**	**0**	**3**

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Components

CO1: relate the basic concepts of computational intelligence (CDL1)

CO2: apply fuzzy logic principles to handle uncertainty and approximate reasoning (CDL1)

CO3: implement evolutionary algorithms for problem-solving and optimization (CDL2)

CO4: solve multi-objective optimization problems using Evolutionary Algorithms (CDL2)

CO5: integrate hybrid computational intelligence techniques for real-world applications (CDL2)

CO1: Relate the basic concepts of computational intelligence (CDL1) **L:6**

Introduction to Computational Intelligence – Comparison with Artificial Intelligence – Paradigms of Intelligence – Soft Computing vs Hard Computing – Components of CI: Neural Networks, Fuzzy Systems, Evolutionary Computation – Applications and Challenges.

CO2: Apply fuzzy logic principles to handle uncertainty and approximate reasoning (CDL1) **L:9**

Introduction to Fuzzy Logic – Crisp vs Fuzzy Sets – Membership Functions – Fuzzy Set Operations – Fuzzy Rules and Inference – Fuzzy Reasoning – Fuzzy Control Systems – Defuzzification Methods – Case Study: Autonomous Driving : Lane Change & Collision
Avoidance.

CO3: Implement evolutionary algorithms for problem-solving and optimization (CDL2) **L:10**

Introduction to Evolutionary Computation – Genetic Algorithms (GA): Background – Framework – GA Strategies – GA Operators (Selection, Crossover, Mutation) – Encoding: Travelling Salesman Problem (TSP) – Reproduction and parameter tuning in Binary
Coded GA.

CO4: solve multi-objective optimization problems using Evolutionary Algorithms (CDL2) **L:10**

Concept of Domination and its properties – MOEA framework – Classification of MOEA techniques – Pareto and Non-Pareto based approaches – Pareto front visualization and
performance metrics – Case Study: Urban Traffic Signal Control

CO5: integrate hybrid computational intelligence techniques for real-world applications (CDL2) **L:10**

Hybrid Systems: Neuro-Fuzzy, Fuzzy-GA, PSO-ANN, ACO-ANN models – Comparative Analysis of CI Techniques (based on accuracy, convergence rate, and interpretability) –

Recent Advances in Computational Intelligence – Case Study: Environmental Informatics– Air Quality Forecasting using Hybrid Models.

TEXT BOOKS

1. Rudolf Kruse, Christian Borgelt, Frank Klawonn, "Computational Intelligence: A Methodological Introduction", 3rd Edition, Springer International Publishing, 2022.
2. Oscar Castillo, Patricia Melin, "Fuzzy Logic Hybrid Extensions of Neural and Optimization Algorithms: Theory and Applications", 1st Edition, Springer International Publishing, 2022.

REFERENCES:

1. Sugumaran D., Souvik Pal, Dac-Nhuong Le, and Noor Zaman Jhanjhi (Eds.), "Recent Trends in Computational Intelligence and Its Application", CRC Press, 2023.
2. Andries P. Engelbrecht, "Computational Intelligence: An Introduction", 2nd Edition, Wiley, 2021.
3. James M. Keller, Derong Liu, and David B. Fogel, "Fundamentals of Computational Intelligence: Neural Networks, Fuzzy Systems, and Evolutionary Computation", Wiley-IEEE Press, 2016.

L: 45; TOTAL: 45 PERIODS

Course Code	FUNDAMENTALS OF DEEP LEARNING	L	T	P	E	C
23CS25E/ 23IT25E	(Common to CSE and IT Degree Programmes)	2	0	2	0	3

COURSE OUTCOMES

Upon completion of this course, the student will be able to

Theory Component

CO1: apply neural network and machine learning optimization techniques to build, optimize, and evaluate predictive models. (CDL2)

CO2: implement and analyze convolutional and recurrent neural network architectures for processing image and sequence data. (CDL2)

CO3: optimize and fine-tune deep neural networks by applying advanced strategies to improve training efficiency, generalization, and performance. (CDL2)

CO4: apply advanced deep learning models, including attention mechanisms and generative architectures, for complex data representation and generation. (CDL2)

Practical Component

CO5: implement and evaluate deep learning models (pre-trained) for real-world tasks involving classification, image analysis, and sequence generation. (PDL2)

CO6: develop and optimize advanced deep learning models for object detection, graph-based learning, and natural language processing applications using MLOps framework. (PDL2)

CO1: Apply neural network and machine learning optimization techniques to build, optimize, and evaluate predictive models. **L:9**

Machine Learning optimization techniques – Underfitting and Overfitting – Hyper

parameters and validation sets – Estimators – Maximum Likelihood Estimation - Optimization – Evaluating the models - Introduction to Neural Networks – Perceptron – Multilayer Feed forward networks – Back propagation – Activation functions – Loss Function – Regularization. Data Augmentation - Noise Robustness – Early Stopping – Bagging – Dropout – batch normalization.

CO2: Implement and analyze convolutional and recurrent neural network architectures for processing image and sequence data. **L:7; P:15**

Convolutional Neural Networks – Convolution Operation – Architecture Overview – Input layers – Convolutional layers – pooling layers – fully connected layers Recurrent Neural networks – LSTM – Bidirectional RNNs – RNN Language model – Word Level RNN - Deep Recurrent Networks – Recursive Neural Networks.

CO5: Implement and evaluate deep learning models (pre-trained) for real-world tasks involving classification, image analysis, and sequence generation.

Implementation of Linear and Logistic regression models, Create a simple classifier using Feed forward network, Implement CNN for image classification, Implement a language model using RNN, Mini Project - Facemask Detection using CNN , Text Generation using RNN , Text to Speech using LSTM

CO3: Optimize and fine-tune deep neural networks by applying advanced strategies to improve training efficiency, generalization, and performance. **L:7**

Optimization in deep learning– Non-convex optimization for deep networks- Stochastic Optimization- Generalization in neural networks- Spatial Transformer Networks Tuning Deep Networks – Basic Concepts – Matching input data and network architecture – relating model goal and output layers – Working layer count, parameter count and Memory – weight initialization strategies – using activation functions – applying loss function – Dealing with overfitting.

CO4: Apply advanced deep learning models, including attention mechanisms and generative architectures, for complex data representation and generation. **L:7; P:15**

Encoder Decoder Models - Attention Mechanism - Attention over images – Hierarchical Attention – Variational auto encoders – Autoregressive models – NADE – MADE - PixelRNN – Generative Adversarial Networks (GANs) – Graph Convolution Network – Deep Belief Network.

CO6: Develop and optimize advanced deep learning models for object detection, graph based learning applications using MLOps framework.

Implement optimization techniques for CNN and RNN models, Implementation of Encoder Decoder model, Implementation of Graph Convolution network, Implement Faster RCNN, Implementation of object tracking. Mini Project - Object detection using Yolo algorithms, Diseases prediction using GCN models, Sentiment analysis using BERT models.

TEXT BOOKS

1. Francois Chollet, “Deep Learning With Python”, 3rd Edition, Manning Publications, 2024.
2. Josh Patterson, Adam Gibson,” Deep Learning: A Practitioner’s Approach”, 1st Edition, O’Reilly Media, Inc, 2017

REFERENCES

1. Bengio, Yoshua, Ian J. Goodfellow, and Aaron Courville. "Deep learning." An MIT Press, 2016.
2. Antonio Gulli, Amita Kappor, Sujit Pal, “ Deep Learning with TensorFlow 2 and Keras”, 2nd

Edition, Pack Publishers, 2019

2. AurelienGeron, "Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent System", 2nd Edition, O'Reilly Publication, 2017

L: 30; P: 30; TOTAL:60 PERIODS

CourseCode	TIME SERIES ANALYSIS	L	T	P	E	C
23CS26E/ 23IT26E	(Common to CSE, IT and AI & DS Degree Programmes)	**2**	**0**	**2**	**0**	**3**

COURSE OUTCOMES

Upon completion of this course, the student will be able to

Theory Component:

CO1: apply various descriptive analysis techniques for handling time series data.(CDL1)
CO2: analyze the characteristics and preprocessing challenges of physiological and financial time series data. (CDL2)
CO3: apply deep learning architectures such as RNN, LSTM, and GRU to model sequential dependencies in time series data.(CDL2)
CO4: analyze and implement advanced deep learning models, including CNN-LSTM hybrids and Transformer-based architectures, for multi-step time series forecasting. (CDL2)

Practical Component

CO5: implement visualization, preprocessing, decomposition, and transformation techniques on non-stationary time series data. (PDL1)
CO6: evaluate the performance of time series forecasting models to implement end-to-end solutions. (PDL2)
CO7: Implement deep learning models and Transformer-based architectures to build, train, evaluate, and deploy end-to-end time series forecasting solutions. (PDL2)

CO1: Apply the various descriptive analysis techniques for handling time series data. (CDL1) **L: 8;**

Time Series Data: definition and characteristics - Time Series Components: Trend, Seasonality, Cyclic, Irregular - Additive and Multiplicative Models - Smoothing Techniques - Forecast Accuracy Metrics – Transformations. Concept of Stationarity - differencing –Auto Regressive, Moving Average, Autoregressive Integrated Moving Average (ARIMA) models - Auto Correlation Functions **P: 8**

CO5: Implement visualization, preprocessing, decomposition, and transformation techniques on non-stationary time series data. (PDL1)

Visualizing time series data - Apply smoothing and transformations - Testing for stationarity - applying differencing, building ARIMA, SARIMA, ARIMAMAX models - generating forecasts.

CO2: Analyze the characteristics and preprocessing challenges of physiological and financial time series data. (CDL2) **L: 8;**

Physiological signals: ECG, EEG - Filtering - Feature extraction - Frequency Analysis **P: 8**

with Periodograms - Case studies: Heart Rate Variability, Emotion recognition. Financial series: Characteristics - returns vs. prices - volatility - Random Walk - Volatility modelling: Auto Regressive Conditional Heteroshedasticity (ARCH), GARCH - Granger Causality - Case studies: NIFTY 500 and Dow Jones Indexes

CO6:Evaluate the performance of time series forecasting models to implement end-to-end solutions. (PDL2)

Applying basic filters for ECG and EEG and to find cycles - extracting HRV metrics – extracting time, frequency, nonlinear and connectivity metrics. Calculating and plotting returns - volatility clustering - fitting a simple GARCH (1,1) – design of time series model to returns - visualizing conditional variance.

CO3:Apply deep learning architectures such as RNN, LSTM, and GRU to model sequential dependencies in time series data. (CDL2) **L: 7; P: 6**

Need for Deep Learning in Time Series - Recurrent Neural Networks (RNNs): Architecture - Vanishing gradient issues - Long Short-Term Memory (LSTM) Networks: Gates, Cell state - Sequence-to-sequence modeling - Gated Recurrent Units (GRU): Reset and update gates - Comparison with LSTM - Deep learning workflows: Data normalization - Sliding window techniques - Sequence preparation

CO7: Implement deep learning models and Transformer-based architectures to build, train, evaluate, and deploy end-to-end time series forecasting solutions. (PDL2)

Building RNN, LSTM, and GRU models for real-world time series analysis (Kaggle Data sets) - Predicting temporal patterns using deep models - Hyperparameter tuning - Visualizing training curves and predictions

CO4: analyze and implement advanced deep learning models, including CNN-LSTM hybrids and Transformer-based architectures, for multi-step time series forecasting. (CDL2) **L: 7; P: 8**

Encoder–Decoder Models for sequence forecasting - CNN–LSTM Hybrid Models - Transformers for Time Series: Self-attention mechanism - Positional encoding - Multi-head attention - Temporal Fusion Transformers (TFT) - Multivariate Time Series Forecasting and Evaluation

CO7: Implement deep learning models and Transformer-based architectures to build, train, evaluate, and deploy end-to-end time series forecasting solutions. (PDL2)

Implementing encoder–decoder LSTM - Implementing GRU and CNN-LSTM hybrid models - Transformer-based forecasting (Kaggle Data sets) - Multi-step and multi-horizon forecasting.

TEXT BOOKS

1. Rangaraj M. Rangayyan, Biomedical Signal Analysis: A Case-Study Approach, Wiley-IEEE Press, 2nd Edition, 2015.
2. Manu Joseph and Jeffrey Tackes, Modern Time Series Forecasting with Python, 2nd Edition, OReilly - Packt>, 2024.

REFERENCES

1. Rob J. Hyndman and George Athanasopoulos, Forecasting: Principles and Practice 3rd Edition, OTexts, Monash University, Australia, 2021.
2. Jason Brownlee, Deep Learning for Time Series Forecasting Predict the Future with MLPs, CNNs and LSTMs in Python, Machine Learning Mastery, 2019.

L: 30; P: 30 TOTAL: 60 PERIODS

CourseCode	NATURE AND BIO INSPIRED COMPUTING	L	T	P	E	C
23CS27E/ 23IT27E	(Common to CSE, IT and AI & DS degree Programmes)	**3**	**0**	**0**	**0**	**3**

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Component

CO1: demonstrate the fundamental concepts of nature-inspired computing (CDL1)

CO2:apply swarm basednature-inspired optimization algorithms for solving complex problems.(CDL1)

CO3:analyze advanced bio-inspired path planning optimizers for finding global optima. (CDL2)

CO4:apply combinatorial optimization for solving NP-hard problems.(CDL2)

CO1: Demonstrate the fundamental concepts of nature-inspired computing.(CDL1) Nature inspired computing - Philosophy - Branches: Individuals, Entities and agents - Parallelism and Distributive Interactivity: Adaptation, Feedback, Self-Organization - Complexity and Emergence - Bottom-up Vs Top-Down Determination - Chaos and Fractals	**L: 9**
CO2: Apply swarm based nature-inspired optimization algorithms for solving complex problems. (CDL1) Particle Swarm Optimization (PSO): Introduction, concepts, operators, PSO on Generalized Framework, Graphical Examples - Ant Colony Optimization (ACO)**:** Ant Colonies, Ant Foraging Behaviour, ACO algorithm with case studies of ACO – Bacterial Foraging Optimization (BFO): BFO algorithm - Parameter initialization – fitness evaluation	**L:12**
CO3: Analyze advanced bio-inspired path planning optimizers for finding global optima. (CDL2) Path planning optimizers: Bat algorithm, cuckoo search algorithm, grey wolf optimization, pigeon-inspired optimization – case study on hybrid models to solve complex problems.	**L: 12**
CO4: Apply combinatorial optimization for solving NP-hard problems. (CDL2) Introduction to NP-hard problems: Routing problems - assignment problems - scheduling problem - subset problem- Machine learning problem - Traveling salesman problem. Combinatorial Optimization: Meta-heuristics - Local search - Tabu search - Global search.	**L: 12**

TEXT BOOKS:

1. S. Das, A. Abraham, A. Biswas, A. P. Tonda (Eds.), "Applications of Nature-Inspired Computing and Optimization Techniques," Elsevier, 1st Edition, Paperback, 2024.
2. Marco Dorigo,Thomas Stutzle, "Ant Colony Optimization", MIT Press, 2004

REFERENCES:

1. A. K. Verma, "Nature and Bio-Inspired Computing: Algorithms and Applications", Springer, Paperback, 2018
2. S. N. Rao, "Nature-Inspired Algorithms and Intelligent Systems", Cambridge University Press, Paperback, 2017
3. M. Clerc, Particle Swarm Optimization, ISTE, 2006.

L: 45; TOTAL: 45 PERIODS

Course Code	GENERATIVE AI TECHNOLOGIES	L	T	P	E	C
23CS21L/ 23IT21L	(Common to CSE, IT and AI & DS Degree Programmes)	**0**	**0**	**2**	**0**	**1**

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Practical Component

CO1: Design, train, and evaluate machine learning, deep learning, and generative AI models for real-world tasks. (PDL2)

CO2: Implement, deploy, and operationalize AI systems using modern frameworks, APIs, and responsible AI practices. (PDL2)

CO1: Design, train, and evaluate machine learning, deep learning, and generative AI models for real-world tasks. (PDL2) **P: 20**

1. Develop a Python program to load a dataset using Pandas, clean missing values, visualize patterns with Matplotlib/Seaborn, and generate an EDA summary report.
2. Create a deep learning model using TensorFlow/PyTorch to classify images (MNIST/CIFAR), train the model, evaluate accuracy, and save the trained model to disk.
3. Develop an NLP application using Hugging Face Transformers to preprocess text, fine-tune a BERT/GPT model using LoRA or QLoRA, classify text data, and generate performance metrics.
4. Create a generative model (DCGAN/VAE) using PyTorch/TensorFlow to synthesize images, train the generator model, and visualize generated samples at different epochs.

CO2: Implement, deploy, and operationalize AI systems using modern frameworks, APIs, and responsible AI practices. (PDL2) **P: 10**

5. Create a RAG pipeline using FAISS/ChromaDB to store document embeddings, retrieve relevant chunks, integrate with an LLM (OpenAI/HF), and generate context-aware responses.
6. Develop a FastAPI/Flask application exposing a prediction API, containerize it using

7. Docker, configure cloud deployment (AWS/Azure/GCP), and test model inference using REST endpoints.

P: 30; TOTAL: 30 PERIODS

Course Code	AGENTIC AI TOOLS AND FRAMEWORKS	L	T	P	E	C
23CS22L/ 23IT22L	(Common to CSE, IT and AI & DS degree Programmes)	0	0	2	0	1

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Component

CO1: Design, develop, and evaluate autonomous AI agents using advanced reasoning, planning, memory systems, and multi-agent coordination. (PDL1)

CO2: Implement, integrate, and deploy production-ready agentic AI systems using modern frameworks, external tools, cloud platforms, safety guardrails, and responsible AI principles. (PDL2)

CO1: Design, develop, and evaluate autonomous AI agents using advanced reasoning, planning, memory systems, and multi-agent coordination. (PDL1) **P: 15**

1. Create a simple Python program that builds an autonomous agent using LangChain/LangGraph to perform step-by-step reasoning and complete small real-world tasks.
2. Implement basic planning methods (ReAct, Chain-of-Thought, Tree-of-Thought) to help the agent break down tasks and make decisions.
3. Add persistent memory using FAISS/ChromaDB so the agent can store information and use it in future tasks.
4. Develop a small multi-agent framework using LangGraph or CrewAI where each agent has a simple role (Planner, Researcher, Checker). • Make the agents communicate and cooperate to solve a shared task.

CO2: Implement, integrate, and deploy production-ready agentic AI systems using modern frameworks, external tools, cloud platforms, safety guardrails, and responsible AI principles. (PDL2) **P: 15**

5. Connect agents with external tools such as calculators, search APIs, file readers, or databases.
6. Implement context management so the agent can track conversation history and relevant information.
7. Build a simple Retrieval-Augmented Generation (RAG) model using ChromaDB for knowledge lookup.
8. Create a deployable agentic AI system by exposing the agent's functionalities through an API, adding safety filters, and packaging everything using Docker.

Software Requirements:

CO1: Python, LangChain, LangGraph, CrewAI, FAISS/ChromaDB, HuggingFace/LLM APIs

CO2: FastAPI or Flask, Docker, Cloud SDKs (optional), Simple logging tools, content filters, error-handling utilities

P: 30; TOTAL: 30 PERIODS

Course code	ACCELERATED AI DEVELOPMENT USING CUDA	L	T	P	E	C
23CS23L/ 23IT23L/	(Common to CSE, IT and AI & DS Degree Programmes)	0	0	2	0	1

COURSE OUTCOMES

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

Practical Component

CO1: apply the basics of CUDA concepts for High-Performance Computing (PDL1)

CO2: utilize tools like MPI, CUDA and cluster environments for HPC (PDL2)

CO1: Apply the basics of CUDA concepts for High-Performance Computing

1. Perform matrix operations and vector computations on GPU (CuPy/Numba)
2. Image enhancement using CUDA Parallel Processing
3. Train a simple model using CUDA-accelerated PyTorch

P:10

CO2: Utilize tools like MPI, CUDA and cluster environments for HPC

1. Run a deep learning model on GPU environment
2. Build and deploy a real time application using CUDA Acceleration

P:20

Tools Required: Python, CUDA, PyTorch, TensorFlow, NCCL

REFERENCES

1. Matloff, N, Programming on parallel machines: GPU, multicore, clusters and more. University of California, Davis, 2023.
2. NVIDIA Corporation, Deep learning and machine learning with GPGPU and CUDA: Unlocking the power of parallel processing, 2023.

P: 30; TOTAL: 30 PERIODS

Course Code	FUNDAMENTALS OF REMOTE SENSING AND GIS	L	T	P	E	C
23CS24L/ 24IT24L	(Common to CSE, IT and AI & DS Degree Programmes)	1	0	0	0	1

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Component

CO 1: apply the basic concepts of Remote Sensing and GIS for real-world scenario (CDL1)

CO 2: Demonstrate the process pipelines for various types of Satellite Data (CDL2)

CO1: Apply the basic concepts of Remote Sensing and GIS for real-world scenario (CDL1) **L: 6**

Remote Sensing: Fundamental principles - components - electromagnetic spectrum - bands - stages of remote sensing process - Geographic Information Systems (GIS) - Coordinate Systems and Map Projections - Basic Spatial Analysis - Cartographic Design Principles

CO2: Demonstrate the process pipelines for various types of Satellite Data (CDL2)

Remote Sensing Data : Optical - SAR - Hyperspectral - LiDAR - Thermal – Data fusion - Application of remote sensing - Agriculture - Forestry - Urban - Ocean science - Inland water – Case Studies with real-world data. **L: 9**

TEXT BOOKS:

1. Thomas M. Lillesand, Ralph W. Kiefer, and Jonathan W. Chipma, Remote Sensing and Image Interpretation, 7th Edition, Wiley, 2022.
2. James B. Campbell, Randolph H. Wynne, and Valerie A. Thomas, Introduction to Remote Sensing, 6th Edition, Guilford Press, 2022.
3. Joel Lawhead, Learning Geospatial Analysis with Python: Understand GIS fundamentals and perform remote sensing data analysis using Python 3.7, 3rd Edition, Packt>, 2019.

L: 15; TOTAL: 15 PERIODS

Course Code	MINI-CAPSTONE PROJECT	L	T	P	E	C
23CS2ME/ 23IT2ME	(Common to CSE and IT Degree Programmes)	**0**	**0**	**0**	**6**	**3**

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Experiential Component

CO1: analyze real-world problems and preprocess relevant data to identify suitable applications of deep learning, reinforcement learning, and natural language processing. (PDL2)

CO2: design and develop intelligent systems using appropriate models and algorithms from deep learning, reinforcement learning, and NLP to address identified problems. (PDL2)

CO3: evaluate the performance of developed intelligent systems and optimize them using suitable metrics and tuning techniques. (PDL3)

Soft skill Component

CO4: enhance problem-solving and adaptability skills by navigating complex tasks and applying innovative solutions in team-oriented environments.

Each student individually selects any project in their area of interest in engineering design and work under the supervision of their allotted guide approved by Head of the department.

Module 1: Problem Identification & Data Preparation Understand the scope of computational intelligence in solving real-world problems **E: 20**

1. Brainstorm and finalize problem statements across domains (healthcare, finance, education, etc.)
2. Data acquisition and preprocessing (cleaning, normalization, feature extraction)
3. Submission of project proposal including objectives, data description, and expected outcomes

Deliverable: Approved project proposal document with data plan

Module 2: System Design & Model Development

Select appropriate computational techniques (DL, RL, NLP) **E:25**

1. Design model architecture based on the problem domain
2. Train initial models using tools like TensorFlow, Py-Torch, OpenAI Gym, Hugging-Face, etc.
3. Implement smart (automate and optimize) training pipelines

Deliverable: Model design document and training scripts

Module 3: System Integration & Performance Evaluation

1. Integrate model with end-user applications or pipelines **E: 25**
2. Evaluate using performance metrics (accuracy, precision, reward score, BLEU score, etc.)
3. Perform model tuning (hyperparameter optimization, regularization, fine-tuning)
4. Conduct comparative analysis and validation

Deliverable: Working intelligent system with evaluation report

Module 4: Final Demonstration & Reporting

Live demonstration of the system (input to intelligent output) **E:20**

1. Technical documentation and final project report
2. Code repository submission (e.g., GitHub)
3. Viva-voce and Q&A based on COs

Deliverables:

- Live demo session
- Final project report
- Code submission

E: 90; Total: 90 PERIODS

Course code	**COMPUTER GRAPHICS**	**L**	**T**	**P**	**E**	**C**
23CS31E/ 23IT31E/ 23AD31E	(Common to CSE, IT and AI & DS degree Programmes)	**3**	**0**	**0**	**0**	**3**

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Components

CO1**:** apply the basic of computer graphics, line and shape drawing algorithms in computer applications. (CDL1)

CO2: analyze 2D and 3D transformations and perform clipping operations using various algorithms for efficient viewing and rendering in computer graphics. (CDL2)

CO3: analyze and apply object representation methods, and visible surface detection algorithms. (CDL2)

CO4: implement 3D viewing techniques and rendering algorithms for effective visualization. (CDL2)

CO5: apply animation principles, design sequence in gaming scenarios. (CDL2)

CO1: apply the basic of computer graphics, line and shape drawing algorithms in computer applications.

Introduction to Computer Graphics - Application areas of Computer Graphics - Overview of graphics systems - Video display devices -Raster-scan systems - Random scan systems - Graphics monitors - Work stations - Input devices- Principles of Modeling. Output primitives - Points and lines - Line drawing algorithms - Mid- point circle and Ellipse algorithms. **L:9**

Filled area primitives - Scan line polygon fill algorithm - Boundary-fill and Flood-fill algorithms.

CO2: analyze 2D and 3D transformations and perform clipping operations using various algorithms for efficient viewing and rendering in computer graphics.

2D and 3D Geometric Transformations – Scaling - Shearing - Rotation - Reflection, Translation - Composite Transformations - Ray and Path Tracing. **L:9**

2D Viewing - Viewing Pipeline - Window to view-port coordinate Transformation. Clipping Operations - Point Clipping - Cohen Sutherland Line Clipping - Sutherland Hodgeman Polygon Clipping.

CO3: analyze and apply object representation methods and visible surface detection algorithms.

3-D object representation: Polygon surfaces - Quadric surfaces - Spline representation - Hermite curve - Bezier curve and B-Spline curve - Bezier and B-Spline surfaces - Basic illumination models - Polygon rendering methods. **L:9**

3-D viewing: Viewing pipeline - Viewing coordinates and clipping - Visible surface detection methods- Classification - Back-face detection - Depth-buffer scan-line - depth sorting - BSP-tree methods.

CO4: implement 3D viewing techniques and rendering algorithm for effective visualization.

Projections-Parallel and Perspective Projection -Bezier Curves and Surfaces - B-Spline Curves and Surfaces - Visible Surface Detection Algorithms - Back-Face Detection Algorithm - Depth Buffer Method - Scan line Method. **L:10**

Color Models - RGB Color Model - YIQ Color Model -CMY Color Model - HSV Color Model - Conversion Between HSV and RGB Models and HLS Color Model

CO5: apply animation principles, design sequence in gaming scenarios.

Animation: Principles of animation - Design of Animation Sequences - Animation Functions - Raster Animations - Key-Frame Systems - Motion Specifications - Computer Graphics in games. **L:8**

TEXT BOOKS

1. Peter Shirley, Fundamentals of Computer Graphics, A K Peters/CRC Press, 5th Edition, 2022.
2. Edward Angel and Dave Shreiner: Interactive Computer Graphics: A Top-Down Approach with WebGL, Addison-Wesley, 8th Edition, 2020.
3. Sumanta Guha, Computer Graphics through OpenGL: from theory to experiments, 4th Edition, CRC Press, 2023.
4. Hamza Zubair, Computer Graphics, Toronto Academic Press, 2024.
5. John F. Hughes, Computer Graphics: Principles and Practice, Addison-Wesley, 3rd Edition, 2014.

REFERENCE BOOKS

1. Eric Lengyel, 3D Graphics for Game Programming, A K Peters/CRC Press, 2nd Edition, 2023.
2. Roger D. Rogers and David F. McLain: Computer Graphics: A Programmed Approach, Prentice Hall, 3rd Edition, 2019.
3. Donald Hearn and M. Pauline Baker: Computer Graphics Using OpenGL, Prentice Hall, 6th Edition, 2014.

L: 45; TOTAL: 45 PERIODS

Course code		L	T	P	E	C
23CS32E/ 23IT32E/ 23AD32E	**UI/UX DESIGN** (Common to CSE, IT and AI & DS degree Programmes)	**2**	**0**	**0**	**2**	**3**

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Components

CO1: analyze UX principles and processes to create user-centered solutions. (CDL1)
CO2: apply User research and UI design principles for user friendly interface. (CDL2)
CO3: utilize interactive design techniques for an engaging UI design. (CDL2)
CO4: apply responsive design principles to create adaptable digital interfaces.(CDL2)

Experiential component

CO5: design user-friendly, visually appealing interfaces and provide industry-relevant insights by applying UX design principles.(PDL1)

Softskill component

CO6: demonstrate creativity, diverse and inclusive attitude while practicing project component.

CO1: Analyze UX principles and processes to create user-centered solutions. (CDL1).

Introduction about UX - Five Main Ingredients of UX - Three "Whats" of User Perspective -Pyramid of UX Impact - UX Is a Process - UX - Not an Event or Task.Behaviour Basics: Psychology versus Culture - User Psychology - Experience - Conscious vs Subconscious Experience - Emotions - Gain and Loss – Motivations. **L:8**

CO2: Apply User research and UI design principles for user friendly interface. (CDL2) **L:10;**

User research: Design thinking process – Conduct research methodologies - Interviews -Surveys -Card Sorting - Creating User personas- Basic of Information Architecture - UI design principles- Creating UI Design – User Centered Design – site mapping and wireframe techniques – UI Aesthetics – Introduction to Design Tools. **E:8**

CO3: Utilize interactive design techniques for an engaging UI design. (CDL2)

Designing Web Interfaces – Grid and Layout – Design Pattern -Visual Hierarchy- Basics of Interaction design – structure and navigation - UI animation: Loading progress- state changes- structure and navigations Micro interaction – Branding - iterative design - Interactive prototypes **L:8; E:12**

CO4: Apply responsive design principles to create adaptable digital interfaces.(CDL2)

Responsive Design -Adaptive design - Understanding user diversity - Adaptive and Fluid layouts - Performance optimization - Future trends and updates-Create a responsive design for various devices and screen sizes. **L:4; E:10**

CO5: Design user-friendly, visually appealing interfaces and provide industry-relevant insights.

TEXT BOOKS

1. Joel Marsh, "UX for Beginners", O'Reilly Media, Inc., 1st Edition 2015.
2. Alvin Albuero De Luna - Essentials of User Interface Design :e-book edition 2022
3. Xia Jiajia, "UI UX Design", O"Reilly, Artpower International, 2016.

REFERENCE BOOKS

1. David Travis, "Fundamentals of Creating a Great UI/UX", Creative Tim, 2022.
2. Raywenderlich Tutorial Team, Prateek Prasad – "App Design Apprentice _ A Non-designer's Guide to Making Better Mobile UI & UX":e-book edition 2022
3. Alan Dix, Janet Finlay, Gregory Abowd, Russell Beale, "HumanComputer Interaction", Pearson Education, 3rd Edition, 2004.
4. Alan Cooper, "The Essential Of User Interface Design", Wiley Dream Tech Ltd., 2002.

L: 30; E: 30; TOTAL: 60 PERIODS.

Course code	**BLOCKCHAIN GAMING**	**L**	**T**	**P**	**E**	**C**
23CS33E/ 23IT33E/ 23AD33E	(Common to CSE, IT and AI & DS degree Programmes)	**2**	**0**	**2**	**0**	**3**

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Components

CO1: develop foundational skills for Game Development. (CDL1)

CO2: demonstrate proficiency in 2D & 3D graphics by applying Physics, Materials, 3D Character Control and Particle Systems. (CDL2)

CO3: apply 3D Animation, Camera Control, Audio management and UI design for Game Development. (CDL2)

CO4: integrate the concepts of Non-Fungible Tokens to a game using Blockchain. (CDL1)

CO5: design a Blockchain game using backend & frontend tools and frameworks. (CDL1)

Practical Components

CO6: develop games using 2D & 3D graphics with animation techniques and interactions in Unity.(PDL2)

CO7: design and Demonstrate User Interface, Audio in gaming environment. (PDL2)

CO8: create and deploy an NFT Smart Contract on a Blockchain within a Game. (PDL1)

CO1: Develop foundational skills for Game Development.

CO6: Develop games using 2D & 3D graphics with animation techniques and interactions in Unity.

Introduction to Game Development – Working in Unity: Game Objects – Components – Prefabs – Scenes – Assets - Building Unity Projects - Unity Packages – Scripting: life cycle of a script - Accessing Components - Finding Objects – Singletons - Using Coroutines – Managing Objects – Scriptable Object – Input: Keyboard Input – Unity Input system – Input Actions – Locking and Hiding the Mouse Cursor – Mouseover and Click Events. **L:5; P:6**

Develop a Script for 2D games using Unity.

CO2: Demonstrate proficiency in 2D & 3D graphics by applying Physics, Materials, 3D Character Control and Particle Systems.

CO6: Develop games using 2D & 3D graphics with animation techniques and interactions in Unity.

2D Graphics: Creating a Sprite with 2D Physics – Customizing Sprite Collision Shapes – Composite Collider – Sprite Packer – Applying Forces to 2D Objects - 3D Graphics: Creating a Simple Material - Controlling a Material's Property - Creating an Unlit Material - Setting Up a Material Using Textures – Understanding Fixed Update – Mouse Look – Controlling a 3D Character - Interacting with Switches and Objects - Picking Up and Putting Down Objects - Particle system. **L:6; P:8**

Develop a Script for 3D games using Unity.

Create a Unity project to apply animation techniques and particle system.

CO3: Apply 3D Animation, Camera Control, Audio management and UI design for Game Development.

CO7: Design and Demonstrate User Interface, Audio in gaming environment.

Animating an Object - Basic Character Movement - Inverse Kinematics – Masked Movement - Blended Movement - Tracking and switching Camera - Keeping Multiple Objects in View – Dollying a camera - Audio: Playing Sounds – Mixer – Audio Effects – Ducking – Multiple Audio Zones – Playing Audio with Scripts - User Interface: Working with UI Controls – Animating the UI. **L:7; P:8**

Design an animated game to adopt better UI experience.

CO4: Integrate Non-Fungible Token (NFT) concepts into a game using Blockchain.

CO8: Create and deploy an NFT Smart Contract on a Blockchain within a Game.

Introduction to Blockchain Game Development – Decentralized Game Architecture – Benefits of Blockchain Games - Challenges of Blockchain Games - Non-Fungible Tokens (NFTs) in Gaming – NFT Standards (ERC-721, ERC-1155) - Designing and Minting NFTs – Creating a Mint Site – Create an NFT Smart Contract - Implementing Smart Contracts for Games - NFTs **L:6; P:8**

in Unity - NFT Trading and Royalties

Creating a Mint Site and NFT Smart Contract using Blockchain.

CO5: Design a Blockchain game using backend & frontend tools and frameworks.

Blockchain Game Backend Development - Setting Up Game Servers - Integrating Blockchain Nodes - Blockchain Game Development Tools and Frameworks (Unity, Unreal Engine, Web3.js) - Frontend Development for Blockchain Games - Web3-Enabled User Interfaces - Wallet Integration and Management - Play-to-Earn Mechanisms - Real-World Examples of Blockchain Games. **L:6**

TEXT BOOKS:

1. Paris Buttfield-Addison, Jon Manning, and Tim Nugent, "Unity Game Development Cookbook: Real-Time Solutions from Game Development to AI", 2nd Edition, O'Reilly Media, 2023.
2. Ashley Godbold, "Mastering UI Development with Unity", 2nd Edition, Packt Publisher, 2024.
3. Victor G Brusca, "Advanced Unity Game Development", 1st Edition, Apress Publisher, 2022.

REFERENCE BOOKS:

1. Ryley James, "NFT for Beginners", 1st Edition, ISBN-13: 9798502200295, 2021.
2. John P. Doran, "Unity 2022 Mobile Game Development", 3rd Edition, Packt Publisher, 2023.
3. Franz Lanzinger, "3D Game Development with Unity", 1st Edition, CRC Press – Taylor & Francis Group, 2022.
4. Anthony Davis, Travis Baptiste, Russell Craig, "Unity 3D Game Development", 1st Edition, Packt publisher, 2022
5. Swayam NPTEL Course on "Making Learning Engaging Through Interactive Games".
6. Swayam NPTEL Course on "Blockchain Architecture Design and Use Cases".

L:30; P:30; TOTAL:60 PERIODS

Course Code	3D MODELING AND DESIGN	L	T	P	E	C
23CS34E/ 23IT34E/ 23AD34E	(Common to CSE, IT and AI & DS Degree Programmes)	**2**	**0**	**2**	**0**	**3**

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Components

CO1: apply the fundamental concepts of 3D graphics, modeling, and coordinate systems. (CDL1)

CO2: modify the 3D models using advanced sculpting and modeling techniques. (CDL2)

CO3: integrate mapping and unwrapping techniques to create texture based 3D models. (CDL2)

CO4: transform 3D models into realistic scene by using lighting, rendering and camera techniques. (CDL2)

CO5: apply the simulation and architectural visualization techniques to mimic the real world environment. (CDL2)

Practical Component

CO6: formulate 3D object creation, transformation, and scene organization. (PDL2)
CO7: develop 3D models by applying materials, texturing and mesh editing tools. (PDL2) CO8: devise various compositing workflows using 3D models. (PDL2)

CO1: Apply the fundamental concepts of 3D graphics, modeling, and coordinate systems. (CDL1) Overview of 3D computer graphics - applications - Blender interface – Navigating the view – workspace – Regions – viewport and scene organization – hardware support – 3D modeling fundamentals – Mesh – working with primitives – rendering – modifiers – precision modeling techniques. **L:6; P:4**

CO6: Formulate 3D object creation, transformation, and scene organization. (PDL2)
Apply translation, rotation, and scaling transformations to a 3D model.

CO2: Modify the 3D models using advanced sculpting and modeling techniques. (CDL2) Sculpting and Retopology – Dynamic Topology – Multiresolution modifier – Voxel remeshing – Cloth Sculpting – Sculpt mode settings – brush types – Animation techniques – game animation creation – Character Rig for animation. **L:12; P:6**

CO6: Formulate 3D object creation, transformation, and scene organization. (PDL2)
Apply sculpting and animation techniques to a 3D model.

CO3: Integrate mapping and unwrapping techniques to create texture based 3D models. (CDL2) Materials and Texturing: Introduction to PBR materials, types of textures - UV mapping and unwrapping – Painting textures – Procedural textures – Shaders: BSDF, Emission, Volume, Background – Principled BSDF: Layers – Inputs – Subsurface Scattering – Specular level – Sheen – Emission. **L:10; P:6**

CO7: Develop 3D models by applying materials, texturing and mesh editing tools. (PDL2)
Apply texturing, shaders and UV mapping to a 3D model.

CO4: Transform 3D models into realistic scene by using lighting, rendering and camera techniques. (CDL2)
Types of lights : Point light , Spot light , Area light , Sun light - Power of lights - Renderer settings: EEVEE Rendering, Cycles Rendering – Key differences in shading - Global illumination– Reflections – Transparency - Refraction – shadows – Volumetrics – motion blur – depth of field – Light ray data – Polygon count. **L:12; P:8**

CO8: Devise various compositing workflows using 3D models. (PDL2)
Apply lighting and rendering techniques to enhance the visual quality of 3D scenes.

CO5: Apply the simulation and architectural visualization techniques to mimic real world environment. (CDL2)
Fundamentals of Animation - Keyframing – Timeline editor – Armatures and Bone constraints – Object Animation – Physics simulations and VFX – Cloth, fire, smoke and Fluids simulations – Camera effects and motion blur - Introduction to Google SketchUp – Interface and Modeling Workflow - Importing SketchUp Models - Architectural Visualization - Rendering using Google SketchUp Integration **L:5; P:6**

CO8: Devise various compositing workflows using 3D models. (PDL2)

Demonstrate simulation, VFX and Google sketch up for real world scenarios.

TEXT BOOKS

1. Brandy, Mac - Blender 4.3 simplified for professionals "A Comprehensive Guide to 3D Modeling, Texturing, and Rendering in Blender 4.3", 2025
2. Ze-Nian Li, Mark S. Drew, Jiangchuan Liu, Fundamentals of Multimedia", 3rd Edition, Springer Texts in Computer Science, 2021.

REFERENCE

John M Blain, The Complete Guide to Blender Graphics: Computer Modeling & Animation, CRC press, 3rd Edition, 2016.

Course Code	AUGMENTED REALITY AND VIRTUAL REALITY	L	T	P	E	C
23CS35E/ 23IT35E/ 23AD35E	(Common to CSE, IT and AI & DS Degree Programmes)	3	0	0	0	3

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

CO1: utilize the foundations and trends of immersive technologies in relevant applications (CDL1)
CO2: apply appropriate AR tools to highlight interactivity in applications. (CDL1)
CO3: apply VR technologies in emerging applications. (CDL1)
CO4: apply rendering and modeling techniques in VR distributed architectures (CDL2)
CO5: analyze various frameworks and development tools in VR domain. (CDL2)

CO1: Utilize the foundations and trends of immersive technologies in relevant applications.

Introduction to Immersive Technologies - Historical evolution of immersive technologies – AR, VR-Key characteristics and taxonomy - Technological Foundations - Current Trends and Future Directions **L:4**

CO2: Apply appropriate AR tools to highlight interactivity in applications.

Augmented reality: Introduction – Types of AR targets –components – User experience - Field of view – visual perception – Degree Of Freedom (DOF) - 3D formats - applications of AR – AR concepts: sensors - processor – display- Ingredients of AR experience- SDK Tools: Vuforia – AR Toolkit – AR core - AR foundation – AR algorithm: SLAM - AR devices. **L:9**

CO3: Apply VR technologies in emerging applications.

Virtual reality: Introduction-The Three I's of VR- Input devices - Trackers, Navigation, Gesture, Output devices – Graphic displays - Human vision System - Gaze interaction - Characteristics - technologies (LCD,LED,OLED,AMOLED), HMDs, 3D sound, Haptic Displays - Human Auditory System- Emerging applications (e.g., metaverse, digital twins) – VR HMD devices. **L:10**

CO4: Apply rendering and modeling techniques in VR distributed architectures.

Fundamentals of Computer Graphics - Rendering and graphics pipeline- Modern graphics and haptic rendering pipeline - Gaming desktop architecture -Distributed VR **L:10**

architectures-cloud rendering- Geometric modelling - 3D authoring software and scanners - scene illumination - Texture mapping - Transformation invarients - behaviour modelling – Level of detail management (LOD).

CO5: Analyze various frameworks and development tools in VR domain.

VR tools: XR interaction toolkit - OpenXR - Meta XR All-in-one SDK – HP VR launch kit - SteamVR plugin - Amazon Sumerian - Godot Engine - A-Frame(webVR) – CoSpaces Edu - Three.JS – WebGL - Babylon.js. **L:12**

TEXT BOOKS:

1. Jonathan Linowes, "Augmented Reality with unity AR foundation", 1st Edition, Packt publishing, 2021
2. Jonathan Linowes and Krystian Babilinski,"Augmented Reality for Developers", 1st Edition, Packt publishing, 2017.
3. Griore C.Burdea and Philippe Coiffet,"Virtual Reality Technology", 3rd Edition, IEEE press-wiley, 2024.
4. Steven M. LaValle, "Virtual Reality," 1st Edition, Cambridge University Press, 2017

REFERENCE BOOKS:

1. Micheal Lanham. "Learn ARCore – Fundamentals of Google ARCore", 1st Edition, Packt publishing, 2018.
2. Paul Mealy, John Carucci, "Virtual & Augmented Reality for Dummies," 1st Edition, For Dummies, 2017.
3. Tomas Akenine-Möller, Eric Haines, "Real-Time Rendering," 4th Edition, A K Peters/CRC Press, 2018.

L: 45; TOTAL: 45 PERIODS

Course Code	AI FOR HUMAN COMPUTER INTERACTION	L	T	P	E	C
23CS36E/ 23IT36E/ 23AD36E	(Common to CSE, IT and AI & DS degree Programmes)	3	0	0	0	3

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Component

CO1: recognize foundational AI concepts and user behavior patterns that enhance human–computer interaction. (CDL1)

CO2: implement advanced text-correction models by integrating AI-assisted mechanisms. (CDL1)

CO3: analyze design considerations to develop accessible and adaptive interfaces for improved user interaction. (CDL2)

CO4: apply Leap Motion technology to enhance human-computer interaction. (CDL2)

CO5: design and evaluate Human-Centric AI solutions by integrating principles of HAX & XAI. (CDL2)

CO1: Recognize foundational AI concepts and user behavior patterns that enhance human–computer interaction. (CDL1) **L:9**

Introduction to AI in Human – Computer Interaction – Role of AI in improving user interaction and usability – Human behavior and challenges in mobile text entry – Mobile text correction fundamentals – Overview of traditional correction methods: cursor movement, gesture typing, and backspace operations – Types of text entry errors such as substitution, omission, and insertion.

CO2: Implement advanced text-correction models by integrating AI-assisted mechanisms. (CDL1) **L:10**

NLP and Deep Learning Techniques – Traditional NLP correction methods: N-grams and edit distance – Deep learning architectures for text correction such as RNN, CNN, and GRU models – Encoder – decoder sequence modeling for intelligent correction – AI Interaction Techniques – Drag-n-Drop, Drag-n-Throw, and Magic Key mechanisms – Design principles for AI-assisted text correction – Dataset preparation using CoNLL, Yelp, and Amazon datasets – Model training and implementation insights for integrating AI correction within Android keyboard systems.

CO3: Analyze design considerations to develop accessible and adaptive interfaces for improved user interaction. (CDL2) **L:10**

Just Correct framework: post-hoc correction without cursor repositioning - AI algorithms: Edit Distance, Word Embedding Similarity, Sentence Coherence - Multimodal interaction: tap, gesture, voice inputs - Evaluation: accuracy, correction time, success rate, usability (NASA-TLX, SUS) - Design implications: accessibility and adaptive interfaces

CO4:Apply Leap Motion technology to enhance human-computer interaction (CDL2) **L:8**

Introduction to Leap Motion technology – Leap Motion and Gesture-based Interaction – Development Environment and SDK for Leap Motion – System Architecture – principles of Leap Motion - Leap Motion V2 vs Ultraleap Gemini - Hand Gestures Recognition - Leap Motion in Augmented Reality and Virtual Reality – Future trends and applications of Leap Motion in HCI.

CO5: Design and evaluate Human-Centric AI solutions by integrating principles of HAX & XAI. (CDL2) **L:8**

Introduction to Human-AI Interaction (HAX) – Designing Human-Centered AI Systems – Explainable AI (XAI) – Ethical and Responsible AI Design – Emotion, trust and Human-AI Collaboration – Prototyping AI interfaces – Applications. Case Study: Virtual Assistant, Personalized recommendation, Autonomous vehicles.

TEXT BOOKS

1. Yang Li & Otmar Hilliges, Artificial Intelligence for Human–Computer Interaction: A Modern Approach, Springer, 2021.
2. Panagiotis Germanakos, Monika Juhasz, Aparna Kongot, Devashree Marathe, Dimitris Sacharidis (Eds.), "Human-Centered AI: An Illustrated Scientific Quest", 1st Edition, Springer Cham, 2025.

REFERENCES

1. Akshay Kore, "Designing Human-Centric AI Experiences: Applied UX Design for Artificial Intelligence", O' Reilly 1st Edition, Apress, 2022.
2. Ben Shneiderman et al., Designing the User Interface: Strategies for Effective HCI, Pearson, 6th Edition 2018.
3. Yang Li. Dieter Schmalstieg & Tobias Hollerer, Augmented Reality: Principles and Practice, Addison-Wesley, 1st Edition 2016.

L: 45; TOTAL: 45 PERIODS

Course Code	DESIGN OF VIRTUAL REALITY SYSTEMS	L	T	P	E	C
23CS37E/ 23IT37E/ 23AD37E	(Common to CSE, IT and AI & DS degree Programmes)	**1**	**0**	**0**	**4**	**3**

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Component

CO1: apply foundational principles of Virtual Reality (VR) Systems. (CDL2)

CO2: design prototypes with interactive VR experiences for improving usability. (CDL2)

Experiential Component

CO3: deploy customizable immersive solutions for real-world applications**.**(PDL3)

Soft Skill Component

CO4: demonstrate effective communication and teamwork (Multi-disciplinary) in VR project development.

CO1: Apply foundational principles of Virtual Reality (VR) Systems. **L:8**

Introduction – Threes I's of VR – History of VR –AR/VR/MR/XR Difference –Types of VR experience – Types of HMD –VR Input Devices and Tracking - Output Devices – Google Cardboard, Daydream – Gear VR – HTC Vive – Oculus HMD – HoloLens – Apple Vision Pro – VR SDK Packages- Navigation, Teleportation and Locomotion techniques

CO2: Design prototypes with interactive VR experiences for improving usability. **L:7**

Lighting and Rendering Strategies - Create menu panel UI Design – Light Probes and Reflection Probes - Physics Particle system – GameObject Life Cycle – Terrain - ProBuilder and ProGrids - Skyboxes – Animation – Audio – Frame rates - Optimizing performance – LOD – Lighting and baking – Occlusion culling – Rendering pipeline.

CO3: Deploy customizable immersive solutions for real-world applications. **E:45**

Setting up VR Project (Unity/Unreal) – World Space UI Design – Interaction with UI elements - Applications of VR in various industries.

REFERENCE BOOKS

1. Grigore C.Burdea and Philippe Coiffet, "Virtual Reality Technology", 3rd Edition, Wiley-IEEE Press, 2024.
2. Jonathan Linowes, "Unity 2020 Virtual Reality Projects", Packt Publication, 3rd Edition, 2020.
3. William R Sherman and Alan B Craig, "Understanding Virtual Reality", Morgan Kaufma, 2nd Edition, 2018.

L: 15; E:45; TOTAL: 60 PERIODS

Course Code	METAVERSE DEVELOPMENT	L	T	P	E	C
23CS38E/ 23IT38E/ 23AD38E	(Common to CSE, IT and AI & DS degree Programmes)	**3**	**0**	**0**	**0**	**3**

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Components

CO1**:** analyze the key components, technologies and challenges of the Metaverse. (CDL1)
CO2: analyze blockchain, tokens, smart contracts and decentralized applications. (CDL2)
CO3: develop multiplayer metaverse environments using Photon and Meta Quest SDKs. (CDL2)
CO4: apply blockchain and security technologies in the metaverse. (CDL2)
CO5: analyze Digital Twin systems for real-world applications (CDL2)

Soft Skill Component

CO6: demonstrate the ability to design metaverse environment for real-world applications.

CO1: Analyze the key components, technologies and challenges of the Metaverse. (CDL1) **L:8**

Introduction-Metaverse - Rise of the Metaverse – Layers of Metaverse - Milestones of Metaverse –Metaverse and Web 3.0- Virtual Reality in the Metaverse – Digital Twins - Challenges and requirements of Metaverse**.**

CO2: Analyze blockchain, tokens, smart contracts and decentralized applications. (CDL2) **L:8**

Blockchain – Fungible Token (FT) – Non-Fungible Token (NFT) – Smart Contracts – Tokenomics – Decentraland – Cryptovoxels – The Sandbox – Somnium Space – IPFS – Goerli – Chainsafe – dApps – MetaMask – Mining.

CO3:Develop multiplayer metaverse environments using Photon and Meta Quest SDKs. (CDL2) **L:10**

Multiplayer Room – Photon SDK – Install – Life Cycle – Lobby creation – Player Prefab – Network Manager – Adding Text and Voice Chat – Avatar – NPC – Streaming Video in Metaverse - Meta Quest SDK

CO4:Apply blockchain and security technologies in the metaverse.(CDL2) **L:10**

Blockchain Gaming in Metaverse – Roblox Human Co-Experience – P2E Blockchain Gaming – GameFi with NFT – Metaverse Privacy – Zero-Knowledge Proof (ZKP) – Secure Multiparty Computation – Homomorphic Encryption (HE) and Federated Learning (FL) - Metaverse Security -Blockchain and Metaverse -CIA – Challenges for Metaverse community- Public Key Cryptography – Self-Sovereign Identity (SSI) and Data Ownership – Smart Contract Security.

CO5: Analyze Digital Twin systems for real-world applications (CDL2) **L:9**

Digital Twin (DT) concepts, key features and visions – Digital Twin Architecture –

Modeling Approaches for DT – Methodologies, framework and Challenges – Digital Twin Networks (DTN) Concepts, Communications and Applications - Artificial Intelligence in Digital Twin – Deep Reinforcement Learning Enabled DT – Federated Learning for DT – Transfer Learning for DT - Blockchain integration in Digital Twin systems – Block Generation and Consensus Mechanism, Roblox Studio - Case Studies: Driving simulation, Urban planning and Infrastructure management, Realtime monitoring of medical equipment, supply chain and logistics, Robotics.

TEXT BOOKS:

1. David Canton Nadales, "Build Your Own Metaverse with Unity", 1st Edition, Packt Publishing, 2023.
2. Winston Ma Ken Huang, "Blockchain and Web3: Building the Cryptocurrency, Privacy, and Security Foundations of the Metaverse", First Edition, Wiley, 2022.

REFERENCE BOOKS

1. Paul Doherty, "Unlocking the Metaverse: A Strategic Guide for the Future of the Built Environment", 1st Edition, Wiley, 2024
2. Ian Khan, "Metaverse for dummies", 1st Edition, John Wiley & Sons Publisher, 2023.
3. Manisha Vohra, "Digital Twin Technology Fundamentals and Applications", Scrivener Publishing, Wiley, 2023

L: 45;TOTAL: 45 PERIODS

Course Code	CREATIVE DESIGNING USING FIGMA	L	T	P	E	C
23CS31L/ 23IT31L/ 23AD31L	(Common to CSE, IT and AI & DS Degree Programmes)	0	0	0	2	1

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Experiential Component

CO1: Create structured wireframes and interactive UI designs in Figma using UI/UX principles, prototyping, and basic animations (PDL2)

Soft Skill Component

CO2: Demonstrate ethical and user centered decision making in interface design

CO1: Create structured wireframes and interactive UI designs in Figma using UI/UX principles, prototyping, and basic animations (PDL2) **E:30**

1. Create a new Figma file and explore tools like frames, shapes, text, colors, alignment, and layers.
2. Create Lo-Fi wireframes for three screens using basic shapes and text.
3. Develop a simple UI kit with fonts, colors, shapes, buttons, tags, and spacing.
4. Convert the Lo-Fi wireframes into Hi-Fi by applying the UI kit elements.
5. Insert images and edit objects by adjusting strokes, fills, and corner radius and create simple icons.
6. Create a clickable prototype by linking all key screens with tap interactions to

show the main user flow

7. Apply Smart Animate to create smooth transitions for menus, buttons, and card movements

REFERENCE BOOK:

1. Fabio Staiano, "Designing and Prototyping Interfaces with Figma", 1st Edition, Packt> publishing, 2022.

E: 30; TOTAL: 30 PERIODS

Course Code	MINI – CAPSTONE PROJECT	L	T	P	E	C
23CS3ME/ 23IT3ME/ 23AD3ME	(Common to CSE, IT and AI & DS Degree Programmes)	0	0	0	6	3

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Experiential Component

CO1: Apply real-time modeling and optimization techniques to create spatially efficient 3D assets suitable for AR/VR applications. (PDL2)

CO2: Design and prototype immersive user interfaces for AR/VR applications. (PDL2)

CO3: Develop and deploy Augmented Reality mobile application in real-world applications. (PDL2)

CO4: Create and Evaluate Real-time Virtual Reality Scenarios with Environmental Feedback. (PDL3)

Soft Skill Component

CO5: Demonstrate effective communication and Collaboration across Multi-disciplinary teams, integrating 3D Modeling, UI/UX design for AR and VR projects.

Module 1: Real-Time 3D Modeling and Optimization for AR/VR

Introduction to 3D Modeling Techniques:

- Basics of 3D modeling for AR/VR applications.
- Understanding polygon count and optimization strategies.

Optimization Techniques:

- Low-polygon modeling.
- Texture compression and efficient asset management. **E: 15**

Best Practices for Asset Creation:

- Real-time rendering considerations.
- Use of LOD (Level of Detail) and efficient memory usage.

Creating Spatially Efficient 3D Assets:

- Hands-on with creating and optimizing 3D models for AR/VR.

Deliverable: Submission of optimized 3D assets and asset management documentation.

Module 2: Immersive User Interface Design for AR/VR

Principles of AR/VR User Interface Design:

- Understanding the difference between traditional UI and AR/VR UI. **E: 15**
- Interaction techniques for AR/VR (e.g., gaze, gesture, voice).

Prototyping AR/VR Interfaces:

- Tools and software for AR/VR UI prototyping (e.g., Unity, Unreal Engine).

- Design considerations for AR/VR interaction flows.

Usability in AR/VR:

- Accessibility and inclusivity in AR/VR interfaces.
- Immersion and user experience principles.

Deliverable: Submission of a UI prototype, including interaction flows and design documentation.

Module 3: Developing Augmented Reality Mobile Applications

Overview of AR Technologies:

- Introduction to ARCore, ARKit, and Vuforia.
- Understanding the differences in AR platforms.

Integrating 3D Models into AR:

- Steps for importing and positioning 3D models in AR environments.

Real-time Tracking & Interaction: **E: 20**

- Implementing object recognition and tracking in AR.
- Interaction mechanisms: touch, gaze, and gesture.

Deploying AR Applications:

- Testing on AR-capable mobile devices.
- Troubleshooting deployment issues.

Deliverable: Deployment of an AR mobile application with functional real-time interactions.

Module 4: Virtual Reality Scenario Creation with Environmental Feedback

Introduction to VR Technologies:

- Overview of VR platforms and hardware (e.g., Oculus, HTC Vive).
- VR interaction models: hand tracking, room-scale VR, and controllers.

Environmental Feedback in VR:

- Designing haptic feedback, spatial sound, and physical interactions.
- Creating reactive virtual environments based on user actions.

Real-Time Interaction in VR: **E: 20**

- Implementing physics-based interactions and dynamic environmental changes.

Evaluating VR Scenarios:

- Methods for testing and improving the effectiveness of VR feedback.
- Conducting user testing for environmental immersion.

Deliverable: Submission of a VR scenario with environmental feedback, including user feedback analysis.

Module 5: Integration and Final Evaluation

Integrating AR/VR Elements:

- Merging AR and VR systems into cohesive experiences.
- Synchronizing AR/VR applications across platforms.

Final System Evaluation:

- Assessing the effectiveness of the AR/VR system through user testing.
- Performance evaluation and optimization strategies.

Final Presentation and Demo: **E: 20**

- Showcasing the fully developed AR/VR application with functional user interface and environmental feedback.

Deliverable:

- Live demonstration of the AR/VR system.
- Final project report and code repository (GitHub).

To Publish AR/VR application for public access is preferable.

E: 90; TOTAL: 90 PERIODS

Course code		L	T	P	E	C
23CS41E/ 23IT41E/ 23AD41E	**GENERATIVEAI** (Common to CSE, IT and AI & DS degree Programmes)	3	0	0	0	3

COURSE OUTCOMES

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

Theory Components

CO1: understand the concepts of Generative Modeling. (CDL 1)
CO2: apply language models for generating Texts. (CDL 2)
CO3: apply GenAI concepts for generating images. (CDL 2)
CO4: apply variants of GAN models for painting, music and play. (CDL 2)
CO5: apply Prompt engineering concepts using GenAI tools. (CDL 2)

CO1: Understand the concepts of Generative Modeling. (CDL 1)

Overview of Generative modeling - Gen AI Vs Discriminative Modeling–Generative models in AI and ML – Types of Generative models – GANs – Challenges of Generative Modeling – Future of Gen AI. **L: 9**

CO2: Apply language models for generating Texts. (CDL 2)

Language Models Basics – Building blocks of Language models - Transformer Architecture – Encoder and Decoder – Attention mechanisms - Generation of Text – Models like BERT and GPT models - Autoencoding – Exploring ChatGPT – Multimodal LLM – Issues of LLM like hallucination. **L: 9**

CO3: Apply GenAI models for generating images. (CDL 2)

Introduction to Generative Adversarial Networks – Adversarial Training Process – Nash Equilibrium – Variational Autoencoders – Encoder-Decoder Architectures – Transformer-based Image Generation– Visual Transformers ViT- Dall-E2 and Dall-E3, GPT-4V– Issues of Image Generation models like Mode Collapse and Stability. **L: 10**

CO4: Apply variants of GAN models for painting, music and play.(CDL 2)

Variants of GAN – Types of GAN - Cyclic GAN – Using Cyclic GAN to Generate Paintings – Neural Style Transfer – Style Transfer - Music Generating RNN – MuseGAN **L: 8**

CO5: Apply Prompt engineering concepts using GenAI tools. (CDL 2)

Prompt Engineering– Designing Prompts – Revising Prompts using Reinforcement Learning from Human Feedback (RLHF) – Retrieval Augmented Generation. **L: 9**
Open source tools: GPT4All – LangChain – Llama – Hugging Face.

TEXTBOOK

1. Denis Rothman, “Transformers for Natural Language Processing and Computer Vision”, 3rd Edition, Packt Books, 2024.

REFERENCE BOOKS

1. David Foster, “Generative Deep Learning”, O’Reily Books, 2024.
2. Altaf Rehmani, “Generative AI for Everyone”, Blue Rose One, 2024

L: 45; TOTAL: 45 PERIODS

Course code	**AI FOR INDUSTRIAL APPLICATIONS**	**L**	**T**	**P**	**E**	**C**
23CS42E/ 23IT42E/ 23AD42E	(Common to CSE, IT and AI & DS degree Programmes)	**3**	**0**	**0**	**0**	**3**

COURSE OUTCOMES

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

Theory Components

CO1: analyze the concepts and applications of industrial AI (CDL 1)

CO2: analyzethe development of industrial revolutions and its applications(CDL 1)

CO3: analyze and apply digital twin technology for smart manufacturing and other industrial applications. (CDL 1)

CO4: apply Edge Computing concepts in IoT Devices (CDL 2)

CO5: evaluate real-world case studies to explore IoT applications in various industries. (CDL 2)

CO1: analyze the concepts and applications of industrial AI

Introduction: Industrial AI –Scope and challenges -Applications –IMS architecture for industrial AI-Visible and Invisible issues – Building the future with AI-Killer Applications of Industrial AI. **L: 9**

CO2: analyze the development of industrial revolutions and its applications

Introduction: Industrial revolutions 1.0 to 4.0 – Technological advancement in Industry 4.0- Potential Benefits and drawbacks in Industrial 4.0 - Applications of Industry 4.0- Impact on Society - Introduction to Industrial 5.0 – Industrial 4.0 Vs Industrial 5.0. **L: 9**

CO3: analyze and apply digital twin technology for smart manufacturing and other industrial applications.

Data Analytics In Industry4.0: Digital Twins (DT) - Characteristics- Evolution- Data twin - physical world and digital world Classifications-Level of integration – Modelling digital wins- Smart manufacturing and Applications- Digital twins maintenance - predictive maintenance- Planning and operation phase- Hybrid analysis and Fleet data. **L: 9**

CO4: Apply Edge Computing concepts in IoT Devices

Edge Computing: Edge Computing Architecture – IoT Devices – Networking Architecture- Network Management and control - Interfaces and devices- Simulators. **L: 9**

CO5: Evaluate real-world case studies to explore IoT applications in various industries.

Case Studies: Healthcare - Manufacturing Industries - Supply chain Management - Inventory Management - Quality Control. **L: 9**

TEXT BOOKS:

1. Ramin Karim, Diego Galarand Uday Kumar, “AI Factory Theories, Applications and case Studies”, CRC Press, 2023
2. K.Anitha Kumari, G.Sudha Sadasivam, D.Dharani, M.Niranjanamurthy, "Edge Computing Fundamentals, Advances and Applications", CRC Press, 2022

REFERENCE BOOKS:

1. Ella Hassanien, Jyotir Moy Chatterjee and Vishal Jain, “Artificial Intelligence and Industry 4.0”, CRC Press, 2022.
2. Stevan Lawrence Fernandes Tarun K.Sharma “Artificial Intelligence in Industrial Applications”, Springer publication, 2022.

3. Utpal Chakraborty, Amitbanerjee, Jayanta Kumar Saha, Niloy Sarkar, Chinmay Chakraborty, "Artificial Intelligence and the Fourth Industrial Revolution", 2022.

L: 45; TOTAL: 45 PERIODS

Course Code	REINFORCEMENT LEARNING	L	T	P	E	C
23CS43E/ 23IT43E/ 23AD43E	(Common to CSE, IT and AI & DS degree Programmes)	2	0	2	0	3

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Component

CO1:explore the basic concepts of Reinforcement Learning(CDL1)

CO2:illustrate the Markov Decision Process to model reinforcement learning problems.(CDL1)

CO3: apply dynamic programming techniques to solve Monte Carlo problems. (CDL1)

CO4: implement Time Difference learning for real-world problems**.** (CDL1)

Practical Component

CO6: apply the fundamentals concepts of Reinforcement Learning(PDL1)

CO7: design and implement the Markov Decision Process & Monto Carlo techniques for real world application. **(PDL1)**

CO8: implement TD Learning and Q-Learning techniques for real time applications. (PDL1)

CO1: Explore the basic concepts of Reinforcement Learning(CDL1) **L:5; P:6**

Introduction–Elements of Reinforcement Learning–Scope–Agent–Environment Interface– Examples –Need of Reinforcement Learning –Challenges –Multi-Arm Bandit Problem.

CO6: Apply the fundamentals concepts of Reinforcement Learning(PDL1)

Installation of code standards and libraries used in RL(Python/Keras/Tensorflow)

Implement Tic-tac-toe and Armed Bandit Problem.

CO2: Illustrate the Markov Decision Process to model reinforcement learning problems(CDL1) **L:7; P:8**

Overview of Markov Chain-Overview of Markov Decision Process-Model Reinforcement Learning- Problem using MDP–Bellman Equation for value functions–Optimal policy and optimal value functions – Policy Evaluation – Policy Improvement – Policy Iteration – Value Iteration.

CO7: Design and implement the Markov Decision Process & Monto Carlo techniques for real world application. (PDL1)

Find the optimal policy (sequence of actions) that minimizes the number of steps to reach the goal.

Implement a robot control strategy to navigate a maze, collect a diamond, and avoid fire hazards.

CO3: Apply dynamic programming techniques to solve Monte Carlo problems. (CDL1) **L:8; P:8**

Monte Carlo Introduction – Policy Evaluation – Incremental Update – Exploration Vs Exploitation – Policy Improvement – Temporal Differencing Learning – TD

Policy Evaluation – Epsilon-Greedy policy – On-policy Vs Off-policy – Q-Learning – SARSA Learning – Double Q-Learning – Applications of Q-Learning

CO7: Design and implement the Markov Decision Process & Monto Carlo techniques for real world application. (PDL1)

- Implement **Monte Carlo Prediction** to estimate the **state-value function.**
- Implement **On-Policy Monte Carlo Control** to find the **optimal policy** that minimizes the number of steps to the goal and avoids penalties.

CO4: Implement Time Difference learning for real-world problems.(CDL1)

Linear value function approximation–Challenges of Large-scale MDP–Value Function approximations–Stochastic Gradient Descent–Linear value and non-linear value approximation–Naïve Deep-Q Learning–Experience Replay-Advanced Deep RL: Actor-Critic Methods, AlphaZero **L:10; P:8**

CO8: Implement TD Learning and Q-Learning techniques for real time applications.(PDL1)

- Implement a **TD-learning based agent** (e.g., Q-learning or SARSA) to control the robotic arm and learn the **best sequence of actions** to complete the pick-and-place task.
- Implement a **temporal-difference (TD) learning agent** to play Connect Four and learn the **optimal policy**

Text BOOKS:

1. Maxim Lapan; "Deep Reinforcement Learning Hands-On: Apply modern RL algorithms to practical problems using Python and PyTorch", 2nd Edition, Packt Publishing, 2024.
2. Dimitri P. Bertsekas; "Reinforcement Learning and Optimal Control"; 1st Edition, Athena Scientific, 2024.

REFERENCE BOOKS:

1. Laura Graesser and Wah Loon Keng, "Foundations of Deep Reinforcement learning: theory and Practice in Python", Pearson India, 2022.
2. Richard S. Sutton and Andrew G. Barto; "Reinforcement Learning: Introduction"; 2nd Edition, MIT Press, 2020.

L: 30; P:30;TOTAL: 60 PERIODS

Course code	AI FOR ROBOTICS	L	T	P	E	C
23CS44E/ 23IT44E/ 23AD43E	(Common to CSE, IT and AI & DS degree Programmes)	3	0	0	0	3

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Components

CO1: utilize the fundamentals of AI-driven robotics and its applications. (CDL1)
CO2: apply machine learning and computer vision techniques for robotic perception. (CDL1)
CO3: analyze and design intelligent motion and path planning algorithms. (CDL1)
CO4: apply modern AI tools for developing basic robotic solutions. (CDL2)

CO1: utilize the fundamentals of AI-driven robotics and its applications. (CDL1)

Overview of Robotics: components, types, and applications-basics of robot kinematics and dynamics- sensors and actuators overview -robot perception–path and motion planning–action loop- Evolution of AI in Robotics - Intelligent agent architectures for autonomous robots- ROS middleware for robotic control and communication. Robot components & sensor visualization using OpenRobotSimulator/ Gazebo/Webots- Ethics of building Intelligent Robots. **L:11**

CO2: apply machine learning and computer vision techniques for robotic perception. (CDL1)

Fundamentals of Feature extraction for robotic perception-robotic vision using deep learning-behavior cloning and imitation learning-Reinforcement learning for control and navigation-Overview of 3D perception: LiDAR and depth cameras-Visual SLAM - LLMs/VLMs for multimodal perception - case study: camera-based object detection simulation using Webots. **L:15**

CO3: analyze and design intelligent motion and path planning algorithms.(CDL1)

Fundamentals of control theory: PID control, open/closed-loop control systems, feedback and stability- Robot dynamics in control and navigation- Motion planning algorithms: D* and RRT*-Path optimization & obstacle avoidance- Case studies: Path tracking and intelligent navigation using simulation. **L:10**

CO4: apply modern AI tools for developing basic robotic solutions. (CDL 2)

Human-Robot communication using NLP-Cognitive robotics -Industrial Applications- Emerging Trends: Humanoid Robots-RoboGPT- Gen AI in robotics. **L:9**

TEXT BOOKS

1. Azar, A. T., & Koubaa, A, "Artificial Intelligence for Robotics and Autonomous Systems Applications", 1st Edition, Springer, 2023.
2. Imran, A., & Gopalakrishnan, K, "AI for Robotics: Toward Embodied and General Intelligence in the Physical World", 1st Edition, Apress, 2025.

REFERENCE BOOKS

1. Martinez, A., & Fernández, E. "Artificial intelligence for robotics: Build intelligent robots using ROS 2, Python, OpenCV, and AI/ML techniques for real-world tasks", 2nd Edition, Packt Publishing, 2024.
2. Robert R Murphy,Introduction to AI Robotics, 2nd Edition, MIT Press,2019.
3. Siciliano, B., Sciavicco, L., Villani, L., & Oriolo, G, "Robotics: Modeling, Planning and Control", 1st Edition, Springer, 2020.
4. Russell, S. J., & Norvig, P, "Artificial Intelligence: A Modern Approach", 4th Edition, Pearson, 2021.
5. Francis X. Govers III, Artificial Intelligence for Robotics: Build Intelligent Robots Using ROS 2, Python, OpenCV, and AI/ML Techniques for Real-world Tasks, 2nd Edition, Packt Publishing, 2024.

L:45;TOTAL:45PERIODS

Course Code	ETHICAL AI	L	T	P	E	C
23CS46E/ 23IT46E/ 23AD46E	(Common to CSE, IT and AI & DS degree Programmes)	3	0	0	0	3

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Components

CO1: explore the foundational concepts of ethics related to AI.**(CDL1)**

CO2: apply the principles of trust and fairness in AI systems with real-world examples.**(CDL1)**

CO3: analyze various risks associated with AI and strategies for managing risks. **(CDL 1)**

CO4: explore the challenges and opportunities in AIethics, with a focus on societal issues and use cases. **(CDL1)**.

CO1: Explore the foundational concepts of ethics related to AI.(CDL1) **L:10**

Introduction to AI - Descriptive Ethics - Normative Ethics- Meta ethics - Applied ethics - Ethics Vs Law - Introduction to Machine Ethics and Robot Ethics- Ethical dilemmas in AI: Trolley problem and Beyond - Importance of ethics in AI design and deployment.

CO2: Apply theprinciples of trust andfairnessin AIsystemswith real-world examples.(CDL1) **L:10**

User Acceptance and Trust - Functional Elements of Trust - Ethical Principles for Trustworthy and Fair AI - Responsibility and Liability in AI Systems: Use cases - Strict Liability - Complex Liability- Consequences of Liability.

CO3: Analyze various risks associated with AI and strategies for managing risks. (CDL 1) **L:10**

Business Ethics for AI Companies - General Business Risks - Ethical Risks - Managing Risk - Risks of AI to Workers- Psychological Aspects of AI: Problems of Anthropomorphisation - Persuasive AI –Privacy Issues-Data Collection and its Dangers.

CO4: Explore the challenges and opportunities in AI ethics, with a focus on societal issues and use cases. (CDL 1) **L:15**

Challenges and opportunities in implementing ethical AI - Ethical issues in AI across domains- Societal implications: AI in healthcare, education and environment - AI in banking - Judicial AI - Ethical concerns in cyber security: ethical hacking, surveillance, facial recognition - National and international AI strategies and governance policies-Role of AI in industrial decision making and future trends.

TEXT BOOKS

1. Sean Welsh, Alan R.Wagner, Christoph Lutge, Christoph Bartneck, “An Introduction to Ethics in Robotics and AI”, Springer Briefs in Ethics, 2019.
2. Mark Coeckelberg, “AI Ethics”, The MIT Press Essential Knowledge series, 2020.

REFERNCES

1. S.Matthew Liao, “Ethics of Artificial Intelligence”, Oxford University Press, 2020.
2. Eleanor Bird, Jasmin Fox-Skelly, Nicola Jenner, Ruth Larbey, Emma Weitkamp and Alan Winfield, “The ethics of artificial intelligence: Issues and initiatives”, Springer, 2020.

L:45;TOTAL:45 PERIODS

Course Code		L	T	P	E	C
23CS47E/ 23IT47E/ 23AD47E	**RESPONSIBLE AI**	3	0	0	0	3
	(Common to CSE, IT and AI & DS degree Programmes)					

COURSE OUTCOMES

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

Theory Component

CO1: employ the fundamentals of Responsible AI and its principles.(CDL 1)

CO2: apply bias detection and fairness techniques in responsible AI systems. (CDL 2)

CO3: analyse explainable AI methods in real time applications.(CDL 2)

CO4: apply privacy-preserving and security techniques to improve the safety and resilience of AI systems.(CDL 2)

CO1: employ the fundamentals of Responsible AI and its principles.(CDL 1) **L:9**

Overview of Responsible AI – Characteristics – Principles of responsible AI – challenges in implementation – ELSI Framework and AI – Safety and Alignment – Fairness and Privacy.

CO2: apply bias detection and fairness techniques in responsible AI systems. (CDL 2) **L:12**

Human Bias – Types of Bias in AI – Effects of biases on different demographics – Bias vs Fairness – Sources of Biases – Bias Mitigation Techniques – Bias detection tools – Overview of fairness in AI – Demographics parity – Group fairness and Individual fairness – Counterfactual fairness – Case studies: COMPAS Algorithm – Photos Tagging Controversy – Recidivism Predictions.

CO3: analyse explainable AI methods in real time applications (CDL 2) **L:12**

Explainability and Interpretability – Challenges – Visualization techniques for interpretation – Intrinsic interpretable methods – Interpretability Evaluation methods – Explainability through causality –
Model agnostic interpretation – LIME(Local Interpretable Model-agnostic Explanations) – SHAP(Shapley Additive Explanations) – AI Recruiting tool – Case study: Facial Recognition.

CO4: apply privacy-preserving and security techniques to improve the safety and resilience of AI systems.(CDL 2) **L:12**

Overview of privacy – security – safety – resilience – Taxonomy of AI safety and security – Adversarial attacks and mitigation – Model and data security – Model drift – Data drift – Concept drift – Privacy preserving AI techniques – Case studies using AI tools (Claude): Health care – Autopilot and Ethical Implications of Autonomous Vehicles.

TEXT BOOKS

1. Adrian Massood, Heather Dawe, "Responsible AI in the Enterprise", O'Reilly,1st Edition 2023.
2. Beena Ammanath, "Trustworthy AI", Wiley, 1st Edition 2022.

REFERENCES

1. Christoph Molnar, "Interpretable Machine Learning", Wiley, 2nd edition, 2022.

2. Virginia Dignum, "Responsible Artificial Intelligence: How to Develop and Use AI in a Responsible Way", Kindle, 1st Edition 2019.

L: 45;TOTAL: 45 PERIODS

Course Code	AI MODEL DEPLOYMENT USING MLOps	L	T	P	E	C
23CS49E/ 23IT49E/ 23AD49E	(Common to CSE, IT and AI & DS degree Programmes)	2	0	2	0	3

COURSE OUTCOMES:

Theory Components

Upon completion of this course, the students will be able to

CO1: apply the fundamentals of MLOps. (CDL1)

CO2: employ the stages and lifecycle of the machine learning pipeline in MLOps.(CDL1)

CO3: apply CRUD operations using pipelines and MLflow to track and manage machine learning experiments**.**(CDL1)

CO4: integrate and manage machine learning models using cloud services. (CDL2)

Practical Components

CO5: perform MLOps life cycle, ML pipeline using tools.(PDL1)

CO6: design end to end automated MLOps pipeline for real time applications and deploy ML models using modern tools.(PDL2)

CO1: apply the fundamentals of MLOps. (CDL1) **L:5; P:4**

MLOps Introduction-Different Roles of MLOps-MLOps Architecture Overview-MLOps Lifecycle Phases-open source MLOps Tools for MLLC- MLOps Vs DevOps

CO5: perform MLOps life cycle, ML pipeline using tools.(PDL1)

- Explore and implement MLOps concepts and MLOps Life cycle

CO2: employ the stages and lifecycle of the Machine Learning pipeline in MLOps.(CDL1) **L:8; P:6**

Overview of the ML Pipeline-Data Ingestion and Validation-Model Training and Validation-Model Packaging and Versioning-Types of pipeline-Model Deployment and Monitoring Lifecycle-Pipeline Automation Concepts-Introduction to CI/CD in MLOps

CO5: perform MLOps life cycle, ML pipeline using tools.(PDL1)

- Build and execute the ML pipeline: data ingestion, validation, training, packaging and monitoring.

CO3: apply CRUD operations using pipelines and MLflow to track and manage machine learning experiments.(CDL1) **L:8; P:8**

Integrating SQL Data into Machine Learning Pipelines - Introduction to MLflow - Components - Key Concepts in MLflow - Installing MLflow - Tracking ML Experiments using MLflow – Case Study: End-to-End ML Pipeline Integrated with SQL/NoSQL Data Sources

CO6: design end to end automated MLOps pipeline for real time applications and deploy ML models using modern tools.(PDL2)

- Install and explore MLflow, Track experiments and metrics.
- Manage ML model lifecycle using MLflow components.

CO4: integrate and manage machine learning models using cloud services. (CDL2) **L:9;**
Introduction to Cloud Services DevOps for Machine Learning Operations with Cloud Service - MLOps with AWS - MLOps with GCP **P:12**

CO6: design end to end automated MLOps pipeline for real time applications and deploy ML models using modern tools.(PDL2)

- Deploy Machine Learning models in cloud platforms.
- Develop a simple project for a real time application using cloud service.
- Demonstrate CI/CD concepts for model deployment.

L:30;P:30; TOTAL: 60 PERIODS

TEXT BOOKS

1. Mark Treveil and the DATAIKU team " Introducing MLOps: How to scale Machine Learning in the Enterprise" Shroff/O'Reilly, 1st Edition, 2020.
2. William Smith, MLflow for Machine Learning Operations: The Complete Guide for Developers and Engineers**,** HiTeX Press, 2025.

REFERENCES

1. Noah Gift and Alfredo Deza "Practical MLOps: Operationalizing Machine Learning Models", O'Reilly Media, Second Edition, 2023
2. Emmanuel Raj, Engineering MLOps, Packt Publishing, 1st Edition, 2024.

Course Code	BIO INFORMATICS	L	T	P	E	C
23CS4AE/ 23IT4AE/ 23AD4AE	(Common to CSE, IT and AI & DS degree Programmes)	3	0	0	0	3

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Components

CO1: recognize the fundamentals of molecular structure. (CDL1)
CO2: apply various data handling techniques for biological databases. (CDL2)
CO3: evaluate and predict the structures of genes, proteins, and RNA. (CDL1)
CO4: analyze the design, testing, and discovery of medications. (CDL 2)
CO5: apply deep learning models in the bioinformatics domain (CDL 2)

CO1: recognize the fundamentals of molecular structure. (CDL1) **L:9**

Molecules and super-molecules structure, DNA and RNA structures, Proteins: Amino acids, Protein folding and interaction, protein structure determination, Polysaccharides, Lipids, Genomics: DNA Sequencing, Gene Identification methods, protein identification, Protein microarrays.

CO2: apply various data handling techniques for biological databases (CDL2) **L: 9**

Biological Database: Introduction, Databases: sequence, molecular visualization, Genome mapping database, GENBANK: Flatfile, Pairwise alignment, sequence alignment, progressive alignment, database similarity searching, working with FASTA, working with BLAST, comparison of FASTA and BLAST.

CO3: evaluate and predict the structures of genes, proteins, and RNA. (CDL1) **L:9**

Gene sequencing –Motif and Motif findings- gene pattern recognition, gene prediction using bioinformatics tools, DNA Microarrays, RNA prediction methods, Protein folding problem, Protein structure prediction methods.

CO4: analyze the design, testing, and discovery of medications. (CDL 2) **L: 9**

Drug discovery: introduction, parameters, technologies, drug identification strategy, drug target validation, validation of targets, Drug Design: Biomarkers: classification, combinatorial biomarkers, biomarkers in drug development, drug identification, computer-aided drug design.

CO5: apply deep learning models in the bioinformatics domain (CDL 2) **L: 9**

Convolutional neural networks for bioinformatics, recurrent neural networks (RNN) for bioinformatics, Long short term memory (LSTM) networks in bioinformatics, Case studies using Python libraries.

TEXT BOOKS:

1. SC Rastogi, Parag Rastogi, and Namita Mendiratta "Bioinformatics Applications, Genomics, Proteomics and Drug Discovery", 5th Edition, PHI, 2022.
2. Habib Izadkhah, "Deep Learning in Bioinformatics', 1st Edition, Elsevier, 2022.

REFERENCE:

1. Faheem Masoodi, Mohammad Quasim, Syed Bukhari, Sarvottam Dixit, Shadab Alam "Applications of Machine Learning and Deep Learning on Biological Data", CRC Press, 2023.

L: 45; TOTAL: 45 PERIODS

Course Code	INTELLIGENT DASHBOARD DEVELOPMENT USING MODERN TOOLS	L	T	P	E	C
23CS42L/ 23IT42L/ 23AD42L	(Common to CSE, IT and AI & DS degree Programmes)	**0**	**0**	**2**	**0**	**1**

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Practical Components

CO1: design and develop interactive dashboards using modern tools. (PDL2)

CO2: integrate real time data sources, APIs, for dynamic and live dashboards (PDL2).

CO1: design and develop interactive dashboards using modern tools. (PDL2) **P: 12**

Data Modeling Concepts

- Data Modeling
- Manage Relationship
- Cardinality
- Dashboard Options

Dashboard & Group Creation

- Report Visualizations
- Map Visualizations
- Miscellaneous Charts

- Custom Visuals

CO2: integrate real time data sources, APIs, for dynamic and live dashboards P: 18 (PDL2)

1. Implementation of connectivity in dashboards with databases (SQL/NoSQL)
2. Implementation of REST APIs, JSON-based datasets, and streaming multiple data sources
3. Develop & deploy the real-time intelligent dashboards (anomaly detection, NLP, predictive dashboards, etc.) with streaming data/live data

TOOLS REQUIREMENTS

Software: Tableau/ Microsoft PowerBI/Streamlit/Looker Studio, Apache superset, Cursor IDE etc.,

P: 30; TOTAL: 30 PERIODS

Course Code	MINI CAPSTONE PROJECT	L	T	P	E	C
23CS4ME/ 23IT4ME/ 23AD4ME	(Common to CSE, IT and AI & DS degree Programmes)	**0**	**0**	**0**	**6**	**3**

COURSE OUTCOMES

Upon the completion of this course, the students will be able to

Experiential Component :

CO1: formulate problem statements in the Artificial Intelligence domain, perform a feasibility study and design a suitable AI-based solution. (PDL1)

CO2: develop a functional prototype using the forward engineering process. (PDL2)

Soft skill Component:

CO3: demonstrate effective collaboration with peers by exhibiting strong communication, teamwork and problem-solving abilities to improve employability.

The project work shall be undertaken either individually or in a team comprising a maximum of four students, In the case of team projects, each student must submit an individual project report clearly indicating their specific contributions to the work. The project assessment will be based on the presentation, demonstration, and quality of the report. The project progress will be evaluated through three periodic reviews:

- Review I: Evaluation of problem identification, literature survey, and feasibility analysis of the proposed project.
- Review II: Assessment of design methodology, model/framework/prototype development.
- Review III: Final evaluation involving demonstration of the developed prototype/product and report submission.

E: 90; TOTAL: 90 PERIODS

Course Code	SECURITY AND PRIVACY IN IOT	L	T	P	E	C
23CS54E / 23IT54E / 23AD54E	(Common to CSE, IT, AI&DS Degree Programmes)	**2**	**0**	**2**	**0**	**3**

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Components

CO1: utilize IoT architectures, technology stack, and cryptographic design principles for secure IoT systems. (CDL1)

CO2: analyze communication and hardware-based threats in IoT devices, including side- channel and fault attacks. (CDL2)

CO3: evaluate hardware security primitives (PUFs, TRNGs, Trojans) for ensuring trust in IoT hardware. (CDL2)

CO4: apply authentication, attestation, and IoT-specific security protocols in real-world scenarios. (CDL2)

Practical Components

CO5: Design secure IoT networks, monitor traffic using Wireshark and perform vulnerability assessment. (PDL1)

CO6: implement cryptographic primitives, authentication protocols and IoT security mechanisms using AI/ML tools. (PDL1)

CO1: utilize IoT architectures, technology stack, and cryptographic design principles for secure IoT systems. **L:7; P:6**

Overview of IoT – IoT Security Architecture – IoT Communication Models – IoT Protocol Stack – Common IoT Vulnerabilities – Role of Hardware in IoT Security – Secure Boot and Firmware Update – Cryptographic Design for IoT: AES, PRESENT, SIMON, ECC – Lightweight Security Design Principles.

CO5: Design secure IoT networks, monitor traffic using Wireshark and perform vulnerability assessment.(PDL1)

Configure secure communication between IoT nodes (ESP32/Raspberry Pi) - Capture encrypted MQTT/CoAP packets using Wireshark - Identify anomalies in IoT traffic using packet inspection

CO2: analyze communication and hardware-based threats in IoT devices, including side-channel and fault attacks. **L:8; P:8**

Hardware Attack Surfaces – Debug and Communication Interfaces – Reverse Engineering of IoT Devices – Firmware Vulnerabilities – Bus Sniffing and Glitching –Hardware attacks: glitching, fault injection, side-channel attacks - Countermeasures and secure firmware development - IoT vulnerability analysis tools

CO5: Design secure IoT networks, monitor traffic using Wireshark and perform vulnerability assessment.(PDL1)

Network traffic inspection using Wireshark (HTTP/MQTT anomalies) - Perform vulnerability assessment using OpenVAS - AI/ML-based anomaly detection in IoT communication

CO3: evaluate hardware security primitives (PUFs, TRNGs, Trojans) for ensuring trust in IoT hardware. **L:8; P:8**

Physically Unclonable Functions (PUFs): Design Principles, Types, Machine Learning Attacks – True Random Number Generators (TRNG): Design Principles, NIST and AIS Tests, Attacks on TRNG – Hardware Trojans: Impact on IoT supply chain, Design

Methodologies, Detection Techniques.

CO6: implement cryptographic primitives, authentication protocols and IoT security mechanisms using AI/ML tools. (PDL1)

Implement PUF-based authentication on IoT devices - Validate TRNG randomness using Python tests - Use ML techniques to detect hardware-level anomalies

CO4: apply authentication, attestation, and IoT-specific security protocols in real-world use cases. **L:7; P:8**

IoT authentication requirements - WPA2 handshake vulnerabilities - TLS protocol and Logjam attack - PUF-based authentication and bit commitment - Remote attestation: Hardware/Software RoT - Lightweight authentication & privacy methods - End-to-end IoT security

CO6: implement cryptographic primitives, authentication protocols and IoT security mechanisms using AI/ML tools. (PDL1)

WPA2 handshake viewing using Aircrack-ng (capture-only mode) - TLS session analysis using Wireshark - Implement lightweight cryptography (AES-Lite, PRESENT) - Build privacy-preserving IoT pipeline (secure aggregation) - Integrate end-to-end IoT security with cloud

TEXT BOOKS

1. Debdeep Mukhopadhyay, Rajat Subhra Chakraborty, Hardware Security: Design, Threats, and Safeguards, Chapman & Hall/CRC, 2019.
2. Arshdeep Bahga, Vijay Madisetti, Internet of Things: A Hands-On Approach, Universities Press, 2019.

REFERENCES

1. Kasem Khalil, Haytham Idriss, Tarek Idriss, Magdy Bayoumi, Lightweight Hardware Security and Physically Unclonable Functions: Improving Security of Constrained IoT Devices, Springer, 2025.
2. Fei Hu, Security and Privacy in Internet of Things (IoTs): Models, Algorithms, and Implementations, CRC Press, 2016.
3. N. Pundir, N. Vashistha, M. Tehranipoor, F. Farahmandi, Hardware Security Primitives, Springer, 2023.
4. Jakub Szefer, Principles of Secure Processor Architecture Design, Morgan & Claypool / Springer, 2018.

Hardware Requirements

1. ESP32 microcontroller boards (minimum 2 units)
2. Raspberry Pi (optional for cloud/TLS/ML experiments)
3. WiFi Router / Access Point
4. USB WiFi Adapter with monitor mode (for WPA2 capture)
5. Sensors (Temperature/Humidity – optional)
6. Breadboard, jumper wires, USB cables

Software Requirements

1. Arduino IDE / ESP-IDF
2. Python with ML libraries (NumPy, Pandas, Scikit-learn, TensorFlow Lite)
3. Mosquitto MQTT Broker

4. Wireshark (network traffic analysis)
5. Aircrack-ng (WPA2 handshake capture – capture only)
6. OpenVAS (vulnerability assessment)
7. Lightweight cryptography libraries (AES, PRESENT, SIMON, ECC)
8. CoAP/MQTT client tools

L:30;P: 30; TOTAL: 60PERIODS

Course Code	SOCIAL NETWORK SECURITY	L	T	P	E	C
23CS55E / 23IT55E / 23AD55E	(Common to CSE, IT, AI&DS Degree Programmes)	**3**	**0**	**0**	**0**	**3**

COURSEOUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Components

CO1: recognize security challenges and vulnerabilities in social networking platforms. (CDL1)
CO2: apply AI/ML techniques for behavioral analysis and user profiling in social network security. (CDL2)
CO3: detect and prevent cyber threats using AI/ML approaches, including phishing attacks and malware detection.(CDL2)
CO4: develop strategies for securing cloud-based and IoT-integrated social networks. (CDL2)
CO5: design access control mechanisms and ensure compliance with privacy regulations. (CDL2)

CO1: Recognize security challenges and vulnerabilities in social networking platforms. (CDL1) **L:9**

Introduction to Social networks. Types of Social media, Social media platforms, Social media monitoring, Hashtag, Viral content, Social media marketing, Social media privacy, Challenges, opportunities and pitfalls in online social network, Security issues related to social media.

CO2: Apply AI/ML techniques for behavioral analysis and user profiling in social network security.(CDL2) **L:8**

AI/ML for behavioral analysis in social networks – User behavior profiling – Anomaly detection – Fraud detection techniques – Hands-on: K-Means clustering for user behavior, Naive Bayes for email spam detection.

CO3: Detect and prevent cyber threats using AI/ML approaches, including phishing attacks and malware detection. (CDL2) **L:10**

Cyber threat detection using AI/ML – Phishing attack detection with Logistic Regression and Random Forest – Malware classification using SVM – URL threat detection using CNN & NLP – Intrusion detection using Deep Learning (LSTM, Autoencoders).

CO4: Develop strategies for securing cloud-based and IoT-integrated social networks. (CDL2) **L:10**

Cloud security challenges in social platforms–IoT security: Securing device eco systems and dataflow–Integration of cloud and IoT security: End-to-end protectionstrategies Hands-on Applications: Secure Cloud File Storage with AES Encryption - Simulating IoT Attacks and Detection Using Wireshark

CO5: Design access control mechanisms and ensure compliance with privacy regulations. (CDL2) **L:8**

Role-Based and Attribute-Based Access Control in social networks – Fine-grained access policies – Privacy-Enhancing Technologies: Data anonymization – pseudonymization - encryption to safeguard user data – Legal Frameworks and Compliance: GDPR - CCPA – DPDP.

TEXT BOOKS

1. William Stallings, Network Security Essentials: Applications and Standards, 6th Edition, Pearson, 2020.
2. B.A.Forouzan, Cryptography and Network Security, 3rd Edition, Tata McGraw-Hill, 2016.

REFERENCES

1. Soma Halder & Sinan Ozdemir, Hands-On Machine Learning for Cybersecurity, Packt Publishing, 2018.
2. Mark Stamp, Information Security: Principles and Practice, 3rd Edition, Wiley, 2021.
3. Vishal Chopra, Phishing and Countermeasures: Understanding the Threats and Defending Against Attacks, Springer, 2022.
4. Nina Godbole & Sunit Belapure, Cyber Security: Understanding Cyber Crimes, Computer Forensics and Legal Perspectives, Wiley, 2011.
5. Chuanlong Xie, Haibin Zhu, Machine Learning and Security, O'Reilly, 2018.
6. Paul Bradley, Social Engineering: The Art of Human Hacking, Wiley, 2011.

L:45; TOTAL: 45 PERIODS

Course Code	CYBER SECURITY AND ETHICAL HACKING	L	T	P	E	C
23CS56E / 23IT55E / 23AD55E	(Common to CSE, IT, AI&DS Degree Programmes)	2	0	2	0	3

COURSEOUTCOMES

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

Theory Component

CO1: explain the fundamentals of cyber security, including threats, terminologies, and legal frameworks. (CDL1)

CO2: identify system and network vulnerabilities using ethical hacking methodologies. (CDL1)

CO3: apply security measures to mitigate cyber threats such as MITM attacks, session hijacking, and remote exploitation. (CDL2)

CO4: apply exploitation techniques and mitigation strategies for web, wireless, and mobile platform attacks. (CDL2)

Practical Component

CO5: implement fundamental cyber security practices using industry-standard tools like Kali Linux, Wireshark, and Ettercap. (PDL1)

CO6: conduct ethical hacking tasks using reconnaissance and exploitation frameworks such as Nmap, Metasploit, and Burp Suite. (PDL1)

CO1: Explain the fundamentals of cyber security, including threats, terminologies, and legal frameworks.(CDL1)

Introduction to Cyber Security: Evolution, importance, and impact of cyber security in modern systems - Threats & Vulnerabilities: Definition, types of cyber threats - Cyber Security Terminologies: Key concepts including CIA triad - Cybercrime & Legal Frameworks: Classification of cybercrimes, types of cybercriminals, global perspectives on cyber security laws - Cyber Laws & IT Act: Overview of cyber laws, Indian IT Act, cybercrime punishments. **L:6; P:3**

CO5: Implement fundamental cyber security practices using industry-standard tools like Kali Linux, Wireshark, and Ettercap.(PDL1)

Setting up Kali Linux, Password Cracking: Crackstation, hash identification techniques, Google Dorking: Using Google Hacking Database (GHDB) for reconnaissance

CO2: Identify system and network vulnerabilities using ethical hacking methodologies.(CDL1)

Introduction to Ethical Hacking: Concepts, objectives, and methodologies in penetration testing - Footprinting& Reconnaissance: Gathering intelligence, hacker categories, social engineering techniques - Scanning & Enumeration: Port scanning, network scanning, vulnerability assessment methods - Exploitation Techniques: Banner grabbing, OS fingerprinting, scanning methodologies (Nmap, Ping Sweep). **L:6; P: 9**

CO6: Conduct ethical hacking tasks using reconnaissance and exploitation frameworks such as Nmap, Metasploit, and Burp Suite.(PDL1)

Reconnaissance Tools: Amass for OS fingerprinting, NESSUS for vulnerability assessment, Exploitation Tools: Metasploit framework for vulnerability exploitation.

CO3: Apply security measures to mitigate cyber threats such as MITM attacks, session hijacking, and remote exploitation.(CDL2)

Network Security Measures: Network sniffing (promiscuous vs. non-promiscuous modes), intrusion detection methods - MITM Attacks & Countermeasures: ARP spoofing, MAC flooding, SSL stripping, DNS poisoning, protective measures - Session Hijacking & DoS Attacks: TCP session hijacking, denial-of-service attack types, mitigation strategies - Remote Exploitation: SMTP and SQL vulnerabilities, securing servers against common exploits. **L:6; P:6**

CO5: Implement fundamental cyber security practices using industry-standard tools like Kali Linux, Wireshark, and Ettercap. (PDL1)

Network Analysis: Wireshark for packet inspection, Ettercap for MITM attacks.

CO4: Apply exploitation techniques and mitigation strategies for web, wireless, and mobile platform attacks.(CDL2)

Web Security: Web server and web application hacking (SQL injection, parameter tampering, broken authentication) - Wireless Security Threats: Wi-Fi encryption, WPA/WEP cracking, rogue access point detection - Mobile Platform Exploitation: Mobile malware, app vulnerabilities, Android and iOS security models - Mitigation Strategies: Intrusion prevention, secure software development practices, encryption methods. **L:6; P:6**

CO6: Conduct ethical hacking tasks using reconnaissance and exploitation frameworks such as Nmap, Metasploit, and Burp Suite. (PDL1)

Web Application Security: Nuclei scanner for automated security assessments, Burp Suite for web application penetration testing.

TEXT BOOKS:

1. Dr. Erdal Ozkaya, "Cybersecurity: The Beginner's Guide", Packt Publishing, 2020.
2. Dafydd Stuttard & Marcus Pinto, "The Web Application Hacker's Handbook: Finding and Exploiting Security Flaws ", John Wiley & Sons, 2nd Edition, 2020.
3. Georgia Weidman, "Penetration Testing: A Hands-On Introduction to Hacking", No Starch Press, 2024.

REFERENCES:

1. David Kennedy, Jim O'Gorman, Devon Kearns, Mati Aharoni, "Metasploit: The Penetration Tester's Guide", No Starch Press, 2023.
2. Allen Harper, Shon Harris, "Gray Hat Hacking: The Ethical Hacker's Handbook", 6th Edition, McGraw Hill, 2022.
3. Yuri Diogenes & Erdal Ozkaya, "Cybersecurity – Attack and Defense Strategies", 3rd Edition, Packt Publishing, 2022.
4. Donovan, Martin, "Hacking: Learn the Basics of Ethical Hacking and Penetration Testing", 2019.
5. Jon Erickson, "Hacking: The Art of Exploitation", 2nd Edition, No Starch Press, 2020.

WEB REFERENCES

- https://owasp.org/
- https://www.kali.org/docs/
- https://attack.mitre.org/
- https://www.sans.org/security-resources/
- https://www.eccouncil.org/train-certify/certified-ethical-hacker-ceh/

L:30;P:30; TOTAL: 60PERIODS

Course Code	FIREWALL AND INTRUSION DETECTION SYSTEMS	L	T	P	E	C
23CS57E / 23IT57E / 23AD57E	(Common to CSE, IT, AI&DS Degree Programmes)	3	0	0	0	3

COURSE OUTCOMES

Upon completion of this course, the student will be able to

CO1: relate firewall roles, types, architectures, and their integration with secure communication protocols such as VPN and SET. (CDL1)

CO2: analyze NGFW evolution, architecture, deployment, and effectiveness in mitigating modern network threats. (CDL2)

CO3: identify IDS/IPS functions, architectures, and alert systems for cooperative detection mechanisms (CDL1)

CO4: apply IDS concepts and techniques to identify and classify network threats. (CDL2)

CO5: analyze wireless intrusion threats and apply detection, prevention, and risk assessment techniques in WLANs. (CDL2)

CO1: Relate firewall roles, types, architectures and their integration with secure communication protocols such as VPN and SET. (CDL1) **L:9**

Roles and functions of firewalls in network security - Key terminologies: DMZ, stateful and stateless inspection - Types of firewalls - Firewall design principles and deployment strategies - Virtual Private Networks (VPN): architecture, tunneling protocols - Secure Electronic Transaction (SET) for e-commerce security - Configure basic firewall rules using pfSense

CO2: Analyze NGFW evolution, architecture, deployment and effectiveness in mitigating modern network threats. (CDL2) **L:8**

Evolution of firewalls: from traditional to NGFW - Need for NGFWs in modern threat landscapes - Characteristics and capabilities of NGFWs - Architecture and components of NGFWs - Open-source firewall tools: IPFire, OPNsense, Untangle - Deploy and test NGFW features using open-source tools - Analysis of Institutional Firewall layout

CO3: Identify IDS/IPS functions, architectures, and alert systems for cooperative detection mechanisms (CDL1) **L:9**

IDS vs IPS: functional differences and integration - Tiered architecture of IDS/IPS - Key components - Information flow and alert correlation - Defending IDS/IPS against evasion techniques - Cooperative intrusion detection systems - Case Study: Data exfiltration detected by Snort IDS

CO4: Apply IDS concepts and techniques to identify and classify network threats. (CDL2) **L:9**

Introduction to intrusion and threat detection - Need for IDS in layered security architecture
- IDS components and architecture - Classification of IDS - Information sources for IDS - Vulnerabilities and countermeasures - Intrusion detection techniques - Analyze alerts using Suricata

CO5: Analyze wireless intrusion threats and apply detection, prevention and risk assessment techniques in WLANs. (CDL2) **L:10**

Wireless threat landscape: WLAN vulnerabilities and attack vectors - 802.11 infrastructure attacks, WEP weaknesses, rogue access points - Mobile device threats - Detection and prevention techniques for wireless threats - Threat briefing and risk quantification - Return on Investment (ROI) in wireless security - Use Kismet/ Wireshark to detect wireless intrusions

TEXT BOOKS

a. Al-Sakib Khan Pathan, "The State of the Art in Intrusion Prevention and Detection", First Edition, CRC Press, 2024.
b. Chris Sanders and Jason Smith, "Applied Network Security Monitoring Collection, Detection, and Analysis", First Edition, Syngress, 2024.

REFERENCES

1. Ali A.Ghorbani, Wei Lu, "Network Intrusion Detection and Prevention: Concepts and Techniques", First Edition, Springer US, 2019.
2. Earl Carter, Jonathan Hogue, "Intrusion Prevention Fundamentals", First Edition, Pearson Education, 2020.
3. Elizabeth D. Zwicky, Simon Cooper & D. Brent Chapman, "Building Internet Firewalls", O'Reilly, 2019.
4. William Stallings, Network Security Essentials: Applications and Standards, 6th Edition, Global Edition, Pearson, 2017.

L:45; TOTAL: 45 PERIODS

Course Code	THREAT INTELLIGENCE AND RISK MANAGEMENT	L	T	P	E	C
23CS58E / 23IT58E / 23AD58E	(Common to CSE, IT, AI&DS Degree Programmes)	**3**	**0**	**0**	**0**	**3**

COURSE OUTCOMES

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

CO1: utilize the concepts, processes, and lifecycle of threat intelligence for cyber protection. (CDL1)

CO2: analyze cyber threats, attack vectors, and adversary tactics using standard threat-analysis frameworks.(CDL2)

CO3: apply threat intelligence sources, tools, and data collection methods to support cybersecurity analysis. (CDL2)

CO4: apply risk management principles for assessing and mitigating organizational risks. (CDL2)

CO5: experiment intelligence-driven defensive and risk reduction plans for managing cyber incidents.(CDL2)

CO1: Utilize the concepts, processes, and lifecycle of threat intelligence for cyber protection. (CDL1)

Threat Landscape – Types of Threats – Cyber Kill Chain – Phases of Threat Intelligence Lifecycle – Types of Threat Intelligence - Role of Threat Intelligence in Cybersecurity Operations – Threat Actor Classification – Attack Surface Analysis – Threat Modeling Fundamentals – ChatGPT- based Phishing Attacks, Deepfakes in OSINT / HUMINT. **L:6**

CO2: Analyze cyber threats, attack vectors, and adversary tactics using standard threat- analysis frameworks. (CDL2) **L:10**

Adversary Behavior and Motivation – Cyber Attack Vectors – MITRE ATT&CK Framework – Diamond Model of Intrusion Analysis – Indicators of Compromise – Adversary TTP – Advanced Persistent Threat Analysis using YARA – Case Study: Mapping APT29 Techniques Using ATT&CK Navigator, ATT&CK Workbench.

CO3: Apply threat intelligence sources, tools, and data collection methods to support cybersecurity analysis. (CDL2) **L:10**

Open Source Intelligence – Human Intelligence – Technical Intelligence – Dark Web Monitoring Threat Feeds and Intelligence Platforms – Threat Hunting Tools – SIEM and SOAR Integration Threat Data Collection and Correlation using Maltego CE, SpiderFoot and OpenCTI – Case Study: Supply Chain Attacks in SolarWinds.

CO4: apply risk management principles for assessing and mitigating organizational risks.(CDL2) **L:9**

Concepts of Risk and Uncertainty – Risk Identification, Assessment, and Treatment – Quantitative and Qualitative Risk Analysis – Risk Appetite and Tolerance – Business Impact Analysis – Risk Evaluation and Mitigation Planning using OCTAVE Allegro, RiskQuant, and NIST Risk Management Framework tools - Case Study: Ransomware Risk Assessment in Healthcare Organization.

CO5: Analyze wireless intrusion threats and apply detection, prevention and risk assessment techniques in WLANs. (CDL2) **L:10**

Risk Mitigation Strategies – Threat Modeling – Incident Detection and Response Planning – Crisis Management – Post-Incident Analysis – Continuous Monitoring and Improvement of Threat & Risk Programs – Tools: Wazuh, ELK Stack, MITRE D3FEND, NIST CSF Assessment Tools – Real-Time Application: Incident Response and Threat Mitigation Plan Implementation for a Banking System.

TEXT BOOKS

1. Aaron Roberts, "Cyber Threat Intelligence: The No-Nonsense Guide for CISOs and Security Managers", 1st Edition, Apress (Springer Nature), 2021.
2. Youssef Baddi, Mohammed Amin Almaiah, Omar Almomani, and Yassine Maleh, "The Art of Cyber Defense: From Risk Assessment to Threat Intelligence", 1st Edition, CRC Press / Routledge, 2024.

REFERENCES

1. Valentina Costa-Gazcón, "Practical Threat Intelligence and Data-Driven Threat Hunting", 2nd Edition, Packt Publishing, 2025.
2. Wouter Veugelen, "Building an Effective Cybersecurity Program: From Risk Assessment to Incident Response", 2nd Edition, CRC Press, 2023.
3. Christopher J. Alberts and Audrey J. Dorofee, "Managing Information Security Risks: The OCTAVE Approach", Addison-Wesley Professional, 2023.

L:45; TOTAL: 45 PERIODS

Course Code	**DIGITAL FORENSICS**	**L**	**T**	**P**	**E**	**C**
23CS59E / 23IT59E / 23AD59E	(Common to CSE, IT, AI&DS Degree Programmes)	**3**	**0**	**0**	**0**	**3**

COURSE OUTCOMES

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

CO1: identify digital forensics concepts and apply forensic procedures to manage digital evidence. (CDL1)

CO2: analyze cybercrime laws and apply forensic readiness and legal methods to handle digital evidence. (CDL1)

CO3: apply forensic methods to analyze digital, mobile, and embedded evidence. (CDL2)

CO4: apply internet and memory forensic techniques for evidence acquisition and analysis in investigations. (CDL2)

CO5: apply legal and forensic practices to tackle digital investigation challenges. (CDL1)

CO1: Identify digital forensics concepts and apply forensic procedures to manage digital evidence. (CDL1)

History of Forensic Science – Digital Forensics – Digital Evidence – The Digital Forensics Process: Identification Phase – Collection Phase – Examination Phase – Analysis Phase – Presentation Phase. **L:9**

CO2: Analyze cybercrime laws and apply forensic readiness and legal methods to handle digital evidence. (CDL1) **L:9**

Introduction to Cybercrime Law – International Legal Framework of Cybercrime Law – Digital Crime – Substantive Criminal Law – Investigation Methods for Collecting Digital Evidence – International Cooperation in Order to Collect Digital Evidence – Digital Forensic Readiness – Law Enforcement vs. Enterprise Digital Forensic Readiness – Frameworks, Standards, and Methodologies – CERT–In Guidelines for Handling and Reporting Cyber Incidents.

CO3: Apply forensic methods to analyze digital, mobile, and embedded evidence. (CDL2)

Introduction to Computer Forensics - Evidence Collection – Examination – Analysis – Mobile and Embedded Forensics: Collection Phase – Examination Phase – Reverse Engineering and Analysis of Applications – Data Recovery Techniques and Tools. **L:9**

CO4: Apply internet and memory forensic techniques for evidence acquisition and analysis in investigations. (CDL2)

Internet Forensics: Layers of Network Abstraction – The Internet – Tracing Information on the Internet – Collection Phase: Local Acquisition – Network Acquisition – Remote Acquisition – Memory Forensics: Memory Management – Volatility – Memory Analysis in Criminal Investigations – Malware Analysis. **L:9**

CO5: Apply legal and forensic practices to tackle digital investigation challenges. (CDL1)

Legal Challenges in Digital Forensic Investigations: Constitutional issues in digital investigations, Federal Rules of Evidence – The Future of Cybercrime, Terror, and Policy: Introduction – Considering the future of cybercrime – Social movements, technology, and social change – Challenges in Digital Forensics: Computational Forensics – Automation and Standardization – Reinforcement of CERT-In Policies in Future Cybercrime Contexts. **L:9**

TEXT BOOKS:

1. Andre Arnes, Digital Forensic, John Wiley & Sons Ltd, 2019.
2. Thomas J. Holt, Adam M. Bossler, and Kathryn C. Seigfried-Spellar, Cybercrime and Digital Forensics an Introduction, 2nd Edition, Routledge, 2023.

REFERENCE BOOKS:

1. Joakim Kavrestad, Fundamentals of Digital Forensics: Theory, Methods and Real-Life Applications, Springer, 2020.
2. Keshav Kaushik, Rohit Tanwar, Susheela Dahiya, Komal Kumar Bhatia, Yulei Wu, "Unleashing the Art of Digital Forensics, 1st Edition, CRC Press, 2023.
3. Ahmed A. Abd El-Latif, Lo'aiTawalbeh, Manoranjan Mohanty, Brij B. Gupta, Konstantinos E. Psannis, Digital Forensics and Cyber Crime Investigation: Recent Advances and Future Directions, 1st Edition, CRC Press, 2024

L:45; TOTAL: 45 PERIODS

Course code	**TOOLS FOR NETWORK PROTECTION**	**L**	**T**	**P**	**E**	**C**
23CS51L/ 23IT51L/ 23AD51L	(Common to CSE, IT and AI & DS degree Programmes)	**0**	**0**	**2**	**0**	**1**

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Practical Component

CO1: Perform network reconnaissance and vulnerability scanning using tools like Nmap, Nessus, and Metasploit. (PDL2)
CO2: Analyze and monitor network traffic and logs using Splunk and Wireshark. (PDL2)

CO1: Perform network reconnaissance and vulnerability scanning using tools like Nmap, Nessus, and Metasploit. (PDL2)

1. Perform network reconnaissance using Nmap to conduct host discovery, port scanning, service/version detection, OS fingerprinting, firewall evasion, and generate a complete subnet scan report.
2. Use Nmap Scripting Engine (NSE) to detect vulnerabilities, run safe/unsafe scripts, enumerate services, and document discovered weaknesses across multiple hosts.
3. Install and configure Nessus Essentials, perform basic and advanced vulnerability scans, analyze CVSS scores, and generate comprehensive vulnerability assessment reports. **P: 15**
4. Scan a vulnerable machine (Metasploitable2) using Nessus, interpret high- risk vulnerabilities, validate results through manual checks, and export detailed scan findings.
5. Use Metasploit Framework for exploitation: perform service enumeration (FTP/SMB/SSH), import Nmap or Nessus scan results, select suitable exploit modules, and exploit a known vulnerability.

CO2: Analyze and monitor network traffic and logs using Splunk and Wireshark. (PDL2)

6. Analyze logs using Splunk: use SPL queries for DNS/HTTP log analysis, detect suspicious login attempts, create alerts, dashboards, and interpret SOC-style monitoring outputs.
7. Perform packet analysis using Wireshark by capturing/filtering traffic, analyzing TCP handshakes, DNS queries, HTTP sessions, detecting anomalies, reconstructing sessions, and performing GeoIP threat analysis.
8. Perform Security Event Correlation using Splunk by ingesting multiple log sources (Windows logs, Firewall logs, Web server logs), applying correlation rules, detecting multi- stage attacks, and generating an incident summary report. **P: 15**

Software Requirements:

Virtualization

VMware Workstation / VirtualBox

Kali Linux VM

Metasploitable2 VM

Security Tools

Nmap

Nessus Essentials

Metasploit Framework (built into Kali)

Wireshark

Splunk Free

P: 30; TOTAL: 30 PERIODS

Course Code	**MINI – CAPSTONE PROJECT**	**L**	**T**	**P**	**E**	**C**
23CS5ME/ 23IT5ME/ 23AD5ME	**(Common to CSE, IT and AI & DS degree Programmes)**	**0**	**0**	**0**	**6**	**3**

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Experiential Component

CO1: Analyze and detect security vulnerabilities through sensitive-data monitoring, insecure-website detection, directory enumeration, honeypot observation, and web application testing. (PDL2)

CO2: Perform offensive cybersecurity techniques including AD credential attacks, Kerberoasting, Pass-the-Hash/Ticket, and business-logic or client-side exploit. (PDL2)

CO3: Deploy defensive mechanisms such as honeypots, monitoring systems, secure clipboard handlers, and access-control hardening to generate threat intelligence (PDL3)

CO4: Evaluate and document security risks by testing, log analysis, validating exploits, and recommending mitigations aligned with cybersecurity standards. (PDL3)

Soft Skill Component

CO5: Communicate security findings effectively through professional reporting, teamwork, and structured presentation of vulnerabilities and mitigations.

Module 1: Introduction to Security Vulnerability Analysis (Week 1-3)

1. Overview of Web Security Vulnerabilities
    - Common vulnerabilities in web applications (XSS, CSRF, SQL Injection)
    - Overview of threat modeling and security analysis techniques
    - Tools for vulnerability scanning (OWASP ZAP, Burp Suite)
2. Sensitive Data Monitoring and Protection
    - Identifying sensitive data exposure in web applications
    - Monitoring data in transit and at rest
    - Encryption methods (SSL/TLS, AES) to secure sensitive data
3. Insecure Website Detection and Directory Enumeration
    - Identifying misconfigurations and vulnerabilities in web servers
    - Tools for directory enumeration and finding exposed resources (DirBuster, Gobuster)
    - Techniques for testing security flaws in web configurations
4. Honeypots for Threat Intelligence
    - Setting up and using honeypots to detect attacks
    - Analyzing data captured from honeypots for insights on attacker behavior
    - Best practices for configuring honeypots and analyzing logs
5. Web Application Testing Techniques
    - Using automated and manual testing to identify vulnerabilities
    - Common testing frameworks and tools (OWASP ZAP, Nikto, Burp Suite)
    - Introduction to ethical hacking principles and penetration testing

Deliverable: A comprehensive security vulnerability report detailing detected weaknesses, exploitation scenarios, and suggested remediation.

Module 2: Offensive Security Techniques (Week 4–6)

1. Introduction to Offensive Security
    - Overview of penetration testing and ethical hacking
    - Setting up an offensive security lab environment (Kali Linux, Metasploit)
    - Legal considerations and rules of engagement in cybersecurity testing
2. Active Directory (AD) Credential Attacks
    - Exploring AD authentication vulnerabilities (Kerberos, NTLM)
    - Pass-the-Hash (PTH) and Pass-the-Ticket (PTT) attacks
    - Tools and techniques: Mimikatz, CrackMapExec, BloodHound
3. Kerberoasting Attacks
    - Understanding Kerberos authentication and service ticket extraction
    - How to carry out a Kerberoasting attack and crack service tickets
    - Mitigation techniques for preventing Kerberoasting

Module 3: Defensive Cybersecurity Strategies (Week 7–9)

1. Defensive Security Frameworks and Strategies
    - Overview of defensive cybersecurity principles (Defense in Depth, Least Privilege)
    - Defensive tools: Firewalls, Intrusion Detection Systems (IDS), and Intrusion Prevention Systems (IPS)
2. Honeypots and Honeynets for Cyber Defense
    - Setting up honeypots for intrusion detection
    - Analyzing attacker behavior from honeypot interactions
    - Using honeypots to gather threat intelligence and improve defenses
3. Access Control Mechanisms and Hardening
    - Implementing Role-Based Access Control (RBAC) and Least Privilege
    - Hardening operating systems, applications, and networks
    - Best practices for configuring firewalls, VPNs, and secure authentication mechanisms
4. Threat Detection and Response Systems
    - Configuring and managing SIEM (Security Information and Event Management) systems (e.g., Splunk, ELK Stack)
    - Setting up monitoring and alerting for suspicious activities
    - Incident response processes and the role of threat intelligence
5. Secure Clipboard Handlers and Memory Protection
    - Protecting clipboard operations from hijacking and data leakage
    - Techniques for securing application memory and preventing exploitation
    - Tools for detecting clipboard and memory vulnerabilities

Deliverable: Implemented defense mechanisms (honeypots, access-control policies), with a report on monitoring activities and response strategies.

Module 4: Security Risk Assessment and Exploit Validation (Week 10-12)

1. Introduction to Security Risk Assessment
    - Risk management frameworks (NIST, ISO 27001)
    - Identifying and prioritizing security risks based on business impact
    - Security assessment tools and methodologies
2. Validating Exploits and Assessing Impact
    - Penetration testing techniques to validate exploits in a controlled environment
    - Using tools like Metasploit to test vulnerabilities and validate exploits
    - Understanding exploit impacts on confidentiality, integrity, and availability
3. Log Analysis and Threat Detection
    - Analyzing system and application logs for signs of compromise
    - Techniques for filtering and correlating log data to detect potential threats
    - Using tools such as Splunk, ELK Stack, and OSSEC for log aggregation and analysis
4. Evaluating Security Risks and Making Mitigations
    - Documenting security risks and preparing risk assessment reports
    - Recommending actionable mitigation strategies to reduce identified risks
    - Aligning recommendations with cybersecurity standards (e.g., NIST, CIS Controls)
5. Reporting and Communicating Security Findings
    - Writing clear and concise security risk reports for stakeholders
    - Communicating technical findings to non-technical decision-makers
    - Proposing risk reduction measures and implementing security controls

Deliverable: A security risk evaluation document including exploit validation, risk assessment, and detailed mitigation strategies.

Module 5: (Week 13-15)

1. Incident Response
    - Overview of the incident response lifecycle: Preparation, Detection, Containment, Eradication, Recovery
    - Building an incident response plan and team
    - Legal and regulatory considerations in incident handling
2. Incident Detection and Identification
    - Tools and techniques for detecting security incidents (IDS/IPS, SIEM)
    - Identifying indicators of compromise (IOCs)
    - Responding to data breaches, malware infections, and denial-of-service attacks
3. Containment, Eradication, and Recovery
    - Strategies for containing security incidents to prevent further damage
    - Eradicating malicious actors and remediating affected systems
    - Recovering from incidents and restoring business operations
4. Post-Incident Analysis and Reporting
    - Performing a post-mortem analysis to identify lessons learned
    - Documenting the incident timeline and response actions
    - Reporting to stakeholders and authorities (if applicable)
5. Threat Intelligence Sharing and Collaboration
    - Sharing threat intelligence with industry peers and agencies
    - Using threat intelligence platforms and services (e.g., MISP, ThreatConnect)
    - Collaboration between blue teams (defensive) and red teams (offensive) in incident response

Deliverable: An incident response report documenting the incident handling process, actions taken, and lessons learned. A comprehensive final security report for stakeholders.

E:90; TOTAL: 90 PERIODS

Course Code	**PRINCIPLES OF BUSINESS ANALYTICS**	**L**	**T**	**P**	**E**	**C**
23CS61E / 23IT61E / 23AD61E	**Common to CSE, IT and AI&DS Degree Programmes**	**3**	**0**	**0**	**0**	**3**

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Component

CO1: recognize the role of analytics in business for problem solving and decision-making. (CDL1)
CO2: apply hypothesis testing and non-parametric tests for decision-making. (CDL2)
CO3: analyze data distributions using statistical measures for effective data interpretation. (CDL2)
CO4: apply regression modeling to perform hypothesis testing, and predictive analytics in real-world applications. (CDL2)
CO5: apply Prescriptive Analytics Techniques to Optimize Business Processes. (CDL1)

CO1:recognize the role of analytics in business for problem solving and decision-making (CDL1) **L:9**

Introduction to Business Analytics: Evolution of Business Analytics – Scope – Data for Business analytics - Models in Business Analytics: Decision models- Uncertainty and Risk- Prescriptive Decision Models- Problem solving with Analytics: Recognizing and defining the problem – Structuring and analyzing the problem – Interpreting results and making a decision.

CO2: apply hypothesis testing and non-parametric tests for decision-making (CDL2) **L:9**

Hypothesis Testing – One- Sample Hypothesis test – Two tailed test of Hypothesis for the mean- Two sample hypothesis tests- One way and two way analysis of variance(ANOVA)- Non parmetric Tests: Chi-Square Goodness of Fit Tests – Choice of Number of Intervals in Chi- Square- Chi-Square test for Independence.

CO3: analyze data distributions using statistical measures for effective data interpretation (CDL2) **L:9**

Introduction – Data types and Scales – Types of Data measurement scales- Population and sample – Measures of central tendency – Percentile, Decile and Quartile – Measures of variation: Range – Inter-Quartile Distance – Variance and standard Deviation- Chebyshev's Theorem- Measures of shape – skewness.

CO4: apply regression modeling to perform hypothesis testing, and predictive analytics in real-world applications (CDL2) **L:9**

Modeling relationships and Trends in data – Simple linear regression: Finding best fitting regression line – Least squares regression – Testing hypotheses for regression coefficients – Multiple linear regression – Correlation and multi-collinearity. Case Study: Application of predictive analytics in customer retention or sales forecasting.

CO5: apply Prescriptive Analytics Techniques to Optimize Business Processes (CDL1) **L:9**

Introduction to prescriptive analytics –Linear programming model building – Sensitivity analysis in Linear programming problem – Solve the problem using Graphical method – Range of optimality – Range of shadow price- Dual linear programming – Primal Dual relationships – Multi-period models.

TEXT BOOKS

1. U Dinesh Kumar, "Business Analytics: The Science of Data-Driven Decision Making," 2nd Edition, Wiley, 2021.
2. R. Evans James, "Business Analytics", 2nd Edition, Pearson, 2017

REFERENCES:

1. Foster Provost and Tom Fawcett, "Data Science for Business," O'Reilly Media, 2021.
2. Randy Bartlett, "Business Analytics: A Practitioner's Guide," McGraw-Hill Education, 2013.
3. Galit Shmueli, Peter C. Bruce, Nitin R. Patel, "Data Mining for Business Analytics: Concepts, Techniques, and Applications in R," Wiley, 2018.
4. Eric Siegel, "Predictive Analytics: The Power to Predict Who Will Click, Buy, Lie, or Die," Wiley, 2016.

L: 45; TOTAL: 45 PERIODS

Course code		**L**	**T**	**P**	**E**	**C**
23CS62E/ 23IT62E/ 23AD62E	**DATA MINING** (Common to CSE, IT and AI & DS degree Programmes)	**3**	**0**	**0**	**0**	**3**

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Components

CO1: explore the data mining process and apply techniques to mine diverse forms of knowledge.
CO2: apply pattern mining techniques in various applications for the discovery of patterns.
CO3: apply classification techniques for real-world problems.
CO4: design and implement clustering in resource-constrained environments.
CO5: develop practical solutions to implement outlier detection in various applications.

CO1: Design and implement data mining projects and model data using schema.

Data mining: an essential step in knowledge discovery - Diversity of data types for data mining - Mining various kinds of knowledge - Data mining: confluence of multiple disciplines – Application. Data, measurements, and data preprocessing: Data types - Statistics of data - Similarity and distance measures - Data quality, data cleaning, and data integration - Data transformation - Dimensionality reduction. Data warehouse: Modeling - schema and measures - OLAP operations - Data cube computation. **L: 9**

CO2: Apply pattern mining techniques in various applications for the discovery of patterns.

Pattern mining: basic concepts and methods - Frequent itemset mining methods —Pattern evaluation methods - Mining various kinds of patterns - Mining compressed or approximate patterns - Constraint-based pattern mining - Mining sequential patterns - Mining subgraph patterns – Pattern mining: application examples. **L: 9**

CO3: Apply classification techniques for real-world problems.

Classification: basic concepts - Decision tree induction - Support vector machines - Rule-based and pattern-based classification – C4.5 algorithm- Classification with weak supervision – Classification with rich data type. **L: 9**

CO4: Design and implement clustering in resource-constrained environments.

Cluster analysis: basic concepts and methods – k-means clustering - agglomerative clustering – DBSCAN and STING - Biclustering - Semisupervised clustering. **L: 9**

CO5: Develop practical solutions to implement outlier detection in various applications.

Outliers and Outlier Analysis - Outlier Detection Methods - Statistical Approaches - Proximity-Based Approaches - Clustering-Based Approaches - Classification-Based Approaches - Mining Contextual and Collective Outliers - Outlier Detection in High-Dimensional Data. **L: 9**

TEXT BOOKS:

1. Jiawei Han, Jian Pei, Hanghang Tong, “Data Mining: Concepts and Techniques” 4th Edition, Elsevier, 2023.
2. Parteek Bhatia, “Data Mining and Data Warehousing: Principles and Practical Techniques”, 1st Edition, Cambridge University Press, 2019.

REFERENCE BOOKS:

1. Charu C. Aggarwal, “Data Mining: The Textbook”, Springer International Publisher, 2015.
2. Pawel Cichosz, “Data Mining Algorithms: Explained Using R”, John Wiley & Sons, 2015.

3. Pang-Ning Tan, Vipin Kumar, Michael Steinbach, "Introduction to Data Mining", Pearson Education India, 2012.

L: 45; TOTAL: 45 PERIODS

Course code		L	T	P	E	C
23CS63E/ 23IT63E/ 23AD63E	**DATA ENGINEERING** (Common to CSE, IT and AI & DS degree Programmes)	**2**	**0**	**2**	**0**	**3**

COURSE OUTCOMES

Upon completion of this course, the students will be able to

Theory Component

CO1: apply data engineering principles for data management and integration (CDL2)

CO2: analyze data modeling methodologies to design precise and efficient database solutions (CDL2)

CO3: apply resource management techniques for file-oriented systems (CDL2)

Practical Component

CO4: implement data extraction, transformation, and loading processes in database systems (PDL2)

CO5: demonstrate the OLAP modeling techniques (PDL2)

CO6: build data handling solutions for the real-life scenarios (PDL2)

CO1: Apply data engineering principles for data management and integration

CO4: Implement data extraction, transformation, and loading processes in database systems

Introduction to Data Engineering - Big Data and its characteristics - Data Lifecycle – Data Pipeline – Components – ETL process – ELT process – Containers – VM vs Container. **L: 10; P: 10**

Extract data from a CSV file, clean and transform it by removing duplicates, standardizing formats, and aggregating relevant metrics. Load the transformed data into a relational database (e.g., MySQL) for structured reporting and a NoSQL database (e.g., MongoDB) for flexible analytics. Verify data integrity by querying both databases and deploy this application in container

CO2: Analyze data modeling methodologies to design precise and efficient database solutions

CO5: Demonstrate the OLAP modeling techniques

Introduction to analytical data modeling methodologies -ER modeling- Dimensional modeling Star schema-Snowflake schema-Data Vault - One Big table modeling- data lakes- data warehouse. **L: 10; P: 8**

For a dataset in repository list down the data sources and model the data with the appropriate schema. Model the aggregate tables. Populate the aggregate tables with the metrics for a specific time period. Drilldown the metrics using the SQL operators such as Grouping sets, Cube, Roll up, Pivot. Bonus points if a metric uses time windows.

CO3: Apply resource management techniques for file-oriented systems **L: 10;**

CO6: Build data handling solutions for the real-life scenarios **P: 12**

HDFS- Map Reduce- Spark- Advantages of spark over map reduce-Spark's application

execution mechanism - Narrow and wide transformation in spark.-Shuffles in spark-ACID properties to file systems- introduction on Apache Hudi, Apache Iceberg and Delta lake.

Setup and configure spark and read the tables. Replicate the metrics table from the previous experiment in PySpark/Scala/Java. Write the output table back to a file system as parquet. Recreate the tables from previous experiments as Hudi/Iceberg tables in a file system.

SOFTWARE REQUIREMENTS

- MS-SQL / Oracle SQL / PostgreSQL
- Python 3.7+ / 4.0 IDLE,
- APACHE SPARK
- APACHE HUDI, APACHE ICEBERG

TEXT BOOKS:

1. I. Triguero and M. Galar, "Large-Scale Data Analytics with Python and Spark: A Hands-on Guide to Implementing Machine Learning Solutions", Cambridge: Cambridge University Press, 2023.
2. Paul Krickard, "Data Engineering with Python", Packt Publishing, 2020.
3. Andreas Kretz, "The Data Engineering Cookbook", 2019.

REFERENCE BOOKS:

1. Wilfried Lemahieu, Seppe vanden Broucke, Bart Baesens," Principles of Database Management: The Practical Guide to Storing, Managing and Analyzing Big and Small Data", Cambridge University Press, 2018.
2. Silberschatz, Korth, Sudarshan, "Database System Concepts", McGraw Hill Education, 7th Edition, 2020.
3. Ralph Kimball Margy Ross ,"The Data Warehouse toolkit by Ralph Kimball", Wiley Computer Publishing, 2013

L: 30; P: 30; TOTAL: 60 PERIODS

Course code 23CS64E/ 23IT64E/ 23AD64E	**IMAGE AND VIDEO ANALYTICS** (Common to CSE, IT and AI & DS degree Programmes)	**L**	**T**	**P**	**E**	**C**
		3	**0**	**0**	**0**	**3**

COURSEOUTCOMES

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

Theory Components

CO1: utilize the fundamentals of digital image processing.(CDL1)

CO2**: a**pply the knowledge of various techniques in image enhancement and frequency transform. (CDL1)

CO3: apply the restoration and boundary detection techniques to restore images and extract edges. (CDL2)

CO4: elaborate on deep learning based video analytics (CDL2)

CO5**:** develop image and video analytics models for real time application (CDL2)

CO1: Utilize the fundamentals of digital image processing. **L: 8**

Digital Image fundamentals -Components of Image Processing-Image Sensing and

Acquisition-Image Sampling and quantization-Relationship between pixels-Distance Measures-Colour Fundamentals and Models

CO2: Apply the knowledge of various techniques in image enhancement and frequency transform.

Discrete Fourier Transform – Fast Fourier Transform – Wavelet Transforms - Image Enhancement in Spatial and Frequency Domain – Grey Level Transformations – Histogram Processing –Spatial Filtering – Smoothing And Sharpening – Filtering in Frequency Domain **L: 9**

CO3: Apply the restoration and boundary detection techniques to restore images and extract edges.

Image Restoration – Image Degradation Model – Noise Modeling – Blur – Order Statistic Filters - Morphological Operations- Dilation-Erosion-Opening-Closing- Edge Detection-Detection Of Discontinuities Edge Linking and Boundary Detection **L: 9**

CO4: Elaborate on deep learning based video analytics

Video Processing – use cases of video analytics-Vanishing Gradient and exploding gradient problem-RestNet and skip connections - Inception Network - GoogleNet architecture-Improvement in Inception v2-Video analytics-RestNet and Inception v3. **L: 9**

CO5: Develop image and video analytics models for real time application (CDL2)

YOLO,R-CNN,U-Net,CBVR(Content Based Video Retrieval)-Industrial- Transportation & Travel- Remote Sensing - Surveillance- Example based video search-Intelligent traffic video analytics **L: 10**

TEXT BOOKS:

1. EI-Sayed M EL-Alfy, Georage Bebis, MengChu Zhou, "Intelligent Image and Video Applications", 1st Edition 2023.
2. Rafael C Gonzalez, Richard E Woods, "Digital Image Processing", Pearson Education, 4th Edition, 2018.

REFERENCE BOOKS:

1. Wilhelm, Mark J. Burge, "Digital Image Processing An Algorithmic Introduction", Springer; 3rd Edition, 2022.
3. VaibhavVerdhan, "Computer Vision Using Deep Learning Neural Network Architectures with Python and Keras" 1st Edition, Springer 2021.
4. Caifeng Shan, FatihPorikli, Tao Xiang, Shaogang Gong, "Video Analytics for Business Intelligence", 1st Edition, Springer, 2016.

L: 45; TOTAL: 45 PERIODS

Course code		L	T	P	E	C
23CS65E/ 23IT65E/ 23AD65E	**HEALTHCARE ANALYTICS** (Common to CSE, IT and AI & DS degree Programmes)	**3**	**0**	**0**	**0**	**3**

COURSE OUTCOMES

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

Theory Components

CO1: explore various sources of healthcare data and perform basic analytics on those data.(CDL 1)
CO2: analyze various biomedical modalities and Biomedical Signals. (CDL 2)
CO3: apply sensor data for healthcare analytics. (CDL 1)
CO4: apply advanced data analytics concepts for healthcare. (CDL 2)

CO1: Explore the various sources of healthcare data and perform basic analytics on those data.

Overview of Healthcare Data Sources: Electronic Health Records (EHR), Biomedical Images, Senor Data, Biomedical signals, Genomic data, Clinical Data, Social Media data, and its analysis – EHR: Components, Benefits, Barriers to Adopting EHR, Challenges – Phenotyping Algorithms – Overview of Coding Systems: ICD, ICF, UMLS, DICOM – Introduction to Data Analytics for Healthcare: Clinical prediction, Temporal and visual analytics, Clinic – Genomic Data Integration, Privacy Preservation – Data Publishing. **L: 12**

CO2: Analyze various biomedical modalities and Biomedical Signals.

Overview of Biomedical Imaging Modalities: CT, PET, MRI, Ultrasound, Microscopy, Biomedical Imaging Standards and Systems – Object Detection: Template Matching, Model – Based Detection, Data – Driven Detection Methods – Image Segmentation - Introduction to biomedical signals – Types of Biomedical Signals – Denoising of Signals – Multivariate Biomedical Signal Analysis – Recent Trends in Biomedical image and Signal Analysis **L: 12**

CO3: Apply sensor data for healthcare analytics.

Sensor Data in Medical Informatics: Scope – Challenges – Sensor Data Mining Applications: Intensive Care Data Mining – Nonclinical Healthcare Applications: Chronic Disease and Wellness Management, Activity Monitoring and Reality Mining – Data Analytics for Pervasive Health: Body area Networks, Dense/Mesh Sensor Networks, Sensor Technology –Applications: Continuous Monitoring, Assisted Living **L: 9**

CO4: Apply advanced data analytics concepts for healthcare.

Introduction to Clinical Prediction Models: Basic Statistical Prediction Models, Alternative Clinical Prediction Models, Survival Models, Evaluation and Validation – Visual Analytics for Healthcare: Introduction, Public Health and Population Research, Clinical Workflow, Clinicians, Patients – Legal and Ethical Issues in Clinical Decision Support Systems – Fraud Detection in Healthcare. **L: 12**

TEXT BOOKS:

1. Chandan K. Reddy and Charu C. Aggarwal, “Healthcare Data Analytics”, CRC Press, 2020.
2. Jaya, K. Kalaiselvi, Dinesh Goyal,” Handbook on Intelligent Healthcare Analytics: Knowledge Engineering with Big Data”, Wiley publications, 2022.

REFERENCE BOOK:

1. Pantea Keikhosrokiani, “Big Data Analytics for Healthcare: Datasets, Techniques, Life Cycles, Management, and Applications”, Academic Press, Elsevier publications, 2022

L: 45; TOTAL: 45 PERIODS

Course Code	PRESCRIPTIVE ANALYTICS AND OPTIMIZATION	L	T	P	E	C
23CS66E / 23IT66E / 23AD66E	(Common to CSE, IT, AI&DS Degree Programmes)	**3**	**0**	**0**	**0**	**3**

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Component

CO1: Analyze the role of prescriptive analytics in business intelligence and decision-making. (CDL1)

CO2: Solve basic linear programming problems for business applications. (CDL2)

CO3: Apply optimization problems to support data-driven business decision making. (CDL2)

CO4: Apply genetic algorithms and metaheuristic optimization techniques to solve complex problems. (CDL2)

CO5: Develop multi-objective optimization solutions for business intelligence applications. (CDL2)

CO1: Analyze the role of prescriptive analytics in business intelligence and decision-making. (CDL1) **L:9**

Evolution of Business Intelligence – Role of Prescriptive Analytics in Business Intelligence and Decision-Making – Applications – Decision Analytics Framework – Optimization and Model Building – Mathematical Modeling Process – Types of Optimization Problems, Stochastic vs Deterministic Models – Overview of Optimization Tools: Excel Solver, Python Solvers, OR-Tools.

CO2: Solve basic linear programming problems for business applications. (CDL2) **L:9**

Decision Variables, Objective Functions, Constraints – Graphical Solution Method – Feasibility and Optimality – Infeasibility, Unboundedness, Multiple Optimal Solutions – Simplex Algorithm: Theory and Steps – Simplex Tableau Method – Artificial Variables: Big-M and Two-Phase Method – Sensitivity Analysis: Ranging and Shadow Prices.

CO3: Apply optimization problems to support data-driven business decision making. (CDL2) **L:9**

Convex vs. Non-Convex Optimization – Gradient-Based Methods – Lagrange Multipliers and Constrained Optimization – Pricing and Revenue Optimization – Portfolio Optimization with Risk (Markowitz Model) – Supply Chain with Volume Discounts

CO4: Apply genetic algorithms and metaheuristic optimization techniques to solve complex problems. (CDL2) **L:9**

Genetic Algorithm Fundamentals – Population, Chromosomes, Operators: Selection, Crossover, Mutation – Fitness Functions and Convergence – Parameter Tuning – Other Metaheuristics: PSO, Simulated Annealing,– Hyperparameter Tuning for ML – Facility Location Problems – Scheduling and Sequencing – Portfolio Configuration.

CO5: Develop multi-objective optimization solutions for business intelligence applications. (CDL2) **L:9**

Goal-Based Decision Models – Priority-Based and Weighted Decision Strategies - Multi-Objective Business Decision Making - Prescriptive BI Dashboards for Executive Decision Support - Case Study: Prescriptive Business Intelligence in Supply Chain, Healthcare and Finance.

TEXT BOOKS:

1. Jeffrey D. Camm, James J. Cochran, Michael J. Fry, Jeffrey W. Ohlmann, "Business Analytics" 5th Edition, Cengage Learning, 2024.
2. S. Christian Albright and Wayne L. Winston, "Business Analytics: Data Analysis and Decision Making", 7th Edition, Cengage Learning, 2020.

REFERENCES:

1. Walter R. Paczkowski, "Hands-On Prescriptive Analytics: Optimizing Your Decision Making with Python", 1st Edition, O'Reilly, 2024.
2. Dursun Delen, "Prescriptive Analytics: The Final Frontier for Evidence-Based Management and Optimal Decision Making", Pearson, 2019.
3. Hamdy A. Taha, "Operations Research: An Introduction", 11th Edition, Pearson Education, 2022.
4. Frederick S. Hillier and Gerald J. Lieberman, "Introduction to Operations Research" 11th Edition, McGraw-Hill Education, 2021.

L: 45; TOTAL: 45 PERIODS

Course Code	FEATURE ENGINEERING	L	T	P	E	C
23CS67E / 23IT67E / 23AD67E	(Common to CSE, IT, AI&DS Degree Programmes)	**2**	**0**	**0**	**2**	**3**

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Component

CO1: recognize the importance of key feature engineering and data preprocessing techniques for data preparation. (CDL2)

CO2: apply various feature encoding methods to build machine learning models. (CDL2)

CO3: analyze the impact of density-based outlier detection methods to improve model performance. (CDL2)

CO4: apply feature selection and dimensionality reduction techniques to solve real-world problems. (CDL2)

Experiential Component

CO5: design and implement data preparation and feature engineering techniques to enhance model performance on real-world datasets. (PDL3)

CO1: Recognize the importance of key feature engineering and data preprocessing techniques for data preparation. **L:7**

Overview – Importance of Feature Engineering in Data Science – Feature Engineering Pipeline – Feature Engineering in Data Science. Types of data – Levels of data – Types of Feature Engineering – Evaluation metrics. Data preprocessing Techniques for handling missing data. Handling noisy data: Smoothing filters, Denoising algorithms.

CO2: Apply various feature encoding methods to build machine learning models. **L:7**

Feature Extraction – Temporal Features – Textual Features – Numeric Features – Categorical Features – Types of Feature Creation – Polynomial features and interaction terms – Binning and discretization techniques – Feature Encoding – Feature hashing and feature embedding – Function, Exponential, Quantile transformations – Feature Scaling –

Standardization – Normalization.

CO3: Analyze the impact of density-based outlier detection methods to improve model performance. **L:8**

Outlier and Outlier Detection: Univariate and Multivariate Outlier – Distributions to detect Outliers - Techniques for Removal of Outliers: Z-score, Trimming, Capping, Discretization Techniques, Mahalanobis distance. Density-based methods: DBSCAN, HDBSCAN, LOF, HDIOD, TPOD, Isolation Forests, One-class SVM for outlier detection.

CO4: Apply feature selection and dimensionality reduction techniques to solve real-world problems. **L:8**

Classes of feature selection methodologies – Effect of irrelevant features – Filter Methods: Chi-square test, Fisher's Score, Correlation coefficient – Wrapper Methods: Forward Selection, Bi-directional Elimination, Recursive Feature Elimination (RFE).

CO5: Design and implement data preparation and feature engineering techniques to enhance model performance on real-world datasets. **E:30**

Analyze real-world datasets and perform the following:

- **Load and Explore Data:** Import the dataset, identify data types (numeric, categorical, text, temporal) and check for missing or noisy data.
- **Handle Missing and Noisy Data:** Apply imputation (mean, median, KNN), smoothing filters or denoising techniques to clean the dataset.
- **Conduct Review I meeting for evaluating the performance.**
- **Feature Extraction:** Extract features from textual, temporal, numeric and categorical data using suitable methods.
- **Feature Encoding:** Use techniques like one-hot encoding, feature hashing or embeddings to convert categorical/textual data into numerical form.
- **Apply Feature Transformations:** Use transformations such as logarithmic, quantile or exponential to adjust data distribution.
- **Conduct Review II evaluation.**
- **Scale Features:** Normalize or standardize numerical features using min-max scaling, z-score normalization or robust scaling.
- **Detect and Handle Outliers:** Identify outliers using statistical (Z-score, IQR) or density-based methods (LOF, Isolation Forest) and assess their impact.
- **Feature Selection & Model Evaluation:** Use techniques like correlation, PCA or Lasso for feature selection and evaluate model performance using appropriate metrics.
- **Conduct Review III evaluation.**

TEXT BOOKS:

1. Sinan Ozdemir, "Feature Engineering Bookcamp", 1st Edition Manning Publications, 2022.
2. Max Kuhn and Kjell Johnson, "Feature Engineering and Selection: A Practical Approach for Predictive Models", 1st Edition, CRC Press Taylor & Francis Group, 2020.

REFERENCES:

1. Alice Zheng and Amanda Casari, "Feature Engineering for Machine Learning: Principles and Techniques", 1st Edition, O'Reilly Media, 2018.
2. Sinan Ozdemir and Divya Susarla, "Feature Engineering Made Easy: Identify Unique Features

from Your Dataset in Just 30 Minutes", 1st Edition, Packt Publishing, 2018.

L: 30; E: 30; TOTAL: 60 PERIODS

Course Code	**PREDICTIVE ANALYTICS**	**L**	**T**	**P**	**E**	**C**
23CS68E / 23IT68E / 23AD68E	(Common to CSE, IT, AI&DS Degree Programmes)	**3**	**0**	**0**	**0**	**3**

COURSEOUTCOMES

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

Theory Component

CO1: explain the foundational concepts of predictive analytics and their role in business decision-making. (CDL 1)

CO2: identify components of time series data and apply decomposition methods for business forecasting. (CDL 1)

CO3: construct statistical forecasting models and apply discriminant analysis techniques for classification in business scenarios. (CDL 2)

CO4: apply predictive and discriminant analytic techniques in real-world business cases for planning and decision-making. (CDL 2)

CO1: Explain the foundational concepts of predictive analytics and their role in business decision-making. **L:12**

Overview of Predictive Analytics: purpose and scope - Descriptive vs. Predictive vs. Prescriptive Analytics - Predictive Analytics vs. Business Analytics - Data-Driven Decision Making in Business - Applications in finance, HR, marketing, and operations - Role of Predictive Analysts in Organizations.

CO2: Identify components of time series data and apply decomposition methods for business forecasting. **L:12**

Structure and types of time series data - Components: Trend, Seasonality, Cyclical, Irregular-Additive vs. Multiplicative decomposition - Smoothing techniques: moving average, exponential smoothing - Seasonally adjusted series and business use cases - Evaluation of Forecast Accuracy - Graphical Analysis and Visualization of Time Series Data.

CO3: Construct statistical forecasting models and apply discriminant analysis techniques for classification in business scenarios. **L:12**

Naïve, average, and trend models - Holt and Holt-Winters exponential smoothing - ARIMA - Fisher's Linear Discriminant Function - Prior probabilities, misclassification costs - Performance evaluation of discriminant analysis - Prior probabilities and misclassification costs - Classification with more than two groups

CO4: Apply predictive and discriminant analytic techniques in real-world business cases for planning and decision-making. **L:12**

Demand Forecasting for Inventory and Supply Chain - Credit Risk Prediction using Discriminant Logic - Sales and Revenue Forecasting - HR Analytics: Employee Attrition

Forecasting - Ethical issues and limitations in prediction - Case study discussion and model interpretation - Communicating analytical results to business stakeholders - Report structuring and presentation best practices - Business use-case simulation in Excel.

TEXT BOOKS:

1. Dean Abbott, "Applied Predictive Analytics: Principles and Techniques for the Professional Data Analyst", 1st Edition, Wiley, 2014.
2. Rob J. Hyndman & George Athanasopoulos, "Forecasting: Principles and Practice", 3rd Edition, OTexts, 2021.
3. J. Holton Wilson & Barry Keating, "Business Forecasting", 6th Edition, McGraw-Hill Education, 2009.
4. Lind, Marchal, Wathen, "Statistical Techniques in Business and Economics", 18th Edition, McGraw-Hill College, 2020.

REFERENCE:

1. Max Kuhn, Kjell Johnson, "Applied Predictive Modeling", 1st Edition, Springer publications, 2013.

L: 45;TOTAL: 45 PERIODS

Course Code	MINI - CAPSTONE PROJECT	L	T	P	E	C
23CS6ME/ 23IT6ME/ 23AD6ME	(Common to CSE, IT, AI&DS Degree Programmes)	**0**	**0**	**0**	**6**	**3**

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Experiential Component:

CO1: apply data preprocessing for effective data modeling. (PDL2).

CO2: apply feature engineering & data mining techniques to discover hidden patterns that supports strategic decision making in business. (PDL2).

CO3: develop predictive models solutions for the text/image/video analytics based applications. (PDL2).

CO1: Apply data preprocessing for effective data modeling.

Module 1: **E:20**

- Define the business objective and identify KPIs & success metrics. Collect structured/semi-structured/unstructured data.
- Understand distribution, trends, patterns, correlations, and outliers in data.
- Clean the data, handle missing values, normalize, and resolve inconsistencies.

Deliverables:

- Confirming business objective & key performance indicators.
- Guide approved project proposal document.

CO2: Apply feature engineering & data mining techniques to discover hidden patterns that supports strategic decision making in business.

Module 2:
- Apply feature engineering, to derive new attributes. **E:20**
- Encode categorical variables and reduce dimensionality if needed.

Deliverables:
- Reduced dimensionality data sheet after applying feature engineering.

Module 3:
- Apply classification/clustering/association techniques to explore hidden patterns in the data. **E:20**
- Train classification or regression models for future prediction or decision support.
- Measure performance using various metrics.

Deliverables:
- Trained model training using R/Python code report
- Insights and business interpretation.

CO3: Develop predictive model solutions for the text/image/video analytics based applications.

Module 4:
- Apply techniques such as CNNs, motion tracking to extract insights for the image/video analysis projects with machine learning or deep learning. **E:15**

Deliverables:
- Architecture model and visual demonstration.

Module 5:
- Present insights using dashboards, graphs, and visual reports. **E:15**
- Relate patterns and predictions to high-level decisions.

Deliverables:
- Interactive dashboard with identified business decision demonstration.

Example Project titles:

1. Sentiment analysis of customer reviews in text analytics.
2. Fraud detection in financial transactions.
3. Sales forecasting in retail or manufacturing area.
4. Image-based product categorization in image analytics
5. Video surveillance for security in video analytics.
6. Chatbot feedback and Query analysis in text analytics.
7. Employee attrition analysis in HR analytics.
8. Social media digital marketing analytics.
9. Customer lifetime value prediction for the E-commerce applications.

Course Code	DATA ANALYTICS USING SNOWFLAKE	L	T	P	E	C
23CS61L/ 23IT61L/ 23AD61L	(Common to CSE, IT, AI&DS Degree Programmes)	0	0	2	0	1

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Practical Component

CO1: design, build, and manage data pipelines and analytics workflows using the Snowflake cloud data platform. (PDL2)

CO2: integrate, deploy, and manage Snowflake-based analytics solutions using BI tools. (PDL2)

CO1: Design, build, and manage data pipelines and analytics workflows using the Snowflake cloud data platform. (PDL2) **P: 20**

1. Create and modify user roles and privileges in Snowflake. Use the CREATE ROLE and GRANT statements for user identity management.
2. Ingest structured and semi-structured data into Snowflake by utilizing SQL load scripts as well as the Snowflake web console.
3. Data exploration through SQL queries involves handling missing values, performing aggregations, and computing basic statistics.
4. Implementation of ETL pipeline in Snowflake using Streams and Tasks to automate data processing and transformation.
5. Design a data warehouse model in Snowflake using STAR/SNOWFLAKE schema and optimize query performance.

CO2: Integrate, deploy, and manage Snowflake-based analytics solutions using BI tools. (PDL2) **P: 10**

6. Connect Snowflake to BI tools and create interactive dashboards and visualizations to represent business data.
7. Create SQL views or materialized views in Snowflake to optimize reporting queries.

Software Requirements:

CO1: Snowflake Account, Snowflake Worksheets, Snowflake Notebooks, SQL, Snowflake Connector for Python, Python, Pandas, JSON/CSV/Parquet datasets.

CO2: Tableau/Power BI, Cloud SDKs (AWS/Azure/GCP), Snowflake Shares.

P: 30; TOTAL: 30 PERIODS

Course Code	APPROXIMATION ALGORITHMS	L	T	P	E	C
23CS72E / 23IT72E / 23AD72E	(Common to CSE, IT, AI&DS Degree Programmes)	**3**	**0**	**0**	**0**	**3**

COURSEOUTCOMES

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

Theory Component

CO1: apply approximation algorithms, greedy methods, and local search techniques to solve problems. (CDL1)

CO2: apply rounding techniques and dynamic programming to develop approximation schemes for knapsack, bin packing, and scheduling problems. (CDL1)
CO3: apply randomized rounding techniques and the primal-dual method to solve problems. (CDL1)
CO4: analyze approximation algorithms and techniques for solving computational problems. (CDL1)
CO5: apply various techniques to prove the hardness of approximation for NP-hard problems. (CDL1)

CO1: Apply approximation algorithms, greedy methods, and local search techniques to solve problems. (CDL1) **L:9**

Approximation Algorithms- Definition, importance, and applications of approximation algorithms - Performance measures- Approximation ratio, approximation factor - Greedy Algorithms andLocal Search - Vertex cover, set cover, independent set - Job scheduling algorithms

CO2: Apply rounding techniques and dynamic programming to develop approximation schemes for knapsack, bin packing, and scheduling problems. (CDL1) **L:9**

Rounding Data and Dynamic Programming - Approximation schemes for knapsack, bin packing, and scheduling problems – Scheduling jobs on identical parallel machines - Deterministic Rounding of Linear Programs- Scheduling, facility location and optimization problems such as bin packing

CO3: Apply randomized rounding techniques and the primal-dual method to solve problems. (CDL1) **L:9**

Randomized Rounding of Linear Programs- Simple algorithms for Max-SAT and MAX CUT - Steiner tree and TSP – Derandomization - The Primal-Dual Method: Set cover, vertex cover – The shortest path problem - Applications to Steiner tree problems

CO4: Apply various techniques to prove the hardness of approximation for NP-hard problems. (CDL1) **L:9**

Overview of NP-hard problems and significance of approximation for NP-hard problems - Hardness of Approximation - Semidefinite Programming (SDP) for Approximation Algorithms - Lower Bounds on Approximations - Techniques for proving the hardness of approximation

CO5: Analyze approximation algorithms and techniques for solving computational problems. (CDL1) **L:9**

Approximation Algorithms for Vertex Cover, Set Cover and Traveling Salesman - Approximation Algorithms for Knapsack and Job Scheduling - Fully Polynomial-Time Approximation Schemes (FPTAS) - Scheduling algorithms and their approximation

TEXTBOOKS:

1. Teofilo F. Gonzalez “Handbook of Approximation Algorithms and Metaheuristics”, 2nd Edition, CRC Press, 2020.
2. Jarosław Byrka, Andreas Wiese, “Approximation and Online Algorithms”, Springer Nature Publisher, 1st Edition, 2023.

REFERENCE:

1. Bernd Gartner, Jiri Matousek, "Approximation Algorithms and Semidefinite Programming", Springer, 2012.
2. Vijay V. Vazirani, "Approximation Algorithms", 1st Edition, Springer, 2013.
3. David P. Williamson and David B. Shmoys, "The Design of Approximation Algorithms", 1st Edition, Cambridge University Press, 2011.

L: 45;TOTAL: 45 PERIODS

Course Code	COMPUTATIONAL GRAPH THEORY	L	T	P	E	C
23CS74E / 23IT74E / 23AD74E	(Common to CSE, IT, AI&DS Degree Programmes)	2	0	2	0	3

COURSEOUTCOMES

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

Theory Component

CO1: apply the fundamental concepts of graphs to represent and analyze real-world systems. (CDL1)

CO2: analyze connectivity and structure in graphs using spanning trees, circuits, and planar representations. (CDL2)

CO3: evaluate graph properties to solve optimization and scheduling problems.

(CDL1) CO4: construct and analyze complex graph structures and flow networks. (CDL2)

Practical Component

CO5: implement traversal and structural graph concepts in network and design problems. (PDL1)

CO6: apply connectivity, flow, and ordering properties in graphs for optimizing and analyzing complex networks. (PDL1)

CO1: Apply the fundamental concepts of graphs to represent and analyze real-world systems. (CDL1) **L:8; P:10**

Graphs – Introduction – Isomorphism – Sub graphs – Walks, Paths, Circuits – Connectedness – Components – Euler graphs – Hamiltonian paths and circuits – Trees – Properties of trees – Distance and centers in tree.

CO5: Implement traversal and structural graph concepts in network and design problems (PDL1)

Euler Circuit for Garbage Collection - Hamiltonian Path for City Tourist Bus Route

CO2: Analyze connectivity and structure in graphs using spanning trees, circuits, and planar representations (CDL2) **L:8 P:6**

Spanning trees – Fundamental circuits – Spanning trees in weighted graphs – Cut sets – Connectivity and separability – Combinational and geometric graphs – Planar graphs – Representation of planar graphs – Hypergraphs.

CO5: Implement traversal and structural graph concepts in network and design problems. (PDL1)

Network Partitioning Using the Max-Cut Algorithm - Planar Graph Verification for Cable Network Design - Community Detection in Social Networks Using Graph Clustering Methods

CO3: Evaluate graph properties to solve optimization and scheduling problems. (CDL1) **L:7; P:8**

Chromatic number – Chromatic partitioning – Chromatic polynomial – Matching – Covering – Four color problem – Directed graphs – Types of directed graphs – Binary relations – Directed paths and connectedness

CO6: Apply connectivity, flow, and ordering properties in graphs for optimizing and analyzing complex networks (PDL1)

Resource allocation using vertex cover - Connectivity testing in Real - world networks

CO4: Construct and analyze complex graph structures and flow networks (CDL2) **L:7; P:6**

Network Flows: Maximum flow - Minimum cut - Flow algorithms (Ford–Fulkerson, Edmonds– Karp) - Graph Density: Dense graphs - Sparse graphs - Graph Minors - Tree Decomposition: Treewidth – Pathwidth

CO6: Apply connectivity, flow, and ordering properties in graphs for optimizing and analyzing complex networks (PDL1)

Smart City Water Distribution Optimization Using Max-Flow/Min-Cut - Dense & Sparse Region Detection in Online Social Media Networks

TEXT BOOKS:

1. N. P. Shrimali, Nita H. Shah, "Recent Advancements in Graph Theory", CRC Press, 1st Edition 2020.
2. Reinhard Diestel, "Graph Theory", Springer-Verlag Berlin Heidelberg, 5th Edition 2017.

REFERENCES:

1. Jay Yellen, Jonathan L. Gross, Mark, "Graph Theory and Its Applications", CRC Press, 2019.
2. Lowell W. Beineke, Robin J. Wilson, Ortrud R. Oellermann, "Topics in Structural Graph Theory", Cambridge University Press, 2013.
3. Narsingh Deo, "Graph Theory: With Application to Engineering and Computer Science", Prentice Hall of India, 2004.

L: 30;P: 30;TOTAL: 60 PERIODS

R-2023 B.E. CSE

OPEN ELECTIVE COURSES

Course Code	PYTHON PROGRAMMING	L	T	P	E	C
23CS01N		2	0	2	0	3

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Component:

CO1: execute the basic programming constructs in Python environment.(CDL1)

CO2: apply conditional and looping constructs for simple problem solving. (CDL1)

CO3: demonstrate the representation of compound data using list. (CDL2)

CO4: develop simple packages using hierarchical data structures.(CDL2)

CO5: apply python libraries like NumPy, pandas, matplotlib to solve applications. (CDL2)

Practical Component:

CO6: demonstrate programs for simple algorithms using sequential, control structures and modular approach. (PDL1)

CO7: develop application with hierarchical data structures using packages.

CO1: Execute the basic programming constructs in Python environment. **L: 5**

Python interpreter and interactive mode - values and data types: Variables - expressions - statements – operators - precedence of operators – Input and Output - comments – Errors: Syntax Errors - Runtime errors - Logical Errors. **P: 5**

CO6: demonstrate programs for simple algorithms using sequential, control structures and modular approach.

Solve the simple problems using basic programming construct in python.

CO2: Apply conditional and looping constructs for simple problem solving. **L: 5**

Conditionals: Boolean values and operators - conditional (if) - alternative (if-else) – chained conditional (if-elif-else) - Iteration: state – while – for – break – continue - pass - Fruitful functions: Function argument and its types - return values – parameters - local and global scope – function composition – recursion - Documenting Function. **P: 5**

CO6: demonstrate programs for simple algorithms using sequential, control structures and modular approach.

Solve problem with conditional, looping constructs with functions.

CO3: Demonstrate the representation of compound data using list. **L: 7**

Classes and Inheritance - Object Oriented Programming - Class Instances – Class Methods. Strings: string slices – immutability - string functions and methods - string module - Lists: list operations - list slices - list methods - list loop – mutability – aliasing - cloning lists – list Parameters - Lists asarrays. **P: 7**

CO7: Develop application with hierarchical data structures using packages.

Implement problems using string and list concepts.

CO4: Develop simple packages using hierarchical data structures **L: 6**

Tuples: tuple assignment - tuple as return value - Dictionaries: operations and methods – advanced list processing - list comprehension - Files and exception: text files - reading and writing files - format operator - command line arguments - errors and exceptions - handling exceptions – modules – packages - Name space. **P: 6**

CO7: Develop application with hierarchical data structures using packages.

Solve problems using tuples, dictionaries – Implement problem using files and modules.

CO5: Apply python libraries like NumPy, pandas, matplotlib to solve applications. **L: 7**
NumPy Basics: Arrays and Vectorized Computation – Pandas: Data Loading, storage and File Formats – Data Cleaning and Preparation – Data Wrangling: Join, Combine and Reshape – Plotting and Visualization – Data aggregation and Group operations. **P: 7**

CO7: develop application with hierarchical data structures using packages.
Develop simple applications using NumPy and Pandas packages for data exploration.

TEXT BOOKS:

1. Martin C. Brown, "The Complete Reference PYTHON", Mc Graw Hill Publications, 4th Edition, 2018.
2. Wes McKinney, "Python for Data Analysis: Data Wrangling with Pandas, NumPy & Jupyter", O'Reilly publications, 3rd Edition, 2022.

REFERENCE BOOKS:

1. John V.Guttag, "Introduction to Computation and Programming Using Python: With Application to Understanding Data", Prentice-Hall International publishers, 3rd Edition, 2021.
2. Meenu Kohli, "Basic Core Python Programming", BPB Publications, 1st Edition, 2021

Web References:

1. SWAYAM/NPTEL Course – Joy of Computing using Python

Software Requirements:

- Python 3.x
- Google Colaboratory/ Jupyter/ Anaconda navigator

L: 30; P: 30; TOTAL: 60 PERIODS

Course Code	FOUNDATIONS OF OBJECT ORIENTED	L	T	P	E	C
23CS02N	PROGRAMMING	2	0	2	0	3

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Component

CO1: explore the constructs of object oriented program(CDL1)
CO2: apply constructor and overloading mechanisms to simple application (CDL1)
CO3: employee code reusability through inheritance and polymorphism (CDL1)
CO4: develop real-world applications by using files, streams, and exceptions (CDL2)
CO5: construct well-defined, efficient data handling strategies using templates and STL (CDL1)

Practical Component

CO6: implement the basic object oriented principles such as class, objects, and constructors (PDL1)
CO7: establish code reusability through inheritance and polymorphism (PDL1)
CO8: solve problems using templates, STL, files and exception handling (PDL1)

CO1: explore the constructs of object oriented program **L:6;**
Introduction to OOPs - Applications of OOP - Structure of C++ - Program - C++ Basics: Keywords – Constants - Data Types - Dynamic Initialization of Variables – Reference Variables - Operators in C++ - C++ Class Overview: Class Definition Objects – Class Members - Access Control – Scope Resolution operator – Inline Function - Friend **P:6**

Functions - static class members.

CO2: apply constructor and overloading mechanisms to simple programs **L:6;**
CO6: implement the basic object oriented principles such as class, objects, and constructors **P:6**

Constructors: Parameterized Constructors - Multiple Constructors in a Class – Constructors with Default Arguments - Dynamic initialization of Objects - Copy Constructors – Dynamic Constructors - Destructors - Function Overloading - Operator overloading - Rules for Operator overloading - overloading of binary and unary operators.

CO3: employe code reusability through inheritance and polymorphism **L:6;**
CO7: establish code reusability through inheritance and polymorphism **P:6**

Introduction to inheritance - Defining Derived Classes - Single Inheritance – Multiple Inheritance - Multi-Level Inheritance - Hierarchical Inheritance - Hybrid Inheritance - Virtual Base Classes - Abstract Classes - Constructors in Derived Classes – Introduction to pointers - Pointers to Objects - Virtual Functions - Pure Virtual Functions – Virtual Destructors.

CO4: develop real-world applications by using files, streams, and exceptions **L:6;**
Files in C++: File handling in C++ - File I/O - Formatted and Unformatted I/O - Basics of **P:6**
Exception Handling, Types of exceptions - Exception Handing Mechanism - Throwing and Catching Mechanism.

CO5: construct well-defined, efficient data handling strategies using templates and STL **L:6; P:6**
CO8: solve problems using templates, STL, files and exception handling

Class Templates - Class Templates with Multiple Parameters - Function Templates, Function Templates with Multiple Parameters –Member Function Templates - STL.

TEXT BOOKS

1. Herbert Schildt, “C++: The Complete Reference”, 5th Edition, Tata McGraw Hill Publishers, 2014.
2. Paul Deitel, Harvey Deitel, “C++ How to Program”, 8th Edition, Prentice Hall Publisher, 2016.
3. Trivedi, Bhushan “Programming with ANSI C++”, 2nd Edition, Oxford University Press, NASW Press, 2013.

REFERENCE BOOKS

1. Ira Pohl, “Object Oriented Programming using C++”, Pearson Education, 2nd Edition, Reprint 2004.
2. S.B.Lippman, Josee Lajoie, Barbara E.Moo, “C++ Primer”, Pearson Education, 4th Edition, 2012.
3. Bjarne Stroustrup, “The C++ Programming language”, Pearson Education, 4th Edition, 2013.

L: 30; P:30; TOTAL: 60 PERIODS

Course Code		L	T	P	E	C
23CS03N	COMPUTER GRAPHICS AND VIRTUAL REALITY	3	0	0	0	3

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Component

CO1: understanding of computer graphics fundamentals, including computer systems, software types, and programming languages(CDL1)

CO2: understand geometric transformations, object modeling, and clipping techniques in 2D and 3D computer graphics. (CDL1)

CO3: analyze and apply core principles of computer graphics, including hidden surface and hidden line removal techniques (CDL1)

CO4: understand and analyze the fundamental concepts and types of Virtual Reality (VR) (CDL2)

CO5: apply and analyze Augmented Reality (AR) technologies (CDL2)

CO1: Understanding of computer graphics fundamentals, including computer systems, software types, and programming languages

Introduction to Computer Graphics - Computer System - Types of Software - Programming Languages - Computer Graphics and CAD - Geometric Modelling Methods - Benefits - Co-ordinate Representation - Types of Computer Graphics - Output Primitives (Points, Lines, Circles and Ellipses) - Line and Line Generation Algorithm (DDA, Bresenham) - Scan Conversion of a Circle (Eight-Point Symmetry, Bresentham's, Mid Point) - Ellipse Generation Algorithm - Area or Region Filling **L:9**

CO2: Understand geometric transformations, object modeling, and clipping techniques in 2D and 3D computer graphics

Anti-Aliasing Lines - Object Modeling - Representation of a Point in Matrix Form - Two Dimensional Transformation - Representation of Objects in Terms of Position Vectors - Geometric Transformation of a Straight Line - 3D Geometric Transformation and Projection - Windowing and Clipping (Point, Line, Polygon, Curve, Text) **L:9**

CO3: Analyze and apply core principles of computer graphics, including hidden surface and hidden line removal techniques

Hidden Lines and Surfaces - Classification - Hidden Surface Removal Algorithms - Illumination Model and Algorithm - Basic Lighting Model (Emissive, Ambient, Diffused Reflection, Specular Reflection) - Polygon Rendering - Three-Dimensional Viewing - Clipping - Curve and Surface Representation - Geometric Modelling - Representation of Polynomial Curve - 2D & 3D Animation Techniques - Software and its Application **L:9**

CO4: Understand and analyze the fundamental concepts and types of Virtual Reality (VR)

Virtual Reality: An Overview - Types of Virtual Reality - VR Systems and Hardware - Components of VR - Software and tools - Interfaces - HMD, Boom, CAVE - VR Tracking Systems - Visual, Acoustic and Haptic Perception - Audio - McGurk effect - Multi Modality on Moving Target - VR Haptics **L:9**

CO5: Apply and analyze Augmented Reality (AR) technologies

Introduction to Augmented Reality - AR Hardware Technologies - AR Software Platforms **L:9**

and SDKs - Tracking and Registration Techniques - Computer Vision in AR - User Interaction in AR Environments - Rendering Techniques in AR - Networking and Cloud Integration in AR - Ethical, Privacy, and Social Implications of AR - case Studies and Industry Applications

TEXT BOOKS

1. C.S Verma, rajesh Purohit, Koyel Datta Gupta and Harsha Verma, "Computer Graphics and CAD", 1st Edition, CRC Press Taylor and Francis group publishers, 2025.
2. Edward Angel and Dave Shreiner: Interactive Computer Graphics: A Top-Down Approach with WebGL, 8th Edition, Addison-Wesley, 2020
3. Peter Shirley, Fundamentals of Computer Graphics, 5th Edition, A K Peters/CRC Press, 2022.
4. Sumanta Guha, Computer Graphics through OpenGL: from theory to experiments, 4th Edition, CRC Press, 2023.

REFERENCE BOOKS

1. Eric Lengyel, 3D Graphics for Game Programming, A K Peters/CRC Press, 2nd Edition, 2023.
2. Roger D. Rogers and David F. McLain: Computer Graphics: A Programmed Approach, Prentice Hall, 3rd Edition, 2019.
3. Donald Hearn and M. Pauline Baker: Computer Graphics Using OpenGL, Prentice Hall, 6th Edition, 2014.

L: 45; TOTAL: 45 PERIODS

Course Code		L	T	P	E	C
23CS04N	**DATA STRUCTURES AND ALGORITHMS**	2	0	2	0	3

COURSE OUTCOMES

Upon the successful completion of the course, the student will be able to

Theory Component

CO1: apply linear data structures like arrays and linked lists to solve sorting and searching problems.(CDL1)

CO2: demonstrate the usage of stack and queue operations for simple applications.(CDL1)

CO3: analyze tree data structures for hierarchical data representation.(CDL1)

CO4: apply appropriate graph representations and traversal techniques in real-world scenarios.(CDL2)

Practical Component

CO5: implement linear data structure operations through simple problem.(PDL1)

CO6: apply non linear data structures such as tree and graphs to solve complex problems.(PDL2)

CO1: Apply linear data structures like arrays and linked lists to solve sorting and searching problems

List ADT (using Array and Linked List)– Singly, Doubly, Circular Linked Lists – Sorting – Bubble, Insertion, Quick – Searching – Linear, Binary. **L:8; P:10**

CO5: Implement linear data structure operations through simple problem

Implement basic operations using arrays and Singly, Doubly, Circular Linked Lists - Implement sorting and searching algorithms.

CO2: Demonstrate the usage of stack and queue operations for simple applications.

Stack ADT (using Array and Linked List) – Applications: Balancing symbols, Infix to Postfix, Postfix Evaluation - Queue ADT (using Array and Linked List) – Circular Queue – Deque – Application of Queue. **L:8; P:6**

CO5: Implement linear data structure operations through simple problem

Develop stack and queue structures using arrays and linked lists.

CO3: Analyze tree data structures for hierarchical data representation

Tree ADT – Terminologies – Binary Tree, Binary Search Tree – Traversals – Expression Tree – AVL Tree- Case study on B+ tree and Hashing techniques. **L:7; P:8**

CO6: Apply non linear data structures such as tree and graphs to solve complex problems

Construct and traverse binary tree and binary search tree- Create and evaluate expression tree.

CO4: Apply appropriate graph representations and traversal techniques in real-world scenarios

Graphs: Basic Terminologies and Representation – Types of graphs – Breadth first search – Depth first search – Topological Sort – Dijkstra'sAlgorithm –Minimum Spanning Tree – Prim's and Kruskal's Algorithm. **L:7; P:6**

CO6: Apply non linear data structures such as tree and graphs to solve complex problems

Implement BFS and DFS for graph traversal - Apply Prim's and Kruskal's algorithms to find minimum spanning tree.

TEXT BOOKS

1. Mark Allen Weiss, "Data Structures and Algorithm Analysis in C++", 4th Edition, Pearson Education, 2014.
2. Marcello La Rocca, "Advanced Algorithms and Data Structures" 1st Edition, Paperback, 2021.
3. Narasimha Karumanchi, "Data Structures and Algorithms Made Easy", 5th Edition, CareerMonk Publications, 2021.
4. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein, "Introduction to Algorithms", 4th Edition, MIT Press, 2022.

REFERENCE BOOKS

1. Robert Sedgewick and Kevin Wayne, "Algorithms", 4th Edition, Addison-Wesley, 2022.
2. Ellis Horowitz, Sartaj Sahni, and Susan Anderson-Freed, "Fundamentals of Data Structures in C", 2nd Edition, Universities Press, 2020.
3. Reema Thareja, "Data Structures Using C", 2nd Edition, Oxford University Press, 2021.

L: 30;P:30; TOTAL: 60 PERIODS

Course Code		L	T	P	E	C
23CS05N	**SQL PROGRAMMING**	**2**	**0**	**2**	**0**	**3**

COURSE OUTCOMES

Upon the completion of the course the students will be able to

Theory Component

CO1:demonstrate the usage of Data Definition Language (DDL) and Data Manipulation Language (DML) statements in SQL.(CDL1)

CO2: construct and analyze complex SQL queries using advanced query structures like subqueries, set operators, and aggregate functions.(CDL1)

CO3: apply advanced SQL operations such as Joins and Stored Procedures to analyze and solve real-time database scenarios. (CDL1)

CO4: integrate NoSQL database concepts and develop real-time applications using non-relational data models. (CDL1)

Practical Component

CO5: design and implement real-time database solutions using SQL by applying DDL, DML, clauses, subqueries, joins, stored procedures, and triggers. (PDL1)

CO6: design and implement real-time applications using NoSQL databases by configuring and performing CRUD operations in Firebase and MongoDB. (PDL2)

CO1: Demonstrate the usage of Data Definition Language (DDL) and Data Manipulation Language (DML) statements in SQL.

Introduction – Data types: DDL Commands: Create – Rename – Alter –Truncate – Drop – Temp table –SQL Keys: Primary Key –Foreign Key – Composite key – Unique Key – Alternate Key. DML Commands: Select – Insert – Where clause –Sql Aliases –Operators - Update – Delete – SQL Aggregate Functions. **L:8**

CO2: Construct and analyze complex SQL queries using advanced query structures like subqueries.

SQL Clauses: Order by- Asc – Desc-Random - Limit - Group By – Having - Sub Query – Nested Sub Query –Data Control Language: Privileges – Grant –Revoke – SQL Case statements -Null Functions – Sequences – Clone tables –SQL Injections. **L:7**

CO3: Apply advanced SQL operations such as Joins and Stored Procedures to analyze and solve real-time database scenarios.

SQL Joins –Types:Self Join -Inner Join - Outer Join - Left Join – Right Join-Full -outer Join – views – Case studies: Customer Sale Scenario –SQL Procedures - PL/SQL Functionalities - Control statements –Functions – Cursor -Triggers – Exceptions **L:7;**

CO5: Design and implement real-time database solutions using SQL by applying DDL, DML, clauses, subqueries, joins, stored procedures, and triggers. **(PDL1)** **P:15**

Perform Database Manipulations operations using DDL, DML commands, Writing SQL Queries to retrieve data from multiple relations, Develop PL/SQL Programs to demonstrate the concept of Control statements –Procedures -Functions – Cursors – Triggers – Exceptions.

CO4: Integrate NoSQL database concepts and develop real-time applications using non-relational data models

FIRE BASE: Real time Database – Setup and Configuration –Data Organization – CRUD Operations – MONGO DB : Setup and Configuration – CRUD Operations – Case Study : Real time applications using FIREBASE / Mongo DB . **L:8;**

CO6: Design and develop real-time applications using NoSQL databases by configuring and performing CRUD operations in Firebase and MongoDB.(PDL2) **P:15**

Firebase Setup and Configuration, Firebase Real-time Database – CRUD Operations, MongoDB Setup and Configuration, MongoDB – CRUD Operations,Developing a real-time Application Using Firebase or MongoDB.

TEXT BOOKS

1. Silberschatz, A, Henry F. Korth, and S. Sudharshan, ―Database System Concepts‖, 7th Edition, Tata McGraw Hill, 2019.
2. David M. Kroenke, David J. Auer, Scott L. Vandenberg, Robert C. Yoder, Database Concepts‖, 9th Edition, Pearson Education, 2020.
3. R.Elmasri, S.B.Navathe, "Fundamentals of Database Systems", 6th Edition, Pearson Education, Addison Wesley, 2017.
4. Sadalage, P. & Fowler, NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence‖, 1st Edition, Wiley Publications, 2019.
5. Adam Fowler, NoSQL for Dummies, 2nd Edition, Wiley Publications, 2020.

REFERENCE BOOKS

1. Ramez Elmasri and Shamkant B. Navathe, Fundamentals of Database Systems, 7th Edition, Pearson, Addision Wesley, 2019.
2. Peter Membrey, "MongoDB Basics", Apress; 1st Edition, 2014.
3. Ian Robinson, Jim Webber, Emil Eifrem "Graph Databases: New Opportunities for Connected Data", 2nd Edition, "O'Reilly Media, Inc.", 2015.

L: 30; P: 30 TOTAL: 60 PERIODS

Course Code		**L**	**T**	**P**	**E**	**C**
23CS06N	**FULLSTACK DEVELOPMENT**	**2**	**0**	**2**	**0**	**3**

COURSE OUTCOMES

Upon the completion of the course the students will be able to

Theory Component

CO1: employ the MVC architecture and the DOM to develop interactive and dynamic websites (CDL1)

CO2: apply React and effective state management to design robust and responsive user interfaces (CDL1)

CO3: design robust React applications using Redux and modern styling frameworks (CDL2)

CO4: develop high performance and scalable web applications using Node.js (CDL2)

Practical Component

CO5: construct client-side interactive applications using JavaScript and React (PDL2)

CO6: develop robust web applications using Node.js (PDL2)

CO1: Employ the MVC architecture and the DOM to develop interactive and dynamic websites (CDL1)

MVC Architecture – JavaScript: Data types – Control statements– Functions – Hoisting – Objects and Arrays – Array :static and instance methods – Immutable arrays– Event handling– Form Validation – Document Object Model (DOM) – Arrow function - Promises – Async/Await – Modules. **L:8; P:8**

CO5: Construct client-side interactive applications using JavaScript and React (PDL2)

A simple application using forms in JavaScript – Design a client side interactive website.

CO2: Apply React and effective state management to design robust and responsive user interfaces (CDL1)

Introduction to React – React DOM and Virtual DOM – JSX – Components: Functional and Class components – Components Lifecycle – Event Handling - **State Management**: Context API, Prop drilling problem- React Hooks: useState, useEffect, useContext, useRef, Custom Hooks. Conditional rendering - List and Keys – Routing. **L:8; P:8**

CO5: Construct client-side interactive applications using JavaScript and React (PDL2)

A simple web application to implement routing for seamless navigation between different tasks.

CO3: Design robust React applications using Redux and modern styling frameworks (CDL2) Introduction to Redux, Redux Architecture: State – Action - Reducers- Color Reducers – Sort Reducers – Store - Action Creators – Passing the store - API calls – Error Handling –Connecting Redux with React - Styling Frameworks. **L:8; P:6**

CO6: Develop robust web applications using Node.js (PDL2)

A simple application for theme changer using Redux with Reacts and Styling Frameworks

CO4: Develop high performance and scalable web applications using Node.js (CDL2)

Introduction to Node.js - Event driven Architecture – Event Loop – Modules - Node Package Manager (NPM) - Understanding callbacks –REST API architecture –REST Methods - Introduction to Express framework –Server side rendering with Template engines – static files – Fetching JSON from Express.js – MongoDB – CRUD operation – Container – Docker. **L:6; P:8**

CO6: Develop robust web applications using Node.js(PDL2)

Setup an Express.js application – Implement CRUD Operation using MongoDB.

TEXT BOOKS

1. Mario Casciaro and Luciano Mammino, “Node.js Design Patterns”, 3rd Edition, O'Reilly Media, 2022.
2. Alex Banks,Eve Porcello, “Learning React – Modern Patterns for developing React apps”, 2nd Edition,O’Reilly Media, 2020.

REFERENCE BOOK

1. Paul Deitel, Harvey Deitel and Abbey Deitel, “Internet and World Wide Web-How to Program”, Prentice Hall, 5th Edition, 2018.

L: 30; P: 30 TOTAL: 60 PERIODS

Course Code	LINEAR ALGEBRA, MATHEMATICAL LOGIC AND SET THEORY	L	T	P	E	C
23SH01E		2	1	0	0	3

COURSE OUTCOMES:

Upon successfully completing the course, the students will be able to:

CO1: analyze concepts of vector spaces. (CDL 1)

CO2: measure the similarity between different datasets using Inner product spaces. (CDL 1)

CO3: decompose the matrix for computational convenience. (CDL 1)

CO4: illustrate the validity of the arguments. (CDL 1)

CO5**:** analyze the concepts of Sets, Relations and Functions. (CDL 1)

CO1: analyze concepts of vector spaces

Vector spaces – Subspaces – Linear combinations – linear span - Linear independence and linear dependence – Bases and dimensions. **L:6; T:3**

CO2: measure the similarity between different datasets using Inner product spaces

Linear transformation - Null spaces and ranges – Rank Nullity theorem - Matrix representation of a linear transformations - Inner product space - Norms - Orthonormal Vectors - Gram Schmidt orthogonalisation process**.** **L:6; T:3**

CO3: decompose the matrix for computational convenience

Generalized eigenvector - QR decomposition- generalized inverse - Singular value decomposition and applications – Pseudo Inverse. **L:6; T:3**

CO4: illustrate the validity of the arguments.

Propositional Logic – Equivalences and Implications – Normal forms – Predicate Calculus and Quantifiers - Rules of inference – Proof methods and Strategies - Forward Proof, Proof by Contradiction, Proof by Contraposition, Proof of Necessity and Sufficiency. **L:6; T:3**

CO5: analyze the concepts of Sets, Relations and Functions

Basic Definitions - Set operations – Laws of set theory – Relations – Properties of relations - Partial Ordering Relation - Equivalence Relation - Matrices of relations - Closure of relations – Functions – Bijective functions - Inverse and Compositions of functions. **L:6; T:3**

TEXT BOOKS

1. Kenneth H.Rosen, Discrete Mathematics and its Applications (with Combinatory andGraph Theory), Special Indian Edition, Tata McGraw-Hill Publishing CompanyLimited, New Delhi, 8th Edition, 2021.
2. Bernard Kolman and David Hill, "Elementary Linear Algebra with Application" Pearson India, 9th Edition, 2019.
3. Erwin Kreyszig, "Advanced Engineering Mathematics", 10thEdition, Wiley India, 2017.

REFERENCE BOOKS

1. Trembly J.P and Manohar.R. Discrete Mathematical Structures with Applications to Computer Science, 1st Edition, Tata McGraw-Hill Pub. Company Limited, New Delhi,2017.
2. Friedberg, A.H., I nsel,A.J.and Spence, L., Elementary Linear Algebra, a matrix approach, 2nd Edition pearson Publication.
3. Raju.K.George and Abhijith Ajayakumar, A course in Linear Algebra, Springer, 2024.
4. Seymour Lipschutz Marc Lipson., "Linear Algebra" Schaum's Out lines series, 6th Edition, McGraw – Hill Education, 2018.

L : 30; T :15; TOTAL : 45 PERIODS

Course Code	LINEAR STRUCTURES AND	L	T	P	E	C
23SH02E	TRANSFORMATIONS	2	1	0	0	3

COURSE OUTCOMES:

Upon successfully completing the course, the students will be able to:

CO1: solve the linear system of equations. (CDL 1)

CO2: determine the dimension of vector spaces. (CDL 1)

CO3: find the orthonormal vectors using Inner product spaces. (CDL 1)

CO4: illustrate Jordan canonical form on a finite dimensional vector space. (CDL 1)

CO5:decompose the matrix using Generalized Eigen vectors for computation.(CDL 1)

CO 1: solve the linear system of equations **L:6**

General system of linear equations – Matrices– Echelon form of matrix- Solving linear systems- Consistency of a system of linear equations -LU factorization. **T:3**

CO2: determine the dimension of vector spaces

Vector spaces – Subspaces – Linear combinations – linear span - Linear independence and linear dependence – Bases and dimensions. **L:6 T:3**

CO3: find the orthonormal vectors using Inner product spaces

Linear transformation - Null spaces and ranges – Rank Nullity Theorem - Matrix representation of a lineartransformations - Inner product space - Norms - Orthonormal Vectors - Gram Schmidt orthogonalisation process. **L:6 T:3**

CO4: illustrate Jordan canonical form on a finite dimensional vector space

Generalized eigenvector- Chains- Canonical basis the minimum polynomial- - Algebraic and Geometric multiplicity of Eigen Values - Similar matrices-Modal matrix-Jordan canonical form. **L:6 T:3**

CO5: decompose the matrix using Generalized Eigen vectors for computation

Eigen-values using QR transformations – Generalized Inverse Eigen vectors – Canonical forms – Singular value decomposition and applications – Pseudo inverse **L:6 T:3**

TEXT BOOKS

1. Bernard Kolman and David Hill, "Elementary Linear Algebra with Application" Pearson India, 9th Edition 2019.
2. Seymour Lipschutz Marc Lipson., "Linear Algebra" Schaum's Out lines series, Six edition, McGraw – Hill Education, 2018.

REFERENCE BOOKS

1. Friedberg, A.H., I nsel,A.J.and Spence, L., Elementary Linear Algebra, A Matrix Approach, 2nd Edition, Pearson 2019.
2. Jim Defranza. Daniel Gagiardi "Introduction to Linear Algebra with Applications" Waveland Pr Lnk, 2015.
3. Eggar.Goodaire "Linear Algebra Pure & Applied", World Sceintific, New Delhi, first edition, 2015.
4. Raju.K.George and Abhijith Ajayakumar, A course in Linear Algebra, Springer, 2024.

L : 30; T :15; TOTAL : 45 PERIODS

Course Code		L	T	P	E	C
23SH03E	**NUMBER THEORY**	**2**	**1**	**0**	**0**	**3**

COURSE OUTCOMES:

Upon successfully completing the course, the students will be able to:

CO1: acquire the concepts of theory of numbers. (CDL 1)

CO2: apply the fundamental propositions to interpret solutions of congruence. (CDL 1)

CO3: find the primitive roots for the congruence. (CDL 1)

CO4: analyze the inter-relation between arithmetical functions. (CDL 1)

CO5: determine quadratic residues of congruence. (CDL 1)

CO1 : acquire the concepts of theory of numbers

Introduction – Divisibility- Greatest common divisor - Prime numbers - The fundamental theorem of arithmetic - The series of reciprocals of the primes - The Euclidean algorithm(without Proof) - The greatest common divisor of more than two numbers. **L:6; T:3**

CO2 : apply the fundamental propositions to interpret solutions of congruence

Congruence - Linear congruence - Euler-Fermat theorem - Polynomial congruence modulo p – Wilson's Theorem **L:6; T:3**

CO3: analyze the inter-relation between arithmetical functions.

The Mobius function μ(n) – The Euler Totient function φ(n)– A relation connecting φ and μ – A product formula for φ(n) – properties of φ(n) –Multiplicative functions– completely multiplicative function. **L:6; T:3**

CO4: determine quadratic residues of congruence

Quadratic Residues – Legendre's symbol and its properties – Evaluation of (-1|p) and (2|p) – Gauss lemma – The Quadratic Reciprocity law – Applications – The Jacobi symbol. **L:6; T:3**

CO5: implement the concepts of congruence in cryptography

Chinese remainder theorem - Applications of Chinese remainder theorem - Cryptography and its application – RSA algorithm and Rabin Cryptosystem. **L:6; T:3**

TEXT BOOKS

1. Tom M.Apostol, "Introduction to Analytic Number Theory", Springer International Edition, Narosa Publishing House, New Delhi, 2013.
2. G.A.Jones& J.M.Jones, "Elementary Number Theory", Springer publications, 2012.

REFERENCE BOOKS

1. David M.Burton, "Elementary Number Theory", McGraw Hill, 7th Edition,2023
2. Joseph H.Silverman, "A Friendly Introduction to Number Theory", Pearson Education, 4th Edition, 2019.
3. Titu Andreesu, Gabriel Dospinescu, Oleg MushKarov, Number Theory: concepts and problems, Springer Science & Business Media, 2017.
4. S B Malik , "Basic Number Theory", S Chand publications, 2nd Edition, 2018

L : 30; T :15; TOTAL : 45 PERIODS

Course Code	NUMERICAL ANALYSIS	L	T	P	E	C
23SH04E		2	1	0	0	3

COURSE OUTCOMES:

Upon successfully completing the course, the students will be able to:

CO1: solve algebraic and transcendental equations using numerical methods. (CDL 1)

CO2: interpolate and approximate the polynomial of data. (CDL 1)

CO3: perform numerical differentiation and integration. (CDL 1)

CO4: find numerical solution of ordinary differential equation. (CDL 1)

CO5: classify and find numerical solution of partial differential equations. (CDL 1)

CO 1: solve algebraic and transcendental equations using numerical methods

Solutions of Algebraic linear equations Newton - Raphson Method, Fixed Point Iteration method - Solutions of algebraic simultaneous linear equations - Gauss Elimination –Gauss Seidel Methods. **L:6; T:3**

CO 2: interpolate and approximate the polynomial of data

Curve Fitting – Method of Least Squares – Fitting a Straight Line – Fitting a Second Degree Parabola - Finite differences - Newton's Forward & Backward Difference Formulae - Central Differences - Stirling's Formula - Lagrange's Formula. **L:6; T:3**

CO 3: perform numerical differentiation and integration

Derivatives using forward and backward difference Formulae - Trapezoidal rule - Simpson's rules - Double integration using Trapezoidal and Simpson's rules. **L:6; T:3**

CO 4: find numerical solution of ordinary differential equation

Taylor's Series Method - Euler's Method – Runge Kutta fourth order Method – Predictor - corrector Methods - Milne's Method - Finite difference for solving ordinary differential equation. **L:6; T:3**

CO 5: classify and find numerical solution of partial differential equations

Classification of Partial Differential Equations of second order - Finite difference solution of one dimensional heat equation by explicit and implicit methods (Crank Nichelson and Bender Schmidth methods) - One dimensional wave equation and two dimensional Laplace and Poisson equations. **L:6; T:3**

TEXT BOOKS

1. Grewal, B.S., "Numerical Methods in Engineering & Science: With Programs in C, C++ & MATLAB", 11th Edition, Khanna Publishers, New Delhi, 2014.
2. M.K.Jain, S.R.K.Iyengar, R.K.Jain "Numerical Methods for scientific and Engineering Computation", 6th Edition, New age International Publishers, 2019.
3. Erwin Kreyszig, Advanced Engineering Mathematics, 10th Edition, J. Wiley and Sons, 2023.

REFERENCE BOOKS

1. Chapra, S. C and Canale, R. P. "Numerical Methods for Engineers", 8th Edition, Tata McGrawHill, New Delhi, 2021.
2. Saumyen Guha, Rajesh Srivastava "Numerical Methods: For Engineering and Science", Oxford University Press, New Delhi, 1st Edition with third impression, 2015.
3. K.Sankara Rao , "Numerical Methods For Scientists And Engineers", 5th Edition, New age International Publisher, 2018
4. Dr Chaitanya Kumar, Dr Harinderjit Kaur Chawla, Dr Indarpal Singh "A Textbook on Numerical Methods and Analysis" Sultan Chand and SonsPublisher, 2024

L : 30; T :15; TOTAL : 45 PERIODS

Course Code		L	T	P	E	C
23SH05E	**OPTIMIZATION TECHNIQUES**	**2**	**1**	**0**	**0**	**3**

COURSE OUTCOMES:

Upon successfully completing the course, the students will be able to:

CO1: find optimum solution of linear programming problem. (CDL 1)

CO2: determine the optimum schedule for assignment and transportation problems. (CDL 1)

CO3: acquire decision making in Pure and Mixed Strategies. (CDL 1)

CO4: analyze the network for optimal schedule. (CDL 1)

CO5:compute optimum solution of non-linear programming. (CDL 1)

CO1: Find optimum solution of linear programming problem

Linear Programming Problem – Mathematical Formulation of Linear Programming Problems (LPP) – Graphical Solution Method - Canonical and Standard Forms of LPP - Simplex Method - Linear Programming using Artificial Variables - Two Phase Method. **L:6; T:3**

CO2: Acquire decision making in Pure and Mixed Strategies

Basic Terms in Game Theory - Two-Person Zero-Sum Games - Maximin-Minimax Principal - Games without Saddle Points - Mixed Strategies-Pure and Mixed Strategies with Saddle Point- Mixed Strategy Problems by Arithmetic Method- Graphic Solution of $2 \times n$ and $m \times 2$ Games. **L:6; T:3**

CO3: Analyze the network for optimal schedule

Development of Network Analysis - Network Analysis and Rules of Network Construction - Critical Path Method (CPM) - Programme Evaluation and Review Technique (PERT). **L:6; T:3**

CO4: Compute optimum solution of non – linear programming

Formulating a Non-Linear Programming Problem – Constrained Optimization with equality Constraints- Graphical Solution – Kuhn- Tucker Conditions with Non negative constraints- Quadratic Programming – Wolfe's modified Simplex method. **L:6; T:3**

CO5: Solve non-linear constrained optimization

Optimization using Gradient Descent – Constrained optimization - Lagrange Multipliers - Convex optimization - Non linear Constrained Optimization. **L:6; T:3**

TEXT BOOKS

1. KantiSwarup, Gupta P.K and Man Mohan, Operations Research: Introduction to management Science, Sultan Chand &Sons, 20th Revised Edition, 2022.
2. Hamdy A Taha, Operations Research - An Introduction, 10th Edition, Pearson Education, 2019.

REFERENCE BOOKS

1. Sharma JK., Operations Research, Trinity, New Delhi, 6th Edition, 2017.
2. Sundaresan.V, Ganapathy Subramanian. K.S. and Ganesan.K, Resource Management Techniques, A.R. Publications, 11th Edition, 2017.
3. Gupta P K, Mohan Man, Problems in Operations Research, Sultan Chand & Sons, 2014
4. V K Kapoor, Operations Research , Concept problems & solutions, Sultan Chand & Sons, 2017

L : 30; T :15; TOTAL : 45 PERIODS

Course Code		L	T	P	E	C
23SH06E	**PRINCIPLES OF DISCRETE MATHEMATICS**	2	1	0	0	3

COURSE OUTCOMES:

Upon successfully completing the course, the students will be able to:

CO1: illustrate the validity of the arguments. (CDL 1)
CO2: analyze the concepts of Sets, Relations and Functions. (CDL 1)
CO3: perform the principles of counting and solve recurrence relations. (CDL 1)
CO4: interpret the basic concepts of graphs. (CDL 1)
CO5: compute minimum Spanning Trees and shortest route for the graph. (CDL 1)

CO1: Illustrate the validity of the arguments.

Propositional Logic – Equivalences and Implications – Normal forms – Rules of inference – Proof methods and Strategies - Forward Proof, Proof by Contradiction, Proof by Contraposition, Proof of Necessity and Sufficiency. **L:6; T:3**

CO2: Analyze the concepts of Sets, Relations and Functions

Basic Definitions - Set operations – Laws of set theory – Relations – Properties of relations - Equivalence Relation - Matrices of relations - Closure of relations – Functions – Bijective functions - Inverse and Compositions of functions. **L:6; T:3**

CO3: Perform the principles of counting and solve recurrence relations.

Mathematical induction - Strong induction and well ordering -The basics of counting – The pigeonhole principle - Recurrence relations – Solving linear recurrence relations – Generating functions - Inclusion and exclusion principle. **L:6; T:3**

CO4: Interpret the basic concepts of graphs

Graphs and their properties - Special types of graphs – Matrix representation of graphs and graph isomorphism- Euler and Hamiltonian graphs. **L:6; T:3**

CO5: Compute minimum Spanning Trees and shortest route for the graph

Trees – Some properties of Trees – Pendant vertices in a Tree – Distance and centers in a Tree – Rooted and Binary Trees - Spanning Trees- minimum spanning tree–Prim's algorithm. **L:6; T:3**

TEXT BOOKS

1. Kenneth H.Rosen, Discrete Mathematics and its Applications (with Combinatory and Graph Theory), Special Indian Edition, Tata McGraw-Hill Publishing CompanyLimited, New Delhi, 8th Edition, 2021.
2. Trembly J.P and Manohar.R. Discrete Mathematical Structures with Applications to Computer Science, first Edition, Tata McGraw-Hill Pub. Company Limited, New Delhi, 2020.
3. Narsingh Deo, Graph Theory with Applications to Engineering and ComputerScience, 1st Edition, Dover Publications Inc., 2016.

REFERENCE BOOKS

1. Ralph .P. Grimaldi, Discrete and Combinatorial Mathematics: An Applied Introduction, 5th Edition, Pearson Education Asia, Delhi, 2019.
2. Bondy, J.A., Murty.U.S.R., Graph Theory with applications, North Holland publication, 2008.
3. K.Balakrishnan, Schaum's Outline of Graph Theory, Tata Mc Graw-Hill Pub, 2020.
4. Richard J.J, Introduction to Graph Theory, 1stEdition, Parker Pub. Company, 2017.

L : 30; T :15; TOTAL : 45 PERIODS

Course Code	**RANDOM PROCESSES AND QUEUEING THEORY**	**L**	**T**	**P**	**E**	**C**
23SH07E		**2**	**1**	**0**	**0**	**3**

COURSE OUTCOMES:

Upon successfully completing the course, the students will be able to:

CO1: interpret the basic characteristic features of Random processes. (CDL 1)

CO2: encapsulate the time averages of uncertain events. (CDL 1)

CO3: evaluate spectral densities of functions. (CDL 1)

CO4: analyze the characteristics of Markovian queues. (CDL 1)

CO5**:**apply the concepts of queuing theory in networks.(CDL 1)

CO1: interpret the basic characteristic features of Random processes **L:6; T:3**

Classification - Stationary process - Markov process - Markov chains - Transition probabilities.

CO2 : encapsulate the time averages of uncertain events **L:6; T:3**

Counting Process - Ergodic process - Poisson Process - Renewal Processes - Gaussian process.

CO3 :evaluate spectral densities of functions **L:6; T:3**

Auto correlation - Cross correlation – Power spectral density–Cross spectral density-Properties–Wiener – Khintchine theorem (without proof).

CO4 : analyze the characteristics of Markovian queues

Markovian models – Birth and Death Queuing models- Steady state results: Single and multiple server queuing models- queues with finite waiting rooms- Finite source models- Little's Formula. **L:6; T:3**

CO5: apply the concepts of queuing theory in networks **L:6; T:3**

M/G/1 queue- Pollaczek- Khintchine formula, series queues- open and closed networks.

TEXT BOOKS

1. Oliver C. Ibe, "Fundamentals of Applied Probability and Random processes", Academic Press, 2nd Edition, 2014.
2. Hwei Hsu, "Schaum's Outline of Theory and Problems of Probability, Random Variables and Random Processes, Tata McGraw-Hill Education, 3rdEdition, 2017.
3. John F Shortle, James M Thompson, Donald Gross and Carl M Harris, "Fundamentals of Queueing Theory", Wiley and Sons Publication Limited, 5thEdition, 2018.

REFERENCE BOOKS

1. Miller.S.L and Childers, S.L, Probability and Random Processes with applications to Signal Processing and Communications, Elsevier Inc., 2nd Edition, 2012.
2. Peyton. Z. Peebles Jr., Probability Random Variables and Random Signal Principles, 4th Edition, Tata McGraw-Hill Publishers, New Delhi, 2017.
3. Erwin Kreyszig, "Advanced Engineering Mathematics", 10thEdition, Wiley India, 2017.

L : 30; T :15; TOTAL : 45 PERIODS

Course Code	STATISTICAL TECHNIQUES AND NUMERICAL	L	T	P	E	C
23SH08E	METHODS	2	1	0	0	3

COURSE OUTCOMES:

Upon successfully completing the course, the students will be able to:

CO1: calculate the various measures of dispersion. (CDL 1)

CO2: apply the principles of hypothesis testing in small and large samples. (CDL 1)

CO3: analyze the variances in design of experiments. (CDL 1)

CO4: find solution of linear equations and to perform differentiation and integration numerically. (CDL 1)

CO5: compute numerical solution of differential equations. (CDL 1)

CO1: calculate the various measures of dispersion

Central tendencies - Mean, median, mode - Measures of Dispersion –Mean deviation, and Quartile deviation–Moments– Skewness –Kurtosis - Correlation and Regression. **L:6; T:3**

CO2: apply the principles of hypothesis testing in small and large samples

Sampling distributions - Statistical hypothesis - Large sample testsbased on Normal distribution for single mean and difference of means -Tests based on t, and F distributions - Chi-square -Contingency table for independent of attributes – Goodness of fit. **L:6; T:3**

CO3: analyze the variances in design of experiments

One way and two way classifications - Completely randomized design – Randomized block design – Latin square design – 2^2 factorial design. **L:6; T:3**

CO4: find solution of linear equations and to perform differentiation and integrationnumerically

Solution of algebraic and transcendental linear equations - Newton - Raphson Method- Solution of simultaneous equations – Gauss Elimination method – Gauss Seidel method – Interpolation – Lagrange's Method - Numerical Differentiation – Newton's forward difference and backward difference formula – Numerical integration - Single integration using Trapezoidal and Simpson's 1/3 rd and 3/8 th rules. **L:6; T:3**

CO5: compute numerical solution of differential equations

Taylor's Series Method – Euler's Method – Runge Kutta fourth order Method – Predictor - corrector Methods – Milne's Method - Solution of one dimensional heat equation by explicit and implicit methods(Crank Nichelson and Bender Schmidth methods) - Two dimensional Laplace and Poisson equations. **L:6; T:3**

TEXT BOOKS

1. Richard A. Johnson, "Miller and Freund's Probability and Statistics for Engineers", 9th Edition, Pearson Education Private Ltd., 2018.
2. Grewal, B.S., "Numerical Methods in Engineering & Science: With Programs in C, C++ &MATLAB", 11th Edition, Khanna Publishers, New Delhi, 2014.

REFERENCE BOOKS

1. Dharmaraja Selvamuthu, Dipayan Das, Introduction to Statistical Methods, Design of Experiments and Statistical Quality Control, Springer Verleg Singapore Pvt. Ltd., 2018.
2. S.C. Gupta and V.K. Kapoor, "Fundamentals of Mathematical Statistics, 12th Edition, Sultan Chand & Sons, Delhi, 2014.
3. M.K.Jain.S.R.K.Iyengar,R.K.Jain "Numerical Methods for scientific and Engineering Computation", 6th Edition, New age International Publishers, 2019.
4. Chapra, S. C and Canale, R. P. "Numerical Methods for Engineers", 8th Edition, Tata McGraw - Hill, New Delhi, 2021.

L : 30; T :15; TOTAL : 45 PERIODS

Course Code	TRANSFORMS, MATHEMATICAL LOGIC AND SET	L	T	P	E	C
23SH09E	THEORY	2	1	0	0	3

COURSE OUTCOMES:

Upon successfully completing the course, the students will be able to:

Theory Components:

CO1: apply Laplace transform to solve ordinary differential equations. (CDL 1)
CO2: compute the Fourier transforms of various functions. (CDL 1)
CO3: solve difference equations using Z-Transform. (CDL 1)
CO4: illustrate the validity of the arguments. (CDL 1)
CO5**:**analyze the concepts of Sets, Relations and Functions. (CDL 1)

CO 1 : Apply Laplace transform to solve ordinary differential equations

Definition of Laplace transform and its inverse – Transforms of elementary functions – Properties – Transforms of periodic functions – Initial and final value theorems – Convolution theorem.- solutions of linear ordinary differential equations with constant coefficients. **L:6; T:3**

CO2 : Compute the Fourier transforms of various functions

Fourier Integral theorem (without proof)–Fourier transform pair–Fourier Sine and Cosine transforms–Properties–Transforms of simple functions–Convolution theorem –Parseval's theorem. **L:6; T:3**

CO3 : Solve difference equations using Z-Transform

Z– transform –Elementary properties – Inverse Z–transform – Convolution theorem- Initial and final value theorem – Formation of difference equations –Solutions of difference equations using Z–transform. **L:6; T:3**

CO4: Illustrate the validity of the arguments.

Propositional Logic – Equivalences and Implications – Normal forms – Rules of inference – Proof methods and Strategies - Forward Proof, Proof by Contradiction, Proof by Contraposition, Proof of Necessity and Sufficiency. **L:6; T:3**

CO5: Analyze the concepts of Sets, Relations and Functions

Basic Definitions - Set operations – Laws of set theory – Relations – Properties of relations - Equivalence Relation - Matrices of relations - Closure of relations – Functions – Bijective functions - Inverse and Compositions of functions **L:6; T:3**

TEXT BOOKS

1. Grewal.B.S. "Higher Engineering Mathematics", 44th Edition, Khanna Publications, Delhi, 2021.
2. Kenneth H.Rosen, Discrete Mathematics and its Applications, Tata McGraw-Hill Publishing Company Limited, New Delhi, 8th Edition, 2021.
3. Erwin Kreyszig, "Advanced Engineering Mathematics", 10thEdition, Wiley India, 2017.

REFERENCE BOOKS

1. Ramana B.V, "Higher Engineering Mathematics", Tata Mc-Graw Hill Education, New Delhi, 2017.
2. Trembly J.P and Manohar.R. Discrete Mathematical Structures with Applications to Computer Science, 1st Edition, Tata McGraw-Hill Pub. Company Limited, New Delhi, 2017.
3. J K Goyal, K.P.Gupta, Laplace and Fourier Transforms, Pragati Prakashan, 2016

L : 30; T :15; TOTAL : 45 PERIODS

Course Code	**FUNDAMENTALS OF LASER TECHNOLOGY**	**L**	**T**	**P**	**E**	**C**
23SH10E		**3**	**0**	**0**	**0**	**3**

COURSE OUTCOMES:

Upon successfully completing the course , the students will be able to:

Theory Components:

CO1: explain the fundamentals of lasers (CDL1)

CO2: demonstrate the laser surface modification process (CDL1)

CO3: describe the laser machining processes (CDL1)

CO4: identify the laser measurement and testing process (CDL1)

CO5: organizethe advanced applications and safety measures of laser (CDL1)

CO1: Explain the fundamentals of lasers

Characteristics of laser -laser principle- population inversion-line broadening mechanisms-Q switching - threshold condition for laser-three-level and four-level systems-conditions for continuous wave (CW) and pulsed laser action- pumping schemes-classification of lasers:Er:YAG - carbon dioxide lasers - argon laser - X-Ray lasers - fiber lasers - Raman lasers. **L:9**

CO2: Demonstrate the laser surface modification process

Laser surface heat treatment: process parameters - advantages and disadvantages of laser surface treatment; laser surface melting - laser direct metal deposition: processing parameters - methods for applying the coating material- laser alloying and cladding - advantages and disadvantages -laser physical vapor deposition - laser shock peening: analysis - advantages and disadvantages **L:9**

CO3: Describe the laser machining processes

Laser welding parameters: beam power, spot diameter and traverse speed; welding efficiency; mechanism of laser welding: conduction mode welding, keyhole welding; laser cutting – process characteristics-fusion cutting, sublimation cutting, photochemical ablation;laser drilling –single pulse drilling-percussion drilling, trepanning applications - laser marking - dot matrix marking, engraving, image micro machining -lasers for marking - application **L:9**

CO4: Identify the laser measurement and testing process

Laser for measurement - distance -length-velocity-acceleration-current-voltage-atmospheric effect-laser application in spatial frequency filtering. **L:9**

Holography: basic principle - methods - Holographic interferometry and applications-holography for non – destructive testing – holographic components

CO5: Organize the advanced applications and safety measures of laser

Laser advanced application in defence-laser weapons- industry for material handling: ASRS and AGV- medicine -laser activated therapy - photodynamic therapy, laser angioplasty, lasers in surgery - photocoagulation, photodisruption and photoablation - laser scanning confocal microscopy - Laser safety - danger - safety limits for eye and skin - class four safety arrangements - electric hazards- chemical hazards - fume hazards - explosion hazards - safety guidelines **L:9**

TEXTBOOKS:

1. William M. Steen, “Laser Material Processing”, Springer Verlag, 2010
2. K.Thyagarajan, AjoyK.Ghatak, “Lasers, Theory and Applications”, Springer, 2nd Edition, 2011.
3. Chunlei Guo, Subhash Chandra SinghHandbook of Laser Technology and Applications Lasers

Applications: Materials Processing and Spectroscopy, 2nd Edition, (Vol.3), 2021

REFERENCES:

1. Uday Shanker Dixit, Shrikrishna N. Joshi, J. Paulo Davim, "Application of Lasers in Manufacturing" Springer Singapore, 1stEdition, 2019
2. Stephan Wieneke and Christoph Gerhard, "Lasers in Medical Diagnosis and Therapy Basics, applications and future prospects" IOP Publishing Ltd, 2018
3. AK Katiyar, CK Pandey and Manisha Bajpai, "Fundamentals of Laser Systems and Applications", Wiley, 2017.

L :45; TOTAL : 45 PERIODS

Course Code		L	T	P	E	C
23SH11E	**NANOMATERIALS FOR ENGINEERS**	**3**	**0**	**0**	**0**	**3**

COURSE OUTCOMES:

Upon successfully completing the course , the students will be able to:

Theory Components:

CO1: explain the fundamentals of nanomaterials (CDL1)
CO2: interpret the different properties of nanomaterials (CDL1)
CO3: demonstrate the synthesis of nanomaterials (CDL1)
CO4: illustrate the characterization of nanomaterials (CDL1)
CO5: organize the applications of nanomaterials(CDL1)

CO1: Explain the fundamentals of nanomaterials

Introduction to nanomaterials - size effect - specific surface area - surface to volume ratio - quantum confinement effects - morphology - density - melting point - wettability - classification based on the dimension - nanoparticles - nanowires - nanoclusters - nanotubes - quantum wells - metal based nanomaterials - nanocomposites - carbon nanotubes - nanosized metals - alloys - semiconductors - ceramics **L:9**

CO2: Interpret the different properties of nanomaterials

Mechanical behavior- comparison of bulk and nano materials - elastic and plastic deformation - tensile strength - superplasticity -hardness - nano hardness -influence of porosity - grain size – thermodynamics of nanoparticles- heat capacity – phase transformation of nanoparticles- electrical and optical properties: electrical conductivity in nano tubes, nano rods and nanocomposites - photoconductivity of nanorods - electroluminiscesnce in nanoparticles- magnetic properties: magnetic hysteresis - superparamagnetism **L:9**

CO3: Demonstrate the synthesis of nanomaterials

Bottom-up and top-down approach - inert gas condensation - plasma arc technique - ion sputtering - ball milling - molecular beam epitaxy - chemical vapour deposition - method - electrodeposition - ultrasonication - microemulsions method - solvothermal synthesis - microwave assisted synthesis. **L:9**

CO4: Illustrate the characterization of nanomaterials

X-ray diffraction - energy dispersive spectrum - atomic force microscopy - high resolution transmission electron microscopy - Raman spectroscopy - x-ray photoelectron spectroscopy - electrochemcial characterization measurements - cyclic voltammetry - linear sweep voltametry - Brunauer-Emmett-Teller - surface area analysis - nanoindentation - determination of nano hardness. **L:9**

CO5: Organize the applications of nanomaterials

Functional graphene - carbon nanotube - polymer composite applications in defence and aerospace - nanomaterials for solar cells - nanoscale catalysts for energy and automobile industries - rechargeable batteries based on nanomaterials - nanomaterials for electrodes and wearable electronics - nano based coating and paints - nanosensors -gas sensors - bio sensors - nano electro mechanical systems **L:9**

TEXTBOOKS:

1. Charles P Poole, Frank J Ownes, Introduction to Nanoscience and Nanotechnology, An Indian Adaption, Wiley, 2020
2. Hornyak, G.Louis, Tibbals, H.F., Dutta, Joydeep, Fundamentals of Nanotechnology, CRC Press, 1st Edition, 2018
3. Dieter Vollath, Nanomaterials an introduction to synthesis, properties and applications, Wiley, 2nd Edition, 2013

REFERENCES:

1. Narendra Kumar, Sunita Kumbhat, Essentials in Nanoscience and Nanotechnology, Wiley, 1st Edition, 2016
2. G. Cao,Ying Wang, Nanostructures and nanomaterials: Synthesis, properties and applications, Imperial College Press, 2nd Edition, 2011
3. B.S. Murty , P. Shankar , Baldev Raj , B B Rath , James Murday, Textbook of Nanoscience and Nanotechnology, Springer, 1st Edition, 2013

L : 45; TOTAL : 45 PERIODS

Course Code		L	T	P	E	C
23SH12E	**PHOTONICS**	**3**	**0**	**0**	**0**	**3**

COURSE OUTCOMES:

Upon successfully completing the course , the students will be able to:

Theory Components:

CO1: explain the basics of photonics (CDL1)
CO2: demonstrate the properties of photonic crystal (CDL1)
CO3: outline the basics of bio photonics (CDL1)
CO4: interpret the quantum confinement in photonic materials(CDL1)
CO5: organize the applications of photonic materials (CDL1)

CO1: Explain the basics of photonics

Wave phenomena – interference, diffraction-photon properties - energy, flux, statistics-Interaction of photons with atoms-optical amplification-three and four level system -EDFA-semiconductor light sources-detectors-light manipulation - birefringence - Faraday's rotation - interaction of light with RF and acoustic waves - Raman-Nath diffraction experiment . **L:9**

CO2: Demonstrate the properties of photonic crystal

Electromagnetic theory of light-electromagnetic properties of material- polarization of light; Reflection and refraction- Fresnel equations; absorption, dispersion, and scattering of electromagnetic waves -Bragg grating; 1D photonic crystals -photonic band structure-real and reciprocal lattices; 2D and 3D photonic crystals-emerging applications of photonic crystals - 1D Bragg grating - periodic dielectric wave guide - 2D photonic crystal slab and fibre. **L:9**

CO3: Outline the basics of bio photonics

Fundamentals of light and matter-basics of light-matter interactions in molecules, cells and tissues -lasers for biophotonics -bioimaging: principles and applications-transmission microscopy, Kohler illumination-optical biosensors-light activated therapy: photo thermal and photo dynamic therapy- tissue engineering with light- optical tweezers, scissors and traps - bio nanophotonics applications - bio chip - DNA micro-arrays - gene chip - lab on chip. **L:9**

CO4:Interpret the quantum confinement in photonic materials

Quantum confined materials: quantum wells, quantum wires, quantum dots, quantum rings, manifestations of quantum confinement, optical properties, quantum confined stark effect, dielectric confinement effect. **L:9**

Nanoplasmonics: optical response of metals, plasmons, optical properties of metal nanoparticles, size dependent absorption and scattering, coupled nanoparticles - metal-dielectric core-shell nanoparticles - local electromagnetic fields in metal nanoparticles.

CO5: Organize the applications of photonic materials

Excitation energy transfer – device operation: nanophotonic AND gate - nanophotonic OR gate – interconnection with photonic devices - metamaterials concept; super lens, hyperbolic metamaterials and application in high-resolution imaging: hyper lens - tunable photonic metamaterials based devices - electro-optical metamaterials - phase-change metamaterials - metamaterials in solar energy harvesting - perfect absorbers and thermal emitter **L:9**

TEXTBOOKS:

1. Bahaa E. A. Saleh, Malvin Carl Teich, Fundamentals of Photonics, 3rdEdition, Wiley,2019.
2. Brian Culshaw, Introducing Photonics, Cambridge University Press, 2020.
3. Gerd Keiser, Biophotonics: Concepts to Applications, second edition, Springer Nature Singapore Pvt. Ltd 2022.

REFERENCES:

1. Joseph W. Haus, Fundamentals and Applications of Nanophotonics, Woodhead Publishing, 2016.
2. W.Cai and V. Shalaev, Optical Metamaterials: Fundamentals and Applications, 2nd Edition, Springer, 2024.
3. P PYupapin, K Srinuanjan, S Kamoldilok, Devices, Circuits and Systems: Nanophotonics, Pan Stanford Publishing, 2013.
4. Paulo Ribeiro, Maria Raposo, "Optics, Photonics and Laser Technology", Springer International publishing, 1stEdition, 2018

L : 45; TOTAL : 45 PERIODS

Course Code		L	T	P	E	C
23SH13E	**BIOLOGY FOR COMPUTING**	**3**	**0**	**0**	**0**	**3**

COURSE OUTCOMES:

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

Theory Components:

CO1: describe the structure, interaction and applications of biomolecules
CO2: interpret the structure and functions of the gene and protein using the bioinformatics data
CO3: simulate the behavior of simple biological models using computational softwares
CO4: identify and design molecules for new drug development by computational methods

CO1: Describe the structure, interaction and applications of biomolecules **L:9**

Biomolecules-I :

Introduction – monomeric units and polymeric structures of carbohydrates, proteins, nucleic acids and lipids. Enzymes: enzymatic action via Lock and key – Enzyme therapy - immune response monitoring – molecular modification – encapsulation. Agarose gel electrophoresis: SDS, PAGE and 2D – Molecular interactions: covalent and non-covalent interactions, antigen – antibody interactions. Methods to measure the interactions: UV-visible and single crystal X-ray diffraction.

Biomolecules -II **L:9**

Chromosome structure and function – chromosome abnormalities – chromosome dynamics – nuclear architecture. DNA transcription, replication and segregation. DNA finger printing. Pedigree analysis. Identifying human disease genes (functional cloning versus positional cloning; mutation screening). Human genome project: introduction – steps – salient features. Hap map project – salient features.

CO2: Interpret the structure and functions of the gene and protein using the bioinformatics data **L:9**

Bioinformatics: introduction – biological databases – types. DNA databases – EMBL, gene bank, DDBJ. Protein databases: Swiss Prot/TrEMBL, PIR. Sequence motif databases - Pfam, PROSITE, Protein structure databases, protein data Bank – SCOP, CATH, and KEGG. Sequence analysis – methods of sequencing: sangar method, maxama - gilbert method and edman degradation method, NGS methods of sequencing. Basic local alignment search tool (BLAST) – types – determining the identity of an organism from its r DNA gene nucleotide sequence. Softwares for handling the databases – ChemDiff.

CO3: Simulate the behavior of simple biological models using computational softwares **L:9**

Quantum mechanics: influence of physics on theoretical chemistry. Semi empirical methods – slater determinants – Hartree – Fock equation. Semi empirical models - Ab-initio calculations: Thermodynamic functions – koopmans's theorem – isodesmic reactions, Density functional theory for larger molecules. Introduction to Gaussian and ADF : Geometry optimization, frequency calculation, location of transition state, intrinsic reaction co-ordinates, molecular orbitals and population analysis, natural bond orbital analysis, calculation of equilibrium constants and rate constants. Introduction to GROMACS: GROMACS input files, simulations of liquid water, water methanol mixtures, S-peptide and free energy of salvation. Introduction to SCILAB- Scilab programming: Curve fitting, integral transforms and introduction to molecular dynamics. Execution of programs for liquid argon.

CO4: Identify and design molecules for new drug development by computational methods **L:9**

Drug design: General approach to discovery of new drugs – lead modification – calculation of the various drug likeness rules like Lipinski's rule, MDDR - like rule, Veber rule, Ghose filter, BBB rule, CMC-50 like rule and Quantitative estimate of drug-likeness (QED) using DruLiTo and Swiss ADMEsoftware. Pharmacokinetic properties of drug using Osiris and Molinspiration software. Structure-based drug designing approaches - target identification and validation - physiochemical principles of drug action – drug stereo chemistry – drug action - 3D database – computer aided drug design. Identification of the suitable target using Pharmmapper - Molecular docking programs using Autovina softwares and visualization tools - Preparation of protein and ligand using ADT and pymol-generation of paper publication-quality images and data analysis-protein-protein docking-Protein DNA docking

TEXT BOOKS

1. Shawn T. O'Neil, A Primer for Computational Biology, Oregon State Campus, Corvallis, USA, 2019.
2. Frank Jensen, Introduction to Computational Chemistry, 3rdEdition, Wiley publishing LLC. USA, 2016
3. Philly Charles, Genes, Genomes, Genetics and Chromosomes, Nottinghamshire, England, 2020.

REFERENCE BOOKS

1. Ariel Fernández Stigliano, Biomolecular Interfaces: Interactions, Functions and Drug Design, 1stEdition, Springer International Publishing AG, London, 2016.
2. S.C. Rastogi, P.Rastogi, N.Mendiratta, Bioinformatics: Methods and Applications - Genomics, Proteomics and Drug Discovery, 5th Edition, PHI Learning Pvt. Ltd., Delhi, 2022.
3. Robert A. Copeland, Enzymes: A Practical Introduction to Structure, Mechanism, and Data Analysis, 3rdEdition, Wiley-Blackwell, New York, 2023.

L: 45; TOTAL: 45 PERIODS

Course Code		L	T	P	E	C
23SH14E	**BIOLOGICAL SYSTEMS FOR ENGINEERS**	**3**	**0**	**0**	**0**	**3**

COURSE OUTCOMES:

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

Theory Components:

CO1: understanding of bio design principles to create novel devices and structures and cell biology

CO2: explain the structure and stability of biomolecules

CO3: describe the principle, components and applications of various instruments for medical diagnosis

CO4: interpret the major bio-energetic pathways

CO5: explain the properties characterization and application of various biomaterials

CO1: Understand the basic principles of biology to create novel devices **L:9**

Cell - prokaryotic and eukaryotic cells - plant cell and animal cell - structural and function of Mitochondria - Chloroplast - Lysosomes - Golgi bodies - Nucleus. Cell cycle: mitosis and meiosis. Bioinspired devices: GPS, aircrafts, swim suits, bullet train, super hydrophobic and self-cleaning surfaces.

CO2: Explain the structure and stability of biomolecules **L:9**

Introduction - monomeric units and polymeric structures of carbohydrates, proteins, nucleic acids and lipids. Molecular interactions: covalent and non-covalent interactions – methods of quantification and determination: UV – visible, CD, and SPR.

Enzymes - classification - specific activity - enzyme activity - chemical nature of enzymes. Protein and non-protein nature of enzymes. Metalloenzymes and metal activated enzymes. Industrial applications of enzymes: biosensors and bio bleaching.

CO3: Describe the principle and applications of various instruments for medical diagnosis **L:9**

Basic concepts of instrumentation: static and dynamic characteristics, design criteria, instrumentation, amplifiers. Biopotential electrodes: fundamentals - body surface electrodes - microelectrodes - Principle, components and applications of microscope: light

and electron microscope. Electrocardiograph, glucometer, CT, magnetic resonance imaging, ultrasonic imaging. Artificial Intelligence for disease diagnosis.

CO4: Interpret the major bio-energetic pathways **L:9**

Thermodynamics in biological systems - exothermic and endothermic versus endergonic and exergonic reactions - concept of K_{eq} and its relation to standard free energy - spontaneity - ATP as an energy currency. Glucose synthesis from CO_2 + H_2O (photosynthesis) – decomposition of glucose (Glycolysis and Krebs cycle). Energy yielding and energy consuming reactions. Concept of energy charge. Regulation of glycogenesis - measurement of blood glucose level.

CO5: Explain the properties, characterization, and applications of various biomaterials. **L:9**

Biomaterials: introduction - types: alloys, polymers. Composites - properties: biocompatibility, elasticity, immune compatibility, resorbability, cytotoxicity, hemocompatibility and biodegradability. Physicochemical characterization: XRD and SEM. Applications: tissue engineering, heart valves, dental and orthopaedic implants.

TEXT BOOKS

1. Y.Nelson, L.David, Lehninger, "Principles of Biochemistry", International Edition. New York, 7th Edition, 2017.
2. Nagata, Kazuhiro, Real-Time Analysis of Biological Interactions, Springer, Japan, 3rd Edition, 2015.
3. I. Bertini, H.B Gray, Bioinorganic Chemistry, University Science Book, California, 4thEdition, 2014.

REFERENCE BOOKS:

1. P.N.Bartlett, Bioelectrochemistry: Fundamentals, Experimental Techniques and Applications, 2nd Edition, John Wiley & Sons, New Delhi, 2014.
2. Ratner and Hoffmann, Biomaterial Science: An Introduction to Materials in Medicine, 2nd Edition, Elsevier Academic Press, London, 2015.
3. Lesile Cromwell, "Bio-medical instrumentation and measurement", Prentice Hall of India, New Delhi, 2nd Edition, Reprint, 2017.

L: 45; TOTAL: 45 PERIODS

Course Code		L	T	P	E	C
23SH15E	POLYMER SCIENCE AND TECHNOLOGY	3	0	0	0	3

COURSE OUTCOMES:

Upon completion of this course, the students will be able to

Theory Components:

CO1: acquire knowledge on structure - property relationship of polymers

CO2: identify the suitable polymerization techniques for the large scale synthesis of polymers

CO3: explain the basic principles of various polymer processing techniques and their applications

CO4: interpret the chemical, thermal, electrical, and mechanical properties of the polymers

CO5: familiar with plastics waste disposal, value addition, associated environmental issues and legislation

CO1: Acquire knowledge on structure - property relationship of polymers **L: 9**

Basic concepts of polymerization - polydispersity - conformation and configuration of macromolecules - stereo isomerism and tacticity in polymers - geometrical isomerism. Structure - property relationship -molecular force and chemical bonding in polymers - effect of polymerization on PDI. General rules for polymer solubility - crystallinity and orientation in polymers. Polymer chain flexibility: concept - factors deciding polymer flexibility - amorphous and crystalline polymers - crystallinity in polymers - factors affecting crystallinity - properties affected by crystallinity of polymers. Glass transition temperature and crystalline melting points. Factors affecting glass transition temperature.

CO2: Identify the suitable polymerization techniques for the large scale synthesis of polymers **L:9**

Basic aspects of polymer synthesis - bulk, solution and suspension polymerization (styrene and MMA) - emulsion polymerization (vinyl acetate, styrene) - preparation of phenolic and epoxy resins. Modern techniques in polymerization: metathesis polymerization - controlled polymerization methods, viz., nitroxide mediated polymerization (NMD), atom transfer radical polymerization (ATRP), group transfer polymerization (GTP), and reversible addition fragmentation termination (RAFT).

CO3: Explain the basic principles of various polymer processing techniques and their applications **L:9**

Plastics technology: raw materials - additives for compounding (fillers, plasticizers and softeners, lubricants, promoters, anti-aging additives, flame retarders, colorants, blowing agents, UV stabilizers,) - requirements and functions of additives. Pre-compounding operations: mixing, drum blenders, ribbon blenders, mixing rolls, internal mixers, mixing extruders, blenders for making organosol and plastisol, granulators, pelletizers.

Advanced fabrication techniques: RTM, RIM, filament winding, BMC/SMC. Post-forming and finishing, machining, welding and design of polymers products. Selections of polymers, additives, mold design. Analysis of defects in moulded products. Processing of reinforced thermoplastics and thermosets: manual processing methods and semi-automatic processing methods. Rubber processing: internal mixer and open mill.

CO4: Interpret the chemical, thermal, electrical, and mechanical properties of polymers **L:9**

Physical testing: density, mechanical behaviour, MFI, and water/solvent adsorption. Chemical testing: ignition - pyrolysis - solvent extraction - elemental analysis. Thermal analysis: vicat softening point - dynamic mechanical thermal analysis. Morphological analysis: atomic force microscopy and chemical force microscopy. Spectroscopic analysis: IR peaks assigned for rubber. Study of hydrogenation, halogention, evidence for cyclization and formation of ionomers. Analysis of carbon filled rubber - Case studies.

CO5: Familiar with plastics waste disposal, value addition, associated environmental issues and legislation **L-9**

Polymer waste: sources, collection, segregation, and identification by simple techniques. Life cycle assessment, risk factor analysis. Plastics waste management techniques: chemical recycling, thermal conversion technologies, microbial, microwave, and ultrasonic. Use of plastics waste for value addition. Plastics waste management rule - environmental issues.

TEXT BOOKS

1. Premamoy Ghosh, Polymer Science and Technology: Plastics, Rubber, Blends and Composites, 3rd Edition, McGraw Hill Education, 2017.
2. Richard A Petherick, Polymer Science and Technology for Engineers and Scientists, Whittles Publishing, 2010.
3. Michael L. Berins, SPI Plastics Engineering Handbook of the Society of the Plastics Industry, Inc. 1st Edition, Springer New York, 2012
4. Vishu Shah, Handbook of Plastics Testing Technology,2nd Revised edition, Wiley–Blackwell, 1998**.**

REFERENCES

1. Gowarikar V R, Polymer science, 5th Edition, New Age International Private Limited, 2023
2. Fred W. Billmeyer, Textbook of Polymer Science, 3rd Edition, John Wiley & Sons, 2007
3. Nayak S.K, Text Book on Fundamentals of Plastics Testing, Springer (I) Private Limited, 2020
4. J S Anand, Recycling & Plastics Waste Management, Central Institute of Plastics Engineering and Technology, 1997.
5. Korschwitz J, Polymer Characterization and Analysis, John Wiley and Sons, 1990.

L: 45; TOTAL: 45 PERIODS

Course Code		L	T	P	E	C
23SH16E	**SENSORS FOR ENGINEERING APPLICATIONS**	**3**	**0**	**0**	**0**	**3**

COURSE OUTCOMES:

Upon completion of this course, the students will be able to

Theory Components:

CO1: Gain knowledge on basic concepts of sensors and Transducer.

CO2: know about the thermal and motion sensors for various applications.

CO3: enumerate the principles and applications of optical and magnetic sensors and transducers used in various field.

CO4: explain the construction, working principle and applications of electrochemical and electric sensors.

CO5: Design the sensors for environmental monitoring

CO1: Gain knowledge on basic concepts of sensors and Transducer. L: 9

Introduction – Historical development of sensors – Human body as a sensor system – sensors and transducers. Principle and classification of sensor. Sensor characteristics – sensor properties – various transducers – piezoelectric effect – pyroelectric effect – seebeck effect and peltier effect. Advantages and limitations of Sensors.

CO2: Know about the thermal and motion sensors for various applications. L:9

Thermal sensors: introduction – types - primary sensor: gas thermometer and He low temperature thermometer. Secondary sensor: Resistance thermometer and NQR thermometer. Temperature sensing technologies: IC sensor, resistive temperature detectors, thermocouples and thermistor.

Motion sensors: Introduction and principle. Types: Infra red and microwave. Specialized motion sensor: proximity and ranging sensor. Motion Sensors in everyday life: The role of

motion sensors in home security.

CO3: Enumerate the principles and applications of optical and magnetic sensors and transducers used in various field **L:9**

Magnetic sensors: Introduction – principle and applications: magnetic field sensors and magneto-resistive Sensors, hall effect sensors.

Optical sensors: light intensity – wavelength and color – light dependent resistors, photodiode, photo transistor, CCD, CMOS sensors. Pulse oximeter, portable pulse oximeter, wearable pulse oximeter; wearable capnometer for monitoring of expired.

CO4: Explain the construction, working principle and applications of electrochemical and electric sensors **L-9**

Electrochemical sensors: Introduction - fundamental concepts – chemiresistors. Conductometric sensor: amperometric sensor - potentiometric sensors - impendence sensors.

Electric sensors: Introduction- conventional volt and ammeters, high current sensors, (current transformers), high voltage sensors, High power sensors. Real time applications: Glucose Monitoring

Devices, GlucoWatch G2 Biographer, GlucoTrackTM; Pulse oximeter, Portable Pulse Oximeter, wearable pulse oximeter.

CO5: Design the sensors for environmental monitoring **L-9**

Environmental Sensor: Introduction - environmental quantities**:** time, moisture acidity/alkalinity, wind-chill, radioactive count rate. Surveying and security. Sensors for environmental monitoring. Smoke and fire detector. Pressure sensor in emission testing, pollution devices, and wind management systems.

TEXT BOOKS

1. Jacob Fraden, Handbook of Modern Sensors: Physics, Design and Applications, 5th edition, Springer Nature, New Delhi, 2016
2. D. Patranabis, Sensors and Transducers, 2nd Edition, PHI Learning Private Limited, New Delhi, 2013.
3. John Veteline, Aravind Raghu, Introduction to sensors, CRC press, New Delhi, 2011.
4. S Nihtianov, A. Luque Smart Sensors and MEMS, 2nd Edition, Woodhead Publishing Limited, New Delhi, 2018.
5. Edward Sazonov and Michael R. Neuman, Wearable Sensors - Fundamentals, Implementation and Applications, Elsevier publishing company, Amsterdam, Netherland, 2014.

REFERENCE BOOKS

1. Shantanu Bhattacharya, A K Agarwal, NripenChanda, Ashok Pandey and Ashis Kumar Sen Environmental, Chemical and Medical Sensors, Springer Verlag, Singapore, 2018.
2. Krzysztof Iniewski, Optical, Acoustic, Magnetic, and Mechanical Sensor Technologies, 1st Edition, CRC Press, New Delhi, 2017.

L: 45; TOTAL: 45 PERIODS