PSG INSTITUTE OF TECHNOLOGY AND APPLIED RESEARCH

COIMBATORE – 641 062

(Autonomous Institution Affiliated to Anna University)

2025 Regulations

Scheme of Assessment and Syllabi

For

B.E. COMPUTER SCIENCE AND ENGINEERING

B.E. COMPUTER SCIENCE AND ENGINEERING

VISION

To foster innovation, research, entrepreneurial thinking and industry collaboration, by empowering students to develop cutting-edge technologies for societal progress and global advancements.

MISSION

- Create a dynamic learning environment that cultivates critical thinking, technical skills and an entrepreneurial mindset in Computer Science and Engineering.
- Stimulate interdisciplinary research and collaboration, lifelong learning to ensure the continuous advancement of knowledge in emerging technologies.
- Empower students to drive innovation and creativity, preparing them to address real-world challenges and ethically contribute to technological progress.

PROGRAM EDUCATIONAL OBJECTIVES (PEOs)

PEO1: Excel in professional careers and higher studies by applying strong foundations in computer science, problem-solving and emerging technologies.

PEO2: Demonstrate innovation, entrepreneurship and research attitude to design sustainable solutions for real-world challenges through industry and interdisciplinary collaboration.

PEO3: Exhibit ethical responsibility, teamwork, leadership and lifelong learning to contribute positively to society and global technological progress.

PROGRAM OUTCOMES (POs)

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

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

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

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

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

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

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

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

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

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

PO11: Life-Long Learning: Recognize the need for, and have the preparation and ability for i) independent and life-long learning ii) adaptability to new and emerging technologies and iii) critical thinking in the broadest context of technological change. (WK8)

PROGRAM SPECIFIC OUTCOMES (PSOs)

PSO1: Design and develop hardware, software systems by applying core principles of computing, advanced algorithms and state-of-the-art computational frameworks.

PSO2: Apply advanced computing concepts and emerging areas in Computer Science and Engineering to create innovative and socially relevant solutions for industry and society.

B.E. COMPUTER SCIENCE AND ENGINEERING

(Minimum No. of credits to be earned: 168)

S. No.	Course Code	Course Title	Hours / Week			C	Maximum Marks			CAT
			L	T	P		CA	ESE	Total	
SEMESTER I										
THEORY										
1	25MA101	Calculus and its Applications	3	1	0	4	40	60	100	BS
2	25CS101	C Programming	3	0	0	3	40	60	100	ES
3	25EE101	Basics of Electrical and Electronic Systems	3	0	0	3	40	60	100	ES
4	25HS101	English Language Proficiency	3	1	0	4	40	60	100	HS
5	25HS102	தமிழர் மரபு / Heritage of Tamils	1	0	0	1	40	60	100	HS
PRACTICALS										
6	25CS111	C Programming Laboratory	0	0	4	2	60	40	100	ES
7	25EE112	Engineering Skills Laboratory	0	0	4	2	60	40	100	ES
8	25GE111	Design Thinking for Innovation	0	0	2	1	100	0	100	ES
MANDATORY COURSES										
9	25GEM01	Induction Programme**	-	-	-	Grade	-	-	-	MC
10	25HSO01/ 25HSO02	Physical Education/Yoga and Health	0	0	2	Grade	-	-	-	MC
Total 27 periods			**13**	**2**	**12**	**20**	**420**	**380**	**800**	

S. No.	Course Code	Course Title	Hours / Week			C	Maximum Marks			CAT
			L	T	P		CA	ESE	Total	
SEMESTER II										
THEORY										
1	25MA202	Transforms and Applications	3	1	0	4	40	60	100	BS
2	25PH204	Sensors for Engineering Applications	3	0	0	3	40	60	100	BS
3	25CY202	Applied Chemistry	3	0	0	3	40	60	100	BS
4	25AD201	Python Programming	3	0	0	3	40	60	100	ES
5	25CS201	Digital Design	3	1	0	4	40	60	100	ES
6	25HS201	தமிழரும் தொழில்நுட்பமும் / Tamils and Technology	1	0	0	1	40	60	100	HS
PRACTICALS										
7	25AD211	Python Programming Laboratory	0	0	4	2	60	40	100	ES
8	25BS212	Physics and Chemistry Laboratory	0	0	4	2	60	40	100	BS
9	25GE211	Engineering Graphics	0	0	4	2	60	40	100	ES
10	25HS21_	Language Elective	0	0	4	2	60	40	100	HS
11	25EEC01	Workplace Communication Skills	0	0	2	Grade	100	0	100	EEC
MANDATORY COURSES										
12	25GEM02	Activity Point Programme I*	-	-	-	Grade	-	-	-	MC
13	25HSO01/ 25HSO02	Physical Education/Yoga and Health	0	0	2	Grade	-	-	-	MC
Total 38 periods			**16**	**2**	**20**	**26**	**580**	**520**	**1100**	

**** As per AICTE Norms**

*** As per AICTE Norms Total: 60 hours, Grade: Non - Credit Course**

CA - Continuous Assessment; ESE - End-Semester Examination; CAT - Category; BS - Basic Science; HS - Humanities and Social Sciences; ES - Engineering Sciences; PC - Professional Core; PE - Professional Elective; OE - Open Elective; EEC - Employability Enhancement Course; MC - Mandatory Course
L-Lecture; T-Tutorial; P-Practical; C-Credits

S. No.	Course Code	Course Title	Hours /Week			C	Maximum Marks			CAT
			L	T	P		CA	ESE	Total	
SEMESTER III										
THEORY										
1	25MA302	Linear Algebra	3	1	0	4	40	60	100	BS
2	25MA305	Discrete Structures	3	1	0	4	40	60	100	BS
3	25CS301	Data Structures	3	0	0	3	40	60	100	PC
4	25CS302	Computer Organization and Architecture	3	1	0	4	40	60	100	ES
5	25CS303	Object Oriented Programming	2	0	2	3	50	50	100	ES
6	25HS301	Project and Finance Management	3	0	0	3	40	60	100	HS
PRACTICALS										
7	25CS311	Data Structures Laboratory	0	0	4	2	60	40	100	PC
8	25EEC02	Foundations for Problem Solving	0	0	2	1	100	0	100	EEC
MANDATORY COURSES										
9	25MC0_ _	Mandatory Course I	2	0	0	Grade	100	0	100	MC
10	25GEM03	Activity Point Programme II*	-	-	-	Grade	-	-	-	MC
Total 30 periods			**19**	**3**	**8**	**24**	**510**	**390**	**900**	

S. No.	Course Code	Course Title	Hours / Week			C	Maximum Marks			CAT
			L	T	P		CA	ESE	Total	
SEMESTER IV										
THEORY										
1	25MA402	Statistical Methods and Stochastic Processes	3	1	0	4	40	60	100	BS
2	25CS401	Database Management Systems	3	0	0	3	40	60	100	PC
3	25CS402	Design and Analysis of Algorithms	3	1	0	4	40	60	100	PC
4	25CS403	Theory of Computation	3	1	0	4	40	60	100	PC
5	25CS404	Software Engineering	3	1	0	4	40	60	100	PC
PRACTICALS										
6	25CS411	Database Management Systems Laboratory	0	0	4	2	60	40	100	PC
7	25CS412	Application Development Laboratory	0	0	4	2	60	40	100	PC
8	25CSE01	Mini Project I	0	0	2	1	100	0	100	EEC
9	25EEC03	Problem Solving	0	0	2	1	100	0	100	EEC
MANDATORY COURSES										
10	25MC0_ _	Mandatory Course II	2	0	0	Grade	100	0	100	MC
11	25GEM04	Activity Point Programme III*	-	-	-	Grade	-	-	-	MC
Total 33 periods			**17**	**4**	**12**	**25**	**620**	**380**	**1000**	

*** As per AICTE Norms Total: 60 hours, Grade: Non - Credit Course**

CA - Continuous Assessment; ESE - End-Semester Examination; CAT - Category; BS - Basic Science; HS - Humanities and Social Sciences; ES - Engineering Sciences; PC - Professional Core; PE - Professional Elective; OE - Open Elective; EEC - Employability Enhancement Course; MC - Mandatory Course
L-Lecture; T-Tutorial; P-Practical; C-Credits

S. No.	Course Code	Course Title	Hours / Week			C	Maximum Marks			CAT
			L	T	P		CA	ESE	Total	
SEMESTER V										
THEORY										
1	25CS501	Operating Systems	3	0	0	3	40	60	100	PC
2	25CS502	Compiler Design	3	0	0	3	40	60	100	PC
3	25CS503	Computer Networks	3	0	0	3	40	60	100	PC
4	25CS504	Artificial Intelligence	3	1	0	4	40	60	100	PC
5	25CSP_ _#	Professional Elective I	3#	0	0#	3	40#	60#	100	PE
PRACTICALS										
6	25CS511	Operating Systems Laboratory	0	0	2	1	60	40	100	PC
7	25CS512	Computer Networks Laboratory	0	0	4	2	60	40	100	PC
8	25CSE02/ 25CSE03	Internship I / Community Project	0	0	0	1	100	0	100	EEC
9	25EEC04	Aptitude Skills	0	0	2	1	100	0	100	EEC
MANDATORY COURSES										
10	25GEM05	Activity Point Programme IV*	-	-	-	Grade	-	-	-	MC
Total 24 periods#			**15#**	**1**	**8#**	**21**	**520#**	**380#**	**900**	

S. No.	Course Code	Course Title	Hours / Week			C	Maximum Marks			CAT
			L	T	P		CA	ESE	Total	
SEMESTER VI										
THEORY										
1	25MA602	Graph Theory	3	1	0	4	40	60	100	PC
2	25CS601	Machine Learning	3	0	0	3	40	60	100	PC
3	25CS602	Embedded Systems	2	2	0	4	40	60	100	ES
4	25CSP_ _#	Professional Elective II	3#	0	0#	3	40#	60#	100	PE
5	25_ _O_ _	Open Elective I	3	0	0	3	40	60	100	OE
PRACTICALS										
6	25CS611	Machine Learning Laboratory	0	0	4	2	60	40	100	PC
7	25CSE04	Mini Project II	0	0	2	1	100	0	100	EEC
8	25EEC06	Enhancing Arithmetic Problem Solving	0	0	2	1	100	0	100	EEC
MANDATORY COURSES										
9	25GEM06	Activity Point Programme V*	-	-	-	Grade	-	-	-	MC
Total 25 periods#			**14#**	**3**	**8#**	**21**	**460#**	**340#**	**800**	

 # will vary for laboratory integrated theory courses

*** As per AICTE Norms Total: 60 hours, Grade: Non - Credit Course**

CA - Continuous Assessment; ESE - End-Semester Examination; CAT - Category; BS - Basic Science; HS - Humanities and Social Sciences; ES - Engineering Sciences; PC - Professional Core; PE - Professional Elective; OE - Open Elective; EEC - Employability Enhancement Course; MC - Mandatory Course
L-Lecture; T-Tutorial; P-Practical; C-Credits

S. No.	Course Code	Course Title	Hours / Week			C	Maximum Marks			CAT
			L	T	P		CA	ESE	Total	
SEMESTER VII										
THEORY										
1	25CS701	Parallel and Distributed Systems	3	0	0	3	40	60	100	PC
2	25CS702	Cryptography and Network Security	3	1	0	4	40	60	100	PC
3	25CSP_ _#	Professional Elective III	3#	0	0#	3	40#	60#	100	PE
4	25CSP_ _#	Professional Elective IV	3#	0	0#	3	40#	60#	100	PE
5	25_ _O_ _	Open Elective II	3	0	0	3	40	60	100	OE
PRACTICALS										
6	25CS711	Parallel and Distributed Systems Laboratory	0	0	4	2	60	40	100	PC
7	25CSE05	Project Work I	0	0	4	2	100	0	100	EEC
8	25CSE06	Internship II	0	0	0	1	100	0	100	EEC
Total 24 periods#			**15#**	**1**	**8#**	**21**	**460#**	**340#**	**800**	

S. No.	Course Code	Course Title	Hours / Week			C	Maximum Marks			CAT
			L	T	P		CA	ESE	Total	
SEMESTER VIII										
THEORY										
1	25CSP_ _#	Professional Elective V	3#	0	0#	3	40#	60#	100	PE
2	25CSP_ _#	Professional Elective VI	3#	0	0#	3	40#	60#	100	PE
PRACTICALS										
3	25CSE07	Project Work II	0	0	8	4	60	40	100	EEC
	Total 14 periods#		**6#**	**0**	**8#**	**10**	**140#**	**160#**	**300**	

Summary of Credit Distribution

B.E. COMPUTER SCIENCE AND ENGINEERING										
S. No.	Course Category	Credits Per Semester								Total Credits
		1	2	3	4	5	6	7	8	
1	HS	5	3	3	0	0	0	0	0	**11**
2	BS	4	12	8	4	0	0	0	0	**28**
3	ES	11	11	7	0	0	4	0	0	**33**
4	PC	0	0	5	19	16	9	9	0	**58**
5	PE	0	0	0	0	3	3	6	6	**18**
6	OE	0	0	0	0	0	3	3	0	**6**
7	EEC	0	0	1	2	2	2	3	4	**14**
8	MC	-	-	-	-	-	-	-	-	**-**
TOTAL		**20**	**26**	**24**	**25**	**21**	**21**	**21**	**10**	**168**

 # will vary for laboratory integrated theory courses

CA - Continuous Assessment; ESE - End-Semester Examination; CAT - Category; BS - Basic Science; HS - Humanities and Social Sciences; ES - Engineering Sciences; PC - Professional Core; PE - Professional Elective; OE - Open Elective; EEC - Employability Enhancement Course; MC - Mandatory Course

L-Lecture; T-Tutorial; P-Practical; C-Credits

LIST OF PROFESSIONAL ELECTIVE COURSES: VERTICALS

S. No.	VERTICAL I Full Stack Development	VERTICAL II Cyber Physical Systems	VERTICAL III Artificial Intelligence and Data Science	VERTICAL IV Frontier Technologies	VERTICAL V Cloud Computing and Data Center Technologies
1	25CSP01 Micro service Architecture	25CSP09 Software Defined Network	25CSP18 Data science Essentials	25ADP11 Immersive Technologies	25CSP27 Cloud Computing Technologies
2	25CSP02 UI/UX Design	25CSP10 Security and Privacy in Cloud	25CSP16 Data Exploration and Visualization	25CSP22 Computer Vision and Image Processing	25ADP18 Cloud Services Management
3	25CSP03 DevOps	25CSP21 Blockchain Technologies	25CSP20 Business Analytics	25CSP17 Soft Computing	25CSP11 Virtualization
4	25CSP04 Software Testing and Automation	25CSP12 Ethical Hacking	25ADP09 Natural Language Processing	25ADP19 Quantum Machine Learning	25CSP31 Managing Big Data
5	25CSP05 Full Stack Web Development	25CSP19 Reinforcement Learning Techniques	25CSP32 Generative Language Models	25CSP24 Neural Networks and Deep Learning	25CSP28 Data Warehousing
6	25CSP06 MERN Stack	25CSP14 Cyber Forensics	25CSP23 Generative AI	25ADP15 Responsible AI	25CSP29 Storage Technologies
7	25CSP07 Agile Methodologies	25CSP15 AI in Cyber Security	25CSP13 Edge AI	25ADP04 Image and Video Analytics	25CSP30 Stream Processing
8	25CSP08 Vibe Coding	25ADP03 Smart Systems	25ADP05 Text and Speech Analysis	25CSP26 Social Network Analysis	25CSP25 High Performance Computing

LIST OF COURSES FOR MINOR DEGREE PROGRAMME IN FULL STACK APPLICATION DEVELOPMENT

S. No.	Course Code	Course Title
1	25CSM01	Micro Services and Cloud Application
2	25CSM02	Java Programming
3	25CSM03	Web frameworks
4	25CSM04	Application Development Technologies
5	25CSM05	Software System Testing
6	25CSM06	User Experience and Interface Development
7	25CSM07	DevOps Systems
8	25CSM08	Intelligent Code Generation

LIST OF LANGUAGE ELECTIVE COURSES

S. No.	Course Code	Course Title
1	25HS211	Communication Skills for Engineers
2	25HS212	Basic German
3	25HS213	Basic Japanese

LIST OF MANDATORY COURSES

S. No.	Course Code	Course Title
1	25MC001	Environmental Sciences
2	25MC002	Indian Constitution
3	25MC003	Industrial Safety
4	25MC004	Disaster Risk Reduction and Management
5	25MC005	Indian Knowledge System for Sustainable Human Development
6	25MC006	Strategic Data Management and Governance
7	25MC007	Professional Ethics
8	25HSO01	Physical Education
9	25HSO02	Yoga and Health

SEMESTER I

25MA101 CALCULUS AND ITS APPLICATIONS

(Common to CIVIL, CSE, EEE, ECE, ICE, MECH, AI&DS and EE-VLSI)

3 1 0 4

DIFFERENTIAL CALCULUS: Functions of two variables, Limits and continuity, Partial derivatives, Chain rule, Extreme values and saddle points, Lagrange multipliers, Taylor's formula for two variables. **(9+3)**

INTEGRAL CALCULUS: Double and iterated integrals over rectangles, Double integrals over general regions, Fubini's theorem, Area and volume by double integration, Reversing the order of integration, Double integrals in polar form. **(9+3)**

FIRST ORDER ORDINARY DIFFERENTIAL EQUATIONS: Basic concepts, Separable differential equations, Exact differential equations, Integrating factors, Linear differential equations, Modeling - Mixing problems, Newton's law of cooling, Decay and growth problems. **(9+3)**

SECOND ORDER LINEAR DIFFERENTIAL EQUATIONS: Homogeneous linear equations of second order, Homogeneous linear ODEs with constant coefficients, Euler– Cauchy equations, Solution by variation of parameters, Free oscillations mass spring systems, Electric circuits. **(9+3)**

VECTOR CALCULUS: Gradient and directional derivative of a scalar field, Divergence and curl of a vector field. Integration in vector field – Line integrals, Path independence of line integrals, Green's theorem in the plane, Divergence theorem of Gauss and Stokes' theorem. **(9+3)**

Total L: 45 + T: 15 = 60 periods

TEXT BOOKS

1. J. Hass, C. Heil, and D. W. Maurice '*Thomas' Calculus*'. Pearson Education, New Delhi, 2018.
2. Erwin Kreyszig '*Advanced Engineering Mathematics*'. Wiley India, New Delhi, 2018.

REFERENCES

1. H. Anton, I. Bivens, and S. Davis '*Calculus*'. John Wiley and Sons, USA, 2016.
2. C. R. Wylie and L. C. Barrett '*Advanced Engineering Mathematics*'. Tata McGraw-Hill, New Delhi, 2019.
3. D. G. Michael '*Foundations of Applied Mathematics*'. Dover Publications, New York, 2013.
4. Gilbert Strang '*Calculus'*. Wellesley Cambridge Press, USA, 2017.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the concepts related to Calculus, Differential Equations and Vector Calculus.	**K2**
CO2	Apply the techniques of Calculus, Differential Equations and Vector Calculus to solve engineering problems.	**K3**
CO3	Analyze the solutions of engineering problems employing Calculus, Differential Equations and Vector Calculus.	**K4**
CO4	Use modern tools to solve engineering problems with the help of Calculus, Differential Equations and Vector Calculus.	**-**

COs-POs & PSOs MAPPING

CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PSO1**	**PSO2**
CO1													
CO2	3												
CO3		1											
CO4					1								
	3	**1**			**1**								

1-low, 2-medium, 3-high

25CS101 C PROGRAMMING
(Common to CSE and AI&DS)

3 0 0 3

INTRODUCTION: Evolution of programming languages, Algorithm, Pseudocode and Flowchart, Programming paradigms: Structured programming – Object oriented programming – Functional programming, Static and Dynamic type checking, Strongly and Weakly typed language, Program execution, Embedded C. **(9)**

C PRELIMINARIES AND CONTROL STATEMENTS: Character set – Identifiers and keywords, Data types, Enumeration, Declarations, Expressions, Statements, Preprocessor directives and symbolic constants – Operators and expressions – Input and output Statements, Control constructs: Unconditional statements – Conditional Statements – Branching and Looping Statements. **(9)**

FUNCTIONS AND POINTERS: Library functions and user-defined functions, Call by value, Recursion: Function call – Tail recursion – Tree recursion – Linear recursion, Macros, Storage types, Pointers: Operations on pointers – Pointer arithmetic, Call by reference, Function pointer. **(9)**

ARRAYS: One dimensional and multi-dimensional array, Defining and processing array, Array as function argument, Strings – Pointer to an Array, Array of pointers, Command line arguments, C99-Standards. **(9)**

STRUCTURES, UNION AND DATA FILES: Defining and processing a structure, Nested structures, passing structure to functions – Array of structure, Pointer to structure – Union – Dynamic memory allocation – Files: Streams, Opening and closing a file, Reading and writing a file. **(9)**

Total L: 45 periods

TEXTBOOKS:

1. H. M. Deitel and P. J. Deitel, *'C: How to Program'.* 9th edition, Pearson, New Delhi, 2022.
2. Herbert Schildt, *'C: The Complete Reference'.* 4th edition, McGraw Hill, Noida, 2017.

REFERENCES:

1. Pradip Dey and Manas Ghosh, *'Programming in C'*. Oxford University, New Delhi, 2018.
2. Mike McGrath, *'C Programming in Easy Steps Limited'*. 5th edition, United Kingdom, 2018.
3. Gottfried B, *'Programming with C'*. McGraw Hill, 4th edition, Noida, 2018.
4. B. W. Kernighan and D. M. Ritchie, *'The C Programming Language'*. 2nd edition, Pearson, New Delhi, 2015.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the application of C programming in engineering problems.	**K2**
CO2	Apply programming knowledge to solve engineering problems through simple projects.	**K3**
CO3	Analyze data generated through C programs and effectively communicate the results.	**K4**
CO4	Evaluate different problem-solving approaches for effectiveness in given engineering tasks.	**K5**

COs-POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1												3	3
CO2	3											3	3
CO3		3										3	3
CO4			2									2	2
	3	**3**	**2**									**3**	**3**

1-low, 2-medium, 3-high

25EE101 BASICS OF ELECTRICAL AND ELECTRONIC SYSTEMS
(Common to CSE and AI&DS)

3 0 0 3

DC CIRCUIT: current-voltage –power-energy, electrical circuit elements: resistors-inductor- capacitor, source of electrical energy. Ohm's law-Kirchhoff's laws, series and parallel circuits, Maxwell's loop current method, Network theorems: superposition theorem-Thevenin's theorem-Norton's theorem-maximum power transfer theorem. **(9)**

AC CIRCUITS: Single phase AC circuits: Average and RMS values of sinusoidal wave form- RLC Circuit-Phasor representation-active, reactive apparent power –power factor, analysis of RLC Circuit, three phase circuit: star and delta connection-phase and line quantities-balance and unbalance systems. **(9)**

ELECTROMAGNETISM AND MAGNETIC CIRCUITS: Electromagnetic induction; induced currents, Faraday's law, induction and energy, motional emf and Lenz's law. Magnetic field-magnetic circuit-inductance and mutual inductance-magnetic materials –ideal transformers and real transformers. **(8)**

SEMICONDUCTOR DEVICES: Basic diode concepts-diode circuit: half wave rectifier-full wave rectifier-bridge rectifier-special purpose diodes-Zener diode –transistor fundamentals – transistor biasing-bipolar junction transistors-basis amplifier concept-loading effect-power supplies and efficiency. **(10)**

OPERATIONAL AMPLIFIERS: Op-Amp Basics, Ideal characteristics of Op-Amp, practical Op-Amp circuits, differential and common mode operation, Inverting and non- inverting amplifiers, Op-Amp as Adder, Subtractor, integrator and differentiator. **(9)**

Total L: 45 periods

TEXT BOOKS

1. Edward Hughes, John Hiley and Keith Brown '*Electrical and Electronic Technology*'. Pearson Education, 2020.
2. K. Murugesh Kumar '*Basic Electrical Science and Technology*'. Vikas Publishing House, 2016.
3. B. L. Theraja '*Basic Electronic Solid State*'. S. Chand and Company Ltd., New Delhi, 2010.

REFERENCES

1. D. P. Leach '*Digital Principles and Applications*'. Tata McGraw Hill, 2021.
2. A. R. Hambley '*Electrical Engineering Principles and Applications*'. Pearson education, 2018.
3. R. L. Boylestad and L. Nashelsky '*Electronic Devices and Circuit Theory*'. Pearson Education, Noida, 2013.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the basic principles of DC and AC circuits, magnetic circuits, and semiconductor devices.	**K2**
CO2	Apply electrical laws, network theorems, and Electromagnetic principles to analyze electrical circuits, magnetic circuits, and transformers.	**K3**
CO3	Analyze the performance characteristics of electrical and electronic systems using phasors, load analysis, and Op-Amp configurations.	**K4**

COs-POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1													
CO2	3		2										
CO3		3	3		2								
	3	**3**	**3**		**2**								

1-low, 2-medium, 3-high

25HS101 ENGLISH LANGUAGE PROFICIENCY
(Common to CIVIL, CSE, EEE, ECE, ICE, MECH, AI&DS and EE-VLSI)

3 1 0 4

VOCABULARY: Etymology-Prefixes and suffixes–Synonyms–Antonyms–Guessing meanings from context– Word formation- Single-word substitutes- Different forms of a word– Phrasal verbs–Collocations. **(9+3)**

LISTENING AND SPEAKING: Understanding listening – Listening techniques - Introducing oneself and others – Seeking and sharing information– Description-Conversation skills– Extempore speaking– Speech practice in varied formal contexts. **(9+3)**

GRAMMAR: Wh-questions – Yes/no questions– Parts of speech – Articles– Prepositions– Gerunds–Conjunctions- Degrees of comparison– Tenses– Modal verbs – Adverbs - Direct and indirect questions. **(9+3)**

READING: Reading strategies: Skimming and scanning, predicting– Reading comprehension: techniques – Practice reading. **(9+3)**

WRITING: Discourse markers – Dialogue writing - Completing sentences – Jumbled sentences – Paragraph writing –Writing compare & contrast paragraphs – Letter writing. **(9+3)**

Total L: 45 + T: 15 = 60 periods

TEXTBOOKS:

1. K. N. Shoba and Lourdes Joavani Rayen '*Communicative English*'. Cambridge University press, Cambridge, 2021.
2. Raymond Murphy '*Intermediate English Grammar*'. Cambridge University Press, New Delhi, 2020.
3. Dr M. Sambaiah '*Technical English an integrated text book*'. Wiley India Pvt. Ltd., 2025.

REFERENCES:

1. Raymond Murphy '*English Grammar in Use*'. Cambridge University Press, New Delhi 2020.
2. N. P. Sudharshana and C. Savitha '*English for Engineers*'. Cambridge University Press, New York, 2018.
3. Helen Naylor with Raymond Murphy '*Essential English Grammar*'. Cambridge University Press, New Delhi, 2019.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Demonstrate the ability to recognize and use a wide range of vocabulary and key grammatical structures accurately, while developing inferential reading skills to comprehend, interpret, and analyze written texts across diverse contexts.	**K2**
CO2	Organize their ideas logically in essay writing, develop paragraphs with clear topic sentences and adapt their letter- writing skills to various real-world scenarios.	**K3**
CO3	Develop and demonstrate clear and confident speaking skills in formal and informal contexts.	

COs-POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1									3		3		
CO2									3		3		
CO3									1		1		
									3		**3**		

1-low, 2-medium, 3-high

25HS102 *தமிழர் மரபு*
(Common to all B.E./B.Tech./B.Des. Degree Programmes)

1 0 0 1

மொழி மற்றும் இலக்கியம்: *இந்திய மொழிக் குடும்பங்கள் – திராவிட மொழிகள் – தமிழ் ஒரு செம்மொழி – தமிழ் செவ்விலக்கியங்கள் – சங்க இலக்கியத்தின் சமயச்சார் பற்ற தன்மை – சங்க இலக்கியத்தில் பகிர்தல் அறம் – திருக்குறளில் மேலாண்மைக் கருத்துக்கள் – தமிழ்க்காப்பியங்கள், தமிழகத்தில் சமண பௌத்த சமயங்களின் தாக்கம் – பக்தி இலக்கியம், ஆழ்வார்கள் மற்றும் நாயன்மார்கள் – சிற்றிலக்கியங்கள் – தமிழில் நவீன இலக்கியத்தின் வளர்ச்சி – தமிழ் இலக்கிய வளர்ச்சியில் பாரதியார் மற்றும் பாரதிதாசன் ஆகியோரின் பங்களிப்பு.* **(3)**

மரபு – பாறை ஓவியங்கள் முதல் நவீன ஓவியங்கள் வரை – சிற்பக்கலை: *நடுகல் முதல் நவீன சிற்பங்கள் வரை – ஐம்பொன் சிலைகள் – பழங்குடியினர் மற்றும் அவர்கள் தயாரிக்கும் கைவினைப் பொருட்கள், பொம்மைகள் – தேர் செய்யும் கலை – சுடுமண் சிற்பங்கள் – நாட்டுப்புறத் தெய்வங்கள் – குமரி முனையில் திருவள்ளுவர் சிலை – இசைக்கருவிகள் – மிருதங்கம், பறை, வீணை, யாழ், நாதஸ்வரம் – தமிழர்களின் சமூக பொருளாதார வாழ்வியல் கோவில்களின் பங்கு.* **(3)**

நாட்டுப்புறக்கலைகள் மற்றும் வீரவிளையாட்டுகள்: *தெருக்கூத்து, கரகாட்டம், வில்லுப் பாட்டு, கணியான் கூத்து, ஒயிலாட்டம், தோல்பாவைக் கூத்து, சிலம்பாட்டம், வளரி, புலியாட்டம், தமிழர்களின் விளையாட்டுகள்.* **(3)**

தமிழர்களின் திணைக்கோட்பாடுகள்: *தமிழகத்தின் தாவரங்களும், விலங்குகளும் – தொல்காப்பியம் மற்றும் சங்க இலக்கியத்தில் அகம் மற்றும் புறக்கோட்பாடுகள் – தமிழர்கள் போற்றிய அறக்கோட்பாடு – சங்க காலத்தில் தமிழகத்தில் எழுத்தறிவும், கல்வியும் – சங்க கால நகரங்களும் துறைமுகங்களும் – சங்க காலத்தில் ஏற்றுமதி மற்றும் இறக்குமதி – கடல் கடந்த நாடுகளில் சோழர்களின் வெற்றி.* **(3)**

இந்திய தேசிய இயக்கம் மற்றும் இந்திய பண்பாட்டிற்குத் தமிழர்களின் பங்களிப்பு: *இந்திய விடுதலைப் போரில் தமிழர்களின் பங்கு – இந்தியாவின் பிறப் பகுதிகளில் தமிழ்ப் பண்பாட்டின் தாக்கம் – சுயமரியாதை இயக்கம் – இந்திய மருத்துவத்தில், சித்த மருத்துவத்தின் பங்கு – கல்வெட்டுகள், கையெழுத்துப் படிகள் – தமிழ்ப் புத்தகங்களின் அச்சு வரலாறு.* **(3)**

Total L: 15 Periods

25HS102 HERITAGE OF TAMILS
(Common to all B.E./B.Tech./B.Des. Degree Programmes)

1 0 0 1

LANGUAGE AND LITERATURE: Language Families in India - Dravidian Languages – Tamil as a Classical Language - Classical Literature 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 Modern literature in Tamil - Contribution of Bharathiyar and Bharathidhasan. **(3)**

HERITAGE - ROCK ART PAINTINGS TO MODERN ART – SCULPTURE: Hero stone to modern sculpture - Bronze icons - Tribes and their handicrafts - Art of temple car making - Massive Terracotta sculptures, Village deities, Thiruvalluvar Statue at Kanyakumari, Making of musical instruments - Mridhangam, Parai, Veenai, Yazh and Nadhaswaram - Role of Temples in Social and Economic Life of Tamils. **(3)**

FOLK AND MARTIAL ARTS: Therukoothu, Karagattam, Villu Pattu, Kaniyan Koothu, Oyillattam, Leather puppetry, Silambattam, Valari, Tiger dance - Sports and Games of Tamil. **(3)**

THINAI CONCEPT OF TAMILS: 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 of Sangam Age - Export and Import during Sangam Age - Overseas Conquest of Chola. **(3)**

CONTRIBUTION OF TAMILS TO INDIAN NATIONAL MOVEMENT AND INDIAN CULTURE: Contribution of Tamils to Indian Freedom Struggle - The Cultural Influence of Tamils over the other parts of India – Self-Respect Movement - Role of Siddha Medicine in Indigenous Systems of Medicine – Inscriptions & Manuscripts – Print History of Tamil Books. **(3)**

Total L: 15 Period

TEXT – CUM – REFERENCE BOOKS:

1. *கே. கே. பிள்ளை, 'தமிழக வரலாறு - மக்களும் பண்பாடும்'. தமிழ்நாடு பாடநூல் மற்றும் கல்வியியல்*

பணிகள் கழகம்.
2. முனைவர் இல. சுந்தரம், 'கணினித்தமிழ்'. விகடன் பிரசுரம்
3. கீழடி - வைகை நதிக்கரையில் சங்ககால நகர நாகரிகம், தொல்லியல் துறை வெளியீடு
4. பொருநை - ஆற்றங்கரை நாகரிகம், தொல்லியல் துறை வெளியீடு
5. Dr. K.K. Pillay, '*Social Life of Tamils*'. A joint publication of TNTB, ESC and RMRL
6. Dr. S. Singaravelu, '*Social Life of the Tamils – The Classical Period*'. International Institute of Tamil Studies.
7. Dr. S.V. Subaramanian and Dr. K.D. Thirunavukkarasu, '*Historical Heritage of the Tamils*'. International Institute of Tamil Studies
8. Dr. M. Valarmathi, '*The Contributions of the Tamils to Indian Culture*'. International Institute of Tamil Studies.
9. Keeladi – Sangam '*City Civilization on the banks of river Vaigai*'. Department of Archaeology, Tamilnadu Text Book and Educational Services Corporation, Tamilnadu
10. Dr. K.K. Pillay, Studies in the History of India with Special Reference to Tamilnadu
11. '*Porunai Civilization*'. Department of Archaeology, Tamil Nadu Text Book and Educational Services Corporation, Tamilnadu
12. R. Balakrishnan, '*Journey of Civilization Indus to Vaigai*'. RMRL, Tamilnadu

COURSE OUTCOMES:

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Describe the Dravidian language family, outline the features of Tamil classical literature, and explain the development of Tamil art, sculpture, and temple-related traditions in a historical context.	**K2**
CO2	Demonstrate the cultural relevance of Tamil folk and martial arts, apply the concepts of Sangam landscape classification to social contexts, and relate Tamil contributions to India's freedom struggle, cultural legacy, and Siddha medicine.	**K3**

COs-POs & PSOs MAPPING

CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PSO1**	**PSO2**
CO1							3				3		
CO2							3				3		
							3				**3**		

1-Low, 2-Medium, 3-High

25CS111 C PROGRAMMING LABORATORY
(Common to CSE and AI&DS)

0 0 4 2

1. Formatted I/O statements.
2. Decision Making statements: Simple If, If – else, Switch- case.
3. Looping Statements: For, While, Do – while.
4. Single dimensional arrays and multi-dimensional arrays.
5. Operations on Strings.
6. Pass by value and pass by address, Recursion using functions.
7. Structures and nested structures.
8. String handling operations using pointers.
9. Operations on arrays using pointers.
10. File operations using command line arguments

Total P: 60 periods

REFERENCES:

1. Byron S. Gottfried and Jitendar Kumar Chhabra '*Programming with C'*. Tata McGraw Hill Publishing Company, New Delhi, 2018.
2. Herbert Schildt '*C-The Complete Reference'*. Tata McGraw Hill Publishing Company, New Delhi, 2010.
3. Pradip Dey and Manas Ghosh '*Programming in C'*. Oxford University Press, New Delhi, 2018.
4. Yashavant P. Kanetkar '*Let Us C'*. BPB Publications, 2017.
5. H. M. Deitel and P. J. Deitel '*C How to Program'*. Pearson Education, New Delhi, 2013.

COURSE OUTCOMES:

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain how programs use logical flow and data organization to produce correct and meaningful results.	**K2**
CO2	Apply structured problem-solving approaches to develop programs that handle data efficiently.	**K3**
CO3	Analyze program behavior and outputs to manage data effectively and identify possible improvements.	**K4**

COs-POs & PSOs MAPPING

CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PSO1**	**PSO2**
CO1												3	3
CO2	3											3	3
CO3		2										2	2
	3	**2**										**3**	**3**

1-low, 2-medium, 3-high

25EE112 ENGINEERING SKILLS LABORATORY
(Common to CSE and AI&DS)

0 0 4 2

1. Residential house wiring.
2. Stair case wiring.
3. Measurement of electrical quantities Voltage, Current, Power and Power factor.
4. Measurement of energy using single-phase energy meter.
5. Study of iron Box/heater and fan regulator.
6. Verification of Ohms law and Kirchhoff's law.
7. Construction of half-wave and full wave rectifier.
8. Verification of logic gates.
9. Fabrication of electronic circuit using general purpose PCB.
10. Measurement of signal parameters using oscilloscope.
11. Construct amplifier circuit using Op. Amp.

Total P: 60 periods

REFERENCES:

1. D. P. Kothari and I. J. Nagrath, '*Basic Electrical Engineering'.* New Delhi, India: Tata McGraw-Hill, 2010.
2. S. K. Bhattacharya, '*Basic Electrical and Electronics Engineering'.* New Delhi, India: Pearson Education, 2011.
3. R. A. Gayakwad, '*Op-Amps and Linear Integrated Circuits*'. 4th edition. Delhi, India: Pearson, 2015.
4. R. L. Boylestad and L. Nashelsky, '*Electronic Devices and Circuit Theory*'. 11th ed. Upper Saddle River, NJ, USA: Pearson, 2013.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the basics of electrical wiring, measurement of electrical and electronic circuits.	**K2**
CO2	Construct and verify basic circuits using electrical and electronics components.	**K3**
CO3	Collaborate in teams to conduct experiments, and present findings through reports.	-

COs-POs & PSOs MAPPING

CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PSO1**	**PSO2**
CO1													
CO2	3												
CO3		3			2								
	3	**3**			**2**								

1-low, 2-medium, 3-high

25GE111 DESIGN THINKING FOR INNOVATION
(Common to CIVIL, CSE, EEE, ECE, ICE, MECH, AI&DS and EE-VLSI)

0 0 2 1

Foundations of Design Thinking: History & Origins: Roots in Creative Problem Solving: Traces back to mid-20th century practices in architecture, engineering, and psychology. Herbert Simon's "Sciences of the Artificial" (1969): Introduced design as a way of thinking distinct from scientific inquiry. IDEO and the Rise of Human-Centered Design: Popularized design thinking as a repeatable, user-focused innovation process. Stanford school's Influence: Helped institutionalize design thinking in education and entrepreneurship.

Variations of Design Thinking Phases: IDEO's 3-Phase Model: Inspiration, Ideation, Implementation A flexible, non-linear approach emphasizing creativity and action. Stanford school's 5-Phase Model: Empathize, Define, Ideate, Prototype, Test A structured yet iterative framework centered on user empathy. Double Diamond Model (Design Council UK): Divides the process into Discover, Define, Develop, and Deliver—highlighting divergent and convergent thinking.

Related Concepts & Frameworks: Human-Centered Design (HCD): Focuses on designing solutions that deeply resonate with users' needs and contexts. Systems Thinking: Encourages understanding the broader ecosystem and interdependencies within a problem space. Agile & Lean UX: Integrates design thinking with iterative development and minimal viable experimentation. Service Design: Applies design thinking to orchestrate holistic user experiences across touchpoints. Participatory Design: Involves stakeholders directly in the design process to ensure relevance and inclusivity.

EMPATHIZE: Apply Human-Centric Design Principles: Focus on designing solutions that prioritize user needs, experiences, and values throughout the process. Consult Experts: Engage with subject matter experts to gain foundational knowledge about the problem space. Competitive Analysis: Identify & studying similar products or services to identify gaps and opportunities. Stakeholder Interviews: Engaging with people who influence or are affected by the product or service. Conduct Observations: Observe users in their natural environment to understand behaviors, challenges, and interactions. Engage with Users: Use interviews, conversations, and other methods to connect with users and hear their stories. Immerse Yourself: Step into the users' context to experience their environment and challenges firsthand. Create Empathy Maps: Visualize what users say, think, feel, and do to synthesize insights. Identify User Needs and Pain Points: Extract meaningful patterns and needs from user interactions and observations. Set Aside Assumptions: Approach the research with an open mind, suspending personal biases and preconceptions. Document Insights: Capture quotes, observations, and emotional cues to inform the next stage (Define). **(6)**

DEFINE: Organize Research Findings: Review and structure the data collected during the Empathize stage. Analyze Observations: Identify patterns, themes, and insights from user interactions and behaviors. Craft a Human-Centered Problem Statement: Frame the problem from the user's perspective, focusing on their needs—not business goals. Avoid Business- Centric Framing: Refrain from defining problems based on company objectives alone (e.g., market share). Persona Development: Synthesizing research into user personas to guide design decisions. Use Empathy to Guide Definition: Ensure the problem statement reflects real user challenges and motivations. Develop Point-of-View Statements: Create concise summaries that capture who the user is, what they need, and why. Prepare for Ideation: Formulate "How Might We" questions to spark creative thinking in the next phase. **(6)**

IDEATE: Review the Problem Statement: Revisit the user-centric problem defined in the previous stage to guide ideation. Explore Multiple Perspectives: Encourage diverse viewpoints to broaden the range of potential solutions. Use Ideation Techniques: Apply methods like Brainstorming, Brain writing, SCAMPER, and Worst Possible Idea to spark creativity. Encourage Free Thinking: Create a judgment-free space to generate as many ideas as possible without filtering. Expand the Problem Space Push boundaries and explore unconventional or extreme ideas to uncover hidden opportunities. Refine and Select Ideas: Use evaluation techniques to identify promising concepts that address user needs effectively. Prepare for Prototyping: Choose ideas that are feasible and impactful to develop into tangible prototypes in the next stage. **(6)**

PROTOTYPE: Build Low-Cost Prototypes: Create simple, scaled-down versions of the product or its features to explore ideas. Experiment with Solutions: Implement different solutions from the Ideate stage into prototypes for testing. Test Internally and Externally: Share prototypes with team members, other departments, or a small group of users. Observe User Interactions: Watch how users engage with the prototypes to uncover usability issues and insights. Evaluate and Iterate: Accept, refine, or discard prototypes based on user feedback and performance. Identify Limitations: Discover constraints and challenges in the proposed solutions through hands-on testing. Gain Deeper User Understanding: Learn how users think, feel, and behave when interacting with the product. **(6)**

TEST: Conduct Rigorous Testing: Evaluate the complete product using the most promising prototypes. Observe Real User Interactions: Study how users behave, think, and feel while using the product. Gather Feedback and Insights: Collect qualitative and quantitative data to assess usability and effectiveness. Identify Remaining Issues: Detect limitations, pain points, and areas for improvement. Refine and Iterate: Use test results to improve the product and revisit earlier stages if needed. Redefine Problems if Necessary: Reframe or adjust problem statements based on new insights. Enhance Understanding of Users: Deepen empathy and knowledge of user needs through real-world testing. **(6)**

Design Thinking & Customer Centricity: A human-centered approach that blends empathy and innovation to create solutions that truly resonate with customer needs. Practical Examples of Customer Challenges: Real-world scenarios where customers face friction, unmet needs, or emotional disconnects in their product or service journey. Use of Design Thinking to Enhance Customer Experience: Applying iterative problem-solving and user insights to craft experiences that are intuitive, delightful, and deeply relevant. Parameters of Product Experience: Key dimensions like usability, accessibility, emotional impact, and consistency that shape how customers perceive and interact with a product. Alignment of Customer Expectations with Product Design: Ensuring that every design decision reflects what customers value, expect, and aspire to achieve through the product.

Total P: 30 periods

TEXT BOOKS

1. T. Brown '*Change by Design*'. Harper Business, 2009.
2. J. Liedtka and T. Ogilvie '*Designing for Growth*'. Columbia Business School Publishing, 2011.

REFERENCES

1. T. Kelley and D. Kelley '*Creative Confidence*'. Crown Business, 2013.
2. Stanford d. School resources: https://dschool.stanford.edu/
3. https://apphaus.sap.com/toolkit/methods#design-thinking

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Apply empathy-driven research to understand user needs.	**K3**
CO2	Frame actionable problem statements and generate creative ideas.	**K6**
CO3	Develop and test prototypes to refine innovative solutions to the real-world problems.	**K4**

COs-POs & PSOs MAPPING

CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PSO1**	**PSO2**
CO1													
CO2	3							3	3		3	3	3
CO3			2			2		2	2		2	2	2
	3		**2**			**2**		**3**	**3**		**3**	**3**	**3**

1-low, 2-medium, 3-high

25GEM01 INDUCTION PROGRAMME
(Common to CIVIL, CSE, EEE, ECE, ICE, MECH, AI&DS and EE-VLSI)

All students shall undergo an induction programme at the beginning of the first semester for a duration of three weeks as per the guidelines of All India Council for Technical Education (AICTE). A student completing the induction programme will be awarded a completed grade in the grade sheet, and only the students who complete the induction programme shall be considered as eligible for award of degree subject to satisfying other conditions. A student who does not complete the induction programme in the first semester shall redo the same in the subsequent semester.

SEMESTER II

25MA202 TRANSFORMS AND APPLICATIONS
(Common to AI&DS and CSE)

3 1 0 4

LAPLACE TRANSFORMS: Laplace transform, inverse transform, linearity, s-shifting, transforms of derivatives and integrals, unit step function, t – shifting, Dirac's delta function, periodic functions, differentiation and integration of transforms. **(9+3)**

APPLICATION OF LAPLACE TRANSFORMS: Convolution, solving differential equations with constant coefficients and variable coefficients, integral equations, systems of ODEs by using Laplace transform technique. **(9+3)**

Z TRANSFORM: Z-transform, the relationship of the Z-transforms to the Laplace transform, some useful properties, inverse Z-transforms, solution of difference equations. **(6+2)**

FOURIER SERIES: Fourier series – even and odd functions, half range expansion, convergence of Fourier series, basic concepts of PDE's, wave equation, solution of one- dimensional heat equation and steady state two-dimensional heat equation. **(12+4)**

FOURIER TRANSFORMS: Fourier integral, Fourier cosine and sine integrals, Fourier transform, Discrete Fourier transform, Fast Fourier transform – DIT algorithm. **(9+3)**

Total L: 45 + T: 15 = 60 periods

TEXT BOOKS

1. Erwin Kreyszig, '*Advanced Engineering Mathematics*'. Wiley India, New Delhi, 2018.
2. G. D. Dean '*Advanced Engineering Mathematics with MATLAB*'. CRC Press, USA, 2017.

REFERENCES

1. Peter V. O. Neil '*Advanced Engineering Mathematics*'. Cengage, New Delhi, 2018.
2. C. R. Wylie and L. C. Barrett '*Advanced Engineering Mathematics*'. Tata McGraw-Hill, New Delhi, 2019.
3. R. K. Jain and S. R. K. Iyengar '*Advanced Engineering Mathematics*'. Narosa Publishing House, New Delhi, 2018.
4. Alexander D Poularikas '*Transforms and Applications Primer for Engineers with Examples and MATLAB*'. CRC press, USA, 2010.

COURSE OUTCOMES

At the end of this course students will be able to:		**Bloom's Level**
CO1	Explain the concepts related to Laplace, Fourier and Z- Transforms and its Applications.	**K2**
CO2	Apply the techniques of Laplace, Fourier and Z- Transforms to solve engineering problems.	**K3**
CO3	Analyze the solutions of engineering problems employing Laplace, Fourier and Z-Transforms.	**K4**
CO4	Use modern tools to solve engineering problems with the help of Laplace, Fourier and Z- Transforms.	

COs-POs & PSOs MAPPING

CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PSO1**	**PSO2**
CO1													
CO2	3												
CO3		1											
	3	**1**											

1-low, 2-medium, 3-high

25PH204 SENSORS FOR ENGINEERING APPLICATIONS
(Common to CSE, ECE and EE-VLSI)

3 0 0 3

STRAIN AND PRESSURE MEASUREMENT: Resistance strain gauge, piezoelectric pressure sensor, characteristics. Electronic circuits for strain gauge, load cells. Interferometer, Fibre-optic pressure sensor. capacitance pressure sensor. **(9)**

ELECTRONIC SENSORS: Inductive, capacitive and ultrasonic based proximity sensors Reed switch, Hall-effect switching sensors, capacitive based humidity sensor, liquid level detectors, flow sensors, smoke sensors. **(9)**

MOTION SENSORS: Capacitor plate sensor, Inductive sensors, LVDT Accelerometer systems, rotation sensors, piezoelectric devices for motion sensing, Hall effect-based speed sensor. **(9)**

LIGHT Sensors: Color temperature, light flux, photo sensors, photo resistor and photoconductors, photodiodes, phototransistors, photovoltaic devices, fiber-optic sensors and their applications. LIDAR working principle and automotive applications. **(9)**

Thermal Sensors: Bimetallic strip, semiconductor-based temperature sensor, thermocouples, Resistance thermometers, thermistors, PTC and NTC thermistors and their applications. Infrared sensors: bolometer, Pyroelectric detector, photodiodes and phototransistor. **(9)**

Total L: 45 periods

TEXT BOOKS

1. Ian R Sinclair '*Sensors and Transducers, 3rd edition*'. Newnes publishers, 2011.
2. Krzysztof Iniewski '*Smart Sensors for Industrial Applications*'. CRC Press Taylor and Francis, 2019.
3. E. O. Doebelin '*Measurement Systems, Application and Design*'. McGraw Hill, 7th Edition, 2019.

REFERENCES

1. Jack P Holman '*Experimental Methods for Engineers'*. 8th edition, McGraw Hill, USA, 2012.
2. Jacob Fraden '*Handbook of Modern Sensors: Physics, Design, and Applications*'. Springer, 5th edition, 2016.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the working principles and characteristics of various sensors, including strain, pressure, motion, light, and thermal sensors, and their applications in engineering systems.	**K2**
CO2	Apply theoretical concepts to calculate the response of various sensors, such as strain gauges, capacitive sensors, and thermistor, in practical engineering applications.	**K3**
CO3	Analyze sensor data to assess performance in different environments, using appropriate methods to measure strain, motion, temperature, light, and other physical parameters.	**K4**
CO4	Prepare a report or presentation on the applications of different types of sensors in real-world engineering systems, emphasizing the comparison of their operating principles and advantages.	

COs-POs & PSOs MAPPING

CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PSO1**	**PSO2**
CO1													
CO2	3												
CO3		1											
	3	**1**											

1-low, 2-medium, 3-high

25CY202 APPLIED CHEMISTRY
(Common to CSE and AI&DS)

3 0 0 3

ELECTRONIC MATERIALS: Inorganic semiconductors – Elemental – Si and Ge - band theory, doping, compound semiconductors – band gap engineering – applications. Organic semiconductors – conjugated polymers – mechanism of charge transport, doping, states of aggregation, material properties – thermal, mechanical, electrical, chemical, electrochemical. Applications – OLED, OPV – working principle. Liquid crystalline materials – display application. **(9)**

PROCESSES IN ELECTRONICS MANUFACTURE: Microchip fabrication – overview, photoresists – chemistry, types. Fabrication facilities – clean rooms - maintenance, ultrapure water– specification, production processes – ion exchange, reverse osmosis, continuous electrode ionisation. PCB fabrication – electroless and electroplating of copper – principle, bath chemistries and process parameters, formation of copper track on plastic board. **(9)**

ELECTRONICS PACKAGING AND PROTECTION: Packaging materials-encapsulants and underfills - adhesives – chemical types, application methods, factors influencing adhesion, soldering alloys – phase diagrams, lead free alloys, phase change materials for cooling. Conducting inks for printed electronics - metal and carbon based – graphene, CNT– synthesis, structure, electrical properties. Corrosion in electronics – types, protection – vapour phase inhibitors. **(9)**

ELECTROCHEMICAL POWER SOURCES: Electrochemical cells – emf, electrode potential, dependence of emf on electrolyte concentration – Nernst equation. Batteries– performance characteristics. Materials, construction, reactions, characteristics of Leclanche cell, primary lithium batteries, lead - acid battery and lithium-ion batteries. Supercapacitors – EDLC – fundamentals, electrode materials, electrolytes, pseudo capacitors – materials. **(9)**

CHEMICAL SENSORS: Sensors – basic components. Electrochemical sensors- potentiometric transducers – principle, ion-selective electrodes – configurations, response functions and selectivity, applications – potentiometric titrations, water quality monitoring - pH, Hardness, fluoride ion sensors Amperometric transducers – principle, application - glucose biosensors, conductivity sensors – principle – application in conductometric titrations. Colorimetric sensors - Beer-Lambert's law, components, application - determination of ferric ion in water sample. Chemi-resistive sensors - principle, application – environmental monitoring – CO_2 sensor. Microelectrodes for sensors – fabrication. **(9)**

Total L: 45 periods

TEXT BOOKS

1. Shashi Chawla '*A Textbook of Engineering Chemistry'.* Dhanpat Rai and Co, New Delhi, *6th edition*,2022.
2. J. M. G. Cowie and Valeria Arrighi '*Polymers: Chemistry and Physics of Modern Materials*'. CRC Press, London, 3rd edition, 2016.

REFERENCES

1. Bansi D. Malhotra '*Handbook of Polymers in Electronics*'. Rapra Technology Ltd., UK, 1st edition, 2002.
2. Peter Van Zant '*Microchip Fabrication: A Practical Guide to Semiconductor Processing*'. McGraw Hill,6th edition, 2014.
3. Derek Pletcher and Frank C. Walsh. '*Industrial Electrochemistry'*. 2nd Edition, London, Chapman and Hall, 1993.
4. Florinel-Gabriel Banica '*Chemical Sensors and Biosensors – Fundamentals and Applications'*. 1st edition, John Wiley and Sons Ltd, 2012.

COURSE OUTCOMES:

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Learn the chemistry of engineering materials and analytical devices.	**K2**
CO2	Utilize the suitable materials for electronics engineering applications.	**K3**
CO3	Analyze the properties of electronics materials for the fabrication of electronic devices	**K4**

COs-POs & PSOs MAPPING

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

1-low, 2-medium, 3-high

25AD201 PYTHON PROGRAMMING
(Common to CSE and AI&DS)

3 0 0 3

INTRODUCTION TO PYTHON PROGRAMMING: Introduction to Python, Demo of Interactive and script mode, Tokens in Python – Variables, Keywords, Comments, Literals, Data types, Indentation, Operators and its precedence, Expressions, Input and Print functions. Selective statements – if, if-else, nested if, if –elif ladder statements. Iterative statements - while, for, Nested loops, else in loops, break, continue, and pass statements. **(10)**

FUNCTIONS AND STRINGS: Functions: Types, parameters, arguments: positional arguments, keyword arguments, parameters with default values, functions with arbitrary arguments, Scope of variables: Local and global scope, Recursion and Lambda functions. Strings: Formatting, Comparison, Slicing, Splitting, Stripping, Pattern matching. **(8)**

COLLECTIONS: List: Create, Access, Slicing, Negative Indices, List Methods, and comprehensions Tuples: Create, Indexing and Slicing, Operations on tuples. Dictionary: Create, add, and replace values, and operations on dictionaries. Sets: Create and operations on set. **(9)**

FILES, EXCEPTIONS, AND PACKAGES: Files: text files, file handling modes, reading, writing, and appending to files, handling file exceptions, with statement; command line arguments, errors, exception handling with try, handling multiple exceptions; Modules and Packages, Python's Standard Library. **(9)**

OBJECT-ORIENTED PROGRAMMING USING PYTHON: Classes and Objects in Python, properties and methods, Instantiation, __init() function, self-parameter, pass statement, inheritance in Python, __init() function in inheritance, super() method, types of inheritance, function overriding, Polymorphism. **(9)**

TOTAL L: 45 periods

TEXTBOOKS:

1. Eric Matthes '*Python Crash Course A Hands-On, Project - Based Introduction to Programming'.* 3rd edition, No Starch Press, 2024.
2. Mark Summerfield '*Programming in Python 3: A Complete Introduction to the Python Language'. 2nd* edition, Pearson Education, 2018.

REFERENCES:

1. Paul Deitel and Harvey Deitel *'Python for Programmers'*. 1st edition, Pearson Education, 2021.
2. John V Guttag *'Introduction to Computation and Programming Using Python: With Applications to Computational Modeling and Understanding Data.*'. 3rd edition, MIT Press, 2021.
3. Allen B. Downey '*Think Python: How to Think like a Computer Scientist'*. 2nd edition. O'Reilly Publishers, 2016.
4. Martin C. Brown *'Python: The Complete Reference'*. 4th edition, Mc-Graw Hill, 2018.
5. Karl Beecher *'Computational Thinking: A Beginner's Guide to Problem Solving and Programming'*. 1st edition. BCS Learning and Development Limited, 2017.

COURSE OUTCOMES:

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the potential usage of Python in engineering applications.	**K2**
CO2	Apply the concepts of Python programming to solve engineering problems and formulate simple projects.	**K3**
CO3	Analyse data using Python and effectively communicate results while working in groups.	**K4**
CO4	Assess Python programming concepts, libraries, and emerging technologies for practical use.	**K5**

COs-POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1												3	3
CO2	3											3	3
CO3		3										3	3
CO4				2								2	2
	3	**3**		**2**								3	3

1-low, 2-medium, 3-high

25CS201 DIGITAL DESIGN

3 1 0 4

NUMBER SYSTEM: Number Systems and Number-Base Conversion - Complements of Numbers (Diminished Radix Complement, Radix Complement) - Signed Binary Numbers - Arithmetic Operation with the Binary Numbers - fixed- and floating-point representation, Binary Codes (BCD, 2421Code, Gray Code, ASCII) Activities:

- Numerical problem solving on number system conversion, complements
- Short numerical quizzes on complements and signed binary arithmetic **(10+4)**

BOOLEAN ALGEBRA: Boolean Algebra - Basic Theorems and Properties of Boolean Algebra - Simplification of Boolean Functions - Canonical and Standard Forms - Other Logic Operation
Activities:

- Numerical problem solving on simplification of Boolean functions using basic theorems and properties of Boolean algebra **(8+3)**

DESIGN OF COMBINATIONAL CIRCUITS: Introductory Digital Concepts - Digital Logic Gates - Karnaugh Map Method - Don't Care Conditions - The Tabulation Method - NAND and NOR Implementation - Design Procedure - Adder - Subtractor - Magnitude Comparator - Decoders - Encoders - Priority Encoder - Multiplexers - Demultiplexers - Three State Gates - Design Example.
Activities:

- Step-by-step design of combinational circuits from given specifications
- Simulation and verification of digital logic gates using truth tables in a logic simulator
- Modelling and verification of basic digital logic gates/combinational circuits using Verilog HDL/VHDL **(10+3)**

DESIGN OF SEQUENTIAL CIRCUITS: Introduction - Storage Elements: - Latch (S-R Latch, D-Latch) - Flip-Flops (S-R Flip Flop, D-Flip Flop, J-K Flip Flop, T-Flip Flop) - Master Slave Configuration of J-K Flip Flop - Shift Registers - Design of Asynchronous and Synchronous Counter. Mealy and Moore Models of Finite State Machines (FSM) - Synchronous Sequential Logic - State Reduction and Assignment - Design Procedure Algorithmic
Activities:

- Design and simulation of synchronous counters with specified counting sequences
- Numerical problem solving on state transition analysis in synchronous sequential circuits **(11+3)**

MEMORY AND PROGRAMMABLE LOGIC: Introduction - Random Access Memory - Memory Decoding - Read Only Memory - Programmable Logic Array (PLA) - Programmable Array Logic (PAL) - Sequential Programmable Devices.
Activities:

- Comparative analysis of PLA and PAL architectures
- Interpretation of memory and programmable logic device specifications **(6+2)**

Total L: 45 +T: 15 = 60 periods

TEXT BOOKS

1. M. Morris Mano and Michael D. Ciletti. '*Digital Design: With an Introduction to the Verilog HDL, VHDL and System Verilog Sixth Edition'*. USA: Pearson Education, 2018.
2. M. Morris Mano. '*Digital Logic and Computer Design*'. Pearson Education, 2017.
3. Thomas L. Floyd. '*Digital Fundamentals*'. Pearson Education, 2015.

REFERENCES

1. Charles H. Roth, Jr and Larry L. Kinney. '*Fundamentals of Logic Design*'. Cengage Learning, 2014.
2. John F. Wakerly. '*Digital Design: Principles and Practices, Pearson Education*'. 2018.
3. Roger L Tokheim. '*Digital Electronics: Principles and Applications'*. McGraw-Hill Education, 2013.
4. Ronald Tocci, Neal Widmer and Greg Moss '*Digital Systems*'. Pearson Education, 2016.
5. Donald D. Givone '*Digital Principles and Design*'. McGraw-Hill Education, 2003.

COURSE OUTCOMES:

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the concepts of Digital Electronic circuit s and Systems.	**K2**
CO2	Apply the concepts and simplification methods of Digital circuits for the given application.	**K3**
CO3	Analyze the given digital electronic circuits and arrive at suitable conclusions.	**K4**
CO4	Design a digital circuit for the given application and constraints.	**K5**

COs-POs & PSOs MAPPING

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

1-low, 2-medium, 3-high

25HS201 தமிழரும் தொழில்நுட்பமும்

1 0 0 1

நெசவு மற்றும் பானைத் தொழில்நுட்பம்: *சங்க காலத்தில் நெசவுத் தொழில் – பானைத் தொழில்நுட்பம் – கருப்பு சிவப்பு பாண்டங்கள் – பாண்டங்களில் கீறல் குறியீடுகள்.* **(3)**

வடிவமைப்பு மற்றும் கட்டிடத் தொழில்நுட்பம்: *சங்க காலத்தில் வடிவமைப்பு மற்றும் கட்டுமானங்கள், சங்க காலத்தில் வீட்டுப் பொருட்களில் வடிவமைப்பு – சங்க காலத்தில் கட்டுமான பொருட்களும் நடுகல்லும் – சிலப்பதிகாரத்தில் மேடை அமைப்பு பற்றிய விவரங்கள் – மாமல்லபுரச் சிற்பங்களும், கோவில்களும் – சோழர் காலத்துப் பெருங்கோயில்கள் மற்றும் பிற வழிபாட்டுத் தலங்கள் – நாயக்கர் காலக் கோயில்கள் – மாதிரி கட்டமைப்புகள் பற்றி அறிதல், மதுரை மீனாட்சி அம்மன் ஆலயம் மற்றும் திருமலை நாயக்கர் மஹால் – செட்டிநாட்டு வீடுகள் – பிரிட்டிஷ் காலத்தில் சென்னையில் இந்தோ-சாரோசெனிக் கட்டிடக் கலை.* **(3)**

உற்பத்தித் தொழில் நுட்பம்: *கப்பல் கட்டும் கலை – உலோகவியல் – இரும்புத் தொழிற்சாலை – இரும்பை உருக்குதல், எஃகு – வரலாற்றுச் சான்றுகளாக செம்பு மற்றும் தங்க நாணங்கள் – நாணயங்கள் அச்சடித்தல்- மணி உருவாக்கும் தொழிற்சாலைகள் – கல்மணிகள், கண்ணாடி மணிகள் – சுடுமண் மணிகள் – சங்கு மணிகள் – எலும்புத் துண்டுகள் – தொல்லியல் சான்றுகள் – சிலப்பதிகாரத்தில் மணிகளின் வகைகள்.* **(3)**

வேளாண்மை மற்றும் நீர்ப்பாசனத் தொழில் நுட்பம்: *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.* **(3)**

அறிவியல் தமிழ் மற்றும் கணினித்தமிழ்: *அறிவியல் தமிழின் வளர்ச்சி – கணினித்தமிழ் வளர்ச்சி – தமிழ் நூல்களை மின்பதிப்பு செய்தல் – தமிழ் மென்பொருட்கள் உருவாக்கம் – தமிழ் இணையக் கல்விக்கழகம் – தமிழ் மின் நூலகம் – இணையத்தில் தமிழ் அகராதிகள் – சொற்குவைத் திட்டம்.* **(3)**

Total L: 15 periods

25HS201 TAMILS AND TECHNOLOGY

1 0 0 1

WEAVING AND CERAMIC TECHNOLOGY: Weaving Industry during Sangam Age – Ceramic technology – Black and Red Ware Potteries (BRW) – Graffiti on Potteries. **(3)**

DESIGN AND CONSTRUCTION TECHNOLOGY: Designing and Structural construction House & Designs in household materials during Sangam 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)-Thirumalai Nayakar Mahal - Chetti Nadu Houses, Indo - Saracenic architecture at Madras during British Period. **(3)**

MANUFACTURING TECHNOLOGY: 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 - Gem stone types described in Silappathikaram. **(3)**

AGRICULTURE AND IRRIGATION TECHNOLOGY: Dam, Tank, ponds, Sluice, Significance of Kumizhi Thoompu of Chola Period, Animal Husbandry - Wells designed for cattle use - Agriculture and Agro Processing - Knowledge of Sea - Fisheries – Pearl - Conch diving - Ancient Knowledge of Ocean - Knowledge Specific Societ. **(3)**

SCIENTIFIC TAMIL & TAMIL COMPUTING: 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. **(3)**

Total L: 15 Periods

TEXT – CUM – REFERENCE BOOKS

1. *கே. கே. பிள்ளை, தமிழக வரலாறு - மக்களும் பண்பாடும், தமிழ்நாடு பாடநூல் மற்றும் கல்வியியல் பணிகள் கழகம்.*
2. *முனைவர் இல. சுந்தரம், கணினித்தமிழ், விகடன் பிரசுரம்*
3. *கீழடி - வைகை நதிக்கரையில் சங்ககால நகர நாகரிகம், தொல்லியல் துறை வெளியீடு*
4. *பொருநை - ஆற்றங்கரை நாகரிகம், தொல்லியல் துறை வெளியீடு*

5. Dr. K.K. Pillay, '*Social Life of Tamils*'. A joint publication of TNTB, ESC and RMRL
6. Dr. S. Singaravelu, '*Social Life of the Tamils – The Classical Period*'. International Institute of Tamil Studies.
7. Dr. S.V. Subaramanian and Dr. K.D. Thirunavukkarasu, '*Historical Heritage of the Tamils*'. International Institute of Tamil Studies
8. Dr. M. Valarmathi, '*The Contributions of the Tamils to Indian Culture*'. International Institute of Tamil Studies.
9. Keeladi – '*Sangam City Civilization on the banks of river Vaigai*'. Department of Archaeology, Tamilnadu Text Book and Educational Services Corporation, Tamilnadu
10. Dr. K.K. Pillay, Studies in the History of India with Special Reference to Tamilnadu
11. '*Porunai Civilization*'. Department of Archaeology, Tamil Nadu Text Book and Educational Services Corporation, Tamilnadu
12. R. Balakrishnan, '*Journey of Civilization Indus to Vaigai*'. RMRL, Tamilnadu

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Identify the significance of ancient Tamil technologies in weaving, pottery, metallurgy, and architecture, with emphasis on traditional design and construction methods across historical periods.	**K2**
CO2	Use insights from traditional Tamil knowledge systems in agriculture, irrigation, and marine sciences, and connect the development of Tamil language to its applications in digital platforms and computing.	**K3**

COs-POs & PSOs MAPPING

CO/PO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PSO1**	**PSO2**
CO1						1			1		1		
CO2						1			1		1		
						1			**1**		**1**		

1-Low, 2-Medium, 3-High

25AD211 PYTHON PROGRAMMING LABORATORY
(Common to CSE and AI&DS)

0 0 4 2

1. Algorithm and Flowchart for Real World Problems
2. Programs using Decision Making statements and Looping Statements
3. Applications using Set, Lists, Tuples, Dictionary
4. Applications using Functions
5. Python Libraries
6. Text processing
7. Searching & sorting algorithms
8. Application Debugging
9. Build a Simple Python Application

Total P: 60 periods

REFERENCES

1. Romano, Fabrizio '*Learn Python Programming: A Beginners Guide to Learning the Fundamentals of Python Language to Write Efficient, High-Quality Code'.* 2nd edition, India, Packt Publishing, 2018.
2. R. Nageswara Rao *'Core Python Programming'.* 2nd edition, Dreamtech Press, 2019
3. Vijay Kumar Sharma, Vimal Kumar, Swati Sharma and Shashwat Pathak '*Python Programming: A Practical Approach*'. CRC Press, 2021.
4. Meenu Kohli '*Basic Core Python Programming: A Complete Reference Book to Master Python with Practical Applications'.* 1st edition', BPB Publications, 2021.
5. J. Cogliati, *Non-Programmers Tutorial for Python 3'*. Platypus Global Media, 2019.

COURSE OUTCOMES:

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Apply programming skills to solve real-world problems using structured and logical approaches.	**K3**
CO2	Analyze program behavior to identify errors and improve solution efficiency.	**K4**

COs-POs & PSOs MAPPING

CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PSO1**	**PSO2**
CO1	3											3	3
CO2		2										2	2
	3	**2**										**3**	**3**

1-low, 2-medium, 3-high

25BS212 PHYSICS AND CHEMISTRY LABORATORY
(Common to CSE and ICE)

0 0 4 2

PHYSICS (ANY EIGHT EXPERIMENTS)

1. Determination of specific resistance of a given wire – Carey foster's bridge.
2. Determination of Young's modulus of the material- Uniform Bending.
3. Measurement of Hall coefficient of a semiconductor using Hall effect setup.
4. Determination of electrical resistivity of a given material using four probe setups.
5. Determination of wavelength of laser using diffraction grating - LASER.
6. Determination of Thickness of a thin wire – Air wedge method.
7. Study of I-V characteristics of solar cell and determination of its efficiency.
8. Determination of velocity of sound and compressibility of liquid - Ultrasonic Interferometer.
9. Determination of Planck's constant and work function of a metal -Photoelectric Effect.
10. Determination of bandgap of a semiconductor – Post office box.
11. Validation of Faraday's Law of induction.
12. Interpreting the working mechanism of spirometer, CO_2 sensor, Venturi tube and heart rate sensor.

DEMONSTRATION:

1. Determination of Numerical Aperture and Acceptance angle - Optical Fiber
2. Study the energy loss of a ferrite magnetic material specimen by B-H curve.

Total P: 60 periods

REFERENCES:

1. Department of Physics, '*Physics laboratory observation'*. 2025, PSG Institute of Technology and Applied Research.
2. Jerry D Wilson; Cecilia A Hernandez Hall, '*Physics laboratory experiments*'. Boston, MA: Cengage Learning, 2016.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Relate the scientific principles, compare the experimental results with theoretical calculations, and apply graphical analysis to visualize the importance of precise measurements.	**K3**
CO2	Analyses the experimental result outcomes using analytical and experimental skills for various engineering materials and applications.	**K4**

CHEMISTRY (ANY EIGHT EXPERIMENTS):

1. Determination of total, temporary & permanent hardness of water by EDTA method.
2. Determination of strength of acids in a mixture of acids using conductivity meter.
3. Determination of strength of given hydrochloric acid using pH meter.
4. Estimation of iron content of the given solution using potentiometer.
5. Corrosion experiment-weight loss method.
6. Electroplating of copper and Nickel and determination of coulombic efficiency.
7. Designing a battery and determination of its characteristics.
8. Construction of phase diagram of a simple eutectic system.
9. Determination of kinematic viscosity and acid value of a lubricating oil.
10. Anodizing of aluminium and determination of thickness of anodised film.

Total P: 60 periods

REFERENCES:

1. J Mendham '*Vogel's Textbook of Quantitative Chemical Analysis*'. 6th Ed., Pearsons Education 2009.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Demonstrate the measurement of water quality parameters in the given water sample.	**K3**
CO2	Analyze the properties of materials for Engineering applications.	**K4**

COs-POs & PSOs MAPPING

CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PSO1**	**PSO2**
CO1	3								3		3		
CO2				3				3				1	
	3			**3**				**3**	**3**		**3**	**1**	

1-low, 2-medium, 3-high

25GE211 ENGINEERING GRAPHICS
(Common to AI&DS and CSE)

0 0 4 2

INTRODUCTION TO ENGINEERING GRAPHICS **(4)**

1. Introduction to Engineering Graphics.
2. Lettering practice as per BIS.
3. Principles of Dimensioning.

ORTHOGRAPHIC PROJECTIONS **(40)**

1. Introduction to Orthographic Projections.
2. Drawing multiple views from pictorial views of objects.
3. Projection of points.
4. Projection of straight lines (only First angle projections) inclined to both the principal planes.
5. Projection of planes (polygonal and circular surfaces) inclined to both the principal planes by rotating object method.
6. Projection of simple solids when the axis is inclined to one of the principal planes and parallel to the other by rotating object method.

SECTION OF SOLIDS **(8)**

1. Section of simple solids in simple vertical position when the cutting plane is inclined to one of the principal planes and perpendicular to the other and obtaining true shape of section.

ISOMETRIC PROJECTIONS **(8)**

1. Isometric projection of simple solids in simple vertical positions.

Total P: 60 periods

TEXT BOOKS:

1. N. D. Bhatt '*Engineering Drawing*'. Charotar Publishing House Pvt. Ltd., 55th Edition, 2025.
2. K. C. John '*Engineering Graphics for Degree*'. Prentice Hall India Publishers, 2009.
3. K. V. Natarajan '*A Text book of Engineering Graphics*'. Dhanalakshmi Publications, 34th Refined Edition, 2021.

REFERENCES:

1. K. Venugopal and V. Prabhu Raja '*Engineering Graphics, New Age International Publishers, 17th Edition*'. 2024.
2. Bureau of Indian Standards '*Engineering Drawing Practices for Schools and Colleges SP 46-2003*'. BIS, New Delhi, 2003.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Use the BIS standards and specifications for drawing the engineering components and structures.	**K2**
CO2	Apply orthographic projection principles to draw projection of points, lines, planes, solids, and sectioned solids, isometric and perspective projection of regular solids.	**K3**

COs-POs & PSOs MAPPING

CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PSO1**	**PSO2**
CO1													
CO2	3												
	3												

1-low, 2-medium, 3-high

25EEC01 WORKPLACE COMMUNICATION SKILLS
(Common to CIVIL, CSE, EEE, ECE, ICE, MECH, AI&DS and EE-VLSI)

0 0 2 0

BUILDING COMMUNICATION SKILLS:

1. Introduction to Workplace Communication
2. Profile Building for Internships
3. English in the Workplace (Grammar & Vocabulary)
4. Professional Communication (Speaking & Writing)
5. Workplace Communication Tools
6. Career Exploration
7. Resume Update

Total P: 30 periods

SEMESTER III

25MA302 LINEAR ALGEBRA

3 1 0 4

VECTOR SPACES: General vector spaces - real vector spaces - Euclidean n-space - subspaces – linear independence- coordinates and basis - dimension – change of basis - row space, column space and null space – consistency of linear systems - rank and nullity. **(9+3)**

LINEAR TRANSFORMATIONS: General linear transformation - matrix transformations- geometry of linear operators on R^2- matrices for general linear transformations – similarity. **(9+3)**

INNER PRODUCT SPACES: Inner products - angle and orthogonality in inner product spaces, Gram - Schmidt process: QR decomposition, best approximation: least squares - least squares fitting to data. **(9+3)**

EIGENVALUES AND EIGENVECTORS: Eigenvalues and eigenvectors, diagonalization - orthogonal diagonalization - quadratic forms. **(9+3)**

MATRIX DECOMPOSITION AND OPTIMIZATION: Spectral decomposition - singular value decomposition - reduced singular value decomposition - optimization using quadratic forms. **(9+3)**

Total L: 45 + T: 15 = 60 Periods

TEXT BOOKS

1. Howard Anton, Chris Rorres, Anton Haul, '*Elementary Linear Algebra*'. Wiley India, New Delhi, 2019.
2. David C Lay,Judi J McDonald, Steven R. Lay, '*Linear Algebra and its Applications*'. Pearson Education, New Delhi, 2021.

REFERENCES

1. Gareth Williams, '*Linear Algebra with Applications*'. Narosa Publishing House, New Delhi, 2017.
2. Gilbert Strang, '*Linear Algebra and Learning from Data*'. Wellesley-Cambridge Press, USA, 2019.
3. Friedberg, Insel and Spence, '*Linear Algebra*'. Pearson Education, USA, 2015.
4. Steven J Leon, 'Linear Algebra with Applications'. Pearson Education, USA, 2015.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the concepts related to Vector spaces, Linear transformations, Inner product spaces, Eigenvalues, Eigenvectors and Matrix decomposition.	**K2**
CO2	Apply the techniques of Vector spaces, Linear transformations, Inner product spaces, Eigenvalues, Eigenvectors and Matrix decomposition to solve engineering problems.	**K3**
CO3	Analyse the solutions of engineering problems employing Vector spaces, Linear transformations, Inner product spaces, Eigenvalues, Eigenvectors and Matrix decomposition	**K4**
CO4	Use modern tools to solve engineering problems with the help of Vector spaces, Linear transformations, Inner product spaces, Eigenvalues, Eigenvectors and Matrix decomposition.	

COs-POs & PSOs MAPPING

CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PSO1**	**PSO2**
CO1													
CO2	3												
CO3		1											
CO4					1								
	3	**1**			**1**								

1-low, 2-medium, 3-high

25MA305 DISCRETE STRUCTURES

3 1 0 4

LOGIC AND PROPOSITIONAL CALCULUS: Propositional logic - connectives- compound propositions - logic and bit operations - propositional equivalences - normal forms - rules of inference.

(9+3)

PREDICATE CALCULUS AND PROOF STRATEGY: Predicates - quantifiers - quantifiers with restricted domain - logical equivalences involving quantifiers - rules of inference for quantified statements - introduction to proofs - direct proof - contraposition - contradiction - mathematical induction - program correctness.

(9+3)

RELATIONS AND LATTICES: Relations and their properties – representing relations – types of relations - closures of relations – partial orderings - lattices as posets – Hasse diagram - properties of lattices.

(9+3)

ALGEBRAIC STRUCTURES: Semi groups and monoids - groups – subgroups – group homomorphism - permutation groups - rings - integral domain – fields.

(9+3)

CODING THEORY: Encoding and decoding functions - Hamming distance - error correction and detection group codes - maximum likelihood decoding technique - polynomial rings and codes.

(9+3)

Total L: 45 + T: 15 = 60 Periods

TEXT BOOKS

1. Kenneth H Rosen, '*Discrete Mathematics and its Applications*'. Tata McGraw Hill, New Delhi, 2021.
2. Tremblay J P and Manohar R, '*Discrete Mathematical Structures with Applications to Computer Science*'. Tata McGraw Hill, New Delhi, 2017.

REFERENCES

1. Bernard K, Robert C. B and Sharon C. R, '*Discrete Mathematical Structures*'. Prentice Hall, New Delhi, 2018.
2. Ralph P. G,' *Discrete and Combinatorial Mathematics - An Applied Introduction*'. Addison Wesley, USA, 2019.
3. Richard J, '*Discrete Mathematics*'. Pearson Education, New Delhi, 2018.
4. Herstein I. N, '*Topics in Algebra*'. Wiley India, New Delhi, 2013.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the concepts related to Logic, Predicate Calculus, Relations, Lattices, Algebraic Structures and Coding theory.	**K2**
CO2	Apply the techniques of Logic, Predicate Calculus, Relations, Lattices, Algebraic Structures and Coding theory to solve engineering problems.	**K3**
CO3	Analyze the solutions of engineering problems employing Logic, Predicate Calculus, Relations, Lattices, Algebraic Structures and Coding theory.	**K4**
CO4	Use modern tools to solve engineering problems with the help of Logic, Predicate Calculus, Relations, Lattices, Algebraic Structures and Coding theory.	

COs-POs & PSOs MAPPING

CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PSO1**	**PSO2**
CO1													
CO2	3												
CO3		1											
CO4					1								
	3	**1**			**1**								

1-low, 2-medium, 3-high

25CS301 DATA STRUCTURES

3 0 0 3

INTRODUCTION: Need for data structures – Types of data structures – Abstract Data Type – Algorithm complexity analysis of linear and non-linear data structures – Best case and worst-case complexities – Asymptotic notations. **(9)**

ARRAYS AND LISTS: Array representation and operations – Matrix representation using multi-dimensional arrays – Linked list representation – Operations on singly linked list – Types of linked list – Polynomial addition – Sparse matrices. **(9)**

STACKS AND QUEUES: Stack ADT – Representation and operations – Expressing handling – Role of stack in implementing recursive algorithms – Queue ADT – Representation and operations – Types of queues – Circular queue – Dequeue – Priority queue. **(9)**

TREES: Terminologies – Binary Tree – Traversal – Expression trees – Binary Heap – Priority queue implementation using Binary Heap – Binary Search Tree – AVL Tree – m-way search trees – B Tree – B+ Tree – Application – Tree structure. **(9)**

HASHING AND GRAPHS: Hash table – Hash functions – Resolving collisions – Rehashing. GRAPHS: Graph terminologies – Types of graphs – Representation – Breadth First Search – Depth First Search – Topological sort. **(9)**

Total L: 45 periods

TEXTBOOKS:

1. Mark Allen Weiss, '*Data Structures and Algorithm Analysis in C*'. 2nd Edition, Pearson Education, 2019.
2. Jean Paul Tremblay, Sorenson, '*An Introduction to Data Structures with Applications*'. 2nd Edition, McGraw Hill Publishing Company, New Delhi, 2017.

REFERENCES:

1. Ellis Horowitz, SartaiShani, Sanguthevar Rajasekaran, '*Fundamentals of Computer Algorithms*'. 2nd edition, Universities Press, 2019.
2. Salaria R S, '*Data Structures and Algorithms using C*'. 5th Edition, Khanna Book Publishing, New Delhi, 2017.
3. Amol M. Jagtap, Ajit S. Mali, '*Data Structures Using C – A Practical Approach for Beginners*'. Chapman and Hall/CRC, 1st edition, 2021.
4. Aaron M Tanenbaum, Moshe J Augenstein and Yedidyah Langsam, '*Data Structures Using C and C++*'. 2nd edition, Prentice Hall, 2021.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the fundamental concepts, classifications, and principles of data structures and algorithm complexity.	**K2**
CO2	Apply appropriate data structures and their operations to solve computational problems.	**K3**
CO3	Analyze the complexity of data structures and associated algorithms.	**K4**
CO4	Design and develop efficient solutions using appropriate linear and non-linear data structures.	

COs-POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1												3	3
CO2	3											3	3
CO3		3										3	3
CO4			2					2				2	2
	3	**3**	**2**					**2**				**3**	**3**

1-low, 2-medium, 3-high

25CS302 COMPUTER ORGANIZATION AND ARCHITECTURE

3 1 0 4

COMPUTER ORGANIZATION & DESIGN: Stored program organization (Von Neumann architecture) – Computer registers – Computer instructions – Timing and control (Hardwired and Micro programmed) – Instruction cycle – Memory reference instructions – Input / Output Instructions – Quantitative principles of computer design.
Activities

- Execution flow analysis of instructions in a processor
- Numerical problem-solving on quantitative principles (execution time, speedup, CPI). **(10+3)**

PROCESSOR DESIGN: Stack organization – Instruction formats – Addressing modes – Fixed point arithmetic: addition, subtraction, multiplication (Booth's algorithm) and division (restoring and non-restoring division algorithm) – RISC vs. CISC.
Activities

- Stepwise demonstration of Booth's algorithm for signed multiplication.
- Comparative execution of restoring and non-restoring division algorithms. **(9+3)**

MEMORY AND I/O SYSTEMS: Memory Hierarchy – Associative memory – Cache memory – Mapping policies – Cache optimization, I/O Systems: Introduction – Interrupts – Modes of transfer – DMA.
Activities

- Numerical exercises on cache mapping techniques (Direct, Associative, Set-Associative).
- Case study on cache optimization techniques in modern processors. **(8+3)**

PARALLELISM: Pipelining – Pipelining hazards – Overcoming hazards – Instruction level parallelism – Dependencies.
Activities

- Problem-solving on pipeline performance and speedup calculations.
- Demonstration of hazard resolution techniques (stalling, forwarding, branch prediction)**.** **(10+3)**

MULTIPROCESSOR SYSTEMS: Symmetric and distributed shared memory architectures – Challenges – Cache Coherence – Snooping protocol.
Activities

- Illustration of cache coherence problem using example scenarios.
- Group presentation on multiprocessor architecture used in modern CPUs **(8+3)**

Total L: 45 + T: 15 = 60 Periods

TEXTBOOKS:

1. Morris Mano, Rajib Mall, '*Computer System Architecture*'. Revised 3rd edition, Prentice Hall of India, 2017.
2. John L. Hennessey, David A. Patterson, '*Computer Architecture: A Quantitative Approach*'. 7th edition, Elsevier India Pvt. Ltd, New Delhi, 2025.

REFERENCES:

1. Carl Hamacher, '*Computer Organization*'. 6th Edition, Tata McGraw Hill Publishing, New Delhi2023.
2. Kai Hwang, '*Advanced Computer Architecture Parallelism, Scalability, Programmability.* 3rd edition, Tata McGraw Hill, New Delhi, 2017.
3. William Stallings, '*Computer Organization and Architecture*'. 11th edition, Pearson Education/Prentice Hall of India, New Delhi, 2022.
4. Kai Hwang, Faye A Briggs, '*Computer Architecture and Parallel Processing'*. Tata McGraw Hill Publishing Company Limited, New Delhi, 2017.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the basic principles and organization of computer systems and architectures	**K2**
CO2	Apply appropriate design techniques and computational methods to solve problems related to computer systems and architectures	**K3**
CO3	Analyze the performance and efficiency of computing systems using architectural and operational parameters.	**K4**
CO4	Prepare a report or presentation demonstrating system design considerations and their practical applications in modern computing environments.	

COs-POs & PSOs MAPPING

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

1-low, 2-medium, 3-high

25CS303 OBJECT ORIENTED PROGRAMMING

2 0 2 3

CLASSES AND OBJECT-ORIENTED CONCEPTS: Object oriented programming paradigms – Java buzzwords – Data types, variables and arrays – Operators – Control statements – Programming structures in Java – Classes – Constructors – Methods – Access specifiers – Static members –- Overloading methods - Objects as parameters - Returning objects. **(6)**

INHERITANCE, PACKAGES AND INTERFACES: Inheritance Basics – Types of inheritance – Super keyword - Method overriding - Dynamic method dispatch - Abstract classes - Final keyword with inheritance – Packages - Packages and member access - Importing packages – Interfaces. **(6)**

EXCEPTION HANDLING AND MULTITHREADING: Exception Handling basics – Multiple catch clauses – Nested try statements – Java's built-in exceptions – User defined exception. Multithreaded programming: Java thread model – creating a thread and multiple threads – Priorities – Synchronization – Inter thread communication-Suspending – Resuming and Stopping threads. . **(6)**

I/O AND STRING HANDLING: I/O Basics – Reading and Writing console I/O – Reading and writing files. Strings: Basic String class, methods and String buffer class – Wrappers – Auto boxing. **(6)**

GENERICS AND COLLECTIONS: Generics: Generic programming – Generic classes – Generic Methods – Bounded Types – Restrictions and Limitations. Collection Framework in Java: List – Array list – Hash table – Hash set – Tree set – Stack – Priority queue. **(6)**

LIST OF EXPERIMENTS

1. Classes and Methods
2. Inheritance
3. Abstract Classes
4. Interfaces and Packages
5. Exception Handling
6. Multithreading Concepts
7. I/O and Files
8. Generics and Lambda Expressions
9. Collection Classes

Total L: 30 + P: 30 = 60 periods

TEXTBOOKS:

1. Herbert Schildt, *'Java: The Complete Reference'*. 11th edition, McGraw Hill Education, New Delhi, 2019.
2. Deitel P and Deitel H, '*Java: How to Program*'. 11th edition, Prentice Hall, 2018.

REFERENCES:

1. Cay S. Horstmann, *'Core Java Fundamentals'.* Volume 1, Eleventh Edition, Prentice Hall, 2018.
2. James Gosling, Bill Joy, Guy Steele, Gilad Bracha, Alex Buckley and Daniel Smith, '*The Java Language Specification – Java SE*'. Thirteenth Edition, Oracle America Inc., USA, 2019.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the core Object Oriented Programming concepts and their applications.	**K2**
CO2	Apply Java constructs to develop various applications.	**K3**
CO3	Analyze solutions involving code reusability and complexity management	**K4**
CO4	Develop modular and reusable applications by integrating OOP principles and advanced programming concepts.	
CO5	Demonstrate problem-solving skills through the development of well-structured and maintainable applications.	

COs-POs & PSOs MAPPING

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

1-low, 2-medium, 3-high

25HS301 PROJECT AND FINANCE MANAGEMENT

3 0 0 3

INTRODUCTION TO PROJECT MANAGEMENT: Project: Trends in project management, project management versus general management, agile project management, the three goals of a project, life cycle of projects, project selection methods, project portfolio process, case study – friendly assisted living facility. **(9)**

ROLE OF PROJECT MANAGER AND ORGANISATION: Project manager's roles and responsibilities, selection of a project manager, project management as a profession, fitting projects into the parent organization, the project team and agile team roles, case study – the company with traditional functional organizational structure setting up teams for the new initiatives. **(9)**

PROJECT ACTIVITIES: The planning process, work-breakdown structure and other aids, risk management, methods of budgeting, cost estimation, scheduling the project with PERT and CPM networks, allocating resources, resource loading and leveling, Goldratt's Critical Chain, application – using Project Libre for project management, case study – success of Chandrayan-3. **(9)**

INTRODUCTION TO FINANCE MANAGEMENT: Overview - finance and related disciplines, scope and objectives of financial management, time value of money, valuation of securities, and risk and return and calculations with spread sheet, analysis using cash flow statement and financial statements. **(9)**

PERSONAL FINANCE: Compounding, debt, equity and financial markets and investments- debt and bonds. Equity, mutual funds, hedge funds, real estate, and commodities, Personal financial plan to enhance wealth and job marketability, components of a financial plan, tools for planning – financial statements, applying time value concept of money and tax planning. **(9)**

Total L: 45 Periods

TEXTBOOKS

1. Jack R. Meredith and Scott M. Shafer, '*Project Management in Practice*'. Wiley, 2021.
2. Khan M. Y. and Jain P. K., '*Basic Financial Management*'. Tata McGraw Hill, 2012.
3. Michael Fisher, '*Saving and Investing*'. Author House, 2005.
4. Jeff Madura, '*Personal Finance*'. Pearson, 2020.

REFERENCES

1. National Finance Olympiad, '*Personal Finance Handbook*'. Pockvue Solutions, 2024.
2. Glen Arnold, '*Investing*'. Financial Times Guides, 2020.
3. Rachel Siegel and Carol Yacht, '*Personal Finance*'. Open Textbook Library, Saylor Foundation, 2009.
4. Google, '*Google Project Management: Professional Certificate*'. Google Project Management: Professional Certificate, Coursera.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Identify various budgeting and cost estimation techniques suited to different project scenarios and the uses of project scheduling methods.	**K1**
CO2	Understand the basic concepts of project management, phases of project life cycle, the roles and responsibilities of project manager and how projects are integrated into different types of organizational structures.	**K2**
CO3	Apply theoretical knowledge and practical tools to support sound financial decision-making in real-world scenarios.	**K3**
CO4	Differentiate between various financial instruments and application of financial planning to enhance personal wealth.	**K4**

COs-POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1										2			
CO2										3			
CO3										2	2		
CO4										3	3		
										3	**3**		

1-low, 2-medium, 3-high

25CS311 DATA STRUCTURES LABORATORY

0 0 4 2

1. Solving problems using arrays.
2. Searching and Sorting algorithms.
3. Implementation of linked list.
4. Applications of linked list.
5. Implementation of stack and queue.
6. Applications of stack.
7. Operations on Binary Search Trees.
8. Applications of Binary Search Tree, AVL tree.
9. Graphs – Depth First Search and Breadth First Search.
10. Hashing and Collision Resolution.

Total P: 60 periods

REFERENCES:

1. Ellis Horowitz, Sartaj Sahni and Sanguthevar Rajasekaran, '*Fundamentals of Computer Algorithms*'. 2nd Edition, Universities Press, 2019.
2. Jean Paul Tremblay and Sorenson, '*An Introduction to Data Structures with Applications*'. McGraw Hill Publishing Company, New Delhi, 2017.
3. Thomas H Cormen, Charles E Leiserson, Ronald L Rivest and Clifford Stein, '*Introduction to Algorithms*'. MIT Press, England, 2009.
4. Salaria R. S, '*Data Structures and Algorithms Using C*'. 5th Edition, Khanna Book Publishing, New Delhi, 2017.
5. Amol M. Jagtap, Ajit S. Mali, '*Data Structures Using C – A Practical Approach for Beginners*'. Chapman and Hall/CRC, 1st Edition, 2021.

COURSE OUTCOMES:

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Apply appropriate data structures to find solutions for computational problems.	**K3**
CO2	Analyze optimized solutions to improve algorithmic efficiency and performance.	**K4**

COs-POs & PSOs MAPPING

CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PSO1**	**PSO2**
CO1	3											3	3
CO2		3										3	3
	3	**3**										**3**	**3**

1-low, 2-medium, 3-high

25EEC02 FOUNDATIONS OF PROBLEM SOLVING

0 0 2 1

1. Speed Mathematics (SAW, Oz, Mirror methods)
2. Speed Mathematics (High5, Minion, Butterfly methods)
3. Speed Mathematics (Inception, Goldeneye methods)
4. Thinking with Numbers
5. Problem Solving with Visual information
6. Words Puzzles
7. Resume Writing Essentials

Total P: 30 periods

REFERENCES

1. R. S. Agarwal, '*Quantitative Aptitude for Competitive Examination'*. S Chand Publishing, New Delhi, 2017.

SEMESTER IV

25MA402 STATISTICAL METHODS AND STOCHASTIC PROCESSES

3 1 0 4

PROBABILITY AND DISCRETE RANDOM VARIABLES: Probability, axioms, conditional probability, law of total probability, Baye's theorem, discrete random variables, probability mass function, cumulative distribution function, binomial, Poisson and geometric random variables, expected values. **(9+3)**

CONTINUOUS RANDOM VARIABLES: Cumulative distribution function, probability density function, uniform, exponential and Gaussian random variables, expected values. **(9+3)**

PAIRS OF RANDOM VARIABLES: Joint cumulative distribution function, joint probability mass function, marginal probability mass function, joint probability density function, marginal probability density function, independent random variables, expected values, covariance, correlation and independence, central limit theorem, weak law of large numbers. **(9+3)**

STOCHASTIC PROCESSES: Types of stochastic processes, Bernoulli process, Poisson process, discrete time Markov chain, classification of states, limiting state probabilities. **(9+3)**

STATISTICAL INFERENCE: Point estimation, Bayesian estimation, interval estimation. Hypotheses testing: one and two tailed tests, tests concerning mean, proportion, and variance, single and two samples, tests for goodness of fit and independence of attributes. **(9+3)**

Total L: 45 + T: 15 = 60 Periods

TEXT BOOKS

1. Roy D Yates and David J Goodman, '*Probability and Stochastic Processes*'. Wiley India, New Delhi, 2021.
2. Ronald E. W, Raymond H M, Sharon L M and Keying Ye, '*Probability and Statistics for Engineers and Scientists*'. Pearson Education, New Delhi, 2016.

REFERENCES

1. Saeed Ghahramani, '*Fundamentals of Probability with Stochastic Processes*'. CRC Press, USA, 2018.
2. Dougles C Montgomery and George C Runger, '*Applied Statistics and Probability for Engineers*'. Wiley India, New Delhi, 2018.
3. Athanasios P and Unnikrishna P. S, '*Probability, Random Variables and Stochastic Processes'*. Tata McGraw Hill, New Delhi, 2017.
4. D. Bertsekas and J. Tsitsiklis, '*Introduction to Probability*'. Athena Scientific, USA, 2008.

COURSE OUTCOMES

At the end of this course students will be able to:		**Bloom's Level**
CO1	Explain the concepts related to Probability, Random variables, Stochastic processes and Statistical inference.	**K2**
CO2	Apply the techniques of Probability, Random variables, Stochastic processes and Statistical inference to solve engineering problems.	**K3**
CO3	Analyze the solutions of engineering problems employing Probability, Random variables, Stochastic processes and Statistical inference.	**K4**
CO4	Use modern tools to solve engineering problems with the help of Probability, Random variables, Stochastic processes and Statistical inference.	**K4**

COs-POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1													
CO2	3												
CO3		1											
CO4					1								
	3	**1**			**1**								

1-low, 2-medium, 3-high

25CS401 DATABASE MANAGEMENT SYSTEMS

3 0 0 3

DATA MODELS: Databases and database users, Database system concepts and architecture, Data models: Entity relationship model - Enhanced entity relationship model - Relational Model, Relational database constraints, Relational algebra, Conceptual to relational mapping, SQL, DB Vault. **(9)**

DATABASE DESIGN THEORY AND METHODOLOGY: Functional dependencies, Axioms, Normal Forms: First normal form - Second normal form - Third normal form - Boyce Codd normal form, Multi-valued dependency, Join dependency. **(9)**

DATA STORAGE AND INDEXING: Overview, Record storage, Primary file organization, Caching, Index structures for files: Single level Indexing - Multilevel Indexing, B-Trees. **(9)**

SYSTEM IMPLEMENTATION TECHNIQUES: Query processing, Query optimization, Transaction management: Transaction - Concurrency control - Recovery system. **(9)**

NON-RELATIONAL DATABASES: Need for NOSQL Databases, Types, MongoDB - Data types, Creating, Updating and Deleting Documents, Querying the database. **(9)**

Total L: 45 periods

TEXTBOOKS:

1. Ramez Elmasri and Shamkant B Navathe, '*Fundamentals of Database Systems*'. Pearson Education, 7th edition, New Delhi, 2017.
2. Abraham Silberchatz, Henry F Korth and Sudarshan S, '*Database System Concepts*'. Tata McGraw-Hill, 7th edition, New Delhi, 2021.

REFERENCES:

1. Atul Kahate, '*Introduction to Database Management Systems*'. 3rd Edition, Pearson Education, New Delhi, 2011.
2. Raghu Ramakrishnan and Johannes Gehrke, '*Database Management Systems*'. Tata McGraw Hill, 4th edition, New Delhi, 2013.
3. Kristina Chodorow and Michael Dirolf, '*MONGODB: The Definitive Guide, Powerful and Scalable Storage*'. 3rd Edition, Shroff Publishers, Mumbai, 2019.
4. Thomas M Connolly Z Carolyn E, '*Database Systems: A Practical Approach to Design, Implementation, and Management*'. 6th Edition, Pearson Education, New Delhi, 2019.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the basic concepts of database management systems and data storage techniques.	**K2**
CO2	Apply database management system techniques to solve practical application problems.	**K3**
CO3	Analyse database concepts for the given scenario.	**K4**
CO4	Evaluate appropriate SQL queries and data models to optimize the performance and effectiveness of database applications.	**K5**

COs-POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1												2	2
CO2	3											3	3
CO3		2										2	2
CO4				2								2	2
	3	**2**		**2**								**3**	**3**

1-low, 2-medium, 3-high

25CS402 DESIGN AND ANALYSIS OF ALGORITHMS

3 1 0 4

DIVIDE AND CONQUER: Introduction to algorithm design techniques, Divide and conquer methodology: Solving recurrence relations – Finding maximum and minimum element – Quick sort – Merge sort – Convex hull.
Activities:

- Time and space complexity analysis and amortized analysis of divide and conquer algorithms
- Implement merge sort and quick sort methods for array of size N. Experiment for different values of N and compare their time complexity. **(9+3)**

GREEDY METHOD: Greedy strategy – Knapsack problem – Minimum spanning trees – Single source shortest path method – Huffman trees.
Activities:

- Implementation of minimum spanning tree algorithms and comparison of their time complexity
- Implementation of dijkstra's algorithm and its complexity analysis. **(9+3)**

DYNAMIC PROGRAMMING: Principle of optimality – Knapsack problem – All pairs shortest path – Optimal binary search tree – Multistage graphs
Activities:

- Implementation of 0/1 knapsack problem using dynamic programming and analysis of time and space complexity.
- Implementation of all pair shortest path algorithm and derive its complexity. **(9+3)**

BACKTRACKING: State space tree – Knapsack problem – The eight queens problem – Sum of subsets – Graph coloring.
Activities:

- NP complete and NP hard problems
- Implement N queen's problem using backtracking. **(9+3)**

BRANCH AND BOUND: Bounding functions – 0/1 Knapsack problem – Traveling salesman problem – Assignment problem.
Activities:

- Implementation of assignment problem
- Implementation of traveling salesman problem
- Comparison of different approaches for solving knapsack problem. **(9+3)**

Total L: 45 + T: 15 = 60 Periods

TEXT BOOKS

1. Anany Levitin, '*Introduction to the Design and Analysis of Algorithm*'. 3rd, Prentice Hall of India, New Delhi, 2017.
2. Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran, '*Fundamentals of Computer Algorithms*'. Galgotia Publications, New Delhi, 2010.

REFERENCES:

1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, '*Introduction to Algorithms*'. 3rd Edition, MIT Press, England, 2009.
2. Donald E. Knuth, '*The Art of Computer Programming*'. Volumes 1& 3, Pearson Education, 2009.
3. Jeffrey J. McConnell, '*Analysis of Algorithms*'. Jones and Bartlett Publishers, 2008.
4. Parag Himanshu Dave, Himanshu Bhalchandra Dave, '*Design and Analysis of Algorithms*'. Pearson Education, 2008.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the fundamental principles of algorithm design and performance analysis.	**K2**
CO2	Develop algorithmic solutions for computational problems using appropriate design strategies.	**K3**
CO3	Analyze the correctness and computational complexity of algorithms using a suitable approach.	**K4**
CO4	Evaluate alternative algorithmic solutions to determine the most efficient approach for a given problem.	**K5**

COs-POs & PSOs MAPPING

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

1-low, 2-medium, 3-high

25CS403 THEORY OF COMPUTATION

3 1 0 4

FINITE AUTOMATA: Need for automata theory – Finite automaton – DFA and NFA – Finite automaton with ε-moves – Regular languages – Properties – Regular expression – Equivalence of NFA and DFA – Equivalence of NFA 's with and without ε-moves – Equivalence of finite automaton and regular expressions – Minimization of DFA – Pumping lemma for regular sets.

Activities:

- Construct DFA for real-world patterns (e.g., valid binary strings, identifiers in programming languages).
- Practice minimization with multiple examples.
- Online simulator for DFA/NFA: FSM2Regex – convert FSM ↔ Regex. **(9+4)**

GRAMMARS: Types of grammar – Chomsky 's hierarchy of languages – Context free grammars and languages– Properties – Derivations and languages – Ambiguity – Relationship between derivation and derivation trees – Simplification of CFG – Greiback normal form – Chomsky normal form – pumping lemma for CFL.

Activities:

- Construct parse trees for arithmetic expressions and language strings.
- Identify ambiguity in given grammars (e.g., " dangling else" problem).
- Simplify grammars: Practice identifying and removing null, unit, and useless productions.

(8+2)

PUSH DOWN AUTOMATA: Definitions – Moves – Instantaneous descriptions – Deterministic and non-deterministic pushdown automata – Equivalence of pushdown automata and CFL.

Activities:

- Convert CFGs to PDAs interactively in pairs.
- PDA design challenge: Recognize palindromes or balanced parentheses.
- Class demonstration: Simulate acceptance by empty stack and final state. **(8+4)**

TURING MACHINE: Definitions of turing machines – Models – Computable languages and functions – Turing machine construction – Multi head and multi tape turing machines – The halting problem.

Activities:

- Design simple turing machines for unary increment, unary decrement.
- Design TM for binary addition, binary subtraction.
- Write the sequence of instantaneous descriptions (IDs) for a given turing machine and input. **(8+3)**

UNSOLVABLE PROBLEMS AND COMPUTABLE FUNCTIONS: Unsolvable problems and computable functions – Primitive recursive functions – PCP – MPCP – Recursive and recursively enumerable languages – Properties – Universal turing machine – Measuring and classifying complexity – Tractable and intractable problems – Tractable and possibly intractable problems – complexity classes – Polynomial time reductions – P and NP completeness – SAT problem – Clique problem – Space complexity.

Activities:

- Solve small instances of PCP and MPCP manually.
- Solve Tractable vs Intractable Problems real world scenario mapping. **(12+2)**

Total L: 45 + T: 15 = 60 periods

TEXT BOOKS

1. John C Martin, '*Introduction to Languages and the Theory of Computation*'. 4th edition, Tata McGraw Hill Publishing Company, New Delhi, 2011.
2. Hopcroft J. E., Motwani R. and Ullman J.D, '*Introduction to Automata Theory, Languages and Computations*'. 3rd edition, Pearson Education, New Delhi, 2008.

REFERENCES

1. Harry R Lewis and Christos H Papadimitriou, '*Elements of the Theory of Computation*'. 2nd edition, Prentice Hall of India, New Delhi, 2015.
2. Peter Linz, '*An Introduction to Formal Language and Automata*'. 6th edition, Narosa Publishers, New Delhi, 2016.
3. Ganesh Gopalakrishnan, '*Automata and Computability'*. 1st edition, Chapman and Hall/CRC, USA, 2019.
4. D Shanthi, N Uma Maheshwari, S Jeyanthi, '*Theory of Computation*'. 1st edition, Yesdee, India, 2017.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain fundamental concepts and models used to describe computation and language formation and their role in defining computational processes.	**K2**
CO2	Apply theoretical concepts to model, construct and solve problems related to language recognition and syntactic analysis in computing systems.	**K3**
CO3	Analyze computational problems to determine properties, decidability, complexity and limitations of computational approaches.	**K4**
CO4	Evaluate computational problems and solution approaches based on computability, complexity and efficiency.	**K5**

COs-POs & PSOs MAPPING

CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PSO1**	**PSO2**
CO1												3	1
CO2	3											3	3
CO3		3										3	3
CO4								1				1	1
	3	**3**						**1**				**3**	**3**

1-low, 2-medium, 3-high

25CS404 SOFTWARE ENGINEERING

3 1 0 4

PRELIMINARIES: Definition of Software – Software characteristics – Types of Software – Evolution of Software – Software Development Life Cycle – Waterfall Model – Incremental Model – Prototyping – Spiral Model – Unified Process – Agile Development Approach – Software Myths. **(8+2)**
Activities: Case Study on selection of a suitable process model for a given scenario.

REQUIREMENTS ENGINEERING: Requirements elicitation – Functional and Non-functional Requirements– Prioritization – Use cases – Use case diagram — Data flow diagrams – UML — Object Model - Class diagram– State diagram – Sequence diagram – Activity diagram - CRC cards - Software Requirements Specification document (SRS). **(12+4)**
Activities: Develop SRS document, CRC cards & UML Diagrams for the chosen case study.

SOFTWARE DESIGN: Architectural Design: Views and Viewpoints – Styles and Patterns – Layered Architecture. Detailed Design: Functional decomposition – Object oriented design – User interface design – Good design attributes – Design specifications – Coding standards. **(8+3)**
Activities: Develop the UI Design as per specification for the chosen case study.

SOFTWARE TESTING: Testing fundamentals: Black Box and White Box Testing – Test Cases – Equivalence partitioning – Boundary value analysis – Basis path testing – Cyclomatic complexity. Testing Strategies: Unit testing – Integration testing – System testing – Acceptance testing. Special testing: Regression testing – Smoke testing – Stress testing - User Interface testing – Test automation – Test documenting and Reporting – Testing object-oriented system. **(8+3)**
Activities: Test Case Design – Equivalence Partitioning – Boundary Value Analysis– Basis Path Testing – Cyclomatic Complexity - Design Patterns

SOFTWARE QUALITY: Views of quality – Quality attributes – Cost of quality – Quality control vs. Quality assurance – Formal technical reviews – Guidelines for reviews – Quality metrics. Software configuration management: Software baselines – Version control – Change control – Software configuration audit. Software Maintenance: Maintenance phase – Activities – Reengineering – Metrics. **(9+3)**
Activities: Risk Management and Estimations for the chosen case study.

Total L: 45 + T: 15 = 60 periods

TEXT BOOKS:

1. Roger S Pressman and Bruce R Maxim, *'Software Engineering - A Practitioner's Approach'*. 9th edition, McGraw Hill International, Singapore, 2023.
2. Pankaj Jalote, *'A Concise Introduction to Software Engineering'*. 2nd edition, Springer, New Delhi, 2025.

REFERENCES

1. Ian Sommerville, *'Software Engineering'*. Pearson Addison Wesley, Boston, 2018.
2. Shari Lawrence Pfleeger, *'Software Engineering: Theory and Practices'*. Pearson Education, New Delhi, 2009.
3. Orlando Karam, Frank Tsui, *'Essentials of Software Engineering'*. 5th edition, SBS Publishers and Distributors (P) Ltd, New Delhi, 2022.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain how systematic approaches are used to develop dependable computing solutions in professional environments.	**K2**
CO2	Apply appropriate representation methods to communicate system behavior and structure clearly among stakeholders.	**K3**
CO3	Analyze software design, testing strategies, and quality practices to ensure effective software development.	**K4**
CO4	Evaluate the reliability and long-term effectiveness of software-based solutions through structured review and improvement practices.	

COs-POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1												3	3
CO2	3											3	3
CO3		3										3	3
CO4													
	3	**3**										**3**	**3**

1-low, 2-medium, 3-high

25CS411 DATABASE MANAGEMENT SYSTEMS LABORATORY

0 0 4 2

1. Create a database table, add constraints (primary key, unique, check, not null), insert rows, update and delete rows using SQL DDL and DML commands.
2. Execute DCL and TCL commands.
3. Create a set of tables, add foreign key constraints and incorporate referential integrity.
4. Query the database tables using different 'where' clause conditions and also implement aggregate functions.
5. Implement sub queries, simple join operations and explore outer joins.
6. Write user defined functions and stored procedures in SQL.
7. Write SQL Triggers for insert, delete, and update operations in a database table.
8. Create view and index for database tables with a large number of records.
9. Create an XML database and validate it using XML schema.
10. Create document, column and graph-based data using NOSQL database tools.

Total P: 60 periods

REFERENCES:

1. Abraham Silberschatz, Henry F Korth and Sudarshan S, '*Database System Concepts*'. Tata McGraw-Hill, 7th edition, New Delhi, 2021.
2. Ramez Elmasri and Shamkant B Navathe, '*Fundamentals of Database System'*. Addison Wesley, 7th edition, USA, 2016.
3. Ivan Bayross, '*SQL, PL/SQL the Programming Language of Oracle*'. BPB Publications, 4th edition, New Delhi, 2020.
4. Benjamin Rosenzweig and Elena Rakhimov, '*Oracle PL/SQL by Example*'. Pearson Education, 5th edition, India, 2015.
5. Raghu Ramakrishnan and Johannes Gehrke, '*Database Management Systems*'. Tata McGraw-Hill, 4th edition, New Delhi, 2015.

COURSE OUTCOMES:

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Apply database concepts to create and manage databases using SQL and connect them with applications by maintaining data accuracy and consistency.	**K3**
CO2	Analyze XML and NoSQL data models through schema validation and multi-model database tools.	**K4**

COs-POs & PSOs MAPPING

CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PSO1**	**PSO2**
CO1	3											3	3
CO2		2										2	2
	3	**2**										**3**	**3**

1-low, 2-medium, 3-high

25CS412 APPLICATION DEVELOPMENT LABORATORY

0 0 4 2

LIST OF EXPERIMENTS

1. Using react native, build a cross platform application for a BMI calculator.
2. Build a cross platform application for a simple expense manager.
3. Develop a cross platform application to convert units from imperial system to metric system (km to miles, kg to pounds etc.,)
4. Design and develop a cross platform application for day to day task (to-do) management.
5. Design an android application using Cordova for a user login screen with username, password, reset button and a submit button. Also, include header image and a label. Use layout managers.
6. Design and develop an android application using Apache Cordova to find and display the current location of the user.
7. Develop a React Native component using the useState hook to manage a counter value and provide buttons to increment and decrement the displayed count.
8. Write programs using Java to create Android application having Databases
    - For a simple library application.
    - For displaying books available, books lend, book reservation. Assume that student information is available in a database which has been stored in a database server.

Total P: 60 periods

REFERENCES:

1. Dawn Griffiths and David Griffiths, *'Head First Android Development: A Brain-Friendly Guide'*. 2nd edition, O'Reilly Media, 2017
2. Raymond K. Camden, *'Apache Cordova in Action'*. Manning Publications, 2015.
3. Anthony Accomazzo, Houssein Djirdeh, Sophia Shoemaker, and Devin Abbott, *'Full Stack React Native: Create Beautiful Mobile Apps with JavaScript and React Native'*. Fullstack.io, 2017.
4. Carlson, John R, '*Cross-Platform Mobile Application Development: A Beginner's Guide Using the Corona SDK'*. Amazon Digital Services LLC – KDP Print US, 2019.

COURSE OUTCOMES:

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the fundamental concepts, tools and development processes used in application development.	**K2**
CO2	Implement core application features using appropriate development techniques.	**K3**
CO3	Design applications by integrating interface design, program logic, and data management.	

COs-POs & PSOs MAPPING

CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PSO1**	**PSO2**
CO1												2	2
CO2	3											3	3
CO3			2		2							2	2
	3		**2**		**2**							**3**	**3**

1-low, 2-medium, 3-high

25CSE01 MINI PROJECT I

0 0 2 1

- Identification of a real time problem in thrust areas.
- Developing a mathematical model for solving the above problem.
- Finalization of system requirements and specification.
- Implementation of different solutions for the problem based on literature survey Future trends in providing alternate solutions.
- Consolidated report preparation of the above.

25EEC03 PROBLEM SOLVING

0 0 2 1

1. Algorithmic Thinking, Branching & Repetition Problems
2. Logical Reasoning - Data Arrangements & Relations
3. Solving problems based on Coding & decoding, Series, Analogy, Odd man out and Visual reasoning
4. Problems based on Ages, Logical Connectives, Syllogisms, Data Interpretation & Data Sufficiency
5. Solving problems on Clocks Calendars, Direction Sense & Cubes
6. Problems based on Number system, Percentages, Simple & Compound Interest
7. Resume Update

Total P: 30 periods

REFERENCES

1. R. S. Agarwal, '*Quantitative Aptitude for Competitive Examination'*. S Chand Publishing, New Delhi, 2017.

SEMESTER V
25CS501 OPERATING SYSTEMS
(Common to CSE and AI&DS)

3 0 0 3

INTRODUCTION: Functions – History – Operating system concepts – System calls – Services – User operating system interface – Design and implementation – Introduction to virtual machines. **(8)**

PROCESS MANAGEMENT: Process model – Creation – Termination – Hierarchies – States – Implementation – Scheduling criteria – Scheduling algorithms – Multithreading models – Thread libraries – Threading issues – Thread and Multiprocessor scheduling algorithms – Inter–process communication. **(9)**

PROCESS SYNCHRONIZATION AND DEADLOCKS: Race conditions – Critical section – Mutual exclusion – Peterson's solution – Synchronization: hardware – Semaphores – Mutex – Monitor – Message passing, Dining philosopher's problem – Readers writer's problem. Deadlocks: conditions – detection – recovery – prevention – avoidance. **(9)**

MEMORY MANAGEMENT: Main Memory: Swapping – Contiguous memory allocation – Paging – Structure of page table – Segmentation – Examples. Virtual memory: Demand paging – Copy on Write – Page replacement – Allocation of frames – Thrashing – Memory–mapped files – Allocating kernel memory – Memory management utilities. **(10)**

STORAGE MANAGEMENT: Files: Naming – Structure – Types – Access – Attributes – Operations – Implementation. Directories: Operations – Path Names – Hierarchical directory system – Implementation – Allocation methods – Free space management – NFS – Efficiency and Performance – Recovery. Mass storage: Disk structure – Disk scheduling algorithms – Swap space management – Streams. **(9)**

Total L: 45 periods

TEXT BOOKS:

1. Silberschatz A, Galvin P, Gagne G, '*Operating Systems Concepts*'. 10th edition, John Wiley and Sons, Singapore, 2023.
2. William Stallings, '*Operating Systems: Internals and Design Principles*'. 9th edition, Pearson Education, New Delhi, 2019.

REFERENCES:

1. Andrew S. Tanenbaum, *'Modern Operating System'*. 5th edition, PHI Learning, New Delhi, 2023.
2. Dhamdhere, *'Operating Systems: A Concept Based approach'*. 3rd edition, Tata McGraw Hill, New Delhi, 2017.
3. Harvey M Deitel, Paul J Deitel, David R Choffnes, *'Operating Systems'*. 3rd edition, Pearson Education, New Delhi, 2013.
4. Remzi Arpaci–Dusseau, Andrea Arpaci–Dusseau, *'Operating Systems: Three Easy Pieces'*. 1st edition, Arpaci–Dusseau Books, 2018
5. Achyut S Godbole, Atul Kahate, *'Operating Systems'*. 3rd edition, McGraw Hill Education, 2016.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the basic concepts of operating system structures, services and functionalities.	**K2**
CO2	Apply CPU scheduling, process synchronization, deadlock handling, memory management, disk and file system management to support OS functionalities.	**K3**
CO3	Analyze various operating system services to assess their efficiency in improving system performance.	**K4**
CO4	Evaluate CPU scheduling algorithms, process synchronization mechanisms, deadlock handling strategies, memory management techniques, and storage management methods to assess operating system performance.	**–**

COs–POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1													
CO2	3											3	3
CO3		3										3	3
CO4				1								1	1
	3	**3**		**1**								**3**	**3**

1–low, 2–medium, 3–high

25CS502 COMPILER DESIGN

3 0 0 3

INTRODUCTION TO COMPILERS: Translators – Compilation and interpretation – The phases of compiler – Errors encountered in different phases – The grouping of phases – Compiler construction tools – JIT. **(7)**

LEXICAL ANALYSIS: Need and role of lexical analyzer – Input buffering – Lexical errors – Expressing tokens by regular expression – Finite automata: NFA– DFA – Converting NFA to DFA – Minimization of DFA – Converting regular expression to DFA. LEX Tool: Structure of LEX program – Predefined variables – Library routines – Design of lexical analyzer for a sample language. **(9)**

SYNTAX ANALYSIS: Need and role of the parser – Context free grammars – Top down parsing: Recursive descent parser – Predictive parser. Bottom up Parsers: Shift reduce parser – LR parser – LR (0) item – Construction of SLR parsing table – CLR parser – LALR parser. Error handling and recovery in syntax analyzer – YACC tool: Structure of YACC program – Communication between LEX and YACC – Design of a syntax analyzer for a sample language. **(10)**

INTERMEDIATE CODE GENERATION: Benefits– Intermediate languages – Generation of three address code – Declarations – Assignment statements – Arrays – Boolean expressions – Backpatching – Flow of control statements – Procedure calls. **(9)**

RUN–TIME ENVIRONMENT, CODE OPTIMIZATION AND GENERATION: Source language issues – Storage organization – Storage allocation – Symbol tables. Principal sources of optimization – Optimization of basic blocks – Global optimization – Global data flow analysis – Issues in design of a code generator – A simple code generator algorithm. **(10)**

TOTAL L: 45 periods

TEXT BOOKS:

1. Alfred V. Aho, Monica S. Lam, Ravi Sethi and Jeffrey D. Ullman, '*Compilers: Principles, Techniques and Tools'*. 2nd edition, Pearson Education, 2007.
2. Keith D. Cooper and Linda Torczon, *'Engineering a Compiler'*. 3rd edition, Morgan Kaufmann Publishers, Elsevier, 2022.

REFERENCES:

1. Torben Ægidius Mogensen, *'Introduction to Compiler Design'*. 2nd edition, Springer International Publishing, 2017.
2. Dick Grune, Kees van Reeuwijk, Henri E. Bal, Ceriel J.H. Jacobs and Koen Langendoen, *'Modern Compiler Design'*. 2nd edition, Springer, 2012.
3. Michael L. Scott, '*Programming Language Pragmatics'*. 4th edition, Morgan Kaufmann Publishers, 2015
4. Andrew W. Appel and Jens Palsberg, '*Modern Compiler Implementation in C'*. 2nd edition, Cambridge University Press, 2018.
5. Steven S. Muchnick, '*Advanced Compiler Design and Implementation'*. Morgan Kaufmann Publishers, Elsevier, Reprint edition, 2017.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Interpret the phases of the compiler, Lexical Analyzer, Code optimization and Code Generation.	**K2**
CO2	Construct Finite Automata for the Regular Expression with LEX implementation.	**K3**
CO3	Interpret different types of parsing techniques using YACC tool.	**K4**
CO4	Generate three address code for the statements in high level languages.	**–**

COs–POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1												3	3
CO2	3											3	3
CO3		3										3	3
CO4			2									2	2
	3	**3**	2									**3**	**3**

1–low, 2–medium, 3–high

25CS503 COMPUTER NETWORKS

3 0 0 3

INTRODUCTION AND MEDIUM: Building a network – Network edge and core – Layering and protocols – TCP/IP Protocol suit – OSI reference model – Network topologies – Internet architecture – Physical layer – Signal characteristics – Transmission media – Signal encoding techniques – Performance metrics. **(10)**

LINK LAYER SERVICES: Link layer services – Framing – Flow control – Error control – Media access control – Ethernet – Wireless LAN – Introduction about Bluetooth – Zigbee. **(8)**

SWITCHING AND ROUTING: Switching – Circuit switching – Packet switching – IPV4 – Global address – Datagram forwarding – Subnetting – CIDR – ARP – ICMP – Routing algorithms – Distance vector routing and link state routing – IPV6 addressing – IPV6 protocol. **(9)**

CONNECTION–ORIENTED AND CONNECTIONLESS SERVICES: Overview of transport layer – UDP –TCP – Reliable byte stream – Connection management – Flow control – Congestion control – SCTP. **(9)**

APPLICATION LAYER SERVICES: Needs / Principles of application layer protocols – Role of proxy – Web and HTTP - HTTPS – FTP – Electronic Mail (SMTP – POP3 – IMAP – MIME) – DHCP – DNS – DASH – QUIC - VPN. **(9)**

Total L: 45 periods

TEXTBOOKS:

1. James F Kurose, Keith W Ross, *'Computer Networking – A Top–Down Approach Featuring the Internet'*. 9th edition, Pearson Education, India, 2025.
2. Larry L Peterson and Bruce S Davie, *'Computer Networks: A systems approach'*. 6th edition, Morgan Kaufmann Publishers Burlington, USA, 2021.

REFERENCES:

1. Behrouz A Forouzan, '*Data Communications and Networking with TCP/IP Protocol Suite*'. 6th edition, McGraw Hill, 2021.
2. Andrew S Tanenbaum and David J Wetherall, '*Computer Networks'*. 6th edition, Prentice Hall of India/ Pearson Education, New Delhi, 2021
3. Prakash C Gupta, '*Data Communication and Computer Networks*'. Prentice Hall of India, New Delhi, 2014.
4. Ajit Pal, '*Data Communication and Computer Networks'*. 1st Edition, PHI Learning, 2013.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the fundamental principles of computer network architectures, communication models, and protocols.	**K2**
CO2	Apply appropriate techniques to address issues related to reliability, efficiency, and security in networks.	**K3**
CO3	Analyze network behaviour and performance under different operational conditions.	**K4**
CO4	Present effective solutions to complex networking problems based on given requirements.	**–**

COs–POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1												3	3
CO2	3											3	3
CO3		3										3	3
CO4			1									1	1
	3	**3**	**1**									**3**	**3**

1–low, 2–medium, 3–high

25CS504 ARTIFICIAL INTELLIGENCE

3 1 0 4

INTRODUCTION: Artificial intelligence – The state of art – Intelligent agents – Rationality – Nature of environments – Structure of agents – Examples.

Activities:

- Analyze Agent type, Environment type, Rational behaviour, Possible limitations of intelligent agent. **(9+3)**

PROBLEM SOLVING AGENTS: Searching for solutions: Uninformed search – BFS, DFS, Uniform cost search, Iterative deepening search – Informed search – Greedy best first search, A* search, AO* search – Adversarial search – Games – Optimal decisions in games – Alpha–beta pruning

Activities:

- Compare algorithms based on Time complexity, Space complexity, Optimality and Completeness, Apply Min Max Algorithm, Perform alpha–beta pruning. **(9+3)**

KNOWLEDGE AND REASONING: Representations and mappings – Approaches to knowledge representation – Property inheritance algorithm – First order predicate logic – Instance and ISA relationships – Computable functions and predicates – Unification – Resolution – Question answering.

Activities:

- FOPL, Unification and Resolution. **(9+3)**

ACTING UNDER UNCERTAINTY: Quantifying uncertainty – Efficient representation of conditional distributions – Probability and Bayes Theorem – Bayesian Networks – Exact and approximate inferences – Making simple decisions – Utility theory – Decision networks – Value of information – Decisions with multiple agents

Activities:

- Bayes' Theorem Application, Conditional Probability Table, Decision Theory & Utility. **(9+3)**

PLANNING: Classical Planning– Algorithms for planning as state space search – Planning graphs – Classic planning as boolean satisfiability – Analysis of planning approaches – Time, schedules and resources – Hierarchical planning

Activities:

- State Space Search for Planning, Planning Graphs, SAT, Analysis of Planning Approaches. **(9+3)**

Total L: 45+T: 15 = 60 periods

TEXTBOOKS:

1. Stuart J Russell and Peter Norvig, '*Artificial Intelligence – A Modern Approach'*. 3rd edition, Prentice Hall of India, Pearson Education, New Delhi, 2021.
2. Elaine Rich, Kevin Knight and Shivashankar B Nair, '*Artificial Intelligence'*. 3rd edition, Tata McGraw Hill Publishing Company, New Delhi, 2019.

REFERENCES:

1. George F Luger, '*Artificial Intelligence: Structures and Strategies for Complex Problem Solving'*. 6th edition, Pearson Education, New Delhi, 2021.
2. Deepak Khemani, '*A First Course in Artificial Intelligence'*. McGraw Hill Education, New Delhi, 2017.
3. John Paul Mueller, '*Artificial Intelligence for Dummies'*. Wiley, 2018.
4. LavikaGoel, '*Artificial Intelligence: Concepts and Applications'*. Wiley, 2021.
5. Pradeepta Mishra, '*Practical Explainable AI Using Python: Artificial Intelligence Model Explanations Using Python–based Libraries, Extensions and Frameworks'*. Apress, 2022.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the basic concepts of AI, intelligent agents and environments.	**K2**
CO2	Apply search algorithms to solve problems and games.	**K3**
CO3	Analyze knowledge representation and reasoning methods.	**K4**
CO4	Evaluate decision–making under uncertainty and planning techniques.	–

COs–POs & PSOs MAPPING

CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PSO1**	**PSO2**
CO1													
CO2	3											3	3
CO3		3										3	3
CO4			3									2	2
	3	**3**	**3**									**3**	**3**

1–low, 2–medium, 3–high

25CS511 OPERATING SYSTEMS LABORATORY
(Common to CSE and AI&DS)

0 0 2 1

1. Linux commands
2. Shell programming
3. Programs on system calls
4. Process management: creation, synchronization
5. Inter–process communication
6. CPU, GPU scheduling algorithms
7. Deadlock avoidance algorithm
8. Memory Allocation Methods like first fit, best fit and worst fit
9. Page Replacement Algorithms
10. Disk scheduling algorithms

Total P: 30 periods

REFERENCES:

1. Silberschatz A, Galvin P, Gagne G, '*Operating Systems Concepts'*. 10th edition, John Wiley and Sons, Singapore, 2023.
2. William Stallings, '*Operating Systems: Internals and Design Principles'*. 9th edition, Pearson Education, New Delhi, 2019.

COURSE OUTCOMES:

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Apply operating system concepts through practical implementation and experimentation in a programming environment	**K3**
CO2	Analyze the performance and behaviour of operating system mechanisms through experimental results	**K4**

COs–POs & PSOs MAPPING

CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PSO1**	**PSO2**
CO1	3											3	3
CO2		3										3	3
	3	**3**										**3**	**3**

1–low, 2–medium, 3–high

25CS512 COMPUTER NETWORKS LABORATORY

0 0 4 2

1. Basic Network Commands and Network Configuration Commands.
2. Router Configuration and LAN Setup using Cisco Packet Tracer.
3. Capture live network traffic in Wireshark and identify OSI/TCP-IP layers.
4. Capture and analyse Ethernet frames in Wireshark to identify MAC addresses, frame structure and framing process.
5. Build a switched star topology in Packet Tracer and observe MAC-based communication between devices.
6. Capture and analyse ARP and ICMP packets in Wireshark to examine address resolution and ping behaviour.
7. Implement Distance Vector Routing (RIP) and Link State Routing (OSPF) in Packet Tracer and compare routing behaviour.
8. Develop a TCP client-server socket program for reliable message exchange.
9. Capture TCP packets in Wireshark and analyse three-way handshake, flow control, and connection termination.
10. Configuration of DNS, HTTP and SMTP Services in Cisco Packet Tracer.

Total P: 60 periods

REFERENCES:

1. James F Kurose, Keith W Ross, '*Computer Networking – A Top–Down Approach Featuring the Internet'*. 9th edition, Pearson Education, India, 2025.
2. Larry L Peterson and Bruce S Davie, '*Computer Networks: A systems approach'*. 6th edition, Morgan Kaufmann Publishers, USA, 2021.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Apply networking concepts through practical experiments using standard tools and techniques.	**K3**
CO2	Analyze network behaviour and protocol operations using appropriate network analysis tools.	**K4**

COs–POs & PSOs MAPPING

CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PSO1**	**PSO2**
CO1	3											3	3
CO2		3										3	3
	3	**3**										**3**	**3**

1–low, 2–medium, 3–high

25CSE02 INTERNSHIP I / 25CSE03 COMMUNITY PROJECT

0 0 0 1

This course provides students with experiential learning through an Internship I or Community Project, enabling them to identify real-world industry or societal problems and develop innovative solutions. The internship I / community project shall be carried out during the Fourth Semester vacation. At the end of the training, students shall submit a report and present their work for evaluation through a viva-voce conducted by internal examiners.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Execute an internship or community-based project by applying project management, techno-economic analysis, sustainability, ethical practices, lifelong learning, and effectively communicate the outcomes through professional reports and presentations	-

COs–POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1						3	3	3	3	3	3	3	3
						3	**3**	**3**	**3**	**3**	**3**	**3**	**3**

1–low, 2–medium, 3–high

25EEC04 APTITUDE SKILLS

0 0 2 1

1. Algorithmic Thinking, Branching & Repetition Problems
2. Logical Reasoning - Data Arrangements & Relations
3. Solving problems based on Coding & decoding, Series, Analogy, Odd man out and Visual reasoning
4. Problems based on Ages, Logical Connectives, Syllogisms, Data Interpretation & Data Sufficiency
5. Solving problems on Clocks Calendars, Direction Sense & Cubes
6. Problems based on Number system, Percentages, Simple & Compound Interest
7. Resume Update

Total P: 30 Periods

REFERENCES:

1. R. S. Aggarwal, '*Quantitative Aptitude for Competitive Examination*'. S Chand Publishing, New Delhi, 2017.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Solve problems related to vocabulary, critical reasoning, probability and prepare resumes.	-

COs–POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	1										1		
	1										**1**		

1–low, 2–medium, 3–high

SEMESTER VI

25MA602 GRAPH THEORY

3 1 0 4

INTRODUCTION: Review on Definition and Basic Terminologies of Graphs – Representations of Graphs – Walks in Graphs and Digraphs- Subgraphs-Vertex Degrees - Path and Cycles - Regular and Bipartite Graphs-Representations of graphs – adjacency and incidence lists – adjacency and incidence matrices -Graph Traversals-Applications: Four Cubes Problem- Social Networks. **(9+3)**

EULERIAN AND HAMILTONIAN GRAPHS: Exploring and Travelling – Eulerian Graphs – Konigsberg bridge problem - Hamiltonian Graphs – Applications: Dominoes – Chinese Postman Problem - Travelling salesman problem**.** **(8+3)**

PATHS AND CONNECTIVITY: Connected Graphs and Digraphs-Menger's Theorem for Graphs-Applications: Reliable Telecommunication Networks. Network flows and applications- Flows and cuts in Networks, Maximum-flow problem, flows and connectivity– applications. **(10+3)**

VERTEX-COLORING: Vertex-coloring - chromatic number of a graph, vertex coloring algorithms – sequential vertex coloring, largest degree first algorithm, applications - scheduling problem, fast register allocation for computer programming. **(10+3)**

MATCHING AND FACTORS: Matching, Perfect matching, Tutte's 1-factor theorem, weighted Bipartite matching, Hall's theorem. **(8+3)**

TOTAL L: 45 + T: 15 = 60 periods

TEXTBOOKS:

1. Jonathan L. Gross and Jay Yellen, *'Graph Theory and its Applications'*. CRC Press, New York, 2016.
2. Douglas B West, *'Graph Theory'*. Prentice Hall, New Delhi, 2017.

REFERENCES:

1. Bondy J.A. and Murty U.S.R., '*Graph Theory*'. Springer, London, 2016.
2. Narsingh Deo, '*Graph Theory with Applications to Engineering and Computer Science*'. Prentice Hall, New Delhi 2017.
3. Joan M Aldous and Robin J Wilson, '*Graphs and Applications: An Introductory Approach, Springer-Verlag*'. New York, 2014.
4. Reinhard Diestel, '*Graph Theory*'. Springer-Verlag, Berlin Heidelberg, 2012.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the concepts related to Basic Terminologies of Graphs, Eulerian Graphs - Hamiltonian Graphs, Vertex-coloring and Matching.	**K2**
CO2	Apply the techniques of Basic Terminologies of Graphs, Eulerian Graphs - Hamiltonian Graphs, Vertex-coloring and Matching to solve engineering problems.	**K3**
CO3	Analyze the solutions of engineering problems employing Basic Terminologies of Graphs, Eulerian Graphs - Hamiltonian Graphs, Vertex-coloring and Matching.	**K4**
CO4	Use modern tools to solve engineering problems with the help of Basic Terminologies of Graphs, Eulerian Graphs - Hamiltonian Graphs, Vertex-coloring and Matching.	**–**

COs–POs & PSOs MAPPING

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

1–low, 2–medium, 3–high

25CS601 MACHINE LEARNING

3 0 0 3

INTRODUCTION: Designing a learning system – concept learning – Find-s algorithm – Candidate elimination – Polynomial curve fitting – Bayesian probabilities – The gaussian distribution – Curve fitting with probabilistic perspective – Bayesian curve fitting. **(9)**

LINEAR AND PROBABILISTIC MODELS: Linear regression models – Maximum likelihood estimation – Least squares – Bias variance decomposition – Linear models for classification – Linear discriminant analysis – Probabilistic generative models – Probabilistic discriminative models. **(9)**

NEURAL NETWORKS AND DECISION TREES: Feed–forward networks – Network training – Delta rule– Gradient descent – Error back propagation – Regularization in neural networks – Decision tree learning– Representation – Inductive bias– Issues. **(9)**

KERNEL AND GRAPHICAL METHODS: Constructing kernels – Radial basis function networks – Maximum margin classifiers – SVM – Bayes theorem – Naive bayes – Bayesian networks. **(9)**

UNSUPERVISED AND REINFORCEMENT LEARNING: Measures of similarity and dissimilarity – Clustering – Partitioning methods – K-Means – Hierarchical methods – Reinforcement learning – Reinforcement learning tasks – Q-learning. **(9)**

Total L: 45 periods

TEXTBOOKS:

1. Tom Mitchell, '*Machine Learning'*. McGraw Hill, 2017.
2. Christopher M Bishop, '*Pattern Recognition and Machine Learning*'. Springer, 2018.

REFERENCES:

1. Ethem Alpaydin, '*Introduction to Machine Learning'*. 3rd edition, PHI Learning, 2020.
2. Trevor Hastie, Robert Tibshirani and Jerome Friedman, '*The Elements of Statistical Learning'*. 2nd edition, Springer, 2017.
3. Kevin Murphy, '*Machine Learning – A Probabilistic Perspective'*. MIT Press, 2012.
4. Yaser S. Abu–Mostafa, '*Learning from Data*'. AML, 2017.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the fundamental concepts, principles, and techniques of machine learning.	**K2**
CO2	Apply machine learning algorithms across various learning paradigms to solve computational problems.	**K3**
CO3	Analyze learning models and algorithms to evaluate their efficiency and relevance for solving learning problems.	**K4**
CO4	Evaluate machine learning techniques across learning paradigms to select suitable models for learning problems.	**–**

COs–POs & PSOs MAPPING

CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PSO1**	**PSO2**
CO1												3	3
CO2	3											3	3
CO3		3										3	3
CO4			2					2				2	2
	3	**3**	**2**					**2**				**3**	**3**

1–low, 2–medium, 3–high

25CS602 EMBEDDED SYSTEMS

2 2 0 4

INTRODUCTION: Fundamental Components of Embedded Systems – Architecture of Embedded systems– Embedded design life cycle – Development environment

Activities:

- Study and identification of fundamental components of an embedded system using development boards.
- Setup and exploration of embedded development environment
- Demonstration of embedded design life cycle using a sample project. **(5+5)**

MEMORY AND INTERRUPTS: Types of memory – Memory access procedure – Memory management techniques – Memory testing – Common memory problems – Interrupts – Interrupt Service Routines.

Activities:

- Implementation of basic memory management techniques.
- Sensor data logging and storage management using development boards.
- External interrupt handling using push button event detection. **(6+6)**

COMMUNICATION INTERFACES AND PROTOCOLS: Interfacing buses – Serial interfaces – RS232/UART – UART Programming – RS422/RS485 – I2C Interface – SPI Interface – I2C/SPI Programming – USB.

Activities:

- Embedded Networking
- Wireless communication using WiFi/Bluetooth module for device control. **(7+7)**

REAL TIME OPERATING SYSTEMS: Real–Time concepts – Task management – Task scheduling – Classification of scheduling algorithms – Clock driven scheduling – Event driven scheduling – Resource sharing – Priority inheritance protocol – Priority Ceiling Protocol – Commercial RTOS.

Activities:

- Timer–based task scheduling.
- Periodic sensor data acquisition.
- Multitasking and resource sharing using multiple sensors. **(7+7)**

HOST and TARGET TESTING: Validation – Host and Target testing – Debugging tool – Case Study – Development of automatic vending machine – Protocol converter – Development of a navigation.

Activities:

- Material Selection and Safety Regulations for Embedded Systems
- Embedded Systems: Indian and Foreign Compliance
- Development and testing of Embedded/IoT–based systems. **(5+5)**

TOTAL L: 30 + T: 30 = 60 periods

TEXTBOOKS:

1. Arnold S Berger, '*Embedded Systems Design – An Introduction to Processes, Tools and Techniques*'. Elsevier, New Delhi, 2011.
2. Prasad K V K K, *'Embedded/Real–Time Systems: Concepts, Design and Programming – The Ultimate Reference*'. Himal Impressions, New Delhi, 2003.

REFERENCES:

1. Raj Kamal, *'Embedded Systems – Architecture, Programming and Design*'. 3rd edition, Tata McGraw Hill, 2017.
2. Rajib Mall, '*Real–Time Systems: Theory and Practice'*. 1st edition, Pearson, 2009.
3. Sriram V Iyer and Pankaj Gupta, '*Embedded Real–time Systems Programming'*. Tata McGraw–Hill Publishing Company Limited, New Delhi, 2006.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the fundamental concepts of embedded system architecture, memory management, communication interfaces, real–time operating systems, and testing methodologies.	**K2**
CO2	Apply embedded system design principles, memory and interrupt handling, communication protocols, RTOS concepts, and testing techniques to solve engineering problems.	**K3**
CO3	Analyze embedded system performance using architectural design, communication interfaces, scheduling algorithms, and validation methods.	**K4**
CO4	Implement embedded system projects with proper testing, validation, and technical report preparation.	–

COs–POs & PSOs MAPPING

CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PSO1**	**PSO2**
CO1												2	2
CO2	3											3	2
CO3		2										2	2
CO4								2	2			2	2
	3	**2**						**2**	**2**			**3**	**2**

1–low, 2–medium, 3–high

25CS611 MACHINE LEARNING LABORATORY

0 0 4 2

1. Data preparation for machine learning
2. Candidate elimination algorithm
3. Implement linear regression for data fitting
4. Implement perceptron for binary classification
5. Implement K-means algorithm
6. Comparing regression models (OLS, ridge, lasso) for a real–world dataset
7. Comparing classification algorithms (decision tree, SVM, Naïve bayes etc.,) for a real–world dataset
8. Comparing hierarchical clustering methods for a real–world dataset
9. Reinforcement learning – Q Learning
10. Mini project

Total P: 60 periods

REFERENCES:

1. Ethem Alpaydin, '*Introduction to Machine Learning'*. 4th edition, PHI Learning, 2020.
2. Willi Richert, Luis Pedro Coelho, '*Building Machine Learning Systems with Python'*. Packt Publishing, 2018.
3. Aurelien Geron, '*Hands–On Machine Learning with Scikit–Learn, Keras and Tensor Flow: Concepts, Tools and Techniques to Build Intelligent Systems'*. 3rd edition, O'Reilly, 2022.
4. Tom Mitchell, '*Machine Learning'*. McGraw Hill, 2017.
5. Richard O. Duda, Peter E. Hard, David G. Stork, '*Pattern Recognition'*. 2nd edition, Wiley, 2020.

COURSE OUTCOMES:

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Apply various machine learning methods to implement, compare, and analyze models on real–world datasets.	**K3**
CO2	Analyze different machine learning models by evaluating their effectiveness across different datasets.	**K4**

COs–POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3											3	3
CO2		3										3	3
	3	**3**										**3**	**3**

1–low, 2–medium, 3–high

25CSE04 MINI PROJECT II

0 0 2 1

This course enables students to apply engineering knowledge to solve real-world problems through project-based learning. Students identify a problem in an emerging area, conduct a literature survey, develop and implement an appropriate solution and evaluate its effectiveness. The course concludes with the submission of a technical report and presentation of the project outcomes

Total P: 30 periods

COURSE OUTCOMES:

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Analyze the literature to identify a research problem and formulate appropriate objectives for solution development.	-
CO2	Design and develop an engineering solution by applying principles of mathematics, computing, engineering and appropriate analytical tools.	-
CO3	Develop and communicate project outcomes through technical reports and presentations by integrating project management, techno-economic feasibility, sustainability, engineering ethics and lifelong learning.	-

COs–POs & PSOs MAPPING

CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PSO1**	**PSO2**
CO1		3									3	3	3
CO2	3		3	3	3							3	3
CO3						3	3	3	3	3		3	3
	3	**3**	**3**	**3**	**3**	**3**	**3**	**3**	**3**	**3**	**3**	**3**	**3**

1–low, 2–medium, 3–high

25EEC06 ENHANCING ARITHMETIC PROBLEM SOLVING

0 0 2 1

1. Logarithms, Progressions, Geometry and Quadratic equations
2. Attention to detail Image Interpretation
3. Voices and Speech
4. Algebraic Expressions
5. Surds, Indices and Simplification
6. Set Theory & Functions
7. Cryptarithmetic & Trigonometry
8. Advance Quantitative
9. Advance Verbal
10. Advance Logical
11. Preparing for Business roles, Case studies, Guesstimation techniques
12. Handling Stress
13. Handling Peer pressure
14. Resume progress check

Total P: 30 Periods

REFERENCES:

1. R. S. Aggarwal, '*Quantitative Aptitude for Competitive Examination'*. S Chand Publishing, New Delhi, 2017.
2. P. C. Wren and H. Martin, '*High school English grammar & composition'*. S Chand Publishing, New Delhi, 2017.
3. Norman Lewis, *'Word Power Made Easy'*. Goyal Publisher, New Delhi, 2011.
4. Arun Sharma, '*How to Prepare for Quantitative Aptitude for the CAT'*. 8th edition, McGraw Hill Education, 2018.

COURSE OUTCOMES:

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Apply quantitative aptitude techniques to solve placement aptitude problems accurately.	-

COs–POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1		2											
		2											

1–low, 2–medium, 3–high

SEMESTER VII

25CS701 PARALLEL AND DISTRIBUTED SYSTEMS

3 0 0 3

INTRODUCTION TO PARALLEL COMPUTING: Flynn's Taxonomy – Types of parallel computers – Characteristics of parallel systems – Parallel computing models: Shared memory models – Interconnection network models – Dataflow model – Parallel algorithms: Classes of problems solvable through parallelization – Types of parallelization–Assigning computational tasks to processors – Map Reduce – Spark. **(9)**

PARALLEL COMPUTING ARCHITECTURE: Introduction – Shared memory architecture: Shared memory paradigm – Message–passing architecture: Message passing paradigm – Routing – Switching – Stream processing architecture: Dataflow paradigm – Parallel accelerators – Stream processors – Kafka. **(9)**

PARALLEL PROGRAMMING WITH OpenMP and MPI: Overview of features – Additional feature details: OpenMP directives – Synchronization – Runtime library routines – MPI: Introduction to MPI – Basic point–to–point communication routines – Basic MPI collective communication routines–Environment management routines – Point–to–point communication routines – Collective communication routines. **(9)**

DISTRIBUTED COMPUTING: Definition, Motivation, Coordination – Clocks – Mutual exclusion – Deadlock – Leader election, Replication management – Consistency protocols. **(9)**

GPU COMPUTING: CPU vs GPU, Architecture of CUDA capable GPU – GPU computing – Data parallelism – CUDA program structure – Vector addition – CUDA thread organisation – Matrix–matrix multiplication – Blocks – Open CL data parallel model, Device architecture. **(9)**

Total L: 45 periods

TEXTBOOKS:

1. Vivek Kale, '*Parallel Computing Architectures and APIs: IoT, big data stream processing'*. CRC Press, 2020.
2. Andrew S. Tanenbaum and Maarten van Steen, *'Distributed Systems, Principles and Paradigm'.* Prentice Hall, 2013.

REFERENCES:

1. Ananth Grama, Anshul Gupta, George Karypis and Vipin Kumar, *'Introduction to Parallel Computing'*. 2nd edition, Pearson Education, 2009.
2. David Kirk, Wen Mei W Hwu, '*Programming Massively Parallel Processors*: *A Hands– on Approach*'. Morgan Kaufmann Publishers Inc, 3rd edition, 2016.
3. Michael J Quinn, *'Parallel Computing: Theory and Practice'*. Tata McGraw–Hill, 2004.
4. Shane Cook, '*CUDA Programming: A Developer's Guide to Parallel Computing with GPU's (Applications of GPU Computing)'.* 1st edition, Morgan Kaufmann, 2012.
5. Wan Fokkink, *'Distributed Algorithms: An Intuitive Approach'*. MIT Press, 2013.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain fundamental concepts and classifications of high–performance computing systems.	**K2**
CO2	Implement basic programs using suitable parallel programming techniques.	**K3**
CO3	Examine system architectures and coordination mechanisms for efficient execution.	**K4**
CO4	Evaluate computing approaches for solving complex problems.	**–**

COs–POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1												3	3
CO2	3											3	3
CO3		3										3	3
CO4			2									2	2
	3	**3**	**2**									**3**	**3**

1–low, 2–medium, 3–high

25CS702 CRYPTOGRAPHY AND NETWORK SECURITY

3 1 0 4

COMPUTER SECURITY CONCEPTS: The OSI security architecture – Security attacks – Security services – Security mechanisms – A model for network security – Number theory concepts: Fermat's and Euler's theorems, Euclidean algorithm – Classical encryption techniques.

Activities:

- Predict the plain text from the given cipher text and identify the technique used
- Play fair cipher technique
- Vigenere cipher
- Vernam cipher
- Rail fence technique **(9+3)**

SYMMETRIC CIPHERS: Block ciphers and stream ciphers – Random bit generation and stream ciphers: Principles of pseudorandom number generation – Pseudorandom number generators: Block cipher modes – Data encryption standard.

Activities:

- Linear congruential generator
- ECB mode encryption **(9+3)**

PUBLIC–KEY CRYPTOGRAPHY: Principles of public key cryptosystems – The RSA algorithm – Diffie–Hellman key exchange – Elliptic curve cryptography.

Activities:

- RSA algorithm **(9+3)**

CRYPTOGRAPHIC HASH FUNCTIONS: Secure Hash Algorithm (SHA) – Message authentication codes – Message authentication requirements – Message authentication functions – Digital signatures – Digital Signature Standard (DSS).

Activities:

- Digital signature
- SHA-256 hashing **(9+3)**

ROLE OF CRYPTOGRAPHY IN SECURITY PROTOCOLS: Network and internet security protocols: Transport level security – Secure Sockets Layer (SSL) – Email Security: Pretty Good Privacy (PGP) – Firewalls: characteristics and types.

- Firewall
- Ciphertext using XOR **(9+3)**

TOTAL L: 45 + T: 15 = 60 Periods

TEXT BOOKS:

1. Hans Delfs, Helmut Knebl, '*Introduction to Cryptography Principles and Applications'*. 3rd edition, Springer–Verlag, Berlin Heidelberg, 2015.
2. William Stallings, '*Cryptography and Network Security: Principles and Practice'*. 7th edition, Prentice Hall of India, Pearson Education, New Delhi, 2017

REFERENCES:

1. Behrouz A Forouzan, '*Cryptography and Network Security'*. 3rd edition, Tata McGraw Hill Ltd, New Delhi, 2015.
2. Atul Kahate, '*Cryptography and Network Security'*. 3rd edition, Tata McGraw Hill Ltd, New Delhi, 2013.
3. Douglas Robert Stinson, Maura Paterson, '*Cryptography: Theory and Practice'*. 4th edition, Chapman and Hall/CRC, 2018.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Interpret cryptographic techniques to ensure data confidentiality, integrity and authentication.	**K2**
CO2	Apply secure communication protocols to protect sensitive data transmission over public networks.	**K3**
CO3	Compare various symmetric and asymmetric cryptosystems and types of attacks.	**K4**
CO4	Examine complex network security challenges and potential vulnerabilities.	**–**

COs–POs & PSOs MAPPING

CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PSO1**	**PSO2**
CO1												3	3
CO2	3											3	3
CO3		3										3	3
CO4			2									2	2
	3	**3**	**2**									**3**	**3**

1–low, 2–medium, 3–high

25CS711 PARALLEL AND DISTRIBUTED SYSTEMS LABORATORY

0 0 4 2

1. Program using multithreading
2. Map–Reduce program with Spark
3. Stream processing using Kafka
4. Simple application using MPI broadcasting (Matrix Multiplication, Pi calculation)
5. Point – to – Point application in MPI (Matrix Multiplication, Pi calculation)
6. Matrix operations / PI calculation using OpenMP
7. Implementation of election algorithms
8. Implementation of distributed deadlock detection algorithms
9. Exploration and Matrix operations of CUDA
10. Simulate distributed debugging, failure detection and leader election in distributed systems.

Total P: 60 periods

COURSE OUTCOMES:

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Apply parallel and distributed computing techniques to develop solutions for computational problems.	**K3**
CO2	Analyze the performance of computing solutions across different architectures and application domains.	**K4**

COs–POs & PSOs MAPPING

CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PSO1**	**PSO2**
CO1	3											3	3
CO2		2										2	2
	3	**2**										**3**	**3**

1–low, 2–medium, 3–high

25CSE05 PROJECT WORK I

0 0 4 2

This course enables students to apply engineering knowledge to solve real-world problems through project-based learning. Students identify a problem in an emerging area, conduct a literature survey, develop and implement an appropriate solution and evaluate its effectiveness. The course concludes with the submission of a technical report and presentation of the project outcomes.

Total P: 60 periods

COURSE OUTCOMES:

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Analyze the literature to identify a research problem and formulate appropriate objectives for solution development.	-
CO2	Design and develop an engineering solution by applying principles of mathematics, computing, engineering and appropriate analytical tools.	-
CO3	Develop and communicate project outcomes through technical reports and presentations by integrating project management, techno-economic feasibility, sustainability, engineering ethics and lifelong learning.	-

COs–POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1		3									3	3	3
CO2	3		3	3	3							3	3
CO3						3	3	3	3	3		3	3
	3	**3**	**3**	**3**	**3**	**3**	**3**	**3**	**3**	**3**	**3**	**3**	**3**

1–low, 2–medium, 3–high

25CSE06 INTERNSHIP II

0 0 0 1

This course provides students with experiential learning through an Internship II, enabling them to identify real-world industry problems and develop innovative solutions. The internship II shall be carried out during the Sixth Semester vacation. At the end of the training, students shall submit a report and present their work for evaluation through a viva-voce conducted by internal examiners.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Execute an internship or community-based project by applying project management, techno-economic analysis, sustainability, ethical practices, lifelong learning and effectively communicate the outcomes through professional reports and presentations.	**-**

COs–POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1						3	3	3	3	3	3	3	3
						3	**3**	**3**	**3**	**3**	**3**	**3**	**3**

1–low, 2–medium, 3–high

SEMESTER VIII

25CSE07 PROJECT WORK II

0 0 8 4

This course enables students to apply engineering knowledge to solve real-world problems through project-based learning. Students identify a problem in an emerging area, conduct a literature survey, develop and implement an appropriate solution and evaluate its effectiveness. The course concludes with the submission of a technical report and presentation of the project outcomes

Total P: 120 periods

COURSE OUTCOMES:

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Analyze the literature to identify a research problem and formulate appropriate objectives for solution development.	-
CO2	Design and develop an engineering solution by applying principles of mathematics, computing, engineering and appropriate analytical tools.	-
CO3	Develop and communicate project outcomes through technical reports and presentations by integrating project management, techno-economic feasibility, sustainability, engineering ethics and lifelong learning.	-

COs–POs & PSOs MAPPING

CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PSO1**	**PSO2**
CO1		3									3	3	3
CO2	3		3	3	3							3	3
CO3						3	3	3	3	3		3	3
	3	**3**	**3**	**3**	**3**	**3**	**3**	**3**	**3**	**3**	**3**	**3**	**3**

1–low, 2–medium, 3–high

PROFESSIONAL ELECTIVE COURSES
25CSP01 MICRO SERVICE ARCHITECTURE

2 0 2 3

INTRODUCTION TO MICROSERVICES AND ARCHITECTURE STYLES: Monolithic vs Microservices Architecture – Principles and characteristics of Microservices – Benefits and challenges – Decomposition of applications – Role of domain–driven design. **(6+6)**

DESIGNING MICROSERVICES: Designing Services: Business capabilities and bounded context – Data Management: Database per service, Event sourcing – API design and communication: REST, gRPC, Message brokers – Service contracts and versioning – Circuit breaker, Service discovery. **(6+6)**

IMPLEMENTATION AND INTEGRATION: Implementing Microservices with Spring Boot / Node.js – Inter– Service Communication Patterns – Integrating third–party APIs. **(6+6)**

DEPLOYMENT AND ORCHESTRATION: Containerization using Docker – Orchestration with Kubernetes – CI/CD Pipelines for Microservices logging, Monitoring (ELK, Prometheus, Grafana) – Configuration management (Spring Cloud Config, etc.,). **(6+6)**

EMERGING TRENDS AND CASE STUDIES: Security in Microservices: OAuth2, JWT, API Gateway – Microservice Anti–Patterns – Event–driven architecture and CQRS – Serverless Microservices. **(6+6)**

LIST OF EXPERIMENTS:

1. Creating a simple Microservice using Spring Boot / Node.js
2. Developing a RESTful Microservice using Spring Boot
3. Implementing Inter–Service Communication using REST and RabbitMQ
4. Containerizing Microservices with Docker
5. Deploying Microservices on Kubernetes

Total L: 30 + P: 30 = 60 periods

TEXTBOOK:

1. Sam Newman, *'Building Microservices: Designing Fine–Grained Systems'*. 2nd edition, O'Reilly Media, 2021.

REFERENCES:

1. Mike Amundsen, Ronnie Mitra, Matt McLarty, Irakli Nadareishvili, *'Microservice Architecture: Aligning Principles Practices and Culture'*. O'Reilly, 2016.
2. Chellammal Surianarayanan, Gopinath Ganapathy, Raj Pethuru, *'Essentials of Microservices Architecture'*. CRC Press, 2019.
3. Sam Newman, *'Monolith to Microservices–Evolutionary Patterns to Transform Your Monolith'*. O'Reilly, 2019.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Describe microservices architecture concepts, principles, and application decomposition using domain–driven design.	**K2**
CO2	Apply micro service design practices including service design, data management, APIs, communication patterns, and resilience mechanisms.	**K3**
CO3	Analyze micro service implementation, integration, deployment, containerization, orchestration, and monitoring techniques.	**K4**
CO4	Design microservices applications using modern tools, containerization, and prepare technical documentation or demonstrations.	–

COs–POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1												2	2
CO2	3											3	3
CO3		2										2	2
CO4								2	2			2	2
	3	**2**						**2**	**2**			**2**	**2**

1–low, 2–medium, 3–high

25CSP02 UI/UX DESIGN

2 0 2 3

FOUNDATIONS OF UX AND UI: Introduction to Human–Computer Interaction (HCI) and User Experience (UX) – Principles of Usability and UX – Difference between UI and UX – Human–Centered Design Process – Stages in UX Design – Empathy Mapping – User Personas – Journey Mapping. **(6+6)**

INTERACTION DESIGN AND WIREFRAMING: Information Architecture – Interaction Design Principles – Task Flows – Storyboarding – Wireframing – Mockups – UI Design Patterns – Paper Prototyping – Low–Fidelity and High–Fidelity Prototypes. **(6+6)**

VISUAL DESIGN AND FRONTEND TECHNOLOGIES: Typography – Color Theory – Layout Design – Responsive Design Principles – HTML – CSS – JavaScript Basics – Frontend Frameworks – Bootstrap – Introduction to Cross-Platform UI Development using Flutter – Building UI Components. **(6+6)**

USABILITY TESTING AND EVALUATION: Usability Testing Methods – Heuristic Evaluation – Usability Metrics (SUS – NPS – Time–On–Task) – A/B Testing – Moderated and Unmoderated Testing – Cognitive Walkthroughs – User Feedback Collection – Iterative Design Process. **(6+6)**

EMERGING TECHNOLOGIES: UX Design for Mobile and Wearable Devices – Gesture–Based Interaction – Voice User Interfaces – Conversational Interfaces (Chatbots) – Introduction to AI in UX Design. **(6+6)**

LIST OF EXPERIMENTS:

1. Analyze UI interaction patterns in existing applications.
2. Design a user interface using UI design principles and style guides.
3. Create wireframes and wire flows using tools like Figma.
4. Apply design thinking process to solve a given problem.
5. Identify a user problem and design a prototype solution.

Total L: 30 + P: 30 = 60 periods

TEXT BOOKS:

1. Alan Dix, Janet Finlay, Gregory Abowd and Russell Beale, *'Human–Computer Interaction'*. 5th edition, Pearson, 2020.
2. Joel Marsh, *'UX for Beginners'*. O'Reilly Media, 2022.

REFERENCES:

1. Ben Shneiderman, Catherine Plaisant, Maxine Cohen, Steven Jacobs, Niklas Elmqvist and Nicholas Diakopoulos, *'Designing the User Interface: Strategies for Effective Human–Computer Interaction'*. 6th edition, Pearson, 2016.
2. Jennifer Tidwell, Charles Brewer and Aynne Valencia, '*Designing Interfaces: Patterns for Effective Interaction Design*'. 2nd edition, O'Reilly Media, 2020.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain principles of human–computer interaction, user experience, and human–centered design.	**K2**
CO2	Apply interaction design techniques to create wireframes, prototypes, and user interfaces.	**K3**
CO3	Analyze usability of interfaces using evaluation methods and metrics to improve user experience.	**K4**
CO4	Design user–centric solutions incorporating modern UX trends and emerging interaction technologies.	-

COs–POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1												3	3
CO2	3											3	3
CO3		3										3	3
CO4												3	3
	3	**3**										**3**	**3**

1–low, 2–medium, 3–high

25CSP03 DEVOPS

2 0 2 3

INTRODUCTION: Dev Vs Ops Practices – Devops Principles and Culture – Overview of Devops Lifecycle – Adopting Devops – Identifying the Target State – Basic Transformation Strategies – Distributed Vs Centralized Version Control Systems – Git Fundamentals (Repository, Commits, Branching, Merging). **(6+6)**

CONTINUOUS INTEGRATION AND DELIVERY: CI/CD Concepts – Benefits and Best Practices – Stages Of A CI/CD Pipeline – Build Automation – Test Automation – Artifact Management – Introduction to CI/CD Tools – Pipeline Creation Concepts using Jenkins and Github Actions – Testing in CI/CD – Unit Testing – Automated Test Suites – Test Pyramid – Overview of Deployment Strategies. **(6+6)**

CONTINUOUS DEPLOYMENT AND INFRASTRUCTURE AUTOMATION: Concepts of Infrastructure Automation – Configuration Management – Introduction to Ansible – Features – Benefit – Terminology Workflow and Usage – Introduction to Containers Using Docker – Container Lifecycle and Images – Introduction to Cloud Foundry and Application Deployment – Monitoring – Prometheus. **(6+6)**

DEVSECOPS: Devops to Devsecops – Integration of Security in Devops Lifecycle – Types of Attacks – Common Vulnerabilities – Layered Security Approach – Methodologies – Automation– Security Principles in Docker and Kubernetes – Kubernetes Security Architecture, RBAC, Network Policies, Secrets Management, Pod Security Standards – Image and Runtime Security. **(6+6)**

MLOPS AND ADVANCED PRACTICES: Overview – Key Features – MLOps Lifecycle – Developing Models – Preparing for Production – Monitoring and Feedback Loop – Model Governance – MLOps in Practice **(6+6)**

LIST OF EXPERIMENTS:

1. Version Control and Branch Management using Git (Includes basic Git commands, branch creation, merging, and conflict resolution).
2. Remote Repository Management and Collaboration using GitHub (Includes pushing to remote repositories, creating pull requests, code reviews, and merging PRs).
3. Build a simple application using a build automation tool.
4. Create CI pipeline using a CI tool, integrate with GitHub repository and automate build process.
5. Extend CI pipeline to CD, deploy application to local server and Validate deployment.

Suggested Tools: Maven / Gradle, Jenkins / GitHub Actions, Docker, Cloud Foundry

Total L: 30 + P: 30 = 60 periods

TEXT BOOKS:

1. Sanjeev Sharma, *'The DevOps Adoption Playbook: A Guide to Adopting DevOps in a Multi–Speed IT Enterprise'*. John Wiley & Sons, Inc., IBM Press, 2017.
2. Gene Kim, Jez Humble, Patrick Debois and John Willis, *'The DevOps Handbook: How to Create World –Class Agility, Reliability and Security in Technology Organizations'*. O'Reilly Media, 2nd Edition, 2021.

REFERENCES:

1. Noah Gift and Alfredo Deza, *'Practical MLOps'*. O'Reilly Media, Inc., 2021.
2. Mark Treveil, Nicolas Omont, Clement Stenac, Kenji Lefevre, Du Phan, Joachim Zentici, Adrien Lavoillotte, Makoto Miyazaki and Lynn Heidmann, *'Introducing MLOps'*. O'Reilly Media, Inc., 2020.
3. Sricharan Vadapalli, *'DevOps – Continuous Delivery, Integration, and Deployment with DevOps'*. Packt Publishing, 2018.
4. Ankita Patil and Mitesh Soni, *'Hands–on Pipeline as Code with Jenkins: CI/CD Implementation for Mobile, Web, and Hybrid Applications Using Declarative Pipeline in Jenkins'*. BPB Publications, 2021.

COURSE OUTCOMES

At the end of this course students will be able to:		**Bloom's Level**
CO1	Explain the principles, lifecycle, and practices of DevOps for modern software development.	**K2**
CO2	Apply DevOps practices to design and implement automated workflows for software development and deployment.	**K3**
CO3	Analyze and utilize tools and techniques for automation, containerization, monitoring, and security in software systems.	**K4**
CO4	Evaluate modern practices including DevSecOps and MLOps for building reliable and scalable software systems.	-

COs–POs & PSOs MAPPING

CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PSO1**	**PSO2**
CO1												3	3
CO2	3											3	3
CO3		3										3	3
CO4				3								3	3
	3	**3**		**3**								**3**	**3**

1–low, 2–medium, 3–high

25CSP04 SOFTWARE TESTING AND AUTOMATION

2 0 2 3

INTRODUCTION TO TESTING: Introduction to testing as an engineering activity – The role of process in Software Quality – Testing as a process – Overview of Testing Maturity Model (TMM) – Testing fundamentals: Basic definitions, Software testing principles, Tester's role in a Software development organization. **(6+6)**

TEST DESIGN STRATEGIES-I: Black Box approach to test case design – Random testing – Equivalence class partitioning – Boundary value analysis – API Testing – Other Black Box test design approaches. **(6+6)**

TEST DESIGN STRATEGIES-II: White Box approach to test case design – Test adequacy criteria – Coverage and control flow graphs – Covering code logic – Paths – Additional white box test design approaches – White box testing methods. **(6+6)**

ADVANCED TESTING CONCEPTS: Performance testing: Load testing, Stress testing, Volume testing, Fail-Over testing, Testing the documentation, Security testing, Testing in the agile environment, Testing web and mobile applications. Types of reviews – Inspection – Walkthroughs – Components of review plan. **(6+6)**

TEST AUTOMATION AND TOOLS: Automated software testing – Automate testing of web applications, Selenium: Introducing web driver and web elements, locating web elements, Actions on web elements, Different web drivers, understanding web driver events, Testing: Understanding testing, Methods to test, Test reports, Testim tool for AI-based test automation. **(6+6)**

LIST OF EXPERIMENTS

1. Design the test cases for testing the e-commerce application
2. Test the e-commerce application and report the defects in it.
3. Develop the test plan and design the test cases for an inventory control system.
4. Test the performance of the e-commerce application.
5. Automate the testing of e-commerce applications using Selenium.

Total L: 30 + P: 30 = 60 periods

TEXTBOOK:

1. Yogesh Singh, *'Software Testing'*. Cambridge University Press, 2012.
2. Roger. S. Pressman, *'Software Engineering A Practitioner's Approach'*. 9th edition, McGraw Hill International Edition, 2020.
3. Unmesh Gundecha, Satya Avasarala, *'Selenium WebDriver 3 Practical Guide'*. 2nd edition, Packt Publishing, 2018.

REFERENCES:

1. Kshirasagar Naik, Priyadhrshini Tripathy, *'Software Testing and Quality Assurance'*. John wiley & sons Inc Publication, 2008.
2. Paul C. Jorgensen, *'Software Testing: A Craftsman's Approach'*. 4th edition, Taylor & Francis Group, 2014.
3. Nina S Godbole, *'Software Quality Assurance–Principles and Practice'*. 2nd edition, Narosa Publishing, 2016.
4. Carl Cocchiaro, *'Selenium Framework Design in Data–Driven Testing'*. Packt Publishing, 2018.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the fundamental concepts of software testing, defects and testing processes.	**K2**
CO2	Apply black–box and white–box strategies to design effective test cases for software systems.	**K3**
CO3	Analyze software systems using advanced testing methods and review techniques.	**K4**
CO4	Design automated testing solutions for web and mobile applications using modern testing tools and frameworks	**–**

COs–POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1												2	2
CO2	3											3	3
CO3		2										2	2
CO4								2	2			2	2
	3	**2**						**2**	**2**			**2**	**2**

1–low, 2–medium, 3–high

25CSP05 FULL STACK WEB DEVELOPMENT

3 0 0 3

PRINCIPLES OF SECURE FULL STACK DEVELOPMENT: Full stack architecture overview – Client-server communication – HTTP/HTTPS – HTML5 – CSS3 – JS security – Secure DOM handling – Input validation and sanitization – Basics of secure backend frameworks (Node.js / Django / Spring). **(9)**

SECURE FRONTEND ENGINEERING: Component security – Secure routing – State handling risks – Avoiding inline scripts – Prevention of Cross-site scripting (XSS) – Cross-site request forgery (CSRF) – Content security policy (CSP) – Preventing clickjacking – Secure API consumption. **(9)**

BACKEND SECURITY: JSON web tokens (JWT) – Oauth2 – Single sign-on (SSO) – Password hashing (bcrypt/Argon2) – Role-based and attribute-based access control (RBAC/ABAC) – Secure session management – Rate limiting – Logging and monitoring – Secure error handling – API gateway security – STRIDE Analysis. **(9)**

SECURE DATABASE AND DEVSECOPS: SQL injection prevention – Prepared statements – ORM security – NoSQL injection – Secrets management (vault environment encryption) – Static/Dynamic code analysis (SAST/DAST) – CI/CD pipelines (Jenkins/GitHub action) – Docker security – Package vulnerability scanning. **(9)**

SECURE DEPLOYMENT AND CLOUD SECURITY: TLS/SSL – Secure headers (CSP, HSTS, X-Frame-Options) – Server hardening – Reverse proxy security (Nginx) – Cloud IAM (AWS/Azure/GCP) – Kyma – Cloud foundry – API keys and secret rotation – Logging and alerting – Zero trust principles. **(9)**

Total L: 45 periods

TEXTBOOKS:

1. Andrew Hoffman, '*Web Application Security: Exploitation and Countermeasures*'. O'Reilly Media, 2020.
2. Neil Madden, '*API Security in Action*'. 1st edition, Manning Publications, 2020.

REFERENCES:

3. Prabath Siriwardena and Nuwan Dias, *'Advanced API Security: Oauth 2.0 and Beyond'*. Apress, 2020.
4. Dafydd Stuttard and Marcus Pinto, *'The Web Application Hacker's Handbook'*. 2nd edition, Wiley Publishing, 2011.
5. Steve Fulton and Jeff Fulton, *'HTML5 Canvas'*. O'Reilly Media, 2013.
6. Frank Zammetti, *'Modern Full-Stack Development: Using TypeScript, React, Node.js, Webpack, and Docker'*. Apress, New Delhi, 2020.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Describe secure software architectures, communication models, and development principles used in modern full stack applications.	**K2**
CO2	Apply secure development methodologies, programming practices, and tools for building and managing full stack systems.	**K3**
CO3	Analyze system interactions, security challenges, and operational factors affecting the reliability and performance of software applications.	**K4**
CO4	Evaluate techniques, frameworks, and deployment strategies for ensuring security, scalability, and efficiency throughout the software development life cycle.	-

COs–POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1												3	3
CO2	3											3	3
CO3		3										3	3
CO4			2					2				2	2
	3	**3**	**2**					**2**				**3**	**3**

1–low, 2–medium, 3–high

25CSP06 MERN STACK

2 0 2 3

MERN STACK FOUNDATIONS & ENVIRONMENTS: Full Stack Development – Benefits and Responsibilities of Full Stack Developers – Overview of MongoDB: Document-Oriented NoSQL Database, Express.js: Lightweight Framework for Web APIs, React.js: Virtual DOM, and Node.js: Event Loop – Environment Setup: Node.js, npm, Git, MongoDB Atlas. **(6+6)**

UI DESIGN WITH REACT: Components, Props, and State Functional and Class Components – Passing props between components – React router – Dynamic routing with params – Event Handling – Hooks (useState, useEffect) – Axios for API Calls. **(6+6)**

SERVER-SIDE LOGIC & API ARCHITECTURE: Introduction to Node.js and Express.js – Express Routing Basics – RESTful API Design – REST Principles – Middleware and Routing – Express Router for Modular Code – JWT Authentication & Authorization – JSON Web Token Workflow. **(6+6)**

RELATIONAL & NOSQL DATA MANAGEMENT: Introduction to Relational vs. Non-Relational (NoSQL) Databases – Structured Database Fundamentals (SQL Basics, Tables, and Joins) – Overview of Connecting Node.js to Relational Databases – MongoDB Basics and CRUD Operations – Creating Databases and Collections – Insert, Read, Update, Delete Documents – Mongoose ODM Connecting to MongoDB – Schema and Model Design – Aggregation and Relationships – Population in Mongoose. **(6+6)**

SYSTEM INTEGRATION & CLOUD DEPLOYMENT: Integrating Frontend and Backend – Environment Variables and CORS Setup – State Management (Context API or Redux) – Vercel – Render – Preparing Projects for Deployment – Deploying Backend to Heroku – Deploying Frontend to Vercel and Render – Version Control with Git and GitHub. **(6+6)**

LIST OF EXPERIMENTS

1. Build an interactive application (e.g., a Voting Dashboard or a multi-functional Calculator) using React.js.
2. Create a robust backend HTTP server, interacting with client using Node.js and Express.js.
3. Develop an application to store, maintain data using Node.js and relational database.
4. Develop advanced data-driven backend feature using NoSQL (MongoDB Atlas) and Mongoose ODM.
5. Build a Full stack live cloud integrated app with React, Node, Express & MongoDB.

Total L: 30 + P: 30 = 60 periods

TEXT BOOK:

1. Andrew Mead, '*Learning Full–Stack JavaScript Development: MongoDB, Node, and React'.* Packt Publishing, 2017.

REFERENCES:

1. Vasan Subramanian, *'Pro MERN Stack: Full Stack Web App Development with Mongo, Express, React, and Node'*. 2nd edition, Apress, 2019.
2. Shama Hoque, '*Full–Stack React Projects: Modern Web Development Using React 16, Node, Express, and MongoDB'.* 2nd edition, Packt Publishing, 2020.
3. Kristina Chodorow, *'MongoDB: The Definitive Guide'*. 3rd edition, O'Reilly Media, 2019.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the fundamentals of MERN stack architecture and the role of Node.js event loops	**K2**
CO2	Apply dynamic routing using react.js components and hooks to design responsive user interfaces	**K3**
CO3	Integrate MongoDB with Node.js and Express.js using CRUD operations and Mongoose ODM.	**K4**
CO4	Evaluate full–stack deployment strategies using cloud platforms like Heroku, Vercel, and Render.	**–**

COs–POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1												3	3
CO2	3											3	3
CO3		3										3	3
CO4			3									3	3
	3	**3**	**3**									**3**	**3**

1–low, 2–medium, 3–high

25CSP07 AGILE METHODOLOGIES

2 0 2 3

AGILE METHODOLOGY: Agile software development – Traditional model vs. Agile model – Classification of agile methods – Agile manifesto and principles – Agile project management – Ethics in Agile teams. **(6+6)**

AGILE PROCESSES: Lean production – SCRUM – Crystal – Adaptive software development – Extreme programming (XP): Method overview – Lifecycle – Work products, roles and practices. **(6+6)**

AGILE PROJECT MANAGEMENT: Agile project management – Traditional model Vs RAD model for project management – Task planning and effort tracking – The project managers new work – Agile development management – Agile product management. **(6+6)**

AGILE SOFTWARE TOOL: Introduction to JIRA – JIRA version – JIRA cloud instance setup – JIRA administration: Creating a new user and group – Permission levels – Global permission – Project roles and permissions – JIRA schemes: Issue types, screens, custom fields and workflow. **(6+6)**

AGILITY AND QUALITY ASSURANCE: Agile product development – Agile metrics – Feature driven development (FDD) – Quality assurance in FDD– Financial and production metrics in FDD – Agile approach to quality assurance. **(6+6)**

LIST OF EXPERIMENTS

1. Creating and Managing an Agile Scrum Board.
2. Sprint Planning and Effort Tracking in Agile Projects.
3. Implementing Agile Roles, Permissions and designing Agile Workflow and Issue Types.
4. Implementing Extreme Programming (XP) Practices.
5. Agile Metrics and Quality Assurance using Feature Driven Development (FDD).

Total L: 30 + P: 30 = 60 periods

TEXTBOOKS:

1. Mark C. Layton, Steven J Ostermiller, and Dean J. Kynaston, *'Agile Project Management for Dummies'*. John Wiley & Sons, Inc 3rd Edition, 2020.
2. David J. Anderson and Eli Schragenheim, *'Agile Management for Software Engineering: Applying the Theory of Constraints for Business Result'.* Prentice Hall, 2009.

REFERENCES:

1. Andrew Stellman, Jennifer Greene, *'Learning Agile: Understanding Scrum, XP, Lean and Kanban'*. O Reilly, 2015
2. Hazza and Dubinsky, *'Agile Software Engineering, Series: Undergraduate Topics in Computer Science'*. Springer, 2009.
3. Robert C. Martin, *'Agile Software Development, Principles, Patterns and Practices'*. First International Edition, Prentice Hall, 2014.
4. Rubin K, *'Essential Scrum: A practical guide to the most popular Agile process'*. Addison-Wesley, 2013.
5. Jira Software Tutorials: *https://www.atlassian.com/agile/tutorials/how-to-do-scrum-with-jira*

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Describe the core principles of the Agile manifesto, compare traditional and agile software development models and identify techniques to enhance team collaboration	**K2**
CO2	Apply agile process frameworks including SCRUM, XP, Lean and Crystal to plan, execute and track software development activities.	**K3**
CO3	Analyze agile project management strategies, effort tracking approaches and the evolving role of the project manager in agile environments	**K4**
CO4	Evaluate agile quality assurance practices, feature-driven development metrics, and utilize JIRA for project configuration and workflow management.	**K5**
CO5	Design and implement agile artefacts, scrum boards, sprint plans, XP practices and FDD-based quality assurance setups using JIRA and related tools.	**K6**

COs–POs & PSOs MAPPING

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

1–low, 2–medium, 3–high

25CSP08 VIBE CODING

2 0 2 3

VIBE CODING FOUNDATIONS: AI in programming – Evolution – Co-creator Role – Vibe coding concepts – Programming with emotion, Tone, and Intention – Clarity and expressiveness – Cognitive flow in coding – Coding environments and themes – Music and mood integration – Developer states and creativity. **(6+6)**

EXPRESSIVE CODING & VISUAL AR: Clean and creative code – ASCII art – Generative art – Audio-reactive visuals – Tools: Processing, p5.js, Python turtle – Aesthetic terminal outputs – Mood - Aware design – Code poetry – Emotional signatures in code – Weekly vibe coding plan. **(6+6)**

HUMAN-AI COLLABORATION IN CODE: AI as partner – AI tools like ChatGPT, Copilot, Claude – Context-aware generation - Code refactoring – Coder to curator shift – Ethical considerations – Authorship and AI bias – personal expression in AI outputs – Responsible use of AI tools. **(6+6)**

WEB APIS & CREATIVE APPLICATIONS: Web APIs and Mood-based projects – Creative problem- solving – Playlist boards – Visual journals – Journaling logic – Weather/Mood apps – Real-World projects – User-Focused design – Quizzes and challenges. **(6+6)**

ETHICAL DIMENSIONS: The creative coder in an AI world – Bias, Ethics, and Authorship – Personal expression in AI-generated output – AI as teammate or tool. **(6+6)**

LIST OF EXPERIMENTS

1. AI-Assisted Prompt-Based Code Generation using ChatGPT and GitHub Copilot.
2. Creative Generative Art and Interactive Visual Design using p5.js and Processing.
3. Emotion-Based User Interface Development and Mood-Responsive Design.
4. Web API Integration and Dynamic Content Display using Flask and Postman.
5. Basic Workflow Automation and Continuous Integration using GitHub Actions.

Total L: 30 + P: 30 = 60 periods

TEXT BOOKS:

1. Michael A. Angelo, '*Vibe Coding #1: Revolution in AI Driven Programming – Comprehensive Reports*'. Kindle Edition, 2025.

REFERENCES:

1. Addy Osmani, '*Vibe Coding: The Future of Programming'*. O'Reilly Media, 2024.
2. GitHub Copilot, '*Documentation and Real–World Case Studies'*. GitHub Inc., 2023.
3. Daniel Shiffman, '*The Nature of Code'.* D. Shiffman Publishing 2nd edition, 2021.
4. Addy Osmani, '*Beyond Vibe Coding'*. O'Reilly Media, Inc., 2025.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the fundamental concepts of vibe coding, AI–assisted programming, and creativity in coding practices.	**K2**
CO2	Apply expressive coding techniques using tools for aesthetic and interactive outputs.	**K3**
CO3	Analyze the role of AI tools in coding workflows and collaboration.	**K4**
CO4	Evaluate creative Web API–based applications in terms of user experience, functionality, and ethical considerations.	**-**

COs–POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1													
CO2	3											3	3
CO3		3										3	3
CO4			1									1	1
	3	**3**	**1**									**3**	**3**

1–low, 2–medium, 3–high

25CSP09 SOFTWARE DEFINED NETWORKS

3 0 0 3

TRADITIONAL NETWORKS AND SDN: Traditional networks vs SDN – Modern networking: Elements of modern networking – SDN definition – Key concepts: Control plane vs Data plane – Centralized vs Distributed control – SDN architecture – Advantages and challenges. **(9)**

SDN ARCHITECTURE AND INTERFACES: Southbound APIs: OpenFlow protocol – Flow table structures – Switch and controller communication – Northbound APIs – Application plane – Controller placement issues – Overview of Popular controllers –Ryu, ONOS – OpenDaylight. **(9)**

NETWORK VIRTUALIZATION AND EMULATION: Network virtualization concepts – Virtual LANs – Nested VLANs – Network virtualization architecture – Mininet emulator – Virtual switches – Open vSwitch – Introduction to network function virtualization – NFV Concepts – NFV principles – NFV reference architecture: Management and orchestration, reference points. **(9)**

SDN PROGRAMMING AND APPLICATIONS: Flow programming – Reactive and proactive flow rules – Use cases: Traffic engineering, QoS, Load balancing – Introduction to P4 language: Function blocks and interfaces – protocol – Independent switch architecture: Language elements, Data types – Creating the server – Configuring the P4 software switch – Compiling and executing P4 program. **(9)**

RECENT DEVELOPMENTS AND SECURITY: Intent–Based networking, AI in SDN – SDN for 5G, Data centers – SDN security – NFV security: Attack surfaces, security techniques – Cloud security: Data protection in cloud, Cloud security as a service – IoT security: patching vulnerability, IoT security framework. **(9)**

Total L: 45 periods

TEXT BOOKS:

1. William Stallings, '*Foundations of Modern Networking: SDN, NFV, QoE, IoT and Cloud'*. 1st edition, Pearson Education, 2015.
2. Paul Goransson, Chuck Black, Timothy Culver'. *Software Defined Networks: A Comprehensive Approach*'. 2nd edition, Morgan Kaufmann, 2022.

REFERENCES:

1. Ken Gray, Thomas D. Nadeau, '*Network Function Virtualization'*. Morgan Kauffman, 2016.
2. Thomas D Nadeau, Ken Gray, '*SDN: Software Defined Networks'*. O'Reilly Media, 2013.
3. Fei Hu, '*Network Innovation through OpenFlow and SDN: Principles and Design'*. 1st edition, CRC Press, 2014.
4. Oswald Coker, Siamak Azodolmolky, *'Software–Defined Networking with OpenFlow'*. 2nd edition, O'Reilly Media, 2017.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Summarize the architecture and motivation for adopting Software Defined Networking.	**K2**
CO2	Apply SDN concepts to configure flows and manage networks using controllers.	**K3**
CO3	Examine SDN environments by analyzing the role of network virtualization, emulation tools, APIs, controller platforms, and security mechanisms.	**K4**
CO4	Present a technical seminar on software defined networks applications by demonstrating domain understanding, and technical insights.	**–**

COs–POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1												3	3
CO2	3											3	3
CO3		3										3	3
CO4								2	2			2	2
	3	**3**						**2**	**2**			**3**	**3**

1–low, 2–medium, 3–high

25CSP10 SECURITY AND PRIVACY IN CLOUD

3 0 0 3

CLOUD SECURITY & PRIVACY: Cloud architecture & security overview – Shared responsibility model – Threats and vulnerabilities – Security services – Confidentiality, Integrity, Authentication, Nonrepudiation, Access Control – Cloud–specific risk assessment – Key security challenges – Cloud security reference models – CSA – NIST. **(9)**

IDENTITY, ACCESS, AND AUTHENTICATION: Access control models RBAC, ABAC, MAC – Identity and Access Management (IAM) – Single Sign–On (SSO) – Multi–Factor Authentication (MFA) – Federated identity– Identity providers and service consumers – OAuth, OpenID, SAML – IAM in AWS, Azure. **(9)**

DATA SECURITY AND CRYPTOGRAPHY: Data confidentiality and integrity – Encryption at rest and in transit – Conventional and public–key cryptography, hash functions, authentication, and digital signatures – Key Management Systems (KMS) – Homomorphic encryption – Secure data deletion – Secure cloud storage mechanisms. **(9)**

INFRASTRUCTURE AND APPLICATION SECURITY: Securing VMs, containers, and serverless functions – Secure APIs – Patch management – Security monitoring and incident response – Network security tools – IDS/IPS – firewall – Cloud Security Posture Management. **(9)**

DATA PRIVACY: Data privacy regulations – GDPR – HIPAA – Cloud compliance standards – ISO 27017 – SOC 2 – Privacy – preserving computation – Incident handling – Cloud forensics – Zero trust model – Emerging threats and defenses**.** **(9)**

Total L: 45 periods

TEXT BOOKS:

1. Ronald L. Krutz and Russell Dean Vines, '*Cloud Security: A Comprehensive Guide to Secure Cloud Computing'*. 2nd edition, Wiley, 2023.
2. Tim Mather, Subra Kumaraswamy, and Shahed Latif, '*Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance*'. 2nd edition, O'Reilly Media, 2023.

REFERENCES:

1. Mark C. Chu–Carroll, *'Code in the Cloud'*. CRC Press, 2011.
2. Dave shackleford, '*Virtualization Security*'. SYBEX a Wiley Brand 2013.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Summarize the core concepts, challenges, and models of security and privacy in cloud computing.	**K2**
CO2	Implement access control, authentication, encryption, and identity management mechanisms with secure configuration and monitoring practices to protect cloud applications and data.	**K3**
CO3	Analyze regulatory, legal, and compliance frameworks for data protection and privacy in the cloud.	**K4**
CO4	Present a technical seminar on security and privacy in cloud systems applications by demonstrating domain understanding, and technical insights.	**–**

COs–POs & PSOs MAPPING

CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PSO1**	**PSO2**
CO1												3	3
CO2	3											3	3
CO3		3										3	3
CO4								2	2			2	2
	3	**3**						**2**	**2**			**3**	**3**

1–low, 2–medium, 3–high

25CSP21 BLOCKCHAIN TECHNOLOGIES

3 0 0 3

BLOCKCHAIN DESIGN: History: Digital money to distributed ledgers – Design primitives: Protocols, security, consensus, permissions, privacy: Blockchain architecture and design – Basic crypto primitives: Hash, signature hash chain to block chain – Basic consensus mechanisms. **(9)**

BLOCKCHAIN CONSENSUS PROTOCOLS: Requirements for the consensus protocols – Proof of Work (PoW) – Scalability aspects of block chain consensus protocols: Permissioned blockchains – Design goals – Consensus protocols for permissioned blockchains. **(9)**

HYPERLEDGER FABRICS: Decomposing the consensus process – Hyper ledger fabric components – Chain code design and implementation: Hyper ledger Fabric II – Beyond chain code – Fabric SDK and front end – Hyper ledger composer tool. **(9)**

BLOCKCHAIN IN FINANCE AND TRADE: Blockchain in financial software and systems (FSS): Settlements – KYC – Capital markets – Insurance blockchain in trade/supply chain: provenance of goods, visibility – trade/supply chain finance – Invoice management/discounting. **(9)**

BLOCKCHAIN FOR GOVERNANCE AND SECURITY: Blockchain for Government: digital identity, land records and other kinds of record keeping between government entities, public distribution system / social welfare systems: Blockchain cryptography: Privacy and security on blockchain. **(9)**

Total L: 45 periods

TEXTBOOKS:

1. Mark Gates, '*Blockchain: Ultimate guide to understanding blockchain, bitcoin, crypto currencies, smart contracts and the future of money*'. Wise Fox Publishing and Mark Gates 2017.
2. Imran Bashir, '*Mastering Blockchain: Inner workings of blockchain, from cryptography and decentralized identities, to DeFi, NFTs and Web3*'. Packt Publishing Limited, Fourth Edition, 2023.

REFERENCES:

1. Bahga, Vijay Madisetti, '*Blockchain Applications: A Hands–On Approach'.* Arshdeep Bahga, Vijay Madisetti publishers 2017.
2. Salman Baset, Luc Desrosiers, Nitin Gaur, Petr Novotny, Anthony O'Dowd, Venkatraman Ramakrishna, '*Hands–On Blockchain with Hyper ledger: Building decentralized applications with Hyperledger Fabric and Composer'.* 2018.
3. Melanie Swa, '*Block chain'.* O'Reilly Media 2014.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the fundamental concepts, evolution, architecture and design principles of blockchain technology, including security, privacy and distributed ledgers.	**K2**
CO2	Apply blockchain development techniques, cryptographic primitives and Hyperledger Fabric tools to build basic decentralized applications.	**K3**
CO3	Analyze blockchain consensus mechanisms and permission models to determine their effectiveness in different environments.	**K4**
CO4	Prepare a case study on blockchain applications in finance, trade, governance and security for solving real–world challenges and improving digital trust systems.	**–**

COs–POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1												3	3
CO2	3											3	3
CO3		3										3	3
CO4			1									1	1
	3	**3**	**1**									**3**	**3**

1–low, 2–medium, 3–high

25CSP12 ETHICAL HACKING

3 0 0 3

PENETRATION TESTING: Black Box, White Box, Grey Box, Types of Penetration test: Network Penetration Testing – Web Application Penetration Testing – Wireless Network Testing – Social Engineering. Writing Reports – Structure of a Penetration Testing Report – Vulnerability Assessment Summary – Risk Assessment, Methodology. **(9)**

FOOTPRINTING AND SOCIAL ENGINEERING: Using Web Tools for Foot printing: Search Engines – WHOIS Lookup – DNS Interrogation – Social Media Analysis – Conducting Competitive Intelligence – Introduction to Social Engineering – The Art of Shoulder Surfing, Dumpster Diving, Piggybacking, Phishing. **(9)**

PORT SCANNING AND ENUMERATION: Port Scanning: Basics of Port–Scanning Tools – Nmap, Unicornscan, Nessus and OpenVAS – Conducting Ping Sweeps – Fping – Hping. Crafting IP Packets Enumeration: Enumerating Windows Operating Systems – NetBIOS Basics – NetBIOS Enumeration Tools – Additional Enumeration Tools – Enumerating the NetWare Operating System – Enumerating the unix Operating system. **(9)**

NETWORK HACKING: Network Infrastructure: Network Infrastructure Vulnerabilities – Choosing Tools, Scanning – Poking – and Prodding – Common Router – Switch and Firewall Weaknesses – General Network Defenses. Wireless LANs: Understanding the Implications of Wireless – Choosing Tools – Wireless LAN Discovery – Wireless Network Attacks and Countermeasures – Rogue wireless devices – Countermeasures against rogue wireless devices – MAC spoofing – Countermeasures against MAC spoofing – Countermeasures against DoS attacks. **(9)**

OPERATING SYSTEM HACKING: Windows: Windows Vulnerabilities – Choosing Tools – Information Gathering – Null Sessions – Share Permissions: Assessing Shared Resources – Modifying Share Permissions – Missing Patch Exploitation: Identifying Unpatched Systems – Exploiting Missing Updates – Authenticated Scans. **(9)**

Total L: 45 periods

TEXT BOOKS:

1. Rafay Baloch, '*Ethical Hacking and Penetration Testing Guide'*. 2nd edition, Tata Mc Graw Hill, 2023.
2. Michael T. Simpson, Kent Backman, and James E. Corley, '*Hands–On Ethical Hacking and Network Defense'*. 2nd edition, Delmar Cengage Learning, 2020.
3. Kevin Beaver, '*Hacking for Dummies'*. 3rd edition, Wiley Publishing, Inc, 2022.

REFERENCES:

1. Stuart McClure, Joel Scambray and Goerge Kurtz, '*Hacking Exposed Network Security Secrets & Solutions'*. Tata Mc Graw Hill, 2020.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the different stages of penetration testing and social engineering.	**K2**
CO2	Examine the various footprint tools and port scanning tools.	**K3**
CO3	Interpret the vulnerabilities in network services and Windows operating systems using network hacking tools, null sessions and patch analysis.	**K4**
CO4	Investigate vulnerabilities to propose mitigation strategies using authenticated scans.	**–**

COs–POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1												3	3
CO2	3											3	3
CO3		3										3	3
CO4			2									2	2
	3	**3**	**2**									**3**	**3**

1–low, 2–medium, 3–high

25CSP19 REINFORCEMENT LEARNING TECHNIQUES

3 0 0 3

REINFORCEMENT LEARNING PROBLEM: Introduction to RL-Elements of RL-Limitations and Scope - History of Reinforcement Learning - Finite Markov Decision Processes - The Agent - Environment Interface - Goals and Rewards – Returns - Markov Decision Processes. (**9)**

DYNAMIC PROGRAMMING AND MONTE CARLO METHODS: Policy Evaluation - Policy Improvement - Policy Iteration - Value Iteration - Asynchronous Dynamic Programming - Generalized Policy Iteration, Monte Carlo Prediction - Monte Carlo Estimation of Action Values - Monte Carlo Control. **(9)**

APPROXIMATE SOLUTION METHODS: On-policy Approximation of Action Values - Value Prediction with Function Approximation – Gradient - Descent Methods - Linear Methods - Control with Function Approximation – Off - policy Approximation of Action Values - Policy Approximation – Actor - Critic Methods.

(**9)**

DEEP REINFORCEMENT LEARNING: Function Approximation with Neural Networks–Deep Q–Networks (DQN)–Learning Q function in DQN–Action selection in DQN–Experience replay–DQN algorithm –Implementing DQN–Training a DQN agent–Double DQN–Prioritized Experience Replay. **(9)**

APPLICATIONS AND CASE STUDIES: RL in Robotics and Autonomous Systems–Game Playing (e.g., AlphaGo, Atari Games) –Case Studies: TD–Gammon– Samuel's Checkers Player–The Acrobot–Elevator Dispatching– Dynamic Channel Allocation–Job–Shop Scheduling. **(9)**

Total L: 45 periods

TEXTBOOKS:

1. Richard S. Sutton and Andrew G. Barto, '*Reinforcement Learning: An Introduction'*. 2nd edition, MIT Press, 2018.
2. Laura Graesser and Wah Loon Keng, '*Foundations of Deep Reinforcement Learning'*. 1st edition, Addison –Wesley, 2019.

REFERENCES:

1. Maxim Lapan, '*Deep Reinforcement Learning Hands–On'*. 2nd edition, Packt, 2020.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the fundamental concepts of reinforcement learning	**K2**
CO2	Apply reinforcement learning methods to solve decision–making problems.	**K3**
CO3	Analyze approximation and deep reinforcement learning techniques using appropriate tools	**K4**
CO4	Present real–world applications and case studies of reinforcement learning	**–**

COs–POs & PSOs MAPPING

CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PSO1**	**PSO2**
CO1												3	3
CO2	3											3	3
CO3		3										3	3
CO4			1									1	1
	3	**3**	**1**									**3**	**3**

1–low, 2–medium, 3–high

25CSP14 CYBER FORENSICS

3 0 0 3

INTRODUCTION: Definition and Scope of Cyber Forensics – Digital Evidence: Types and Challenges – Categories and Trends in Cyber Crime – Forensic readiness – Incident response and management. **(9)**

EVIDENCE COLLECTION AND PRESERVATION: Handling digital evidence – Acquisition techniques – Disk imaging, cloning – Write blockers – Chain of custody – Ensuring evidence integrity – Use of hashing – Metadata analysis. **(9)**

OPERATING SYSTEM FORENSICS: Windows and Linux Forensics – System logs and artifacts – Registry forensics – File systems – FAT, NTFS, EXT – Data recovery – Live vs Dead forensics. **(9)**

NETWORK AND EMAIL FORENSICS: Fundamental of network forensics – Log analysis – Packet capturing tools – Wireshark, TCPDump – Email structure and header analysis – Phishing and spoofing investigations. **(9)**

LEGAL AND PROFESIONAL ASPECTS: Cyber Laws in India and International standards – Admissibility of digital evidence – Forensics in law enforcement – Ethics in cyber forensics – Professional responsibilities. **(9)**

Total L: 45 periods

TEXTBOOKS:

1. Bill Nelson, Amelia Phillips, Christopher Steuart, *'Guide to Computer Forensics and Investigations'*. 5th edition, Cengage Learning, 2022.
2. John Sammons, *'The Basics of Digital Forensics: The Primer for Getting Started in Digital Forensics'*. 2nd edition, Elsevier, 2014.

REFERENCES:

1. Marjie T. Britz, *'Computer Forensics and Cyber Crime: An Introduction'*. Pearson, 2021.
2. Warren G. Kruse II, Jay G. Heiser, *'Computer Forensics: Incident Response Essentials'*. Wesley, 2020.
3. Eoghan Casey, *'Digital Evidence and Computer Crime: Forensic Science, Computers and the Internet'*. 3rd Edition, Academic Press, 2020.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Describe fundamental principles and investigative approaches for analyzing digital information in complex problems.	**K2**
CO2	Apply appropriate techniques to examine, handle, and interpret digital information for investigative purposes.	**K3**
CO3	Assess investigative approaches and analytical methods to identify and interpret digital evidence effectively.	**K4**
CO4	Evaluate strategies for ensuring secure and dependable digital investigation processes in different environments.	-

COs–POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1												3	3
CO2	3											3	3
CO3		3										3	3
CO4			2					2				2	2
	3	**3**	**2**					**2**				**3**	**3**

1–low, 2–medium, 3–high

25CSP15 AI IN CYBER SECURITY

3 0 0 3

OVERVIEW OF AI IN CYBERSECURITY: Overview of cybersecurity threats – Introduction to AI and ML – Intersection of AI and Cybersecurity – AI use cases in cyber defense – Attack vectors that leverage AI –Threat intelligence. **(9)**

SUPERVISED LEARNING FOR THREAT DETECTION: Classification techniques: SVM, Decision Trees, Random Forest, Naive Bayes – Model training & evaluation – Dataset preparation – Case studies: Malware detection, phishing website classification. **(9)**

UNSUPERVISED LEARNING AND ANOMALY DETECTION: Clustering algorithms: K–Means, DBSCAN, PCA – Autoencoders – Detecting anomalous behaviour in system logs and network traffic – Insider threat detection. **(9)**

DEEP LEARNING AND NLP IN CYBERSECURITY: Introduction to deep learning – DNNs, CNNs, RNNs – Natural Language Processing for spam/phishing detection – AI for fake news and social engineering defense – Overview of Embeddings – Transformers. **(9)**

SECURITY, PRIVACY, AND ETHICS IN AI: Adversarial machine learning – Model poisoning and evasion attacks – Fairness and bias in AI–based systems – Ethical considerations – Recent research and future trends. **(9)**

Total L: 45 periods

TEXT BOOKS:

1. Mark Stamp, *'Introduction to Machine Learning with Applications in Information Security'*. 2nd edition, CRC Press, 2023.
2. Sumeet Dua and Xian Du, *'Data Mining and Machine Learning in Cybersecurity'*. CRC Press, 2023 edition.

REFERENCE BOOKS:

1. Alessandro Parisi, *'Hands–On Artificial Intelligence for Cybersecurity'*. 1st edition, Packt Publishing, 2019.
2. Clarence Chio and David Freeman, *'Machine Learning and Security: Protecting Systems with Data and Algorithms'*. 1st edition, O'Reilly Media, 2018.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Describe the fundamental concepts of AI and machine learning techniques used in cybersecurity.	**K2**
CO2	Demonstrate the use of AI/ML techniques to detect, analyze, and mitigate cybersecurity threats using various algorithms.	**K3**
CO3	Examine the effectiveness and limitations of these techniques in threat detection, anomaly detection, deep learning applications, and ethical and privacy concerns.	**K4**
CO4	Design AI–based models using machine learning and deep learning techniques to detect, prevent, and mitigate cybersecurity threats in real–world applications.	**–**

COs–POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1												3	3
CO2	3											3	3
CO3		3										3	3
CO4			1		1							1	1
	3	**3**	**1**		**1**							**3**	**3**

1–low, 2–medium, 3–high

25ADP03 SMART SYSTEMS

3 0 0 3

INTRODUCTION TO SMART SYSTEMS: Introduction to Smart Systems – Characteristics and Features – Smart Devices and Smart Environments – Cyber Physical Systems (CPS) – Components of CPS – Embedded Systems and Real-Time Systems – Architecture of IoT-based Smart Systems – Sensors, Actuators, and Controllers – Smart System Design Challenges. **(9)**

EMBEDDED PLATFORMS AND CONNECTIVITY: Sensors and Transducers – Sensor Classification – Signal Conditioning – Actuators and Interfacing – Microcontrollers and Embedded Platforms – Arduino, Raspberry Pi, ESP32 Overview – Data Acquisition Systems – Wireless Sensor Networks – Communication Technologies: Wi-Fi, Bluetooth, ZigBee, RFID, NFC – MQTT and CoAP Protocols – Edge Devices and Smart Gateways. **(9)**

INTERNET OF THINGS AND CLOUD INTEGRATION: Internet of Things Architecture – IoT Functional Blocks – IoT Communication Models – Cloud Computing for Smart Systems – Edge and Fog Computing – Data Storage and Device Management – IoT Middleware – Smart System Networking – Security and Privacy Issues – Authentication and Access Control. . **(9)**

INTELLIGENT DATA ANALYTICS FOR SMART SYSTEMS: Data Collection and Preprocessing – Real-Time Data Analytics – Machine Learning for Smart Systems – Supervised and Unsupervised Learning – Predictive Analytics – Intelligent Decision Making – AI-enabled Smart Systems – Stream Processing – Visualization Techniques. **(9)**

APPLICATIONS AND INDUSTRIAL CASE STUDIES: Smart Grid and Energy Management – Intelligent Transportation Systems – Smart Manufacturing and Industry 4.0 – Smart Healthcare Systems – Smart Agriculture – Smart Buildings and Smart Cities. **(9)**

TOTAL L:45 periods

TEXT BOOKS

1. Bahga Arshdeep and Madisetti Vijay, *'Internet of Things: A Hands-On Approach'*. Universities Press, 2015.
2. Rajkumar R., Lee Insup, Sha Ling and Stankovic John, '*Cyber-Physical Systems: Foundations, Principles and Applications*'. Elsevier, 2017.
3. Mahmood Zaigham, '*Smart Systems and IoT: Innovations in Computing'*. CRC Press, 2020.

REFERENCES

1. Vermesan Ovidiu and Friess Peter, '*Building the Internet of Things: Implement New Business Models, Disrupt Competitors, Transform Your Industry'*. Wiley, 2016.
2. Valvano Jonathan W., '*Embedded Systems: Real-Time Interfacing to Arm Cortex-M Microcontrollers'*. CreateSpace Independent Publishing Platform, 2019.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the architecture and components of smart systems and cyber physical systems.	**K2**
CO2	Apply sensors, embedded platforms, and communication technologies in smart applications.	**K3**
CO3	Analyze different tools and techniques to enhance the IoT architectures and cloud-enabled smart systems.	**K4**
CO4	Design smart system applications in industrial and societal domains.	**-**

COs–POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1												2	2
CO2	3											3	3
CO3		2										2	2
CO4			2	2	2							2	2
	3	**2**	**2**	**2**	**2**							**3**	**3**

1–low, 2–medium, 3–high

25CSP18 DATA SCIENCE ESSENTIALS

3 0 0 3

DATA SCIENCE PROCESS: Data science – benefits and uses – Facets of data – Defining research goals and creating a project charter – Retrieving data – Cleansing, integrating and transforming data – Exploratory data analysis – Build the model – Presenting findings and building applications. **(9)**

DESCRIBING DATA: Types of data – Types of variables – Describing data with averages – mean, median, mode – Describing variability – range, variance, standard deviation, Degrees of freedom, Interquartile range, describing data with tables and graphs. **(9)**

DESCRIBING RELATIONSHIPS: Correlation – Scatter plots – Correlation coefficient for quantitative data – Computational formula for correlation coefficient – Regression analysis – A regression line, Least squares regression line, Standard error of estimate, Multiple regression equations**.** **(9)**

LIBRARIES FOR DATA WRANGLING: Introduction to data wrangling – Basics of numpy arrays – Aggregations – Computations on arrays – comparisons, fancy indexing – Data manipulation with pandas – Data indexing, selection and operating on data, handling missing data**.** **(9)**

DATA VISUALIZATION: Principles of data visualization, Types of visualizations – Bar charts, Histograms, Scatter plots, Density and contour plots, Heatmaps, using visualization tools – Matplotlib, Visualization with seaborn – Seaborn versus Matplotlib, Exploring seaborn plots**.** **(9)**

Total L: 45 periods

TEXT BOOKS:

1. David Cielen, Arno D. B. Meysman, and Mohamed Ali, '*Introducing Data Science*'. Manning Publications, 2016.
2. Robert S. Witte and John S. Witte, '*Statistics*'. 11th edition, Wiley Publications, 2017.

REFERENCES:

1. Jake VanderPlas, *'Python Data Science Handbook'*. O'Reilly, 2016.
2. Allen B. Downey, *'Think Stats: Exploratory Data Analysis in Python'*. Green Tea Press, 2014.
3. Foster Provost and Tom Fawcett, *'Data Science for Business'*. O'Reilly, 2013.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the stages of the data analytics process, types of data, and statistical measures used for describing datasets.	**K2**
CO2	Apply statistical methods for various data distribution for ensuring relationship among attributes.	**K3**
CO3	Analyze datasets using Python libraries by performing data wrangling, cleaning, and transformation.	**K4**
CO4	Evaluate data insights using appropriate visualization techniques and tools to support decision–making.	**–**

COs–POs & PSOs MAPPING

CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PSO1**	**PSO2**
CO1												3	3
CO2	3											3	3
CO3		3										3	3
CO4			2									2	2
	3	**3**	**2**									**3**	**3**

1–low, 2–medium, 3–high

25CSP16 DATA EXPLORATION AND VISUALIZATION

2 0 2 3

EXPLORATORY DATA ANALYSIS: EDA fundamentals – Steps in EDA – Data types: Numerical data – Discrete data – Continuous data – Categorical data – Measurement Scales: Nominal – Ordinal – interval – Ratio – Comparing EDA with classical and Bayesian analysis. **(6+6)**

DATA TRANSFORMATION: Transformation techniques: Performing data deduplication – replacing values – Discretization and binning. Introduction to missing data, handling missing data: Traditional methods – Maximum likelihood estimation. **(6+6)**

CORRELATION AND TIME SERIES ANALYSIS: Types of analysis: Univariate analysis – Bivariate analysis – Multivariate analysis. Time series analysis: Fundamentals of TSA – Characteristics of TSA – Time based indexing – Visualizing – Grouping – Resampling. **(6+6)**

STATISTICAL DATA VISUALIZATION: Visualizing Distributions – Histograms and Density Plots – Q-Q Plots – Skewed Distributions – Visualizing Proportions – Pie Charts – Visualizing Time Series – Individual Time Series – Multiple Time Series. **(6+6)**

VISUAL ANALYTICS: Visual Analytics – Visual Analytics vs Data Visualization – Visual Analytics Process – Visual Analytics Applications – Case Studies – Python Data Visualization Libraries – Tableau. **(6+6)**

LIST OF EXPERIMENTS:

1. Write R programs to read data from keyboard or text files and compute summary measures like arithmetic mean, median, mode, variance and standard deviation. Also read a set of X, Y values and find covariance and correlation.
2. Exploratory Data Analysis for Bivariate and Multivariate Structured Data.
3. Explore various variable and row filters in R for cleaning data. Apply various plot features in R on sample data sets and visualize.
4. Perform time series analysis and apply various visualization techniques.
5. Perform Data Analysis and representation on a Map using various Map data sets with Mouse Rollover effect, user interaction, etc.

Total L: 30 + P: 30 = 60 periods

TEXTBOOKS:

1. Suresh Kumar Mukhiya, Usman Ahmed, '*Hands-On Exploratory Data Analysis with Python*'. Packt Publishing, 1st Edition, 2020.
2. Claus O. Wilke, '*Fundamentals of Data Visualization*'. O'Reilly Publications, 2019.

REFERENCES:

1. Kieran Healy, '*Data Visualization: A Practical Introduction'*. Princeton University Press, 2018.
2. Eric Pimpler, '*Data Visualization and Exploration with R*'. GeoSpatial Training Services, 2018.
3. Catherine Marsh, Jane Ellitt, '*Exploring Data: An Introduction to Data Analysis for Social Scientists*'. Wiley Publications, 2nd Edition, 2008.
4. Matthew O. Ward, Georges Grinstein, Daniel Keim, '*Interactive Data Visualization: Foundations, Techniques, and Applications*'. 2nd Edition, CRC press, 2015.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Describe fundamental concepts and systematic approaches for analyzing and interpreting data in different environments.	**K2**
CO2	Apply appropriate methods to process, organize, and examine data for meaningful analysis.	**K3**
CO3	Analyze data characteristics and relationships to extract insights and support informed decision-making.	**K4**
CO4	Evaluate approaches for accurate and effective interpretation and presentation of data.	-

COs–POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1												3	3
CO2	3											3	3
CO3		3										3	3
CO4			2					2				2	2
	3	**3**	**2**					**2**				**3**	**3**

1–low, 2–medium, 3–high

25CSP20 BUSINESS ANALYTICS

2 0 2 3

INTRODUCTION TO BUSINESS ANALYTICS: Analytics and Data Science – Analytics Life Cycle – Types of Analytics – Business Problem Definition – Data Collection – Data Preparation – Hypothesis Generation – Modeling – Validation and Evaluation – Interpretation – Deployment and Iteration. **(6+6)**

BUSINESS INTELLIGENCE: Data Warehouses and Data Mart - Knowledge Management –Types of Decisions - Decision Making Process - Decision Support Systems – Business Intelligence –OLAP – Analytic functions. **(6+6)**

BUSINESS FORECASTING: Introduction to Business Forecasting and Predictive analytics - Logic and Data Driven Models –Data Mining and Predictive Analysis Modelling –Machine Learning for Predictive analytics. **(6+6)**

HR & SUPPLY CHAIN ANALYTICS: Human Resources – Planning and Recruitment – Training and Development - Supply chain network - Planning Demand, Inventory and Supply – Logistics – Analytics applications in HR & Supply Chain - Applying HR Analytics to make a prediction of the demand for hourly employees for a year. **(6+6)**

MARKETING & SALES ANALYTICS: Marketing Strategy, Marketing Mix, Customer Behaviour –selling Process – Sales Planning – Analytics applications in Marketing and Sales - predictive analytics for customers' behaviour in marketing and sales. **(6+6)**

LIST OF EXPERIMENTS

1. Perform numerical, statistical operations and data import/export operations for different file formats
2. Perform Z-test, T-test & ANOVA
3. Perform data pre-processing operations i) Handling Missing data ii) Normalization
4. Perform dimensionality reduction operation using PCA, KPCA & SVD
5. Perform bivariate and multivariate analysis on the dataset.

Total L: 30 + P: 30 = 60 periods

TEXT BOOKS:

1. James R. Evans, '*Business Analytics'*. 3rd edition, Pearson Education, 2020.
2. R. N. Prasad & Seema Acharya, '*Fundamentals of Business Analytics'*. 2nd edition, Wiley India, 2016.

REFERENCES:

1. Philip Kotler, Kevin Lane Keller & Alexander Chernev, '*Marketing Management'*. 16th edition, Pearson Education, 2022.
2. V. S. P. Rao, *'Human Resource Management: Text and Cases'*. 4th edition, Excel Books, 2020.
3. B. Mahadevan, '*Operations Management: Theory and Practice'*. 4th edition, Pearson Education, 2024.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the fundamentals of business analytics techniques used for business decision-making.	**K2**
CO2	Apply business analytics methods, predictive modeling techniques, forecasting approaches and machine learning concepts to solve business problems and support organizational decision-making.	**K3**
CO3	Analyze the effectiveness of analytics applications in marketing, sales, human resources and supply chain management to improve business performance and predict future outcomes.	-

COs–POs & PSOs MAPPING

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

1–low, 2–medium, 3–high

25ADP09 NATURAL LANGUAGE PROCESSING

2 0 2 3

INTRODUCTION: Foundations of natural language processing – Language Syntax and Structure- Text Preprocessing and Wrangling – Text tokenization – Stemming – Lemmatization – Removing stop-words – Feature Engineering for Text representation – Bag of Words model- Bag of N-Grams model – TF-IDF model. **(6+6)**

WORD EMBEDDINGS AND NEURAL NETWORKS Feedforward Neural Networks for NLP - Vector Semantics and Embeddings -Word Embeddings - Word2Vec model – Glove model – FastText model – Overview of Deep Learning models – RNN – Long Short-Term Memory (LSTM) and Gated Recurrent Units (GRU) - Bidirectional RNNs and Deep RNNs - Limitations of recurrent architectures – Transition from recurrent models to transformers. **(6+6)**

TRANSFORMERS AND ADVANCED LANGUAGE MODELS: Transformer Architecture – Self-Attention Mechanism – Multi-Head Attention – Positional Encoding – BERT and GPT (Introduction) – Architecture and Working – Transfer learning and fine-tuning pretrained models – Transformer-based NLP pipelines. **(6+6)**

NLP TASKS AND APPLICATIONS: Text Classification and Sentiment Analysis – Named Entity Recognition (NER) – Question Answering Systems – Text Summarization and Machine Translation – Text Generation – Chatbots and Conversational AI – Evaluation Metrics: Accuracy, Precision, Recall, F1-Score, BLEU, ROUGE. **(6+6)**

EMERGING TRENDS AND ETHICS IN NLP: Next-Generation Pre-trained Models: BART – T5 – XLNet – Overview of Prompt-Based Learning – GPT-3 and Few-shot Learning – Adapter and Prefix Tuning – Multilingual NLP: SentencePiece Tokenization – mBART – XLM-R – BLEU Score – Fairness and Ethics in NLP – Bias in Language Models – Responsible Use of NLP Tools. **(6+6)**

LIST OF EXPERIMENTS

1. Text Preprocessing with BoW & TF-IDF Classification
2. Implementation of Word2Vec & GloVe Embeddings
3. BERT Fine-Tuning for Sentiment Analysis & NER
4. Implementation of MLP & CNN for Document Classification
5. Implementation of Seq2Seq Neural Machine Translation with Attention and BLEU Evaluation

Total L: 30 + P: 30 = 60 periods

TEXTBOOKS:

1. Daniel Jurafsky and James H. Martin, *'Speech and Language Processing'*. 3rd edition, Prentice Hall, 2023.
2. Dipanjan Sarkar, '*Text Analytics with Python: A Practical Real-World approach to Gaining Actionable insights from your data*'. APress,2018.

REFERENCES:

1. Delip Rao and Brian McMahan, *'Natural Language Processing with PyTorch'.* O'Reilly Media, 2019.
2. Yoav Goldberg, *'Neural Network Methods for Natural Language Processing'.* Morgan & Claypool Publishers, 2017.
3. Delip Rao and Brian McMahan, *'Natural Language Processing with PyTorch'.* O'Reilly Media, 2019.
4. Thomas Wolf et al., *'Transformers for Natural Language Processing'.* 2nd edition, Packt Publishing, 2022.
5. Sebastian Ruder, *'NLP Progress'.* Online Resource.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the foundational concepts of natural language structure, syntax, and statistical text processing techniques.	**K2**
CO2	Apply neural network architectures and word embedding techniques to represent and model textual data.	**K3**
CO3	Analyze the architecture and mechanisms of Transformer-based language models and their role in transfer learning.	**K4**
CO4	Evaluate the performance and Develop solutions for NLP tasks such as sentiment analysis, text classification, question answering, and machine translation using state-of-the-art models.	-

COs–POs & PSOs MAPPING

CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PSO1**	**PSO2**
CO1												3	3
CO2	3											3	3
CO3		3										3	3
CO4			2						2			3	3
	3	**3**	**2**						**2**			**3**	**3**

1–low, 2–medium, 3–high

25CSP32 GENERATIVE LANGUAGE MODELS

3 0 0 3

LARGE LANGUAGE MODELS (LLM) AND ITS ARCHITECTURES: Evolution – LLM - Foundation Models – Ecosystem - Development Pipeline – Applications-Language Modeling Fundamentals – Tokenization – Embeddings - Transformer Architecture - Self-Attention Mechanism - Multi-Head Attention - Positional Encoding - Encoder-only Models - Decoder-only Models-Encoder–Decoder Models - Scaling Laws. **(9)**

LLMS IN PRACTICE AND PROMPTING: Selecting an LLM - Local vs Cloud Models - Running Open-source Models-Model APIs-Hugging Face Ecosystem- Inference-Temperature-Top-k and Top-p Sampling-Context Window-Cost Estimation - Prompt Engineering: Fundamentals - Anatomy of a Prompt - Zero-shot, One-shot, Few-shot, Chain-of-Thought and Role Prompting - Structured Outputs - Prompt Templates and Evaluation. **(9)**

RETRIEVAL-AUGMENTED GENERATION(RAG) AND FRAMEWORKS: Hallucination Problem - Knowledge Retrieval – Architecture - Document Processing - Chunking Techniques -Embedding Models - Vector Databases - Similarity Search – Pipeline - LangChain and LlamaIndex - Overview -Components-Chains-Memory-Tools- Agents- Data Connectors-Index Types- Query Engines - Prompt and Chat Prompt – Templates - Output Parsers - Runnable Interfaces – Sequential and Conversational Chains - Memory Management - Streaming Responses. **(9)**

INDEXES, RETRIEVERS, ADVANCED RAG AND AGENTS: Data Collection and Data Cleaning - Text Splitting – Metadata -Embedding Generation - Vector Stores - Retriever Types - Hybrid Search - Compression Retrieval - Evaluation Metrics - Multi-query, Parent-Child and Recursive Retrieval - Re-ranking - Query Transformation - Fusion Retrieval - Graph and Agentic RAG – Evaluation - AI Agents – Architectures - Tools - Planning - Function Calling- Multi-Agent Systems - Autonomous Workflows - Human-in-the-loop. **(9)**

FINE-TUNING, DEPLOYMENT AND OPTIMIZATION: Full Fine-tuning – PEFT – LoRA – QloRA - Instruction Tuning – RLHF -Dataset Preparation - Model Evaluation - Deployment Strategies – Quantization – Pruning – Distillation - GPU Optimization - Model Serving – Monitoring – Guardrails – Security - Production Best Practices. **(9)**

Total L: 45 periods

TEXTBOOKS

1. Louis-François Bouchard and Louie Peters, '*Building LLMs for Production: Enhancing LLM Abilities and Reliability with Prompting, Fine-Tuning, and RAG'*. O'Reilly Media, 2024.
2. Alammar, Jay, Grootendorst, Maarten, '*Hands-On Large Language Models: Language Understanding and Generation'*. O'Reilly Media, 2024

REFERENCES

1. Sebastian Raschka, '*Build a Large Language Model from Scratch*'. Manning Pubns Co, 2024.
2. Radi, Mouad, '*Introduction to Prompt Engineering: Prompt-based Systems Examples. N.p*'. Amazon Digital Services LLC - Kdp, USA, 2023.
3. Hunter, Nathan, '*The Art of Prompt Engineering with ChatGPT: A Hands-on Guide'*. Nathan Hunter, United Kingdom, 2023.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the concepts and architectures of LLMs, prompting, RAG, and fine-tuning techniques	**K2**
CO2	Apply prompt engineering techniques, RAG frameworks, and LLM tools to develop intelligent language processing applications	**K3**
CO3	Analyze the effectiveness of prompting, RAG, fine-tuning, and deployment strategies in developing efficient LLM applications	**K4**
CO4	Implement end-to-end LLM-based solutions by integrating prompt engineering, retrieval mechanisms, AI agents, and deployment techniques	**-**

COs-POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	2											2	2
CO2	3											3	3
CO3	2	2										2	2
CO4	2	2	2		2			2				2	2
	3	**2**	**2**		**2**			**2**				**2**	**2**

1-low, 2-medium, 3-high

25CSP23 GENERATIVE AI

2 0 2 3

SLM & LLM: Concept of Language Modeling – SLMs and LLMs – Generative AI and Large Language Models – Usage of LLMs – Training LLMs – Generative AI & Evolution – Types of Generative AI Models – Challenges and Limitations of Generative AI – Applications – Case Study: Evolution of GPT Models. **(6+6)**

GENERATIVE MODELLING: Generative Model vs Discriminative Models – Rise of Generative Modeling – Generative Modeling Framework – Probabilistic Generative Models – Naive Bayes Model – Deep Generative Models – VAEs, GANs, Diffusion Models. **(6+6)**

PROMPT ENGINEERING: Fundamentals of Prompts – Types of Prompts: Zero-shot, Few-shot, Chain of Thought, ReAct – Creating Effective Prompts – Prompt Optimization Techniques – Ethics and Bias in Prompting – Tools and Platforms – Testing and Evaluation. **(6+6)**

AGENTIC AI: Agentic Systems – Self Governance, Autonomy, Agency – Characteristics of Intelligent agents – Architecture – Multi agent systems – Knowledge representation – Reasoning – Decision making – Planning Algorithms – CWD model – Communication and Collaboration. **(6+6)**

APPLICATIONS: Personalized Learning – Healthcare Transformation – Reinventing Customer Engagement through Intelligent Systems – AI-Assisted Document Creation and Review – Coding and Programming – Data Insights – Generative AI in Media and Entertainment – Marketing – Advertisement – Retail. **(6+6)**

LIST OF EXPERIMENTS:

1. Compare outputs from a Small Language Model and a Large Language Model for:
   - Text summarization
   - Question answering
   - Content generation
2. Build a basic Variational Autoencoder (VAE) for image reconstruction.
3. Develop Chain-of-Thought prompts for solving mathematical reasoning tasks.
4. Build a multi-agent communication simulation.
5. Generate marketing content using LLM prompts.

Total L: 30 + P: 30 = 60 periods

TEXTBOOKS:

1. Biswas, Anjanava, and Talukdar, Wrick. '*Building Agentic AI Systems: Create Intelligent, Autonomous AI Agents that Can Reason, Plan, and Adapt*'. N.p., Packt Publishing, 2025.

REFERENCES:

1. Anand Vemula, '*Mastering Prompt Engineering for Generative AI: Unlocking the Full Potential of AI Technology'*. Amazon Digital Services, 2024.
2. Marr, Bernard. '*Generative AI in Practice: 100+ Amazing Ways Generative Artificial Intelligence is Changing Business and Society*'. United Kingdom, Wiley, 2024.
3. Vladimir Geroimenko, '*The Essential Guide to Prompt Engineering*'. Springer Nature Switzerland,2025.
4. Numa Dhamani, Maggie Engler, *'Introduction to Generative AI'*. Manning Publications, 2024.
5. Jebara, Tony, *'Machine Learning: Discriminative and Generative'*. Switzerland, Kluwer Academic, 2012.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain and explain the fundamental concepts of generative AI, language models, and intelligent agents, including their evolution, architectures, and key characteristics.	**K2**
CO2	Apply generative modeling techniques, prompt engineering strategies, and AI tools to develop effective solutions for diverse problem domains.	**K3**
CO3	Analyze and evaluate the performance, limitations, ethical considerations, and biases of generative AI models and agent–based systems in real–world scenarios.	**K4**
CO4	Design and develop intelligent applications by integrating large language models, agentic systems, and generative AI techniques across various domains.	**–**

COs–POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1												2	2
CO2	3											3	3
CO3		2										2	2
CO4			3									3	3
	3	**2**	**3**									**3**	**3**

1–low, 2–medium, 3–high

25CSP13 EDGE AI

3 0 0 3

INTRODUCTION TO EDGE COMPUTING AND AI: Fundamentals of edge computing: Introduction: Need, Key techniques, Benefits – Systems paradigms of edge computing – Edge computing frameworks – Value scenarios for edge computing – Edge computing system architectures – Applications – Challenges and opportunities. Fundamentals of artificial intelligence: Artificial intelligence and Deep learning – Neural networks in deep learning – Deep reinforcement learning – Potential DL libraries for edge. **(9)**

EDGE AI COMPUTING TOOLS: Role in edge computing: A high-level hardware hierarchy of edge computing paradigm. Virtualization: Virtual machine and container – Network Virtualization. Introduction to DevOps: Understanding the history and evolution – Overview of the benefits and challenges – Introduction to DevOps tools and practices. CI/CD: tools such as Jenkins – Travis CI – CircleCI. **(9)**

INFERENCE AND TRAINING IN EDGE AI: Artificial intelligence inference in edge: Optimizing AI models in Edge: General method – Edge device – Overview and benefits – Introduction to Open Neural Network Exchange (ONNX) format and its advantages – Understanding NVIDIA TensorRT format and its optimizations for inference Segmentation of AI Model – Segmentation of AI Model, Early Exit of Inference (EEoI) – Sharing of AI Computation. Artificial Intelligence Training at Edge: Distributed training at edge, Federated learning (FL) at edge. **(9)**

MOBILE EDGE AI: Overview – Edge inference: On-device inference – Computation offloading– Server-based edge inference – Device-edge joint inference – Edge training: Data partition based – Model partition based – Coded computing. **(9)**

ARTIFICIAL INTELLIGENCE APPLICATIONS ON EDGE: Real-time video analytic – Autonomous Internet of Vehicles (IoVs) – Intelligent Manufacturing – Smart Home and City – Urban Healthcare – Urban Energy Management – Manufacturing – Transportation and traffic. **(9)**

Total L: 45 Periods

TEXTBOOKS:

1. Wang, X., Han, Y., Leung, V. C., Niyato, D., Yan, X., & Chen, X., '*Edge AI: Convergence of edge computing and artificial intelligence*', Singapore: Springer, 2020.
2. Yuanming Shi, Kai Yang, Zhanpeng Yang, Yong Zhou., '*Mobile Edge Artificial Intelligence Opportunities and Challenges'*, Elsevier Science publication, 2021

REFERENCES:

1. Jie Cao, Quan Zhang, Weisong Shi, '*Edge Computing: A Primer'*, Springer International Publishing, 2021.
2. Liu, Deyin; Chen, Xu; Zhou, Zhi; Ling, Qing, HierTrain., '*Fast Hierarchical Edge AI Learning with Hybrid Parallelism in Mobile-EdgeCloud Computing'*, IEEE Open Journal of the Communications Society, 634-645.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the fundamental concepts and recent advancements in modern computing technologies.	**K2**
CO2	Analyze different methodologies and tools used for designing and managing intelligent computing systems.	**K3**
CO3	Apply computational techniques to solve complex real-world problems effectively.	**K4**
CO4	Evaluate the performance and impact of emerging technologies in industrial and societal applications.	**-**

COs–POs & PSOs MAPPING

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

1–low, 2–medium, 3–high

25ADP05 TEXT AND SPEECH ANALYSIS

2 0 2 3

INTRODUCTION: Introduction to Natural Language Processing (NLP) and Levels of NLP Regular Expression Basic Text processing- Text normalization Vector Semantics and embedding: Lexical Semantics, Vector Semantics, Words and Vectors - Pointwise Mutual Information, N-gram Language Models: N-grams, Smoothing. **(6+6)**

PARTS OF SPEECH AND NAMED ENTITIES: Parts of Speech Tagging, Hidden Markov Model, Conditional Random Fields. Constituency Grammars: Constituency, Context-Free Grammars, Dependency Parsing: Dependency Relations, Dependency Formalism, Neural Dependency Parser. Logical Representations of Sentence Meaning, Word Sense and Word Net. **(6+6)**

APPLICATIONS OF TEXT AND NLP: Naive Bayes and Sentiment Analysis: Naive Bayes for text classification, Information Extraction, and Relationship Extraction. Learning Architectures for Sequence Processing: Recurrent Neural Networks for text classification - Long Short-Term Memory (LSTM). **(6+6)**

AUTOMATIC SPEECH RECOGNITION: Speech Sounds and Phonetic Transcription, Articulatory Phonetics, Acoustic Phonetics and Signals - Phonetic Resources, Prosody-Automatic Speech Recognition (ASR) Task - Feature Extraction: Log Mel Spectrum -Speech Recognition Architecture: Introduction: Gaussian Mixture Model Connectionist Temporal Classification (CTC) - ASR Evaluation: Word Error Rate. **(6+6)**

TEXT-TO-SPEECH: Text-to-Speech (TTS) Preprocessing: Text normalization, TTS: Spectrogram Prediction - TTS: Vocoding - TTS Evaluation. **(6+6)**

LIST OF EXPERIMENTS

1. Implement text preprocessing techniques using Python and NLTK for tokenization, word pattern matching, vocabulary analysis, frequency distribution, collocations, and bigrams.
2. Access and analyze text corpora using NLTK and develop a program to identify the 50 most frequent non-stop words from a given text.
3. Implement the Word2Vec model for generating word embeddings and semantic similarity analysis.
4. Use a transformer for implementing classification.
5. Convert text to speech and find accuracy.

Total L: 30 + P: 30 = 60 periods

TEXT BOOKS:

1. Jurafsky, D. and J. H. Martin, '*Speech and language processing: An Introduction to Natural Language Processing'.* Computational Linguistics, and Speech Recognition, 2021.
2. John Atkinson-Abutridy, '*Text Analytics: An Introduction to the Science and Applications of Unstructured Information Analysis'.* CRC Press, 2022.

REFERENCES:

1. Jim Schwoebel, NeuroLex, 'I*ntroduction to Voice Computing in Python'.* 2018.
2. Lawrence R. Rabiner, Ronald W. Schafe, 'T*heory and Applications of Digital Speech Processing'.* 1st edition Pearson, 2010.
3. Srinivasa-Desikan, Bhargav, '*Natural Language Processing and Computational Linguistics: A practical guide to text analysis with Python Gensim, spaCy, and Keras'*. Packt Publishing Ltd, 2018.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Understand the foundations of data mining by identifying various data types, data preprocessing and data transformation techniques.	**K2**
CO2	Apply the basic principles and techniques to discover the knowledge to collect, clean pre-process.	**K3**
CO3	Analyze OLAP and data warehousing concepts including data warehouse models, OLAP operations, and data cube computation methods to support analytical processing.	**K4**
CO4	Demonstrate the ability to mine patterns from data using frequent itemset mining, sequential pattern mining, and constraint-based methods, and evaluate their applicability in real-world scenarios.	**-**

COs–POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1												2	2
CO2	3										3	3	3
CO3		3									3	3	3
CO4			2					2	2		3	3	3
	3	**3**	**2**					**2**	**2**		**3**	**3**	**3**

1–low, 2–medium, 3–high

25ADP11 IMMERSIVE TECHNOLOGIES

3 0 0 3

INTRODUCTION: Evolution of AR and VR – Differences between AR, VR, and Mixed Reality (MR) – Components of AR and VR Systems – Key Terminology: Immersion, Presence, Latency, Field of View – VR Devices: Head-Mounted Displays (HMDs), Controllers, Sensors – AR Devices: Mobile Devices, Smart Glasses, HoloLens – Tracking Techniques: Optical Tracking, Inertial Tracking, Sensor Fusion. **(9)**

AUGMENTED REALITY: System structure of Augmented Reality - Key technologies in AR - Marker-less tracking for Augmented Reality - Enhancing interactivity in AR environment - General solution for calculating geometric & illumination consistency in the augmented environment - Evaluating AR systems. **(9)**

VIRTUAL REALITY: Introduction – modern VR experiences – Bird's Eye View – Geometry of Virtual Worlds – Virtual Perception – Visual Rendering – Tracking – Evaluating VR Systems and Experiences – Types of Virtual Reality – Human-Computer Interaction in VR. **(9)**

AUGMENTED AND MIXED REALITY: Taxonomy, technology and features of augmented reality - difference between AR and VR - Challenges with AR - AR systems and functionality - Augmented Reality methods - Visualization techniques for Augmented Reality. **(9)**

AR/VR APPLICATIONS: AR & VR in Retail – AR & VR in Tourism – AR & VR in Education – AR & VR Medical Applications – VR and Media – AR & VR in Agriculture – Future Trends: Metaverse, AI Integration in AR/VR, Extended Reality Overview. **(9)**

Total L: 45 periods

TEXTBOOKS:

1. Alan B. Craig, '*Understanding Augmented Reality Concepts and Applications*'. Elsevier Science, 2013.
2. Steven M. LaValle, '*Virtual Reality*'. Cambridge University Press, 2023.
3. C.Burdea & Philippe Coiffet, '*Virtual Reality Technology*'. 2nd edition Gregory, John Wiley & Sons, Inc., 2013.

REFERENCES:

1. M. Claudia tom Dieck, Timothy Jung, '*Augmented Reality and Virtual Reality the Power of AR and VR for Business*'. Springer International Publishing,2019.
2. Jitendra Kumar Verma, Sudip Paul, '*Advances in Augmented Reality and Virtual Reality*'. Springer Nature Singapore, 2022.
3. Nawaz Mohamudally, '*State of the Art Virtual Reality and Augmented Reality Knowhow*'. IntechOpen, 2018.
4. Steve Benford, Gabriella Giannachi, '*Performing Mixed Reality*'. MIT Press, 2022.
5. Spencer Grey, '*Mind-Melding Unity and Blender for 3D Game Development*'. Packt Publishing, 2021.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Describe the basic concepts, devices, and tracking methods used in AR, VR, and MR systems.	**K2**
CO2	Apply AR and VR technologies to create simple interactive experiences with development tools and frameworks to build basic AR and VR applications.	**K3**
CO3	Analyze the working and performance of AR and VR systems for different applications.	**K4**
CO4	Examine innovative AR and VR solutions using modern technologies like AI and XR.	**-**

COs–POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1												3	3
CO2	3											3	3
CO3		3										3	3
CO4			2						2			3	3
	3	**3**	**2**						**2**			**3**	**3**

1–low, 2–medium, 3–high

25CSP22 COMPUTER VISION AND IMAGE PROCESSING

2 0 2 3

IMAGE FORMATION AND PROCESSING: Computer vision – Geometric primitives and transformations – Photometric image formation – The digital camera – Point operators – Linear filtering – Nneighborhood operators – Pyramids and wavelets – Global optimization. **(6+6)**

FEATURE DETECTION, MATCHING AND SEGMENTATION: Points and patches – Edges – Lines – Segmentation – Active contours – Split and merge – Mean shift and mode finding – Normalized cuts – Graph cuts and energy-based methods – From traditional vision to deep learning. **(6+6)**

FEATURE–BASED ALIGNMENT: 2D and 3D feature based alignment – Pose estimation – Geometric intrinsic calibration – Triangulation – Two–frame structure from motion – Factorization – Bundle adjustment – Constrained structure. **(6+6)**

3D RECONSTRUCTION: Shape from X – Active range finding – Surface representations – Point-based representations – Volumetric representations – Model-based reconstruction. **(6+6)**

IMAGE–BASED RENDERING: View interpolation Layered depth images – Light fields and Lumigraphs – Environment mattes – Video-based rendering – Instance recognition – Category recognition – Context and scene understanding. **(6+6)**

LIST OF EXPERIMENTS

1. OpenCV Installation and performing basic Image Processing operations (loading images, Cropping, Resizing, Thresholding)
2. Image Enhancement (Histogram equalization, Image smoothing, Edge Detection)
3. Image segmentation and Feature Detection
4. 3D Reconstruction – Creating Depth map from stereo images
5. Image–Based Rendering and View Interpolation

Total L: 30 + P: 30 = 60 periods

TEXTBOOKS:

1. Richard Szeliski, *'Computer Vision: Algorithms and Applications'*. 2nd edition, Springer– Texts in Computer Science, 2022.
2. D. A. Forsyth, J. Ponce, *'Computer Vision: A Modern Approach'*. 2nd edition, Pearson Education, 2015.

REFERENCES:

1. Richard Hartley and Andrew Zisserman, '*Multiple View Geometry in Computer Vision'*. 2nd edition, Cambridge University Press, March 2004.
2. Christopher M. Bishop; '*Pattern Recognition and Machine Learning'*. Springer, 2006.
3. E. R. Davies, '*Computer and Machine Vision'*. Fourth Edition, Academic Press, 2012.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain basics in image processing and computer vision.	**K2**
CO2	Apply 2D and 3D feature–based image alignment, segmentation and motion estimations.	**K3**
CO3	Aanalyze 3D image reconstruction and representations.	**K4**
CO4	Implement image processing techniques in OpenCV.	-

COs–POs & PSOs MAPPING

CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PSO1**	**PSO2**
CO1													
CO2	3											3	3
CO3		3										3	3
CO4			3		3							3	3
	3	**3**	**3**		**3**							**3**	**3**

1–low, 2–medium, 3–high

25CSP17 SOFT COMPUTING

2 0 2 3

INTRODUCTION TO SOFT COMPUTING AND FUZZY LOGIC: Introduction - Fuzzy Logic - Fuzzy Sets, Fuzzy Membership Functions, Operations on Fuzzy Sets, Fuzzy Relations, Operations on Fuzzy Relations, Fuzzy Rules and Fuzzy Reasoning, Fuzzy Inference Systems. **(6+6)**

NEURAL NETWORKS: Supervised Learning Neural Networks – Perceptrons - Backpropagation -Multilayer Perceptrons – Unsupervised Learning Neural Networks – Kohonen Self-Organizing Networks. **(6+6)**

GENETIC ALGORITHMS: Chromosome Encoding Schemes -Population initialization and selection methods – Evaluation function - Genetic operators- Cross over – Mutation - Fitness Function – Maximizing function. **(6+6)**

NEURO FUZZY MODELING: ANFIS architecture – hybrid learning – ANFIS as universal approximator – Coactive Neuro fuzzy modeling – Framework – Neuron functions for adaptive networks – Neuro fuzzy spectrum – Analysis of Adaptive Learning Capability. **(6+6)**

APPLICATIONS: Modeling a two input sine function - Printed Character Recognition – Fuzzy filtered neural networks – Plasma Spectrum Analysis – Hand written neural recognition - Soft Computing for Color Recipe Prediction. **(6+6)**

LIST OF EXPERIMENTS

1. Programming exercise on classification with a discrete perceptron
2. Implementation of XOR with backpropagation algorithm
3. Implementation of self-organizing maps for a specific application
4. Programming exercises on maximizing a function using Genetic algorithm
5. Implementation of two input sine function and three input non-linear function

Total L: 30 + P: 30 = 60 periods

TEXT BOOKS:

1. Jyh-Shing Roger Jang, Chuen-Tsai Sun, and Eiji Mizutani, '*Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence*'. 1st edition, Pearson India, 2015.
2. Himanshu Singh and Yunis Ahmad Lone, '*Deep Neuro-Fuzzy Systems with Python: With Case Studies and Applications from the Industry*'. 1st edition, Apress, 2020.

REFERENCES:

1. Roj Kaushik & Sunita Tiwari, '*Soft Computing: Fundamentals, Techniques and Applications*'. 1st edition, McGraw Hill, 2018.
2. S. Rajasekaran & G. A. Vijayalakshmi Pai, '*Neural Networks, Fuzzy Logic and Genetic Algorithms: Synthesis and Applications*'. 1st reprint edition PHI Learning, 2017.
3. Samir Roy & Udit Chakraborty, '*Introduction to Soft Computing: Neuro-Fuzzy and Genetic Algorithms'*. 1st reprint edition, Pearson, 2024.
4. S. N. Sivanandam & S. N. Deepa, 'Principles of Soft Computing'. 3rd edition, Wiley India, 2018.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the fundamental concepts of soft computing, fuzzy logic, fuzzy inference systems, and neural network architectures.	**K2**
CO2	Apply neural network, genetic algorithm and neuro-fuzzy modeling techniques to solve learning, classification and optimization problems.	**K3**
CO3	Analyze the performance and applicability of soft computing techniques and intelligent system applications.	**-**

COs–POs & PSOs MAPPING

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

1–low, 2–medium, 3–high

25ADP19 QUANTUM MACHINE LEARNING

3 0 0 3

INTRODUCTION: Qubits and Quantum States – Superposition and Entanglement – Quantum Gates and Circuits – Quantum Measurement – Classical Machine Learning Overview – Supervised and Unsupervised Learning – Linear Algebra and Probability for Quantum Computing. **(9)**

QUANTUM ALGORITHMS FOR MACHINE LEARNING: Grover's Search Algorithm – Quantum Fourier Transform – Quantum Phase Estimation – Quantum Principal Component Analysis (qPCA) – Quantum Support Vector Machines (QSVM) – Quantum k-Means Clustering – Speedups and Limitations of Quantum ML Algorithms. **(9)**

VARIATIONAL CIRCUITS AND HYBRID QUANTUM-CLASSICAL MODELS: Parametrized Quantum Circuits – Variational Quantum Eigensolver (VQE) – Quantum Approximate Optimization Algorithm (QAOA) – Cost Function Design – Hybrid Model Architectures – Training Challenges and Optimization Strategies – Gradient Estimation in Quantum Circuits. **(9)**

QUANTUM NEURAL NETWORKS AND DEEP LEARNING ARCHITECTURES : Quantum Perceptron – Quantum Convolutional Neural Networks (QCNN) – Quantum Boltzmann Machines – Quantum Generative Adversarial Networks (QGANs) – Transfer Learning in Quantum Models – Use Cases in Vision, Text, and Healthcare. **(9)**

IMPLEMENTATION TOOLS, TRENDS, AND ETHICS IN QUANTUM ML: Quantum Programming Frameworks: Qiskit, Cirq, PennyLane – Quantum Cloud Platforms: IBM Q, IonQ, Rigetti – Quantum Noise and Error Mitigation – Ethical Issues in Quantum AI – Scalability, Interpretability, and Explainability – Future Trends and Research Directions in Quantum Machine Learning. **(9)**

Total L: 45 periods

TEXTBOOKS:

1. Santanu Ganguly, '*Quantum Machine Learning: An Applied Approach'*. Springer, 2021.
2. Peter Wittek, '*Quantum Machine Learning: What Quantum Computing Means to Data Mining*'. Academic Press, 2014.
3. Michael A. Nielsen & Isaac L. Chuang, '*Quantum Computation and Quantum Information*'. Cambridge University Press, 10th Anniversary Edition, 2010.

REFERENCES:

1. Maria Schuld & Francesco Petruccione, '*Quantum Machine Learning: An Introduction'*. Springer, 2018.
2. Frank Zickert, '*Quantum Machine Learning with Python*'. Packt Publishing, 2022.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the core principles of quantum computing and their applicability in machine learning	**K2**
CO2	Apply quantum algorithms for machine learning tasks and variational circuits	**K3**
CO3	Analyze quantum models and train them using quantum hardware or simulators	**K4**
CO4	Evaluate challenges such as noise, decoherence, scalability, and ethical aspects of quantum ML systems	**-**

COs–POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1												3	3
CO2	3											3	3
CO3		3										3	3
CO4			2						2			2	2
	3	**3**	**2**						**2**			**3**	**3**

1–low, 2–medium, 3–high

25CSP24 NEURAL NETWORKS AND DEEP LEARNING

2 0 2 3

INTRODUCTION: Neural networks – Application scope of neural networks – Artificial neural network: An Introduction – Evolution of neural networks – Basic models of artificial neural network – Important terminologies of ANNs – Supervised learning network. **(6+6)**

ASSOCIATIVE MEMORY AND UNSUPERVISED LEARNING NETWORKS: Training algorithms for pattern association – Auto associative memory network – Hetero-associative memory network – Bidirectional associative memory (BAM) - Hopfield networks – Iterative auto-associative memory networks – Kohonen self – Organizing feature maps. **(6+6)**

THIRD-GENERATION NEURAL NETWORKS: Spiking neural networks – Convolutional neural networks – Deep learning neural networks – Extreme learning machine model – Convolutional neural networks: The Convolution operation – Motivation – Pooling – Variants of the basic convolution function. **(6+6)**

DEEP FEEDFORWARD NETWORKS: History of deep learning – A Probabilistic theory of deep learning – Gradient learning – Chain rule and Backpropagation – Regularization: Dataset augmentation – Noise robustness – Early stopping, bagging and dropout – batch normalization – VC dimension and neural nets. **(6+6)**

RECURRENT NEURAL NETWORKS: Recurrent neural networks: Introduction – Recursive neural networks – bidirectional RNNs – Deep recurrent networks – Complete auto encoder, Regularized autoencoder, Stochastic encoders and decoders. **(6+6)**

LIST OF EXPERIMENTS

1. Implement a perceptron in TensorFlow/Keras Environment.
2. Implement a Feed-Forward Network in TensorFlow/Keras.
3. Implement an Image Classifier using CNN in TensorFlow/Keras.
4. Perform Sentiment Analysis using RNN.
2. Implement an Autoencoder in TensorFlow/Keras.

Total L: 30 + P: 30 = 60 periods

TEXT BOOKS:

1. Ian Goodfellow, Yoshua Bengio, Aaron Courville, *'Deep Learning'*. MIT Press, 2016.
2. Francois Chollet, *'Deep Learning with Python'*. 2nd edition, Manning Publications, 2021.

REFERENCES:

1. Josh Patterson, Adam Gibson, '*Deep Learning: A Practitioner's Approach*'. O'Reilly Media, 2017.
2. Charu C. Aggarwal, *'Neural Networks and Deep Learning: A Textbook'*. 1st edition, Springer International Publishing, 2018.
3. Jojo Moolayil, *'Learn Keras for Deep Neural Networks'*. Apress, 2018.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Summarize the fundamentals, evolution, models, and key terminologies of Artificial Neural Networks.	**K2**
CO2	Implement associative memory models, unsupervised learning techniques, neural networks, and autoencoders to address pattern recognition, image, and natural language processing tasks.	**K3**
CO3	Analyze the design and training of deep feedforward networks.	**K4**
CO4	Present a technical seminar on deep neural networks applications by demonstrating domain understanding, and technical insights.	**–**

COs–POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1												3	3
CO2	3											3	3
CO3		3										3	3
CO4								2	2			2	2
	3	**3**						**2**	**2**			**3**	**3**

1–low, 2–medium, 3–high

25ADP15 RESPONSIBLE AI

3 0 0 3

INTRODUCTION: Need for Responsible AI: Ethical, legal, social, and environmental perspectives - Core principles: Fairness, Accountability, Transparency, Safety, Sustainability, Data and Model Privacy - Fairness and Proxy Features: Key parameters, confusion matrix, common accuracy matrics, fairness matrics, proxy features, methods to detect proxy features, Case studies of AI failures and unintended consequences. **(9)**

BIAS DETECTION AND MITIGATION: Types of bias in AI: Historical bias, Representation bias, Measurement bias, Aggregation bias, Statistical Parity Difference, Disparate Impact, Remove Bias from ML Model: Reweighting the data, Additive Counterfactual Fairness, Remove Bias from ML Output: Reject Option Classifier Tradeoff between fairness and accuracy, What-If Tool. **(9)**

SECURE AND TRUSTWORTHY AI SYSTEMS: Privacy-preserving techniques: Anonymization: K-Anonymity, L-Diversity and T-Closeness, Differential privacy, Differentially private ML Algorithm, Federated learning - Adversarial threats: Model stealing, Adversarial examples, Membership inference - Trust and transparency in AI systems: Interpretability, Explainability (XAI) - Interpretable Models: Linear Regression, Logistic Regression, Decision Tree - Tools for explainability and privacy: LIME, SHAP, Captum, TensorFlow Privacy. **(10)**

SUSTAINABLE AND GREEN AI: Introduction to Green AI: Definition, Red vs Green AI paradigm - Carbon footprint of large-scale AI models: GPT, BERT, Training and inference phases - Estimating emissions: Experiment tracking, Energy profiling tools, ML CO2 - Infrastructure and sustainability: GPU vs TPU, Data centres, Renewable energy use - Strategies for sustainable AI: Efficient model design, Pruning, Quantization, Transfer learning - Green AI tools: Code Carbon, ML CO2 Calculator, Energy monitoring tools **(9)**

ACCOUNTABILITY IN AI: Data Drift: Covariate Drift, Jensen-Shannon Distance, Wasserstein Distance, Stability Index, Concept Drift: Kolmorgorov-Smirnov Test, Brier Score, Page-Hinkley Test, Early Drift Detection Method, Hierarchical Linear Four Rate (HLFR), Future directions in Responsible AI: Human-in-the-loop AI and Ethics by design. **(8)**

Total L: 45 periods

TEXT BOOKS:

1. Agarwal, Sray., Mishra, Shashin, '*Responsible AI: Implementing Ethical and Unbiased Algorithms*'. Springer International Publishing, 2021.
2. Christoph Molnar, '*Interpretable Machine Learning*'. 1st edition, Lulu, 2019.

REFERENCES:

1. Dignum Virginia, '*Responsible Artificial Intelligence: How to Develop and Use AI in a Responsible Way*'. Springer International Publishing, 2019.
2. Ammanath, Beena, '*Trustworthy AI: A Business Guide for Navigating Trust and Ethics in AI*'. Wiley, 2022.
3. Gambelin, Olivia, '*Responsible AI: Implement an Ethical Approach in Your Organization*'. Kogan Page 2024.
4. Alloghani, Mohamed Ahmed, '*Artificial Intelligence and Sustainability*'. Springer Nature, 2023.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the fundamental principles of Responsible AI, including fairness, transparency, and sustainability, while identifying ethical risks and historical failures in AI systems.	**K2**
CO2	Apply bias detection metrics and mitigation techniques to ensure equity in machine learning datasets and model outputs.	**K3**
CO3	Analyze AI systems for security vulnerabilities and interpretability using privacy-preserving techniques and XAI tools.	**K4**
CO4	Evaluate the environmental impact and operational stability of AI models by assessing carbon footprints and designing monitoring strategies for data and concept drift.	**-**

COs–POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1												3	3
CO2	3											3	3
CO3		3										3	3
CO4					2							2	2
	3	**3**			**2**							**3**	**3**

1–low, 2–medium, 3–high

2 0 2 3

IMAGES IN DIFFERENT DOMAINS AND OPERATORS: Images in the Spatial Domain, Images in the Frequency Domain, Operators: Point - Local - Global, Three Procedural Components, Classes of Local Operators, Advanced Edge Detectors. **(6+6)**

IMAGE AND MOTION ANALYSIS: Basic Image Topology, Geometric 2D Shape Analysis, Image Value Analysis, Detection of Lines and Circles,3D-Motion and 2D-OpticalFlow, Algorithms: Horn–Schunck - Lucas–Kanade - BBPW, Performance Evaluation of Optical Flow Results. **(6+6)**

IMAGE AND VIDEO SEGMENTATION: Image Segmentation: Mean-Shift, Image Segmentations as an Optimization Problem, Video Segmentation and Segment tracking. **(6+6)**

OBJECT DETECTION IN IMAGES AND VIDEO: Post Processing methods, multi-frame methods using 3-D convolution, multi-frame methods using recurrent neural networks, Sparse feature propagation, multi-frame feature aggregation. **(6+6)**

APPLICATIONS OF MACHINE LEARNING FOR IMAGE AND VIDEO ANALYSIS: Face recognition using eigen faces, Face recognition using LBP, Face detection with Haar-Cascades in real-time video, Object detection with YOLO. Tracking moving object in video. **(6+6)**

LIST OF EXPERIMENTS

3. Develop a program to compute the T-pyramid and quad tree representation of an image using the homogeneity criterion of equal intensity.
4. Develop programs for the following geometric transformations on images:
   - Rotation
   - Scaling
   - Skewing
   - Affine Transformation using three pairs of corresponding points
   - Bilinear Transformation using four pairs of corresponding points.
5. Develop a program for object detection and recognition in digital images.
6. Develop a program for facial detection and facial recognition using image processing techniques.
7. Develop a program for motion analysis and event detection in video surveillance systems using moving edge analysis on image sequences.

TOTAL L: 30 + P: 30 = 60 periods

TEXT BOOKS

1. Reinhard Klette, '*Concise Computer Vision: An Introduction into Theory and Algorithms'*. Springer, 2014
2. Milan Sonka, Vaclav Hlavac, Roger Boyle, '*Image Processing, Analysis, and Machine Vision'*. 4th edition, Thomson Learning, 2013.
3. Vaibhav Verdhan,'*Computer Vision Using Deep Learning: Neural Network Architectures with Python, Keras and Tensorflow*'. 2021.

REFERENCES

1. Simon J. D. Prince, '*Computer Vision: Models, Learning, and Inference'.* Cambridge University Press, 2012.
2. D. A. Forsyth, J. Ponce, '*Computer Vision: A Modern Approach*'. Pearson Education, 2003

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Understand the basic concepts of Image and Video segmentation and object detection	**K2**
CO2	Apply the fundamentals of image processing and acquire knowledge of image acquisition, storage, and display technologies.	**K3**
CO3	Analyze different tools and techniques to enhance image quality, reduce noise, and extract information from images and videos.	**K4**
CO4	Design and implement image and video analysis algorithms using software tools.	**-**

COs–POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1												3	3
CO2	3											3	3
CO3		3										3	3
CO4			2	2	2							2	2
	3	**3**	**2**	**2**	**2**							**3**	**3**

1–low, 2–medium, 3–high

25CSP26 SOCIAL NETWORK ANALYSIS

3 0 0 3

SOCIAL NETWORK STRUCTURE AND MEASURES: Basics of network structure – Representing networks – Basic network structures and properties Subnetworks – Describing nodes and edges – Describing networks – Tie Strength. **(9)**

ENTITY RESOLUTION AND LINK PREDICTION: Modeling networks – Sampling methods – Egocentric network analysis – Link prediction – Entity resolution – Incorporating network data – Case Study. **(9)**

LOCATION-BASED SOCIAL INTERACTION AND ANALYSIS: Location technology – Mobile location sharing – Location-based social media analysis – Privacy and location-based social media – Social sharing and social filtering – Automated recommender systems – Analyzing public-sector social media-Case study. **(9)**

UNCOVERING BRAND ACTIVITY, POPULARITY AND EMOTIONS: Facebook brand page – Project planning – Analysis – Noun phrases – Detecting trends in time series – Uncovering emotions. **(9)**

SOCIAL MEDIA CONTENT ANALYSIS: Getting the data from Social Network – Sentiment analysis – Customized sentiment analysis – Named entity recognition – Combining NER and sentiment analysis – Trends Mining on GitHub – Data pull – Data processing – Data analysis. **(9)**

Total L: 45 periods

TEXT BOOKS:

1. Jennifer Golbeck, *'Analyzing the Social Web'*. Morgan Kaufmann; 1st edition 2013.
2. Siddhartha Chatterjee, Michal Krystyanczuk, *'Python Social Media Analytics'.* Packt publishers, 2017.

REFERENCES:

1. Charu C. Aggarwal, 'Social Network Data Analytics'. Springer, 2011.
2. Guandong Xu, Yanchun Zhang and Lin Li, *'Web Mining and Social Networking Techniques and applications'*. 1st edition, Springer, 2011.
3. Peter Mika, *'Social networks and the Semantic Web'*. 1st edition, Springer, 2007.
4. Borko Furht, *'Handbook of Social Network Technologies and Applications'*. 1st edition, Springer, 2010.
5. Dion Goh and Schubert Foo, *'Social information retrieval systems: emerging technologies and applications for searching the Web effectively'*. IGI Global snippet, 2008.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the fundamental concepts of social networks, network structures, measures, tie strength and network representation techniques.	**K2**
CO2	Apply network analysis techniques, entity resolution methods, link prediction models and location-based social media analytics to solve real-world problems.	**K3**
CO3	Analyze social interactions, recommendation systems, brand activities, trends and user behaviour using social network and social media data.	**K4**
CO4	Evaluate social media content using sentiment analysis named entity recognition, trend mining, and data analytics techniques to derive actionable insights.	**-**

COs–POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1												3	3
CO2	3											3	3
CO3		3										3	3
CO4					2							2	2
	3	**3**			**2**							**3**	**3**

1–low, 2–medium, 3–high

25CSP27 CLOUD COMPUTING TECHNOLOGIES

3 0 0 3

CLOUD FOUNDATIONS: CONCEPTS, MODELS, AND CHARACTERISTICS: Origins and Influences – Basic Concepts and Terminology – Goals and Benefits – Risks and Challenges – Roles and Boundaries: Cloud Provider, Cloud Consumer, Cloud Broker, Service Owner, Resource Administrator – Cloud Essential Characteristics – Cloud Delivery Models: IaaS, PaaS, SaaS – Cloud Deployment Models: Public Cloud, Private Cloud, Multi Cloud and Hybrid Cloud. **(9)**

VIRTUALIZATION AND CONTAINERIZATION TECHNOLOGIES IN CLOUD: Origins and Influences – Fundamentals of Virtualization and Containerization – Understanding Containers – Container Images – Multi-Container Types. Virtualization: Approaches – Properties of Full Virtualization – Conceptual Organization of VM Systems – Efficient Execution and Processor Privilege Levels – Extending Privileges and Levels of Trust – I/O and Virtual Devices – VM Migration – Live and Cold Migration. **(9)**

ADVANCED CLOUD MECHANISMS AND VIRTUAL MACHINE OPERATIONS: Cloud Infrastructure Mechanisms – Logical Network Perimeter – Virtual Server – Hypervisor – Cloud Storage Device: Levels, Network, Object and Database Storage Interfaces – Cloud Usage Monitor – Resource Replication – Automated Scaling Listener – Load Balancer – SLA Monitor – Pay-Per-Use Monitor – Audit Monitor – Failover System – Resource Cluster – Multi-Device Broker – State Management Database. **(9)**

CLOUD ARCHITECTURES: FUNDAMENTAL CONCEPTS AND ADVANCED STRATEGIES: Fundamental Cloud Architectures: Workload Distribution – Resource Pooling – Dynamic Scalability – Elastic Resource Capacity – Service Load Balancing – Cloud Bursting – Elastic Disk Provisioning – Redundant Storage – Multi Cloud – Hypervisor Clustering – Virtual Server Clustering – Load-Balanced Virtual Server Instances – Nondisruptive Service Relocation – Zero Downtime – Cloud Balancing – Resilient Disaster Recovery – Distributed Data Sovereignty – Resource Reservation – Dynamic Failure Detection and Recovery. **(9)**

CLOUD DELIVERY FRAMEWORKS: COST, QUALITY AND SERVICE LEVLEL AGREEMENTS: Cloud Provider Perspective: Building IaaS, PaaS, SaaS Environments. Cloud Consumer Perspective: Working with IaaS, PaaS, SaaS Environments. Cloud Metrics and Management: Business Cost Metrics – Cloud Usage Cost Metrics: Network, Server, Storage Device, Service Usage Costs – Cost Management Considerations – Service Quality – Availability – Reliability – Performance – Scalability – Resiliency Metrics – SLA Guidelines. **(9)**

Total L: 45 periods

TEXT BOOKS:

1. Thomas Erl, Eric Barcelo Monroy, *'Cloud Computing: Concepts, Technology, Security & Architecture'*. 2nd edition, Pearson Publishers, 2024.
2. Douglas E. Comer, *'The Cloud Computing Book: The Future of Computing Explained'*. 1st edition, CRC Press, Taylor & Francis Group, 2021.

REFERENCES:

1. Dan C. Marinescu, *'Cloud Computing: Theory and Practice'*. 3rd edition, Morgan Kaufmann Publications, 2022.
2. Rajkumar Buyya, Christian Vecchiola, S.TamaraiSelvi, *'Mastering Cloud Computing'*. 1st edition, TMGH Publications, 2017.
3. Rajkumar Buyya, James Broberg, Andrzej Goscinski, *'Cloud Computing: Principles and Paradigms'*. 1st edition, Wiley Publishers, 2016.
4. Kai Hwang, Geoffrey C Fox, Jack G Dongarra, *'Distributed and Cloud Computing, From Parallel Processing to the Internet of Things'*. 2nd edition, Morgan Kaufmann Publishers, 2014.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the concepts, characteristics, service models, deployment models, and enabling technologies of cloud computing.	**K2**
CO2	Apply virtualization, containerization, cloud infrastructure mechanisms, and virtual machine operations to design and manage cloud-based environments.	**K3**
CO3	Analyze cloud architectures, scalability mechanisms, resource management strategies, and disaster recovery solutions for efficient cloud service delivery.	**K4**
CO4	Evaluate cloud delivery frameworks, service quality metrics, cost models, and Service Level Agreements (SLAs) to recommend suitable cloud solutions for organizational requirements.	**-**

COs–POs & PSOs MAPPING

CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PSO1**	**PSO2**
CO1												3	3
CO2	3											3	3
CO3		3										3	3
CO4					2							2	2
	3	**3**			**2**							**3**	**3**

1–low, 2–medium, 3–high

25ADP18 CLOUD SERVICES MANAGEMENT

2 0 2 3

INTRODUCTION: Introduction – Cloud computing at a Glance – Historical Development – Building Cloud Computing Environments – Computing Platform and Technologies – Principles of Parallel and Distributed Computing – Elements of parallel and Distributed Computing. **(6+6)**

VIRTUALIZATION: Introduction - Hypervisors - Main Categories of Virtualization - Application Server - Application Network - Storage Service - Benefits of Virtualization - Cost of Virtualization Case study: Creation of Virtual machines in Google Cloud, AWS. **(6+6)**

CLOUD COMPUTING ARCHITECTURE AND CLOUD SERVICES: Introduction to Cloud Computing Architecture - Cloud Reference Model -Microservices - Architecture - RESTful Web Services- Storage as a Service - Database as a Service - Information as a Service - Process as a Service - Application as a Service - Platform as a Service - Integration as a Service - Security as a Service - Management as a Service - Testing as a Service - Infrastructure as a Service - Case Study - Google Cloud, AWS. **(6+6)**

CLOUD APPLICATIONS: Scientific Applications - Gene Expression Data Analysis for Cancer Diagnosis - Business and Consumer Applications - Social Networking - Media Applications. **(6+6)**

CLOUD COMPUTING INFRASTRUCTURE: Introduction to Dockers Deploying, maintaining and scaling containerized applications using cloud infrastructure - Cloud storage Identity Access Management Cloud monitoring - Case Study: Google Cloud, AWS. **(6+6)**

LIST OF EXPERIMENTS

1. Launch and configure a Virtual Machine using AWS EC2 / GCP Compute Engine.
2. Deploy a static website on AWS S3 or Azure Blob Storage.
3. Set up and deploy a containerized application using Docker.
4. Manage a Kubernetes cluster and deploy microservices using kubectl.
5. Connect a web application to a cloud database (RDS / Firebase / Cosmos DB) and configure Load Balancer with Auto-scaling Group for scalability and high availability.

Total L: 30 + P: 30 = 60 periods

TEXT BOOKS:

1. Rajkumar Buyya, Christian Vecchiola, ThamaraiSelvi, '*Mastering Cloud Computing*'. Tata McGraw Hill Education Private Limited, New Delhi, 2017.
2. Kai Hwang, '*Distributed and Cloud Computing: From Parallel Processing to the Internet of Things*'. 1st edition, Morgan Kaufmann, 2013.

REFERENCES:

1. Shailendra Singh, '*Cloud Computing*'. 1st edition, Oxford University Press, 2018.
2. Sandeep Bhowmik, '*Cloud Computing*'. 1st edition, Cambridge University Press, July 2017.
3. Kris Jamsa, '*Cloud Computing*'. 2nd edition, Jones & BArtiett Learning Publisher, 2022.
4. Priyanka Vergadia, '*Visualizing Google Cloud: Illustrated References for Cloud Engineers & Architects: 101 Illustrated References for Cloud Engineers & Architects'.* 1st edition, John Wiley & sons Inc, May 2022.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Describe the evolution, principles, and elements of cloud computing, parallel, and distributed computing.	**K2**
CO2	Apply cloud computing architecture models and service models (IaaS, PaaS, SaaS, etc.) in real-time scenarios and analyze service integration strategies.	**K3**
CO3	Analyze various scientific, business, and consumer cloud applications including social and media-based use cases.	**K4**
CO4	Design and manage containerized applications and evaluate cloud infrastructure elements such as identity management, monitoring, and storage with tools like Docker, Google Cloud, and AWS.	**-**

COs–POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1												2	2
CO2	3											3	3
CO3		3										3	3
CO4			2	2	3							3	3
	3	**3**	**2**	**2**	**3**							**3**	**3**

1–low, 2–medium, 3–high

25CSP11 VIRTUALIZATION

2 0 2 3

INTRODUCTION TO VIRTUALIZATION: Virtualization and cloud computing – Need of virtualization – cost, administration, fast deployment, reduce infrastructure cost – limitations – Hyperscale Virtualization – Types of hardware virtualization: Full virtualization – partial virtualization – Paravirtualization – Types of Hypervisors. **(6+6)**

SERVER AND DESKTOP VIRTUALIZATION : Virtual machine basics – Types of virtual machines – Understanding Server Virtualization – types of server virtualization – Business Cases for Server Virtualization – Uses of Virtual Server Consolidation – Selecting Server Virtualization Platform – Desktop Virtualization – Types of Desktop Virtualization. **(6+6)**

NETWORK VIRTUALIZATION: Introduction to Network Virtualization – Advantages – Functions – Tools for Network Virtualization VLAN –WAN Architecture –WAN Virtualization. **(6+6)**

STORAGE VIRTUALIZATION: Memory Virtualization-Types of Storage Virtualization – Block, File – Address space Remapping – Risks of Storage Virtualization – SAN – NAS – RAID. **(6+6)**

VIRTUALIZATION TOOLS: VMWare –Amazon AWS –Microsoft HyperV – Oracle VM Virtual Box – IBM PowerVM – Google Virtualization – Case study. **(6+6)**

LIST OF EXPERIMENTS:

1. Create type 2 virtualization in VMWARE or any equivalent Open Source Tool. Allocate memory and storage space as per requirement. Install Guest OS on that VMWARE.
2. a. Shrink and extend virtual disk
   b. Create, Manage, Configure and schedule snapshots
   c. Create Spanned, Mirrored and Striped volume
   d. Create RAID 5 volume
3. a. Desktop Virtualization using VNC
   b. Desktop Virtualization using Chrome Remote Desktop
4. a. Create type 2 virtualization on ESXI 6.5 server
   b. Create Nested Virtual Machine
5. Install KVM in Linux

Total L: 30 + P: 30 = 60 periods

TEXTBOOKS:

1. Anthony T. Velte, Toby J. Velte Robert Elsenpeter, '*Cloud computing a practical approach'*. TATA McGraw–Hill, New Delhi 2017.
2. Rajkumar Buyya, James Broberg, Andrzej Goscinski, John Wiley & Sons, '*Cloud Computing (Principles and Paradigms)*'. Inc. 2011.

REFERENCES:

1. David Marshall, Wade A. Reynolds, '*Advanced Server Virtualization: VMware and Microsoft Platform in the Virtual Data Center*'. Auerbach, 2006.
2. Chris Wolf, Erick M. Halter, '*Virtualization: From the Desktop to the Enterprise'*. APress, 2005.
3. James E. Smith, Ravi Nair, '*Virtual Machines: Versatile Platforms for Systems and Processes'*. Elsevier/Morgan Kaufmann, 2005.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the fundamentals of virtualization, including its need, types (full, partial, para–virtualization), and the role of hypervisors in modern computing environments.	**K2**
CO2	Illustrate and implement server and desktop virtualization concepts by identifying suitable platforms and exploring virtual machine configurations for real–world scenarios.	**K3**
CO3	Apply network and storage virtualization techniques (VLAN, WAN, SAN, NAS, RAID) and analyze their advantages, risks, and performance in enterprise environments.	**K4**
CO4	Evaluate and compare various virtualization tools and platforms (VMware, AWS, Hyper–V, VirtualBox, etc.) through case studies to determine their suitability for different organizational needs.	–

COs–POs & PSOs MAPPING

CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PSO1**	**PSO2**
CO1												3	3
CO2	3											3	3
CO3		3										3	3
CO4			1									1	1
	3	**3**	**1**									**3**	**3**

1–low, 2–medium, 3–high

25CSP31 MANAGING BIG DATA

2 0 2 3

BIG DATA: Evolution of database technology and Big data – Importance, 4 V's of Big data, Big data analytics – Structured data storage and relational databases – Introduction to NoSQL databases: Cassandra – Cassandra data model – Cassandra examples – Cassandra clients. **(6+6)**

HADOOP: Overview of Hadoop – Hadoop ecosystem – Reading and writing files on Hadoop distributed file system – MapReduce workflows – Anatomy of MapReduce job run – Classic MapReduce. **(6+6)**

STREAM PROCESSING AND REAL-TIME DATA ANALYTICS: Batch processing and stream Processing – Lambda and Kappa architecture – Introduction to Kafka: Architecture, Producers, Consumers, Building data pipelines with Kafka – Introduction to Apache Spark – Spark architecture – RDD concepts – Spark for batch and stream processing. **(6+6)**

ADVANCED BIG DATA TOOLS AND FRAMEWORKS: Apache Flume – Overview and architecture – Basics of Sqoop – Integrating Sqoop with Hadoop – Zookeeper – Getting started with Zookeeper API – Machine learning using Apache Mahout. **(6+6)**

BIG DATA APPLICATIONS: Cloud computing and Big data – Big data challenges – Big data applications – Big data in social media – Recommendation systems – Link analysis –Crowdsourcing. **(6+6)**

LIST OF EXPERIMENTS

1. Perform CRUD operations and data modeling concepts in MongoDB.
2. Implement of matrix multiplication with Hadoop MapReduce.
3. Explore the Hadoop distributed file system and implement a basic MapReduce program (word count).
4. Build a basic Kafka pipeline and understand producer–consumer communication.
5. Use Spark and Spark MLlib for batch processing and machine learning.

Total L: 30 + P: 30 = 60 periods

TEXT BOOKS:

1. Mayank Bhushan, *'Big Data and Hadoop'*. BPB Publications, 2nd edition, 2023.
2. Nitin Kumar, *'Big Data Using Hadoop and Hive'*. Mercury Learning and Information, 2021.

REFERENCES:

1. Douglas Eadline, '*Hadoop 2 Quick–Start Guide*'. Pearson Education, 2015.
2. Thangaraj, M., Suguna, S., Sudha, G, '*Big Data Analytics: Concepts, Techniques, Tools and Technologies*'. PHI Learning Pvt. Ltd., 2022.
3. Judith S. Hurwitz, Alan Nugent, Fern Halper, Marcia Kaufman, '*Big Data for Dummies*'. Wiley, 2013.
4. Bill Chambers, Matei Zaharia, '*Spark: The Definitive Guide Big Data Processing Made Simple'.* O'Reilly Media, 2018.
5. Neha Narkhede, Gwen Shapira, and Todd Palino, '*Kafka – Definitive Guide*'. 2017.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the fundamental concepts of Big data, Hadoop ecosystem, streaming frameworks, and related tools and technologies.	**K2**
CO2	Demonstrate the use of Big data tools such as Hadoop, MongoDB, Kafka, and Spark for data storage and processing.	**K3**
CO3	Analyze different Big data architectures and frameworks including batch and stream processing models for effective data handling.	**K4**
CO4	Implement Big data solutions using data modeling and processing techniques for data storage and computation.	**-**
CO5	Develop and implement Big data workflows using Hadoop, MapReduce and streaming tools to process and analyze large–scale datasets	**-**

COs–POs & PSOs MAPPING

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

1–low, 2–medium, 3–high

25CSP28 DATA WAREHOUSING

2 0 2 3

INTRODUCTION TO DATA WAREHOUSE: Data warehouse Introduction - Data warehouse components- operational database Vs data warehouse – Data warehouse Architecture – Three-tier Data Warehouse Architecture – Autonomous Data Warehouse- Autonomous Data Warehouse Vs Snowflake - Modern Data Warehouse. **(6+6)**

ETL AND OLAP TECHNOLOGY: What is ETL – ETL Vs ELT – Types of Data warehouses - Data warehouse Design and Modeling - Delivery Process - Online Analytical Processing (OLAP) - Characteristics of OLAP – Online Transaction Processing (OLTP) Vs OLAP - OLAP operations- Types of OLAP- ROLAP Vs MOLAP Vs HOLAP. **(6+6)**

META DATA, DATA MART AND PARTITION STRATEGY: Meta Data – Categories of Metadata – Role of Metadata – Metadata Repository – Challenges for Meta Management - Data Mart – Need of Data Mart- Cost Effective Data Mart- Designing Data Marts- Cost of Data Marts- Partitioning Strategy – Vertical partition – Normalization – Row Splitting – Horizontal Partition. **(6+6)**

DIMENSIONAL MODELING AND SCHEMA: Dimensional Modeling- Multi-Dimensional Data Modeling – Data Cube- Star Schema- Snowflake schema- Star Vs Snowflake schema- Fact constellation Schema- Schema Definition – Process Architecture- Types of Data Base Parallelism – Datawarehouse Tools. **(6+6)**

SYSTEM & PROCESS MANAGERS: Data Warehousing System Managers: System Configuration Manager- System Scheduling Manager - System Event Manager - System Database Manager - System Backup Recovery Manager - Data Warehousing Process Managers: Load Manager – Warehouse Manager- Query Manager – Tuning – Testing. **(6+6)**

LIST OF EXPERIMENTS:

1. Data exploration integration and validation with WEKA
2. Write the query for schema definition
3. Design data ware house for real time applications
4. Analyse the dimensional Modeling
5. Implementation of warehouse testing

Total L: 30 + P: 30 = 60 periods

TEXTBOOKS:

1. Alex Berson, Stephen J. Smith & Kurt Thearling, '*Data Warehousing, Data Mining and OLAP'*. 1st edition, McGraw-Hill Education, 2004.
2. Ralph Kimball & Margy Ross, '*The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling*'. 3rd edition, Wiley India, 2013.

REFERENCES:

1. Paulraj Ponniah, '*Data Warehousing Fundamentals for IT Professionals'*. 2nd edition, Wiley India, 2010.
2. K. P. Soman, Shyam Diwakar & V. Ajay, '*Insight into Data Mining: Theory and Practice'*. 2nd edition, PHI Learning, 2014.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the concepts, architecture, components and design principles of data warehouses.	**K2**
CO2	Apply dimensional modeling techniques, schema design approaches, partitioning strategies and OLAP operations for data warehouse development and analytical processing.	**K3**
CO3	Analyze data warehouse management functions, system and process managers, database parallelism and data warehousing tools to support efficient data storage, retrieval and decision-making.	**K4**
CO4	Evaluate data warehouse strategies to recommend suitable solutions for organizational decision-support requirements.	–

COs–POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1												3	3
CO2	3											3	3
CO3		3										3	3
CO4			1									1	1
	3	**3**	**1**									**3**	**3**

1–low, 2–medium, 3–high

25CSP29 STORAGE TECHNOLOGIES

3 0 0 3

STORAGE SYSTEM: Digital data and its types - Information storage - Key characteristics of data center - Evolution of computing platforms, Third platform technologies: Cloud computing and its essential characteristics - deployment models - Big data analytics - Imperatives for third platform transformation, Data Center Environment: Building blocks of a data center - Software-defined data center, Intelligent Storage Systems: Components - Types - Scale-up and scale-out storage architecture, RAID. **(9)**

STORAGE NETWORKING TECHNOLOGIES: Block-based Storage System: Storage provisioning and storage tiering, File- based Storage System: Components and architecture of NAS - NAS file sharing methods, Object - based and Unified Storage: Key features of OSD - Storage and retrieval process in OSD system, Software-defined Storage: Attributes of software-defined storage- Architecture. **(9)**

SAN BASED TECHNOLOGY: Fibre Channel SAN: Software-defined networking - FC SAN components and architecture - FC SAN topologies - Zoning - Virtualization in FC SAN environment, Internet Protocol SAN: iSCSI protocol - Switch aggregation - VLAN - FCIP protocol - Connectivity and configuration, Fibre Channel over Ethernet SAN: FCoE SAN connectivity - Converged Enhanced Ethernet - FCoE architecture. **(9)**

BACKUP, ARCHIVE AND REPLICATION: Introduction to Business Continuity: Impact of information unavailability – Business continuity planning lifecycle - Application resiliency, Backup and Archive: Backup architecture - Backup targets and methods - Data deduplication - Cloud-based and mobile device backup, Replication: Uses of replication and its characteristics - Compute-based, storage-based and network-based replication - Data migration - Disaster Recovery as a Service (DRaaS). **(9)**

SECURING AND MANAGING STORAGE INFRASTRUCTURE: Securing the Storage Infrastructure: Information security goals - Storage security domains - Security controls to protect a storage infrastructure, Managing the Storage Infrastructure: Management functions - Management processes. **(9)**

Total L: 45 periods

TEXT BOOKS:

1. Somasundaram G and Alok Shrivastava, '*ISM - Storing, Managing and Protecting Digital Information'*. 3rd edition, EMC Education Services, Wiley, New Delhi, 2016.
2. Robert Spalding, '*Storage Networks: The Complete Reference'*. Tata McGraw Hill, New Delhi, 2006.

REFERENCES:

1. Meeta Gupta, '*Storage Area Network Fundamentals'*. Pearson Education, New Delhi, 2002.
2. Gerald J Kowalski and Mark T Maybury, '*Information Storage and Retrieval Systems: Theory and Implementation*'. BS Publications., New Delhi, 2009.
3. Gerardus Blokdyk, '*Storage and Data Management Services'*. 3rd edition, 5 Star Cooks, New Delhi, 2018.
4. Jamie Watters and Janet Watters, '*Disaster Recovery, Crisis Response, and Business Continuity'*. Apress, London, 2014.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the concepts of storage systems, data center infrastructure, cloud-based storage technologies, intelligent storage systems, and RAID architectures.	**K2**
CO2	Apply storage networking technologies, including NAS, SAN, software-defined storage, Fibre Channel, iSCSI and FCoE, to design and manage enterprise storage solutions.	**K3**
CO3	Analyze storage architectures, backup strategies, archiving methods, replication techniques, and disaster recovery mechanisms to ensure business continuity and data availability.	**K4**
CO4	Evaluate storage security controls, management functions and infrastructure management processes to recommend secure and efficient storage solutions for enterprise environments.	**-**

COs–POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1												3	3
CO2	3											3	3
CO3		3										3	3
CO4					2							2	2
	3	**3**			**2**							**3**	**3**

1–low, 2–medium, 3–high

25CSP30 STREAM PROCESSING

2 0 2 3

FOUNDATIONS OF DATA SYSTEMS: Introduction to Data Processing, Stages of Data processing, Data Analytics, Batch Processing, Stream processing, Data Migration, Transactional Data processing, Data Mining, Data Management Strategy, Storage, Processing, Integration, Analytics, Benefits of Data as a Service, Challenges. **(6+6)**

REAL-TIME DATA PROCESSING: Introduction to Big data, Big data infrastructure, Real-time Analytics, Near real-time solution, Lambda architecture, Kappa Architecture, Stream Processing, Understanding Data Streams, Message Broker, Stream Processor, Batch & Real-time ETL tools, Streaming Data Storage. **(6+6)**

DATA MODELS AND QUERY LANGUAGES: Relational Model, Document Model, Key-Value Pairs, NoSQL, Object-Relational Mismatch, Manyto-One and Many-to-Many Relationships, Network data models, Schema Flexibility, Structured Query Language, Data Locality for Queries, Declarative Queries, Graph Data models, Cypher Query Language, Graph Queries in SQL, The Semantic Web, CODASYL, SPARQL. **(6+6)**

EVENT PROCESSING WITH APACHE KAFKA: Apache Kafka, Kafka as Event Streaming platform, Events, Producers, Consumers, Topics, Partitions, Brokers, Kafka APIs, Admin API, Producer API, Consumer API, Kafka Streams API, Kafka Connect API. **(6+6)**

REAL-TIME PROCESSING USING SPARK STREAMING: Structured Streaming, Basic Concepts, Handling Event-time and Late Data, Fault-tolerant Semantics, Exactly-once Semantics, Creating Streaming Datasets, Schema Inference, Partitioning of Streaming datasets, Operations on Streaming Data, Selection, Aggregation, Projection, Watermarking, Window operations, Types of Time windows, Join Operations, Deduplication. **(6+6)**

LIST OF EXPERIMENTS:

1. Design and Query Simple application using MongoDB
2. Create a Event Stream with Apache Kafka
3. Create a Real-time Stream processing application using Spark Streaming
4. Build a Micro-batch application
5. Real-time Fraud and Anomaly Detection

Total L: 30 + P: 30 = 60 periods

TEXTBOOKS:

1. Tyler Akidau, Slava Chernyak, Reuven Lax, *'Streaming Systems: The What, Where, When, and How of Large-Scale Data Processing'*. 1st edition, O'Reilly Media, 2018.
2. Martin Kleppmann, *'Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable and Maintainable Systems'*. 1st edition, O'Reilly Media, 2017.

REFERENCES:

1. Valentin Crettaz, *'Practical Real-Time Data Processing and Analytics: Distributed Computing and Event Processing using Apache Spark, Flink, Storm and Kafka'*. 1st edition, Packt Publishing, 2018.
2. https://spark.apache.org/docs/latest/streaming-programming-guide.html

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the concepts of data processing systems, stream processing architectures, data models, and real-time analytics frameworks.	**K2**
CO2	Apply data processing techniques, query languages, and streaming frameworks to manage and process large-scale real-time data streams.	**K3**
CO3	Analyze event-driven data processing systems and streaming architectures to ensure scalability, reliability, and efficient data integration.	**K4**
CO4	Evaluate stream processing solutions and real-time analytics platforms based on performance, fault tolerance, and business requirements.	–

COs–POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1												3	3
CO2	3											3	3
CO3		3										3	3
CO4			1									1	1
	3	**3**	**1**									**3**	**3**

1–low, 2–medium, 3–high

Fourth ACM

25CSP25 HIGH PERFORMANCE COMPUTING

3 0 0 3

INTRODUCTION: Artificial Intelligence (AI vs ML vs DL), Fundamentals of deep learning, CPU vs GPU computing (Understanding the GPU computing), Nvidia NGC and Docker to train and deploy models in GPUs. **(9)**

HPC ARCHITECTURE: Basic architecture and organization, Memory hierarchy, Shared and Distributed memory architectures, Multiprocessor architecture, Introduction to thread level parallelism, Accelerators (GPU, Xeon-phi), Performance prediction and evaluation. **(9)**

PARALLEL PROGRAMMING: Parallel programming/computing, Introduction to MPI/ openMP, Basics of CUDA programming, optimizing cluster operation - Running jobs in HPC environment, Job scheduler, Cluster level load balancing. **(9)**

GPU ARCHITECTURE & CUDA: Introduction to Accelerated Data Science, RAPIDS, Introduction to Machine Learning Algorithms, Solving and Benchmarking End to End Data Science Problem using RAPIDS. **(9)**

ACCELERATED COMPUTING: Accelerated Computing for Deep Neural Networks, Accelerating and Scaling Deep Neural Networks using DALI, Mixed Precision and Multi-GPU Scaling Optimizing and Deployment of Neural Networks using TensorRT & Triton Inference Server. **(9)**

Total L: 45 periods

TEXT BOOKS:

1. Gerassimos Barlas, '*Multicore and GPU Programming'*. Morgan Kaufmann publisher, 2014.
2. Magnus Ekman, '*Learning Deep Learning, Theory and Practice of Neural Networks, Computer Vision, Natural Language Processing, and Transformers Using Tensor Flow*'. Addison Wesley Professional, 2021.

REFERENCES:

1. Chao Wang, Chapman and Hall, '*High Performance Computing for Big Data Methodologies and Applications'*. CRC, New York, 2018.
2. Janson Sanders, Edward Kandrot, '*CUDA by Example: An Introduction to General-Purpose GPU Programming*'. O'Reilly Addison-Wesley Professional, USA, 2010.
3. David Kirk, Wen-mei Hwu '*Programming Massively Parallel Processors: A Hands-on Approach'*. Morgan Kaufman, USA, 2010.
4. N D Lewis, '*Deep Learning made easy with R: A Gentle Introduction for Data Science'*. Create space dependent Publishing Platform, New Delhi 2016.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the fundamentals of artificial intelligence, deep learning, GPU computing, HPC architectures, memory hierarchies, and parallel computing concepts.	**K2**
CO2	Apply parallel programming techniques, MPI/OpenMP, CUDA programming, and HPC tools to develop and execute scalable computing applications.	**K3**
CO3	Analyze GPU architectures, accelerated data science frameworks, and machine learning workflows to improve computational efficiency and performance.	**K4**
CO4	Evaluate accelerated computing solutions, deep learning optimization techniques, and deployment frameworks for large-scale AI applications.	**-**

COs–POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1												3	3
CO2	3											3	3
CO3		3										3	3
CO4					2							2	2
	3	**3**			**2**							**3**	**3**

1–low, 2–medium, 3–high

PROFESSIONAL COURSES FOR MINOR DEGREE PROGRAMME
25CSM01 MICRO SERVICES AND CLOUD APPLICATION

3 0 0 3

INTRODUCTION TO MODERN ARCHITECTURE: evolution from monolithic to microservices, key architecture principles: Small footprint, loose coupling, and independent scaling. Applications: smart grids and IoT devices to the cloud. **(9)**

DESIGNING APIS AND DATA FLOW: API Fundamentals, Understanding REST APIs and data exchange in connected systems. Lightweight Protocols: Basics of MQTT protocol and introduction to gRPC. Message Brokers: Introduction to message brokers and event messaging using MQTT. **(9)**

BUILDING A SIMPLE SERVICE LIGHTWEIGHT FRAMEWORKS: Developing basic services using Python (Fast API). Database Basics: Storage and management of engineering data, sensor data, and device logs. **(9)**

CONTAINERS AND DEPLOYMENT: Introduction to cloud computing, Docker fundamentals: containers usage, writing simple Docker files and isolating services. conceptual overview of Kubernetes for container orchestration, introduction to system monitoring, health checks, and visualization of engineering system. **(9)**

BASIC SECURITY AND GATEWAY SYSTEMS: concept of API gateways, Managing a single entry point for microservices. Token Authentication: Basic concepts of user and device authentication. Resilience Patterns: Fundamentals of fault tolerance and resilient systems. **(9)**

Total L: 45 periods

TEXT BOOKS:

1. Sam Newman, '*Building Microservices: Designing Fine–Grained Systems'*. 2nd edition, O'Reilly Media, 2021.

REFERENCES:

1. Mike Amundsen, '*Ronnie Mitra, Matt McLarty, Irakli Nadareishvili, Microservice Architecture: Aligning Principles Practices and Culture'.* O'Reilly, 2016.
2. Chellammal Surianarayanan, Gopinath Ganapathy, Raj Pethuru, *'Essentials of Microservices Architecture'.* CRC Press, 2019.
3. Sam Newman, *'Monolith to Microservices–Evolutionary Patterns to Transform Your Monolith'*. O'Reilly, 2019.
4. Sherwin John C. Tragura, '*Building Python Microservices with FastAPI'*, Packt Publishing, 2023.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Describe microservices architecture concepts, principles, and applications in cloud-based systems.	**K2**
CO2	Apply micro service design practices including service design, data management, APIs, communication patterns.	**K3**
CO3	Analyze deployment, containerization, orchestration, and monitoring techniques for microservices.	**K4**
CO4	Evaluate API gateways, authentication mechanisms, and resilience techniques for secure and reliable microservices.	**–**

COs–POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1													
CO2	3												
CO3													
CO4								2	2				
	3							**2**	**2**				

1–low, 2–medium, 3–high

25CSM02 JAVA PROGRAMMING

2 0 2 3

JAVA FUNDAMENTALS: Java basics – Features of java language – Java virtual machine – Bytecode – Java program structure – Data types – Variables – Operators – Control and looping constructs – Array: one dimensional and multidimensional – Enhanced for loop – Wrapper class. **(6+6)**

CLASSES AND OBJECTS: Class fundamentals – Access and non-access specifiers –Objects and object references – Array of objects – Constructors – Methods – Static members – Usage of this keyword. **(6+6)**

INHERITANCE AND POLYMORPHISM: Method overloading – Objects as parameters – Returning objects – Inheritance: basics – Types of inheritance – Super keyword – Method overriding – Abstract classes – 'final' with inheritance – Interfaces. **(6+6)**

PACKAGES AND EXCEPTION HANDLING: Creating and accessing packages – Importing packages – Exception handling basics – Multiple catch clauses – Nested try statements – Built-in exceptions – User defined exceptions. **(6+6)**

MULTITHREADING, GENERICS AND STRING HANDLING: Thread lifecycle – Creating single and multiple threads – Thread priorities – Generic classes – Generic methods – Strings: string class, commonly used methods and StringBuffer class. **(6+6)**

LIST OF EXPERIMENTS:

1. Develop Java programs using control structures, arrays, and wrapper classes.
2. Implement applications using classes, objects, constructors, methods, and static members.
3. Develop programs demonstrating inheritance, method overloading, method overriding, and interfaces.
4. Create Java packages and implement exception handling using built-in and user-defined exceptions.
5. Develop multithreaded applications and implement thread synchronization concepts.

Total L: 30 + P: 30 = 60 periods

TEXT BOOKS:

1. Herbert Schildt, '*Java: The Complete Reference'*. 13th edition, McGraw Hill Education, New Delhi, 2024.
2. Cay S. Horstmann, '*Core Java Fundamentals'*. 12th edition, Pearson Education (US), 2021.

REFERENCES:

3. Deitel P. and Deitel H., *'Java: How to Program'*. 11th edition, Prentice Hall, 2018.
4. Hortsmann C. S. and Cornell G., *'Core Java: Volume 1 – Fundamentals'*. 11th edition, Prentice Hall, 2018.
5. James Gosling, Bill Joy, Guy Steele, Gilad Bracha, Alex Buckley and Daniel Smith, *'The Java Language Specification – Java SE'*. 13th edition, Oracle America Inc., USA, 2019.
6. Daniel Liang L., *'Introduction to Java Programming'*. 10th edition, Pearson Education, New Delhi, 2015.
7. Matt Weisfeld, *'The Object–Oriented Thought Process'*. 5th edition, Addison–Wesley Professional, US, 2019.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the principles of object-oriented programming and Java programming constructs.	**K2**
CO2	Apply Java programming concepts to develop applications using classes, inheritance, packages, and exception handling.	**K3**
CO3	Analyze object-oriented solutions using polymorphism, multithreading, generics, and string handling techniques.	**K4**
CO4	Evaluate Java applications using object-oriented principles and programming best practices.	**–**

COs–POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1													
CO2	3												
CO3		3											
CO4			3	3									
	3	**3**	**3**	**3**									

1–low, 2–medium, 3–high

25CSM03 WEB FRAMEWORKS **Fourth ACM**

2 0 2 3

FUNDAMENTALS OF WEB APPLICATION DEVELOPMENT: Introduction to web applications – Client-server architecture – Frontend and backend concepts – HTTP methods and status codes – REST architecture – JSON data exchange – Web development workflow. **(6+6)**

FRONTEND FRAMEWORKS: Single page application concepts – Introduction to React – Components and props – State management basics – Event handling – React hooks – Responsive web design using Bootstrap/Tailwind CSS. **(6+6)**

BACKEND FRAMEWORKS: Introduction to backend development – Server-side scripting concepts – MVC architecture — Routing and middleware – Introduction to Express.js framework – Database connectivity using MySQL and MongoDB. **(6+6)**

API DEVELOPMENT & DATABASE INTEGRATION: RESTful API concepts – API endpoint design – CRUD operations – JSON data handling – Authentication basics – Session management – Form validation techniques – API testing basics. **(6+6)**

WEB APPLICATION DEPLOYMENT & SECURITY: Web hosting concepts – Deployment basics – Introduction to cloud deployment – Input validation techniques – Authentication and authorization concepts – Secure coding practices – Fundamentals of web application security –Common web vulnerabilities and protection mechanisms. **(6+6)**

LIST OF EXPERIMENTS:

1. Create a responsive web page using Bootstrap/Tailwind CSS and manage the source code using Git.
2. Develop a React application using components, props, state management, and event handling.
3. Build a Single Page Application (SPA) using React with navigation and responsive design.
4. Develop a backend application using Express.js and connect it to a MySQL database.
5. Design and test RESTful APIs implementing CRUD operations and JSON data exchange.
6. Develop and deploy a full-stack web application integrating React, Express.js, MySQL, authentication, and form validation.

Total L: 30 + P: 30 = 60 periods

TEXT BOOKS:

1. Andrew Hoffman, '*Web Application Security: Exploitation and Countermeasures for Modern Web Applications'*. 1st edition, O'Reilly Media, Inc, 2020.
2. Bryan Sullivan, Vincent Liu, '*Web Application Security: A Beginners Guide'*. The McGraw–Hill Companies, 2012.
3. Neil Madden, '*API Security in Action'*. Manning Publications Co., NY, USA, 2020.

REFERENCES:

1. Michael Cross, '*Developer's Guide to Web Application Security'*. Syngress Publishing, Inc, 2007.
2. Ravi Das and Greg Johnson, '*Testing and Securing Web Applications'*. Taylor & Francis Group, LLC, 2021.
3. Prabath Siriwardena, '*Advanced API Security'*. Apress Media LLC, USA, 2020.
4. Malcom McDonald, '*Web Security for Developers'*. No Starch Press, Inc., 2020.
5. Allen Harper, Shon Harris, Jonathan Ness, Chris Eagle, Gideon Lenkey, and Terron Williams Grey Hat '*Hacking: The Ethical Hackers Handbook'*. 3rd edition, The McGraw–Hill Companies, 2011.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the concepts, architectures, frameworks, and technologies involved in modern web application development.	**K2**
CO2	Apply frontend and backend framework concepts to develop responsive web applications with database connectivity and API integration.	**K3**
CO3	Analyze web application design, deployment approaches, and integration mechanisms to enhance functionality and performance.	**K4**
CO4	Evaluate web application security practices, deployment strategies, and framework-based solutions for developing reliable and secure applications.	**–**

COs–POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1													
CO2	3												
CO3		3											
CO4				3	2								
	3	**3**		**3**	**2**								

1–low, 2–medium, 3–high

25CSM04 APPLICATION DEVELOPMENT TECHNOLOGIES

2 0 2 3

PRINCIPLES OF SECURE FULL STACK DEVELOPMENT: Full stack architecture overview – Client-server communication – HTTP/HTTPS – HTML5 – CSS3 – JS security – Secure DOM handling – Input validation and sanitization – Basics of secure backend frameworks (Node.js / Django / Spring). **(7+6)**

SECURE FRONTEND ENGINEERING: Component security – Secure routing – State handling risks – Avoiding inline scripts – Prevention of Cross-site scripting (XSS) – Cross-site request forgery (CSRF) – Content security policy (CSP) – Secure API consumption. **(6+6)**

BACKEND SECURITY: JSON web tokens (JWT) – Oauth2 – Single sign-on (SSO) – Password hashing (bcrypt/Argon2) – Role-based and attribute-based access control (RBAC/ABAC) – Secure session management. **(5+6)**

SECURE DATABASE AND DEVSECOPS: SQL injection prevention – Prepared statements – ORM security – NoSQL injection – Secrets management (vault environment encryption) – Static/Dynamic code analysis (SAST/DAST) – CI/CD pipelines (Jenkins/GitHub action) – Docker security. **(6+6)**

SECURE DEPLOYMENT AND CLOUD SECURITY: TLS/SSL – Secure headers (CSP, HSTS, X-Frame-Options) – Server hardening – Reverse proxy security (Nginx) – Cloud IAM (AWS/Azure/GCP) – Logging and alerting – Zero trust principles. **(6+6)**

Total L: 30 Periods

LIST OF EXPERIMENTS

1. Secure Frontend Application with XSS and CSRF Protection - Build a React/HTML5 application implementing Content Security Policy (CSP) headers, input sanitization, CSRF tokens, and secure DOM manipulation. Demonstrate prevention of XSS attacks using DOMPurify and secure routing techniques.
2. JWT-Based Authentication and Role-Based Access Control (RBAC) - Develop a Node.js/Express backend with JWT authentication, OAuth2 integration, bcrypt password hashing, and RBAC middleware. Implement secure session management and test access control for different user roles (admin, user, guest).
3. Secure Database Operations and SQL/NoSQL Injection Prevention - Implement a full stack application using prepared statements, parameterized queries, and ORM-based security (Sequelize/Mongoose). Demonstrate SQL injection attacks and defenses, secrets management using environment variables/Vault, and secure CRUD operations.
4. DevSecOps Pipeline with SAST/DAST Integration - Configure a CI/CD pipeline using GitHub Actions or Jenkins integrating static analysis (ESLint security rules, SonarQube) and dynamic analysis (OWASP ZAP). Containerize the application with Docker applying security best practices such as non-root users, minimal base images, and secret scanning.
5. Secure Cloud Deployment with TLS, Reverse Proxy and IAM - Deploy a full stack application on a cloud platform (AWS/Azure/GCP) configuring TLS/SSL certificates, HSTS, secure HTTP headers (X-Frame-Options, X-Content-Type-Options), Nginx reverse proxy hardening, and Cloud IAM policies. Implement logging, alerting, and zero-trust network access principles.

Total P: 30 periods

TEXTBOOKS:

1. Andrew Hoffman, *'Web Application Security: Exploitation and Countermeasures'*. O'Reilly Media, 2020.
2. Neil Madden, *'API Security in Action'*. 1st edition, Manning Publications, 2020.

REFERENCES:

1. Prabath Siriwardena and Nuwan Dias, *'Advanced API Security: Oauth 2.0 and Beyond'*. Apress, 2020.
2. Dafydd Stuttard and Marcus Pinto, *'The Web Application Hacker's Handbook'*. 2nd edition, Wiley Publishing, 2011.
3. Steve Fulton and Jeff Fulton, *'HTML5 Canvas'*. O'Reilly Media, 2013.
4. Frank Zammetti, *'Modern Full-Stack Development: Using TypeScript, React, Node.js, Webpack, and Docker'*. Apress, New Delhi, 2020.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Describe secure full stack architectures, client-server communication models, and security principles underlying modern web application development.	**K2**
CO2	Apply secure frontend and backend development techniques including authentication, authorization, state management and API security to build robust full stack applications.	**K3**
CO3	Analyze system interactions, security challenges, and operational factors affecting the reliability and performance of software applications.	**K4**
CO4	Evaluate techniques, frameworks, and deployment strategies for ensuring security, scalability, and efficiency throughout the software development life cycle.	**K5**
CO5	Design and develop secure, scalable, and cloud-ready full stack web applications by integrating deployment strategies, zero-trust principles, and industry best practices.	**K6**

COs–POs & PSOs MAPPING

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

1–low, 2–medium, 3–high

25CSM05 SOFTWARE SYSTEM TESTING

3 0 0 3

INTRODUCTION TO TESTING: Testing fundamentals: Basic definitions – Software testing principles and Tester's role in a software development organization – Defects, Hypothesis and Test: Origins of defect – Defect Classes – The defect repository and test design. **(9)**

BLACK BOX TESTING: Black box approach to test case design – Random testing – Equivalence class partitioning – Boundary value analysis – Other black box test design approaches. **(8)**

WHITE BOX TESTING: White box approach to test case design – Test adequacy criteria –Coverage and control flow graphs – Covering code logic – Paths – Additional white box test design approaches: Data flow – Loop testing – Mutation testing **(9)**

LEVELS OF TESTING: Unit Test: Functions – Procedures – Classes, and Methods as Units. Integration test: Goals – Integration strategies for procedures – Functions and classes – System test: Functional testing – Performance testing – Stress testing – Configuration testing – Security testing – Recovery testing – Regression testing – Alpha, Beta, Acceptance testing. **(10)**

TEST AUTOMATION AND TOOLS: Automated software testing – Automate testing of web applications – Selenium: Introducing web driver and web elements – Locating web elements – Actions on web elements – Different web drivers – Understanding web driver events– Testing: Understanding Testing.xml – Adding classes – Packages – Methods to test – Test reports. **(9)**

Total L: 45 periods

TEXTBOOKS:

1. Ilene Burnstein, *'Practical Software Testing'*. Springer International Edition, 2010, Reprint.
2. Roger. S. Pressman, *'Software Engineering - A Practitioner's Approach'*. McGraw Hill International Edition, 2014.
3. Daniel Galin, *'Software Quality Assurance'*. Pearson Publication, 2009.

REFERENCES:

1. Yogesh Singh, *'Software Testing'*. Cambridge University Press, 2012.
2. Kshirasagar Naik, Priyadhrshini Tripathy, *'Software Testing and Quality Assurance'*. John wiley & sons Inc Publication, 2008.
3. Nina S Godbole, 'Software Quality Assurance-Principles and Practice'. 2nd Edition, Narosa Publishing, 2016.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain concepts, principles, and defect characteristics in software validation.	**K2**
CO2	Apply test case design techniques to develop effective test scenarios.	**K3**
CO3	Analyze different testing levels and strategies for ensuring software quality.	**K4**
CO4	Evaluate testing tools and automation approaches for efficient validation processes.	**-**

COs-POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1												3	3
CO2	3											3	3
CO3		3										3	3
CO4			2									2	2
	3	**3**	**2**									**3**	**3**

1-low, 2-medium, 3-high

25CSM06 USER EXPERIENCE AND INTERFACE DEVELOPMENT

2 0 2 3

FOUNDATIONS OF UX AND UI: Introduction to HCI and UX, Human-Centered Design - Design Thinking, Principles of Usability and UX - Difference between UI and UX - Stages in UX Design - Empathy Maps - User Personas - Journey Mapping - History of UI and UX - Need of UI and UX. **(6+6)**

INTERACTION DESIGN AND WIREFRAMING: Information Architecture - Navigation Design - Interaction Models - Task Flows – Storyboarding – Wireframes – Mockups - UI Design Patterns - Paper Prototyping - Low-Fidelity and High-Fidelity Prototypes - Design Tools: Figma, Adobe XD. **(6+6)**

VISUAL DESIGN AND FRONTEND TECHNOLOGIES: Typography - Color Theory - Layout Design, - Responsive Design Principles – HTML – CSS - JavaScript Basics - Introduction to Frontend Frameworks: Bootstrap - React basics - Building UI Components - Mobile-First Design. **(6+6)**

USABILITY TESTING AND EVALUATION: Heuristic Evaluation - Usability Metrics: SUS – NPS - Time-on- Task - Usability Testing - Methods - A/B Testing - Moderated User Testing - Unmoderated User Testing - Cognitive Walkthroughs - User Feedback Collection - Iterative Design Process. **(6+6)**

EMERGING TECHNOLOGIES: UX for Mobile and Wearable Devices - Interaction design for gesture controls. Designing interactions for futuristic technologies – Voice – AI - Conversational Interfaces: Chatbots - Voice UI - Multi-Screen Interaction design - UX for AR/VR - Portfolio Building. **(6+6)**

LIST OF EXPERIMENTS:

1. Conduct user research and create empathy maps, user personas, and journey maps for a selected application.
2. Design the information architecture, navigation flow, and low-fidelity wireframes for a web or mobile application.
3. Develop a high-fidelity interactive prototype using Figma or Adobe XD by applying UI design principles.
4. Implement a responsive user interface using HTML, CSS, JavaScript, and Bootstrap for a chosen application.
5. Perform usability testing and evaluation of a prototype using heuristic evaluation, cognitive walkthroughs, and usability metrics.

Total L: 30 + P: 30 = 60 periods

TEXTBOOKS:

1. Alan Dix, Janet Finlay, Gregory Abowd, Russell Beale, '*Human–Computer Interaction'*. 5th edition, Pearson 2020.
2. Joel Marsh, '*UX for Beginners'*. O"Reilly, 2022.

REFERENCES:

1. Ben Shneiderman et al., '*Designing the User Interface'*. 6th edition, Pearson.2016.
2. Jenifer Tidwell, '*Designing Interfaces: Patterns for Effective Interaction Design'*. O"Reilly. 2010.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Understand and explain the principles, processes, and foundations of user experience (UX), user interface (UI), and human–centered design.	**K2**
CO2	Apply design thinking, interaction design methods, and relevant tools/technologies to create effective and user–centric interface solutions.	**K3**
CO3	Analyze and evaluate the usability, accessibility, and performance of interactive systems using standard testing methods and user feedback.	**K4**
CO4	Design and develop responsive and innovative user interfaces by applying modern tools, emerging technologies, and best design practices.	**–**

COs–POs & PSOs MAPPING

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

1–low, 2–medium, 3–high

25CSM07 DEVOPS SYSTEMS

2 0 2 3

INTRODUCTION TO DEVOPS: Software Development and Operations (Dev Vs Ops) – Evolution and Need for DevOps – DevOps Culture and Principles – Benefits of DevOps – DevOps Lifecycle – Business Value of DevOps – DevOps Process Best Practices. **(6+6)**

CONTINUOUS INTEGRATION: Continuous Integration Concepts and Workflow – Version Control Using Git and GitHub – Jenkins Overview – Jenkins Architecture – Integration of GitHub with Jenkins – Creating Build Jobs – Basics of Automated Testing. **(6+6)**

CONTINUOUS DELIVERY: Continuous Delivery Concepts – Build Pipelines – Build Automation Using Maven – Repository Management – Testing in CI/CD Environments – Deployment Workflows – Release Management. **(6+6)**

DEPLOYMENT AUTOMATION AND MONITORING: Deployment Automation Concepts – Configuration Management Fundamentals – Overview of Chef and Ansible – Automation Workflows – Monitoring Fundamentals – Log Management – Application Performance Monitoring. **(6+6)**

DEVSECOPS AND MLOPS FUNDAMENTALS: Security in the Software Development Lifecycle – Common Cyber Threats – Secure Development Practices – DevSecOps Principles – Containerization Concepts and Docker Fundamentals – MLOps Lifecycle – Model Deployment, Monitoring, and Governance. **(6+6)**

LIST OF EXPERIMENTS:

1. Basic Version Control and Branch Management using Git (Includes basic Git commands, branch creation, merging, and conflict resolution.)
2. Remote Repository Management and Collaboration using GitHub (Includes pushing to remote repositories, creating pull requests, code reviews, and merging PRs.)
3. Build a simple application using a build automation tool
4. Create CI pipeline using a CI tool, integrate with GitHub repository and automate build process
5. Extend CI pipeline to CD, deploy application to local server and Validate deployment

Suggested Tools: Maven / Gradle, Jenkins / GitHub Actions, Docker, Cloud Foundry.

Total L: 30 + P: 30 = 60 periods

TEXT BOOKS:

1. Sanjeev Sharma *'The DevOps Adoption Playbook: A Guide to Adopting DevOps in a Multi–Speed IT Enterprise'*. John Wiley & Sons, Inc., IBM Press, 2017.
2. Gene Kim, John, Patrick Debois, Jez Humb, *'The DevOps Handbook: How to create world– class Agility, Reliability and Security in Technology Organization'*. O'Reilly Publications, 2016.

REFERENCES:

1. Noah Gift, Alfredo Deza, '*Practical MLOps'*. O'Reilly Media, Inc.2021.
2. Mark Treveil, Nicolas Omont, Clément Stenac, Kenji Lefevre, Du Phan, Joachim Zentici, Adrien Lavoillotte, Makoto Miyazaki, Lynn Heidmann '*Introducing MLOps*'. O'Reilly Media, Inc, 2020.
3. SricharanVadapalli, '*DevOps – Continuous Delivery, Integration, and Deployment with DevOps*'. Packt Publishing, 2018.
4. Ankita Patil, Mitesh Soni, '*Hands–on Pipeline as Code with Jenkins CI/CD Implementation for Mobile, Web, and Hybrid Applications Using Declarative Pipeline in Jenkins*'. BPB publications, 2021.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the principles, lifecycle, and best practices of DevOps in modern software development and operations.	**K2**
CO2	Apply Continuous Integration and Continuous Delivery concepts using version control, build automation, and CI/CD tools	**K3**
CO3	Analyze deployment automation, configuration management, and monitoring practices for improving software delivery processes.	**K4**
CO4	Analyze the role of security, containerization, and MLOps practices in building reliable and scalable software systems.	**-**

COs–POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1												3	3
CO2	3											3	3
CO3		3										3	3
CO4			1		1							1	1
	3	**3**	**1**		**1**							**3**	**3**

1–low, 2–medium, 3–high

25CSM08 INTELLIGENT CODE GENERATION

2 0 2 3

VIBE CODING FOUNDATIONS: AI in programming – Evolution – Co-creator Role – Vibe coding concepts – Programming with emotion, Tone, and Intention – Clarity and expressiveness in Coding. **(6+6)**

EXPRESSIVE CODING & VISUAL ART: Creative coding principles – Generative art – Introduction to Processing, p5.js, and Python Turtle – Mood-aware design. **(6+6)**

HUMAN-AI COLLABORATION IN CODE: AI coding assistants (ChatGPT, GitHub Copilot, Claude) – AI-assisted code generation – Code refactoring – Responsible use of AI tools. **(6+6)**

WEB APIS & CREATIVE APPLICATIONS: Introduction to Web APIs – API integration – Weather and Mood applications – User-focused application development. **(6+6)**

ETHICAL DIMENSIONS: Ethics in AI-assisted programming – AI bias and authorship – Responsible AI practices – AI as a coding partner. **(6+6)**

LIST OF EXPERIMENTS:

1. Develop a Simple Calculator and Student Grade Calculator using AI-assisted code generation with ChatGPT.
2. Develop Interactive Visual Designs such as Geometric Patterns and Animated Shapes using p5.js or Python Turtle.
3. Develop a Weather Information or Random Quote Application using Public Web APIs.
4. Debug, Refactor, and Optimize an Existing Program using ChatGPT or GitHub Copilot.
5. Design Effective Prompts for AI-Assisted Code Generation, Generate and Implement the Program, and Evaluate the Output.

Total L: 30 +P: 30 = 60 periods

TEXT BOOKS:

1. Michael A. Angelo, '*Vibe Coding #1: Revolution in AI Driven Programming - Comprehensive Reports*'. Kindle Edition, 2025.

REFERENCES:

1. Addy Osmani, '*Vibe Coding: The Future of Programming'*. O'Reilly Media, 2024
2. GitHub Copilot, '*Documentation and Real-World Case Studies'*. GitHub Inc., 2023.
3. Daniel Shiffman, '*The Nature of Code'.* D. Shiffman Publishing 2nd Edition, 2021.
4. Addy Osmani, '*Beyond Vibe Coding'*. O'Reilly Media, Inc., 2025.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the fundamental concepts of vibe coding, AI-assisted programming, and creativity in coding practices.	**K2**
CO2	Apply expressive coding techniques using tools for aesthetic and interactive outputs.	**K3**
CO3	Analyze the role of AI tools in coding workflows and collaboration.	**K4**
CO4	Evaluate creative Web API-based applications in terms of user experience, functionality, and ethical considerations.	**-**

COs–POs & PSOs MAPPING

CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PSO1**	**PSO2**
CO1													
CO2	3												
CO3		3											
CO4			1										
	3	**3**	**1**										

1–low, 2–medium, 3–high

LANGUAGE ELECTIVES

25HS211 COMMUNICATION SKILLS FOR ENGINEERS
(Common to all B.E./B.Tech. Programmes)

0 0 4 2

COMMUNICATION CONCEPTS: Process of Communication – Inter and Intrapersonal Communication – Essentials for effectiveness. **(9)**

ORAL COMMUNICATION: Oral presentations with visual aids and Group discussions. **(16)**

FOCUS ON SOFT SKILLS: Etiquette – Work Place etiquette – Telephone etiquette- Body Language – Critical Reasoning and Conflict Management based on Case Studies – Group Communication- Meetings -Interview Techniques. **(14)**

TECHNICAL WRITING: Technical Writing Principles - Style and Mechanics - Technical Definitions – Physical, Functional and Process Descriptions – Technical Report Writing – Preparing Instructions – Interpretation of Technical Data. **(14)**

BUSINESS CORRESPONDENCE: Writing Emails, Preparing Resumes. **(7)**

Total P: 60 periods

TEXT BOOKS

1. Course materials prepared by the Faculty, Department of English, PSG Institute of Technology and Applied Research, Neelambur, Coimbatore.

REFERENCES

1. Jeff Butterfield, '*Soft Skills for Everyone*'. Cengage Learning, New Delhi, 2020.
2. Sabina Pillai, and Agna Fernandez, '*Soft skills and Employability Skills*'. Cambridge University Press, New Delhi, 2019.
3. Prashant Sharma, '*Soft Skills Personality Development for Life Success*'. BPB Publications, New Delhi, 2021.
4. K. N. Shoba, and D. Praveen Sam, '*Technical English*'. Cambridge University Press, New York, 2020.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Produce clear and concise technical reports, compose professional and effective emails and develop well-structured and impactful resumes	**K2**
CO2	Plan, organize, and deliver engaging and informative presentations using appropriate visual aids and participate positively in group discussions	**K3**
CO3	Resolve disagreements constructively, embody professional conduct and a strong work ethic and apply critical thinking to generate effective solutions	**-**

COs-POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1										2			
CO2										3			
CO3									3	3			
									3	**3**			

1-low, 2-medium, 3-high

25HS212 BASIC GERMAN
(Common to all B.E./B.Tech. Programmes)

0 0 4 2

Guten Tag! - Learning: To greet, learn numbers till 20, practice telephone numbers & e mail address, learn alphabet, speak about countries & languages; **Vocabulary:** related to the topic; **Grammar:** W – Questions, Verbs & Personal pronouns I.

Freunde, Kollegen und ich - Learning: To speak about hobbies, jobs, learn numbers from 20; **Vocabulary:** related to the topic; **Grammar:** Articles, Verbs & Personal pronouns II, sein & haben verbs, ja/nein Frage, singular/plural.

In der Stadt – Learning: To know places, buildings, question, know transport systems, understand international words; **Vocabulary**: related to the topic; **Grammar:** Definite & indefinite articles, Negotiation, Imperative with Sie.

Guten Appetit! – Learning: To speak about food, shop, converse; **Vocabulary:** related to the topic; **Grammar:** Sentence position, Accusative, Accusative with verbs.

Tag für Tag and Zeit mit Freunden – Learning: To learn time related expressions, speak about family, ask excuse, fix appointments on phone, birthdays, understand & write invitations, converse in the restaurant; **Vocabulary:** related to the topic; **Grammar:** Preposition – am, im, um, von…bis, Possessive articles, Modal verbs.

Total P: 60 periods

TEXT BOOK:

1. Dengler, Stefanie et al., '*Netzwerk A1.1*'. Klett-Langenscheidt Gmbh, München, 2013.

REFERENCES:

1. Dengler, Stefanie et al., '*Netzwerk A1'.* Klett-Langenscheidt Gmbh München, 2013.
2. Sandra Evans, Angela Pude, and Franz Specht, '*Menschen A1*'. Hueber Verlag, 2012.
3. Hermann Funk, Christina Kuhn, and Silke Demme, '*Studio d A1*'. Goyal Publishers & Distributors Pvt. Ltd, 2009.
4. Rosa-Maria Dallapiazza, Eduard von Jan, '*Til Schönherr*'. Tangram Aktuell 1 (Deutsch als Fremdsprache), Max Hueber Verlag, 2004.

25HS213 BASIC JAPANESE
(Common to all B.E./B.Tech. Programmes)

0 0 4 2

Orientation Session, Geographic & Socio, economic perspective to Japan, Japanese people and culture and Basic greetings and responses.

Basic script, Method of writing hiragana and katakana, and Combination sounds and simple words.

Topic marker "wa", Desu / dewa arimasen cupolas, Interrogative particle "ka", Grammar particles "mo", "no", "Introducing someone: "Kochira wa ~" and Self introductions: Hajimemashite"

Demonstratives "Kore", "Sore", "Are", Demonstrative "Kono", "Sono", "Ano", Possessive noun particle "no" and Japanese apartments: Greeting your neighbor.

Place markers "Koko", "Soko", "Asoko", Direction markers "Kochira", "Sochira", "Achira" and Japanese department stores: Asking for and buying something.

Asking for and telling the time, Particle "ni (at)" for time, kara (from) ~ made (until), Particle "to (and)", Time periods: Days of the week, months, time of day, Verbs (Present / future and past tense) and Telephone enquiry: Asking for a phone no. And business hours.

Destination particle "e", Particles "de (mode of transportation)" and "to (with) and Japanese train station: Asking for Fare and track no. / types of trains.

Direct object particle "o", Particle "de (place of action)", Verbs ("~masen ka", "~mashou") and "Ohanami" Cherry blossom viewing.

Particle "de (by means of)", Particle "ni (to)", Aaemasu (give) and Moraimasu (receive) and Visiting a Japanese house.

Adjectives ("i" and "na" type), Adjectives (Positive and negative usage), Particle "ga (however, but), "Dore which?)" and Leaving a room, thanking someone for hospitality.

Likes and dislikes, Potential verbs (wakarimasu and dekimasu), "Kara (~ because)", Adverbs and Asking someone out over the phone.

Verbs denoting presence: "Imasu" and "arimasu", Particle "ni (in)", "Dare (who?)", Adverbs ("Chikaku ni ~ "), Particle "dare mo (negative ~ no one)" , Dare ka (anyone), dare ga (who) , Nani ka (anything) , nani ga (what) - ~ya (and) ~ nado (etc.) and Asking for directions.

Counters and Counting suffixes.

Introduction to Adjectives (na and ii type), Different usages of adjectives, Comparison, Likes and dislikes and Going to a trip.

Need and desire (ga hoshii), Wanting to … (Tabeti desu), Going for a certain purpose (mi –ni ikimasu) and Choosing from a menu.
Verb groups, I, II and III and Exercises to group verbs.

Please do (te kudasai), Present continuous tenses (te imasu), Shall I? (~ mashou ka) and Describing a natural phenomenon (It is raining).

To grant permission (~te mo ii desu), Asking for permission (~ te mo ii desu ka) and Should not do (~ te waikemasen) Describing a continuing state and Describing a habitual action.

Roleplays in Japanese.

A demonstration on usage of chopsticks and Japanese tea party.

Total P: 60 periods

TEXT BOOK

1. '*Minna no nohongo'.* – Romaji ban (first 10 lessons of this book).

REFERENCE

1. '*Minna no Nihongo I Honsatsu Roma – ji ban (Main Textbook Romanized Version)'.* International publisher– 3A Corporation,Tokyo, Indian distributor – Goyal Publishers & Distributors, New Delhi.

MANDATORY COURSES
25MC001 ENVIRONMENTAL SCIENCES
(Common to B.E./B.Tech. Programmes)

2 0 0 0

INTRODUCTION TO ENVIRONMENT: Environment - Definition, scope and importance - Types and composition of atmosphere - particles, ions and radicals - Ozone layer - significance, formation and depletion – Ecosystems - Structure and functions, components, energy flow, food chains, food web, Biodiversity-levels, values and threats - India as a mega-diversity nation, hotspots of biodiversity, endangered and endemic species of India, conservation of biodiversity. **(6)**

ENERGY RESOURCES: Introduction - National and International status - exploitation - sustainable strategies - Fossil fuels - classification, composition, physico-chemical characteristics and energy content of coal, petroleum and natural gas; solar energy - introduction, harnessing strategies. Wind energy - availability, wind power plants, wind energy conversion systems, site characteristics, and types of wind turbines - Supporting renewable energy resources - tidal, geothermal, hydroelectric. **(6)**

ENVIRONMENTAL POLLUTION: Definition, Sources, causes, impacts and control measures of air pollution, water pollution, soil pollution, marine pollution, noise pollution, thermal pollution, nuclear hazards, RF hazards, Role of an individual in prevention of pollution - Disaster Management: Floods, earthquake, cyclone and landslides - Case studies, consequences and rescue measures. **(6)**

WASTE MANAGEMENT: Waste water - Characteristics of domestic and industrial wastewater - COD and BOD, Various stages of treatment - primary, secondary, tertiary treatment- Biological and advanced oxidation processes. Solid waste management - Characteristics of municipal solid waste (MSW), biomedical, automobile and e-wastes and their management, landfills, incineration, pyrolysis, gasification and composting**.** **(6)**

SOCIAL ISSUES AND THE ENVIRONMENT: Environmentally Sustainable work practices - Rain water harvesting, Role of non-governmental organizations - Human ethics and rights - impact on environment and human health, role of information technology on environment and human kind. Green IT policies, Process of EIA - ISO 14000 - Legislation - Environment protection act - Air (Prevention and Control of Pollution) act - Water (Prevention and control of Pollution) act - Wildlife protection act - Forest conservation act. **(6)**

Total L: 30 periods

TEXT BOOKS

1. Gilbert M. Masters, '*Introduction to Environmental Engineering and Science*'. Pearson Education, New Delhi, 2004.
2. Deswal S and Deswal A, '*A Basic Course in Environmental Studies*'. Dhanpat Rai and Co, New Delhi, 2004.

REFERENCES

1. Benny Joseph, '*Environmental Science and Engineering*'. Tata McGraw - Hill, New Delhi, 2006.
2. Koteswara Rao M V R, '*Energy Resources: Conventional & Non - Conventional*'. BSP Publications, New Delhi, 2006.
3. Botkin and Keller, '*Environmental Science*'. Wiley India Private Limited, New Delhi, 2013.

COURSE OUTCOMES:

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the basic concepts of environment, energy sources and waste management	**K2**
CO2	Use different renewable energy resources and environment protection measures for sustainable development	**K3**
CO3	Conduct a case study and real-time environmental issues and present as a team	**-**

COs-POs & PSOs MAPPING:

CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PSO1**	**PSO2**
CO1													
CO2	1												
CO3						2	2	2			2		
	1					**2**	**2**	**2**			**2**		

1-low, 2-medium, 3-high

25MC002 INDIAN CONSTITUTION
(Common to B.E./B.Tech. Programmes)

2 0 0 0

INTRODUCTION: Evolution of Indian Constitution; significance of constitution; Composition; Preamble and its Philosophy. **(4)**

RIGHTS, DUTIES AND DIRECTIVE PRINCIPLES: Fundamental Rights - Writs and Duties, Directive Principles of State Policy. **(5)**

UNION GOVERNMENT: Union Government, President and Vice President, Houses of the Parliament and their functions; Types of Bills, Stages of passing of Bill into an Act, Veto Power, Constitution Amendment Procedure, Various Amendments made and their significance for India. **(6)**

STATE GOVERNMENT AND FEDERALISM: Composition of State Legislature; Powers, Functions and Position of Governor, Function of Chief Ministers, Council of Ministers; The Indian Federal System, Administrative Relationship between Union and States. **(8)**

JUDICIARY: Supreme Court, High Court; District Court and Lower Courts - Functions and Powers - Judges - Qualifications and Powers - Judicial Review. **(7)**

Total L: 30 periods

TEXT BOOKS

1. Subash C Kashyap, '*Our Political System*'. National Book Trust, 2011.
2. Praveenkumar Mellalli E, '*Constitution of India, Professional Ethics and Human Rights*'. Sage Publications India Pvt. Ltd., 2015.

REFERENCES

1. Briji Kishore Sharma, '*Introduction to the Constitution of India*'. Prentice Hall of India, 2010.
2. Basu D D, '*Introduction to the Constitution of India*'. Prentice Hall of India, 2016.
3. Jain. M C, '*The Constitution of India*'. Law House, New Delhi, 2001.
4. Shukla V N, '*Constitution of India*'. Eastern Book Company Ltd., New Delhi, 2011.

COURSE OUTCOMES:

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the evolution, significance, and philosophy of the Indian Constitution, including its Preamble, composition, and core principles.	**K2**
CO2	Analyze the structure, powers, and functions of the Union and State Governments, including the roles of the President, Parliament, Governor, and Council of Ministers, as well as the legislative process, types of bills, and constitutional amendments.	**K3**
CO3	Conduct a case study on the Indian Constitution, demonstrating understanding of its evolution, fundamental rights and duties, structure of Union and State governments, federal system, and the role of the judiciary in governance.	**-**

COs-POs & PSOs MAPPING:

CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PSO1**	**PSO2**
CO1													
CO2	2												
CO3								2	2		2		
	2							**2**	**2**		**2**		

1-low, 2-medium, 3-high

25MC003 INDUSTRIAL SAFETY
(Common to B.E./B.Tech. Programmes)

2 0 0 0

SAFETY TERMINOLOGIES: Hazard - Types of Hazards - Risk-Hierarchy of Hazards Control Measures-Lead indicators - lag Indicators - Flammability - Toxicity Time-weighted Average (TWA) - Threshold Limit Value (TLV) - Short Term Exposure Limit (STEL) - Immediately dangerous to life or health (IDLH) - acute and chronic Effects - Routes of Chemical Entry - Personnel Protective Equipment - Health and Safety Policy - Material Safety Data Sheet MSDS. **(6)**

STANDARDS AND REGULATIONS: Indian Factories Act 1948 - Health - Safety - Hazardous materials and Welfare - ISO 45001:2018 occupational health and safety (OH&S) - Occupational Safety and Health Audit IS14489:1998 - Hazard Identification and Risk Analysis - code of practice IS 15656:2006. **(6)**

SAFETY ACTIVITIES: Toolbox Talk - Role of safety Committee - Responsibilities of Safety Officers and Safety Representatives - Safety Training and Safety Incentives - Mock Drills - On-site Emergency Action Plan - Off-site Emergency Action Plan - Safety poster and Display - Human Error Assessment. **(6)**

WORKPLACE HEALTH AND SAFETY: Noise hazard - Particulate matter - musculoskeletal disorder improper sitting poster and lifting Ergonomics RULE & REBA - Unsafe act & Unsafe Condition - Electrical Hazards - Crane Safety - Toxic gas release. **(6)**

HAZARD IDENTIFICATION TECHNIQUES: Job Safety Analysis - Preliminary Hazard Analysis - Failure mode and Effects Analysis - Hazard and Operability - Fault Tree Analysis - Event Tree Analysis Qualitative and Quantitative Risk Assessment - Check list Analysis - Root cause analysis - What-if Analysis - and Hazard Identification and Risk Assessment. **(6)**

Total L: 30 periods

TEXT BOOKS

1. Jain R. K. and Sunil S. Rao, '*Industrial Safety, Health and Environment Management Systems*'. Khanna Publisher, 4th Edition, 2000.
2. Deshmukh L. M., *'Industrial Safety Management: Hazard Identification and Risk Control'*. McGraw-Hill Education, 2007.

REFERENCES

1. John Ridley, John Channing, '*Safety at Work*'. Routledge, 7th Edition, 2008.
2. Dan Petersen, '*Techniques of Safety Management*: *A System Approach*'. Amer Society of Safety Engineers, 4th Edition, 2003.

COURSE OUTCOMES

At the end of the course, students will be able to		**Bloom's Level**
CO1	Describe the safety protocols and standard operating procedures in industrial settings to ensure compliance with safety regulations and minimize hazards in the workplace.	**K2**
CO2	Implement and test emergency response plans tailored to the industrial environments, ensuring effective action during emergencies such as fires, chemical spills or equipment malfunctions.	**K3**
CO3	Review and present on risk assessments and hazards using industry-specific tools to identify potential safety risks and choose appropriate corrective actions to prevent accidents and injuries.	-

COs - POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1													
CO2	2					2	2						
CO3						1	1	1	1		1		1
	2					**2**	**2**	**1**	**1**		**1**		**1**

1 - low, 2- medium, 3 - high

25MC004 DISASTER RISK REDUCTION AND MANAGEMENT
(Common to B.E./B.Tech. Programmes)

2 0 0 0

HAZARDS, VULNERABILITY AND DISASTER RISKS: Definition: Disaster, Hazard, Vulnerability, Resilience, Risks – Types of Disasters: Natural, Human induced, Climate change induced – Earthquake, Landslide, Flood, Drought, Fire, etc. – Technological disasters - Structural collapse, Industrial accidents, oil spills - Causes, Impacts including social, Economic, political, environmental, health, psychosocial, etc.- Disaster vulnerability profile of India and Tamil Nadu - Global trends in disasters: urban disasters, pandemics, Complex emergencies, Inter relations between Disasters and Sustainable development Goals. **(6)**

DISASTER RISK REDUCTION (DRR): Sendai Framework for Disaster Risk Reduction, Disaster cycle - Community Based DRR, Structural – Non-structural measures, Roles and responsibilities of - community, Panchayati Raj Institutions / Urban Local Bodies (PRIs/ULBs), States, Centre, and other stakeholders - Early Warning System – Relevance of indigenous Knowledge, appropriate technology and Local resources. **(6)**

DISASTER MANAGEMENT: Components of Disaster Management – Preparedness of rescue and relief, mitigation, rehabilitation and reconstruction - Disaster Risk Management and post disaster management – Compensation and Insurance- Disaster Management Act (2005) and Policy - Institutional Processes and Framework at State and Central Level - (NDMA – SDMA – DDMA – NRDF - Civic Volunteers). **(6)**

TOOLS AND TECHNOLOGY FOR DISASTER MANAGEMENT: Early warning systems - Components of Disaster Relief: Water, Food, Sanitation, Shelter, Health, Waste Management, Institutional arrangements (Mitigation, Response and Preparedness) – Role of GIS and Information Technology in Disaster Management – Disaster Damage Assessment - Elements of Climate Resilient Development – Standard operation Procedure for disaster response – Financial planning for disaster Management. **(6)**

DISASTER MANAGEMENT: CASE STUDIES: Case studies in the context of disasters - Landslide Hazard Zonation, Earthquake Vulnerability Assessment of Buildings and Infrastructure, Drought Assessment, Coastal Flooding, Storm Surge Assessment, Floods: Fluvial and Pluvial Flooding, Forest Fire, Man Made disasters. **(6)**

Total L: 30 periods

TEXTBOOKS

1. Thomas D. Schneid, and Larry Collins, '*Disaster Management and Preparedness*'. CRC Publications, 2016.
2. R. Singh, '*Disaster Management Guidelines: Earthquakes, Landslides, Avalanches and Tsunami'*. Horizon Press Publications, 2017.
3. J. P. Singhal, '*Disaster Management'*. Laxmi Publications, 2024.
4. T. Bhattacharya, '*Disaster Science and Management'*. McGraw Hill India Education Pvt. Ltd., 2012.

REFERENCES

1. Government of India, '*Disaster Management Act'*. New Delhi, 2005.
2. Government of India, '*National Disaster Management Policy'*. New Delhi, 2009.
3. R. Shaw, '*Community based Disaster risk reduction*'. Natural Hazard Science, Oxford Research Encyclopedias, 2016.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Summarize the concepts, tools, technologies and strategies for disaster risk reduction and management.	**K2**

COs-POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1					1	1	1	1	1		1		
					1	**1**	**1**	**1**	**1**		**1**		

1-low, 2-medium, 3-high

25MC005 INDIAN KNOWLEDGE SYSTEMS FOR SUSTAINABLE HUMAN DEVELOPMENT
(Common to B.E./B.Tech. Programmes)

2 0 0 0

INTRODUCTION TO INDIAN KNOWLEDGE SYSTEMS & HUMAN DEVELOPMENT (6)

- Meaning, scope, and significance of Indian Knowledge Systems
- Indian view of life: Purusharthas (Dharma, Artha, Kama, Moksha)
- Relationship between individual, society, and nature
- Relevance of IKS in modern education and engineering

INDIAN PERSPECTIVES ON SUSTAINABILITY & ECOLOGY (6)

- Indian worldview on nature: Pancha Mahabhutas
- Traditional water management, agriculture, and resource conservation
- Sustainable lifestyles in Indian traditions
- Comparison with modern sustainable development goals (SDGs)

HEALTH, WELL-BEING & HUMAN POTENTIAL IN IKS (6)

- Indian concept of health: Body–Mind–Spirit integration
- Introduction to Ayurveda & Yoga for sustainable health
- Mental health and emotional balance in Indian wisdom
- Food, nutrition, and lifestyle from IKS perspective
- Preventive healthcare through traditional knowledge

ETHICS, VALUES & SOCIAL SUSTAINABILITY (5)

- Work ethics and leadership in Indian tradition
- Gender dignity and social justice in Indian thought
- Professional ethics through IKS

IKS, TECHNOLOGY & SUSTAINABLE DEVELOPMENT (7)

- Science and technology in ancient India
- Sustainable engineering in traditional architecture and town planning
- Metallurgy, textiles, and indigenous manufacturing systems
- IKS and modern innovation ecosystems and ethical responsibility
- Role of engineers in sustainable human development

Total L: 30 periods

TEXTBOOKS

1. Dharampal, '*Indian Science and Technology in the Eighteenth Century'.*
2. Kapil Kapoor, '*Indian Knowledge Systems'.*
3. S Radhakrishnan, '*Indian Philosophy'.*
4. B K Matilal, '*The Character of Logic in India'.*
5. UGC–IKS Documents and AICTE IKS Modules
6. David Frawley, '*Ayurvedic Healing'.*
7. *IIT-IKS Foundation Course Materials*
8. Ministry of Education – IKS Portal Resources

COURSE OUTCOMES:

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Explain the core concepts and evolution of Indian Knowledge Systems.	**K2**
CO2	Interpret Indian views on human development, ethics, and sustainability.	**K3**
CO3	Apply IKS principles to sustainable living and professional practices.	**K3**
CO4	Demonstrate social, ecological, and ethical responsibility in real-life situations.	**K3**

COs - POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1						3	2				3
CO2						3	3				2
CO3			2			3	3		2	2	
CO4				2		3	3	2			
			2	**2**		**3**	**3**	**2**	**2**	**2**	**3**

1 - low, 2- medium, 3 - high

25MC006 STRATEGIC DATA MANAGEMENT AND GOVERNANCE
(Common to B.E./B.Tech. Programmes)

2 0 0 0

INTRODUCTION TO STRATEGIC DATA MANAGEMENT: Data and Its strategic Value; Types of Data, Data as a Source of Competitive Advantage; The Data Lifecycle; Aligning Data Management with Business Goals; Data Management challenges in Contemporary Organizations; Overview of Data driven Decision making. **(6)**

DATA GOVERNANCE AND FRAMEWORK: Introduction to Data Governance; Data Management Frameworks (DAMA-DMBOK, COBIT, ISO Data management Standards); Structuring Data Leadership Roles; Implementation of Data Policies and Standards. Master Data Management (MDM) and API Integration. **(6)**

DATA PRIVACY, ETHICS AND QUALITY MANAGEMENT: Data Privacy and Protection Regulations (DPDP Act, GDPR); Principles and Importance of Ethical data management; Ethical decision-making models; Key data Quality Dimensions; Role of Data Quality in Effective Decision-Making; Tools and techniques for ensuring data quality. **(6)**

DATA LITERACY AND ORGANIZATIONAL TRANSFORMATION: Role of Data Literacy in Modern Organisations; Strategies for implementing data-driven transformation; Communication and training for Overcoming Resistance to Change; Cross-Functional Collaboration across Organisational Departments; Fostering a Data-Driven Culture. **(6)**

DATA ARCHITECTURE AND RECENT TRENDS: Business Intelligence (BI) Architecture; Big Data Ecosystem; AI/ML data pipelines in Business enterprises, Strategic Planning for Data Infrastructure; Build vs. Buy Decisions for Data Platforms; Generative AI in Data Management. **(6)**

Total L: 30 periods

TEXT BOOKS

1. Ladley, J. & Ladley, A. '*Data Governance: How to Design, Deploy, and Sustain an Effective Data Governance Program* (3rd ed.)'. Morgan Kaufmann (Elsevier).
2. Eryurek, E., Gilad, U., Kaye, B., Sicular, S., & Schur, E. (2021). '*Data Governance: The Definitive Guide: People, Processes, and Tools to Operationalize Data Trustworthiness'*. O'Reilly Media.

REFERENCE BOOKS

1. Plotkin, D. (2020). '*Data Stewardship: An Actionable Guide to Effective Data Management and Data Governance* (2nd ed.)'. Academic Press (Elsevier).2020.
2. DAMA International. (2024). '*DAMA-DMBOK: Data Management Body of Knowledge* (2nd ed. Revised)'. Technics Publications.2024.
3. Loukides, M., Mason, H., & Patil, D. J. (2018). '*Ethics and Data Science'*. O'Reilly Media.2018.

COURSE OUTCOMES:

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Understanding the Management role in leveraging data for competitive advantage and strategic decision-making.	**K2**
CO2	Applying governance frameworks to design policies, roles, and standards.	**K3**
CO3	Analyzing ethical challenges in data management to improve data-literacy across organization departments.	**K4**
CO4	Evaluating enterprise data architecture and modern platforms for making strategic decisions on Data Management.	**K5**

COs - POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	2	2				3				3	3
CO2			2			2	3	2		3	
CO3						3	3		3		2
CO4	2		2	2	3	2				3	2
	2	**2**	**2**	**2**	**3**	**3**	**3**	**2**	**3**	**3**	**3**

1 - low, 2- medium, 3 - high

25MC007 PROFESSIONAL ETHICS
(Common to B.E./B.Tech. Programmes)

2 0 0 0

INTRODUCTION TO THE CONCEPTS OF ETHICS: Introduction to ethics – meaning, definition, dimensions of ethics, determinants and consequences of ethics; ethical dilemmas – types of ethical dilemmas. Morality – definition, types of moral values, levels of moral development. Ethics Versus morality. Professional Ethics – definition, components, types and significance. **(6)**

PSYCHOLOGY OF ETHICAL BEHAVIOUR: Attitude – components of attitude in promoting ethical behaviour; Motivation – human drives as a trigger for ethical behaviour; Role of Cognitive biases (self-serving bias, incrementalism and overconfidence bias) in shaping ethical behaviour.

(6)

WORKPLACE ETHICAL BEHAVIOURS: Integrity, honesty, accountability, responsibility, fairness, civic virtue, conscientiousness, empathy, self-confidence, character, respect for others, transparency, loyalty. **(6)**

APPLICATION AREAS OF ETHICS: Application of Ethics in different fields - Engineering ethics, medical ethics, financial ethics, technological ethics, scientific ethics, legal ethics, business ethics, environmental ethics. **(6)**

INDICATORS OF PROFESSIONAL ETHICS IN ORGANIZATIONS: Corporate Governance – Definition, importance, Theories of Corporate Governance: Agency Theory, Stewardship theory, Stakeholder Theory, Resource Dependency Theory.

Corporate Social Responsibility – Definition, Models of CSR. **(6)**

Total L: 30 periods

TEXT BOOKS

1. Govindarajan M, Natarajan S and Senthil Kumar VS, *'Engineering Ethics'*. Prentice Hall of India, New Delhi, 2013.
2. Stephen P. Robbins, *'Essentials of Organizational Behaviour'.* Pearson Education, 2023.
3. Subramanian. R. '*Professional Ethics'*. Oxford Publication, 2013.
4. Nagarasan. R.S. *Professional Ethics and Human Values'.* New Age International Publications, 2006.

COURSE OUTCOMES:

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Understand the fundamental concepts of ethics, morality and the psychological principles behind ethical behaviour.	**K2**
CO2	Demonstrate the application of core ethical values in the different situations of the workplace.	**K3**
CO3	Analyze the ethical issues, challenges in different professional domains and the application of ethical principles to solve the issues	**K4**
CO4	Evaluate the relevance and effectiveness of corporate governance theories and CSR models in addressing ethical and managerial challenges in organizations.	**K5**

COs - POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1						2	3				3
CO2						2	3	3	2		
CO3		2	2			3	3				
CO4						3	3	2		3	2
		2	**2**			**3**	**3**	**3**	**2**	**3**	**3**

1 - low, 2- medium, 3 - high

25HSO01 PHYSICAL EDUCATION
(Common to all B.E./B.Tech. Programmes)

0 0 2 0

INTRODUCTION TO PHYSICAL EDUCATION: Importance and scope of physical education, Physical fitness and exercise fundamentals, Conditioning and calisthenics exercises, Posture, flexibility, balance, and coordination, Warm-up, cool-down, and relaxation techniques, Basics of injury prevention and first aid practices, Importance of teamwork, discipline, and sportsmanship.

BASIC GYMNASTICS AND MOVEMENT EXERCISES: Warm-up exercises, conditioning drills, movement education, balance exercises, floor exercises, tumbling basics, swinging and grip exercises, simple apparatus activities, lead-up games, coordination drills, and safety practices.

SKILL DEVELOPMENT IN ANY ONE OF THE FOLLOWING OUTDOOR GAMES: Basketball, Volleyball, Ball Badminton, and Football.

SKILL DEVELOPMENT IN ANY ONE OF THE FOLLOWING INDOOR GAMES: Shuttle Badminton, Chess, Carrom, and Table Tennis.

Total P: 30 Periods

WEIGHTAGE
Continuous Assessment: 100%

ASSESSMENT
Attendance and Participation – 70%
Practical Assessment of Physical Fitness and Sports Skills – 30%

REFERENCES:

1. Singh, A. '*Essentials of Physical Education'*. Kalyani Publishers.
2. Kamlesh, M.L. '*Psychology in Physical Education and Sport'*.
3. Mangal, S.K. '*Psychology of Sports Performance'*. Sports Publication.
4. Kandappan, K. '*Foundations of Physical Education'*. Friends Publications.

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Demonstrate basic movement, balance, coordination, and fitness exercises with proper technique and safety awareness.	**K2**
CO2	Demonstrate physical fitness parameters such as endurance, agility, flexibility, speed, and strength through regular participation in physical activities.	**K2**
CO3	Develop teamwork, discipline, leadership qualities, and sportsmanship through participation in games and group activities.	**K3**
CO4	Adopt active lifestyle habits and participate regularly in physical and recreational activities for overall well-being.	**K3**

COs - POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1							2	3			2
CO2						2		3			3
CO3							3	3	2		2
CO4						2	2	3			3
						2	**3**	**3**	**2**		**3**

1 - low, 2- medium, 3 - high

25HSO02 YOGA AND HEALTH
(Common to all B.E./B.Tech. Programmes)

0 0 2 0

INTRODUCTION TO YOGA: Yoga as a tool for physical and mental wellness, Importance of posture, breathing, relaxation, and mindfulness, Role of yoga in stress management and lifestyle balance, Basic guidelines and safety precautions for yoga practice, Introduction to healthy habits including sleep hygiene, nutrition, and digital well-being

SURYA NAMASKAR – TYPES OF POSTURES: Pranamasana (Prayer Pose), Hastauttanasana (Raised Arms Pose), Padahastasana / Hastapadasana (Standing Forward Bend), Ashwa Sanchalanasana (Equestrian Pose), Dandasana (Plank Pose), Ashtanga Namaskara (Eight-Limbed Salutation), Bhujangasana (Cobra Pose), Adho Mukha Svanasana (Downward-Facing Dog), Ashwa Sanchalanasana (Equestrian Pose), Padahastasana (Standing Forward Bend), Hastauttanasana (Raised Arms Pose), and Pranamasana (Prayer Pose).

BASIC ASANAS:

1. Standing – Tadasana, Utkatasana, Trikonasana, Virabhadrasana, and Vrikshasana
2. Sitting – Sukhasana, Dandasana, Paschimottanasana, Vajrasana, and Gomukhasana
3. Prone Position – Bhujangasana, Salabhasana, Dhanurasana, Makarasana, and Urdhva Mukha Svanasana
4. Supine Position – Apanasana, Viparita Karani, Supta Baddha Konasana, Ardha Halasana, and Savasana

PRANAYAMA TECHNIQUES & MUDRAS: Ujjayi, Nadi Shodhana, Sheetali, Sitkari, Kapalabhati, Bhastrika, and Bhramari. Gyan Mudra (Chin Mudra), Prana Mudra, Apana Mudra, Vayu Mudra, Prithvi Mudra, Varun Mudra, Surya Mudra, Shunya Mudra, Linga Mudra, and Anjali Mudra.

Total P:30 Periods

WEIGHTAGE: Continuous Assessment: 100%
ASSESSMENT METHODOLOGY: Attendance and Participation – 70%, Practical Demonstration (Surya Namaskar, Asanas, and Pranayama) – 30%

REFERENCES:

1. *'The Heart of Yoga'.* – T.K.V. Desikachar
2. *'Yoga for Holistic Health'.* – Dr. H.R. Nagendra
3. *'Light on Yoga'.* – B.K.S. Iyengar
4. *'Asana Pranayama Mudra Bandha'.* – Swami Satyananda Saraswati

COURSE OUTCOMES

At the end of the course, students will be able to:		**Bloom's Level**
CO1	Demonstrate Surya Namaskar and basic asanas in standing, sitting, prone, and supine positions with correct alignment and breathing.	**K2**
CO2	Demonstrate pranayama techniques for stress management and respiratory health.	**K2**
CO3	Improve flexibility, balance, posture, breathing efficiency, and relaxation through regular yoga practice.	**K3**
CO4	Apply principles of healthy living such as nutrition, sleep hygiene, and digital well-being in daily life.	**K3**

COs - POs & PSOs MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1							2	3			2
CO2						2	2				2
CO3						2	2	3			3
CO4						3	2	2			3
						3	**2**	**3**			**3**

1 - low, 2 - medium, 3 - high