



R.M.D. ENGINEERING COLLEGE

(An Autonomous Institution)

REGULATIONS–2024

CHOICE BASED CREDIT SYSTEM

B.E. COMPUTER SCIENCE AND ENGINEERING

(For the students admitted in the Academic year 2024-2025)

PROGRAM EDUCATIONAL OBJECTIVES

PEO 1

Graduates will apply the principles of Computer Science and Engineering to develop effective computing solutions for real-world problems.

PEO 2

Graduates will build successful careers in industry, pursue higher education or research, or become entrepreneurs in computing-related domains.

PEO 3

Graduates will demonstrate ethical values, social responsibility, and contribute to sustainable development for the benefit of society.

PEO 4

Graduates will adapt to emerging technologies through continuous learning and professional development

PROGRAMME OUTCOMES (POs)

On successful completion of the programme, Engineering Graduates will be able to:

PO 1. Engineering Knowledge (K3):

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.

PO 2. Problem Analysis (K4):

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

PO 3. Design/Development of solutions (K5):

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)

PO 4. Conduct investigations of complex problems (K5):

Conduct investigations of complex engineering problems using research-based knowledge including

design of experiments, modelling, analysis & interpretation of data to provide valid conclusions. (WK8).

PO 5. Engineering Tool Usage (K3):

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)

PO 6. The Engineer and The World (A3):

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).

PO 7. Ethics (A2):

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

PO 8. Individual and Collaborative Team Work (A3):

Function effectively as an individual, and as a member or leader in diverse/multi-disciplinary teams.

PO 9. Communication (A3):

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.

PO 10. Project Management and Finance (A3):

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.

PO 11. Life-long Learning (A2):

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)

PROGRAMME SPECIFIC OUTCOMES (PSOs)

After the successful completion of the program, the graduates will be able to:

PSO 1:

Apply concepts of intelligent computing, data analytics, and advanced programming techniques to analyze and develop innovative computing solutions.

PSO 2:

Design and implement scalable software applications using modern development tools and frameworks, while adapting to emerging technologies through continuous learning.

PSO 3:

Apply Computer Science and Engineering principles to address industrial requirements and societal challenges, emphasizing sustainability, ethics, and social responsibility.

Mapping of POs/PSOs to PEOs

Contribution

1: Reasonable

2: Significant

3: Strong

PEOs & POs

The B.E. Computer Science and Engineering program outcomes leading to the achievement of the objectives are summarized in the following table.

PROGRAM EDUCATIONAL OBJECTIVES	PROGRAM OUTCOMES											
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
I	3	3	3	3	2	2	2	1	1	1	1	1
II	3	3	3	3	2	1	1	1	3	3	1	3
III	3	3	3	2	2	3	3	3	3	3	2	1
IV	3	3	3	3	2	2	2	2	3	2	2	3

PROGRAM EDUCATIONAL OBJECTIVES	PROGRAM SPECIFIC OUTCOMES		
	PSO 1	PSO 2	PSO 3
I	3	3	2
II	3	3	3
III	2	2	3
IV	3	3	2

MAPPING OF COURSE OUTCOMES WITH PROGRAMME OUTCOMES

A broad relation between the Course Outcomes and Programme Outcomes is given in the following table.

Contribution

1: Reasonable

2: Significant

3: Strong



R.M.D. ENGINEERING COLLEGE
B.E. COMPUTER SCIENCE AND ENGINEERING
REGULATION – 2024
CURRICULUM & SYLLABUS
SEMESTER I to VIII

CHOICE BASED CREDIT SYSTEM
 (For the Students admitted in the Academic Year 2024-25)

SEMESTER – I								
Sl. No.	Course Code	Course Title	Category	Contact Periods	L	T	P	C
THEORY COURSES								
1.	24MA101	Matrices and Calculus	BSC	4	3	1	0	4
2.	24GE101	Heritage of Tamils	HSMC	1	1	0	0	1
3.	24HS111	Interpersonal skills, Psychometric Analysis and Career Development	EEC	1	1	0	0	1
THEORY COURSES WITH LABORATORY COMPONENT								
4.	24CS101	Programming in C++ (Lab Integrated)	ESC	6	3	0	3	4.5
5.	24CS102	Software Development Practices (Lab Integrated)	ESC	6	3	0	3	4.5
6.	24CH101	Engineering Chemistry (Lab Integrated)	BSC	5	3	0	2	4
7.	24EC101	Digital Principles and System Design (Lab Integrated)	ESC	5	3	0	2	4
LABORATORY COURSE								
8.	24GE111	Idea Lab I (Non Credit)	EEC	1	0	0	1	0
MANDATORY COURSES								
9.	24MC101	Students Induction Program (Non Credit)	MC	3 Weeks				
10.	24MC102	Programming in C (Non Credit)	MC	40 Periods				
TOTAL				29	17	1	11	23

SEMESTER – II								
Sl. No.	Course Code	Course Title	Category	Contact Periods	L	T	P	C
THEORY COURSES								
1	24GE201	Tamils and Technology	HSMC	1	1	0	0	1
2	24HS211	Innovation and Creativity Skills Development	EEC	1	1	0	0	1
THEORY COURSES WITH LABORATORY COMPONENT								
3	24MA201	Linear Algebra and Applications (Lab Integrated)	BSC	5	3	0	2	4
4	24CS201	Data Structures (Lab Integrated)	ESC	6	3	0	3	4.5
5	24CS202	Java Programming (Lab Integrated)	ESC	6	3	0	3	4.5
6	24PH201	Physics for Information Science (Lab Integrated)	BSC	5	3	0	2	4
7	24AM201	Introduction to Artificial Intelligence (Lab Integrated)	ESC	4	2	0	2	3
LABORATORY COURSE								
8	24GE211	Idea Lab II	EEC	2	0	0	2	1
MANDATORY COURSES								
9	24MC201	Yoga for Stress Management (Non Credit)	AC	1	0	0	1	0
10	24CH102	Environmental Science & Sustainability (Non Credit)	MC	2	2	0	0	0
TOTAL				33	18	0	15	23

SEMESTER – III								
Sl. No.	Course Code	Course Title	Category	Contact Periods	L	T	P	C
THEORY COURSES								
1	24MA301	Discrete Mathematics	BSC	4	3	1	0	4
2	24GE301	Universal Human Values II: Understanding Harmony	HSMC	3	2	1	0	3
3	24CS301	Computer Organization and Architecture	ESC	3	3	0	0	3
THEORY COURSES WITH LABORATORY COMPONENT								
4	24CS302	Advanced Java Programming	PCC	6	3	0	3	4.5
5	24CS303	Database Management Systems	PCC	6	3	0	3	4.5
6	24CS304	Operating Systems	PCC	4	2	0	2	3
LABORATORY COURSES								
7	24GE311	Product Development Lab - 1	EEC	2	0	0	2	1
8	24CS311	Aptitude and Coding Skills I	EEC	3	0	0	3	1.5
9	24CS312	Internship/Seminar (1 Week)	EEC	1	0	0	1	0.5
MANDATORY COURSES								
10	24MC301	Indian Constitution (Non Credit)	MC	1	1	0	0	0
TOTAL				33	17	2	14	25

SEMESTER – IV								
Sl. No.	Course Code	Course Title	Category	Contact Periods	L	T	P	C
THEORY COURSES WITH LABORATORY COMPONENT								
1.	24MA401	Probability and Statistics	BSC	5	3	0	2	4
2.	24CS401	Computer Networks	PCC	5	3	0	2	4
3.	24CS402	Design and Analysis of Algorithms	PCC	5	3	0	2	4
4.		Professional Elective I	PEC	4	2	0	2	3
LABORATORY COURSES WITH THEORY COMPONENT								
5.	24IT402	Web Development Frameworks	PCC	6	3	0	3	4.5
LABORATORY COURSES								
6.	24GE411	Product Development Lab - 2	EEC	2	0	0	2	1
7.	24CS411	Aptitude and Coding Skills II	EEC	3	0	0	3	1.5
AUDIT COURSES								
8	24MC401	Value Education (Non Credit)	AC	1	1	0	0	0
TOTAL				31	15	0	16	22
FOR HONOURS DEGREE								
	24CS981	Capstone Design Project-Level I	EEC	12	0	0	12	6
OR								
		Honour Elective - I	PEC	3	3	0	0	3
		Honour Elective - II	PEC	3	3	0	0	3
OR								
		Honour Elective - I	PEC	3	3	0	0	3
FOR MINOR DEGREE								
		Minor Elective - I	OEC	3	3	0	0	3
		Minor Elective - II	OEC	3	3	0	0	3

SEMESTER – V								
Sl. No.	Course Code	Course Title	Category	Contact Periods	L	T	P	C
THEORY COURSES								
1.	24CS501	Theory of Computation	PCC	3	3	0	0	3
2.		Professional Elective II	PEC	3	3	0	0	3
THEORY COURSES WITH LABORATORY COMPONENT								
3.	24CS502	Distributed and Cloud Computing	PCC	4	2	0	2	3
4.	24CS503	Machine Learning	PCC	5	3	0	2	4
5.		Professional Elective III	PEC	4	2	0	2	3
LABORATORY COURSES								
6.	24HS511	Professional Communication - I (TOEFL)	HSMC	4	0	0	4	2
7.	24GE511	Product Development Lab – 3	EEC	2	0	0	2	1
8.	24CS511	Advanced Aptitude and Coding Skills I	EEC	3	0	0	3	1.5
9.	24CS512	Internship/Seminar (2 Weeks)	EEC	2	0	0	2	1
AUDIT COURSES								
10.	24MC501	Essence of Indian Traditional Knowledge (Non Credit)	MC	1	1	0	0	0
TOTAL				31	14	0	17	21.5
FOR HONOURS DEGREE								
	24CS982	Capstone Design Project-Level II	EEC	12	0	0	12	6
OR								
		Honour Elective - III	PEC	3	3	0	0	3
		Honour Elective - IV	PEC	3	3	0	0	3
OR								
		Honour Elective - II	PEC	3	3	0	0	3
FOR MINOR DEGREE								
		Minor Elective - III	OEC	3	3	0	0	3
		Minor Elective - IV	OEC	3	3	0	0	3

SEMESTER – VI								
Sl. No.	Course Code	Course Title	Category	Contact Periods	L	T	P	C
THEORY COURSES								
1.		Professional Elective IV	PEC	3	3	0	0	3
2.		Open Elective I	OEC	3	3	0	0	3
THEORY COURSES WITH LABORATORY COMPONENT								
3.	24CS601	Compiler Design	PCC	5	3	0	2	4
4.	24CS602	Cryptography and Cyber Security	PCC	5	3	0	2	4
5.	24CS603	Object Oriented Software Engineering	PCC	4	2	0	2	3
6.	24CS604	Design Thinking	HSMC	3	1	0	2	2
LABORATORY COURSES								
7.	24HS611	Professional Communication - II (TOEFL)	HSMC	2	0	0	2	1
8.	24GE611	Product Development Lab - 4	EEC	2	0	0	2	1
9.	22CS611	Advanced Aptitude and Coding Skills II	EEC	3	0	0	3	1.5
AUDIT COURSES								
		Personality Development (Non Credit)	AC	2	2	0	0	0
TOTAL				32	17	0	15	22.5
FOR HONOURS DEGREE								
	24CS983	Capstone Design Project-Level III	EEC	12	0	0	12	6
OR								
		Honour Elective - V	PEC	3	3	0	0	3
		Honour Elective - VI	PEC	3	3	0	0	3
FOR MINOR DEGREE								
		Honour Elective - III	PEC	3	3	0	0	3
FOR MINOR DEGREE								
		Minor Elective - V	OEC	3	3	0	0	3
		Minor Elective - VI	OEC	3	3	0	0	3

SEMESTER – VII								
Sl. No.	Course Code	Course Title	Category	Contact Periods	L	T	P	C
THEORY COURSES								
1.	24CS702	Wireless and Mobile Communication	PCC	3	3	0	0	3
2.		Professional Elective V	PEC	3	3	0	0	3
3.		Open Elective II	OEC	3	3	0	0	3
THEORY COURSES WITH LABORATORY COMPONENT								
4.	24CS701	Data Analytics	PCC	5	3	0	2	4
5.		Professional Elective VI	PEC	4	2	0	2	3
LABORATORY COURSES								
6.		Internship/Seminar (4 weeks)	EEC	4	0	0	4	2
TOTAL				22	14	0	8	18
FOR HONOURS DEGREE								
		Honour Elective - IV	PEC	3	3	0	0	3
		Capstone Project		12	0	0	12	6

SEMESTER – VIII								
Sl. No.	Course Code	Course Title	Category	Contact Periods	L	T	P	C
EMPLOYABILITY ENHANCEMENT COURSES								
1	24CS811	Project	EEC	16	0	0	16	8
TOTAL				16	0	0	16	8

Credit Summary

S. No.	Subject Area	Credits Per Semester								Credit Total	Percentage
		I	II	III	IV	V	VI	VII	VIII		
1.	HSMC	1	1	3		2	3			10	6.13%
2.	BSC	8	8	4	4					24	14.72%
3.	ESC	13	12	3						28	17.18%
4.	PCC			12	12.5	10	11	7		52.5	32.21%
5.	PEC				3	6	3	6		18	11.04%
6.	OEC						3	3		6	3.68%
7.	EEC	1	2	3	2.5	3.5	2.5	2	8	24.5	15.03%
8	MC/AC	□	□	□		□					
Total		23	23	25	22	21.5	22.5	18	8	163	

HSMC - Humanities and Social Sciences including Management Course; BSC – Basic Science Course; ESC – Engineering Science Course; PCC – Professional Core Course; PEC – Professional Elective Course; OEC – Open Elective Course; EEC – Employability Enhancement Course; MC – Mandatory Course; AC – Audit Course.

ELECTIVE VERTICALS

CYBER SECURITY								
S. No.	Course Code	Course Title	Category	Contact Periods	L	T	P	C
1	24CS901	Ethical Hacking (Lab Integrated)	PEC	4	2	0	2	3
2	24CS902	Social Network Security (Lab Integrated)	PEC	4	2	0	2	3
3	24CS903	Blockchain Technology	PEC	3	3	0	0	3
4	24CS904	Cloud and Data Security	PEC	3	3	0	0	3
5	24CS905	Enterprise Cyber Security	PEC	3	3	0	0	3
6	24CS940	Cyber Security Essentials	PEC	3	3	0	0	3
7	24CS941	AI in Cyber Security	PEC	3	3	0	0	3
8	24IT903	Rest Application Development using Spring Boot and JPA(Lab Integrated)	PEC	4	2	0	2	3
CLOUD COMPUTING								
1	24CS911	Cloud Foundations (Lab Integrated)	PEC	4	2	0	2	3
2	24CS912	Virtualization	PEC	3	3	0	0	3
3	24CS913	Data Engineering in Cloud	PEC	3	3	0	0	3
4	24CS914	Devops (Lab Integrated)	PEC	4	2	0	2	3
5	24CS942	Cloud Architecting	PEC	3	3	0	0	3
6	24CS935	Agentic AI in Cloud	PEC	3	3	0	0	3
7	24IT903	Rest Application Development using Spring Boot and JPA(Lab Integrated)	PEC	4	2	0	2	3
FULL STACK TECHNOLOGY								
1	24CS921	UI/UX Design (Lab Integrated)	PEC	4	2	0	2	3
2	24CS922	Software Testing and Automation	PEC	4	3	0	0	3
3	24CS903	Blockchain Technology	PEC	3	3	0	0	3
4	24CS923	Usability Design of Software Application	PEC	3	3	0	0	3
5	24CS914	Devops (Lab Integrated)	PEC	4	2	0	2	3
6	24CS944	AI Integration in Software Development and Operations	PEC	3	3	0	0	3
7	24IT903	Rest Application Development using Spring Boot and JPA(Lab Integrated)	PEC	4	2	0	2	3
8	24AM001	Foundations of Deep Learning	PEC	3	3	0	0	3
EMERGING TECHNOLOGIES								
1	24CS929	Multimedia Computing	PEC	3	3	0	0	3
2	24CS930	GPU Computing	PEC	3	3	0	0	3
3	24CS931	Introduction to Augmented and Virtual Reality	PEC	3	3	0	0	3
4	24CS932	Digital Marketing	PEC	3	3	0	0	3
5	24CS933	Quantum Computing	PEC	3	3	0	0	3
6	24CS934	Intelligent Robots	PEC	3	3	0	0	3
7	24CS936	Principles of 3D Printing and Design	PEC	3	3	0	0	3
8	24CS937	High Performance Computing	PEC	3	3	0	0	3

HONOUR DOMAIN SUBJECTS

CYBER SECURITY								
S. No.	Course Code	Course Title	Category	Contact Periods	L	T	P	C
1	24CS906	Digital and Mobile Forensics		3	3	0	0	3
2	24CS907	Vulnerability Analysis and Penetration Testing		3	3	0	0	3
3	24CS908	Engineering Secure Software Systems		3	3	0	0	3
4	24CS909	Network Design and Programming		3	3	0	0	3
5	24CS910	Fault Tolerant Computing		3	3	0	0	3
6	24CS925	Web Application Security		3	3	0	0	3
CLOUD COMPUTING								
1	24CS915	Machine Learning for NLP in Cloud		3	3	0	0	3
2	24CS916	Microservice Architecture		3	3	0	0	3
3	24CS917	Software Defined Networks		3	3	0	0	3
4	24CS918	Storage Technologies		3	3	0	0	3
5	24CS919	Cloud Security Foundations		3	3	0	0	3
6	24CS920	Cloud Services Management		3	3	0	0	3
FULL STACK TECHNOLOGY								
1	24CS924	MERN Stack Development (Lab Integrated)		4	2	0	2	3
2	24CS925	Web Application Security		3	3	0	0	3
3	24CS926	Generative AI Fundamentals		3	3	0	0	3
4	24CS927	Mobile Architecture and Application Development (Lab Integrated)		4	2	0	2	3
5	24CS916	Microservice Architecture		3	3	0	0	3
6	24CS943	Introduction to Agentic AI		3	3	0	0	3
DATA SCIENCE								
1	24AM901	Foundations of Data Science (Lab Integrated)		4	2	0	2	3
2		Data Visualization (Lab Integrated)		4	2	0	2	3
3	24AM902	Text and Speech Analytics		3	3	0	0	3
4	24CS926	Generative AI Fundamentals		3	3	0	0	3
5	24AM905	Image and Video Analytics		3	3	0	0	3
6	24AM906	Stream Processing and Analytics		3	3	0	0	3
ARTIFICIAL INTELLIGENCE								
1	24CS938	Soft Computing (Lab Integrated)		4	2	0	2	3
2	24AM933	Knowledge Engineering		3	3	0	0	3
3		Reinforcement Learning		3	3	0	0	3
4	24AM932	Recommender Systems		3	3	0	0	3
5	24CS939	Natural Language Processing		3	3	0	0	3
6	24CS926	Generative AI Fundamentals		3	3	0	0	3

OPEN ELECTIVES OFFERED TO OTHER DEPARTMENTS

S. No.	Course Code	Course Title	Contact Periods	L	T	P	C
1	24CS001	Blockchain Technology	3	3	0	0	3
2	24CS002	Cloud Architecting	3	3	0	0	3
3	24CS003	Software Testing and Automation	3	3	0	0	3
4	24CS004	Digital and Mobile Forensics	3	3	0	0	3

R2024 CURRICULUM

MINOR DEGREE CURRICULUM OFFERED BY DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING (FOR OTHER B.E. / B.Tech PROGRAMMES)

Sl. No.	NAME OF THE MINOR DEGREE WITH SPECIALIZATION
1	Cyber Security
2	Cloud Computing
3	Full Stack Development

CYBER SECURITY

S. No.	Course Code	Course Title	Contact Periods	L	T	P	C
1	24CS005	Computer Networks	3	3	0	0	3
2	24CS006	Cryptography and Cyber Security	3	3	0	0	3
3	24CS007	Ethical Hacking	3	3	0	0	3
4	24CS004	Digital and Mobile Forensics	3	3	0	0	3
5	24CS812	Capstone Project	12	0	0	12	6

CLOUD COMPUTING

S. No.	Course Code	Course Title	Contact Periods	L	T	P	C
1	24CS008	Operating Systems	3	3	0	0	3
2	24CS009	Distributed and Cloud Computing	3	3	0	0	3
3	24CS010	Cloud Foundations	3	3	0	0	3
4	24CS011	Cloud Architecting	3	3	0	0	3
5	24CS812	Capstone Project	12	0	0	12	6

FULL STACK DEVELOPMENT

S. No.	Course Code	Course Title	Contact Periods	L	T	P	C
1	24CS012	Web Development Frameworks(Lab Integrated)	4	2	0	2	3
2	24CS013	MERN Stack Development(Lab Integrated)	4	2	0	2	3
3	24CS014	Mobile Architecture and Application Development(Lab Integrated)	4	2	0	2	3
4	24CS015	Web Application Security	3	3	0	0	3
5	24CS812	Capstone Project	12	0	0	12	6

SEMESTER – I

24MA101	MATRICES AND CALCULUS (Common to B.E./B.Tech., all branches except CSBS)	L	T	P	C
		3	1	0	4
OBJECTIVES: The Course will enable learners to: • Explain the concepts of matrix algebra techniques. • Understand various techniques to solve second and higher order differential equations. • Demonstrate simple applications of functions of several variables and vector calculus. • Comprehend the basic concepts of multiple integrals. • Illustrate elementary ideas of vector calculus.					
UNIT I	MATRICES				12
Eigenvalues and Eigenvectors of a real matrix – Properties of eigenvalues and eigenvectors – Statement and applications of Cayley-Hamilton Theorem – Diagonalization of matrices by orthogonal transformation (excluding similarity transformation) – Reduction of a quadratic form to canonical form by orthogonal transformation - Nature of quadratic forms.					
UNIT II	DIFFERENTIAL EQUATIONS				12
Second and Higher order linear differential equations with constant coefficients - Method of variation of parameters - Homogenous equation of Euler's and Legendre's type - System of simultaneous linear differential equations with constant coefficients.					
UNIT III	FUNCTIONS OF SEVERAL VARIABLES				12
Total derivative - Differentiation of implicit functions - Jacobian and properties - Taylor's series for functions of two variables - Maxima and minima of functions of two variables- Lagrange's method of undetermined multipliers.					
UNIT IV	MULTIPLE INTEGRALS				12
Double integrals – Change of order of integration – Area enclosed by Cartesian Coordinates (excluding polar coordinates) - Triple integrals (excluding spherical and cylindrical coordinates) - Volume of solids (Cartesian Coordinates only).					
UNIT V	VECTOR CALCULUS				12
Gradient, divergence and curl (excluding vector identities) – Directional derivative – Irrotational and Solenoidal vector fields – Green's theorem in a plane, Stoke's theorem and Gauss divergencetheorem (Statement only) - Simple applications involving cubes and rectangular parallelopeds.					
TOTAL: 60 PERIODS					

OUTCOMES:

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

- CO1:** Compute the matrix inverse and their higher powers.
CO2: Solve second and higher order differential equations.
CO3: Determine the maxima and minima of functions of two variables.
CO4: Determine the volume and surface area using multiple integrals.
CO5: Evaluate integrals using the concept of vector calculus.
CO6: Apply matrix algebra techniques to diagonalize the matrix.

TEXT BOOKS:

1. Erwin Kreyszig, –Advanced Engineering Mathematics||, John Wiley and Sons, 10th Edition, New Delhi, 2016.
2. B.S. Grewal, –Higher Engineering Mathematics||, Khanna Publishers, New Delhi, 44th Edition, 2014.

REFERENCES:

1. James Stewart, –Calculus: Early Transcendentals||, Cengage Learning, 7th Edition, New Delhi, 2015.
2. M. K. Venkataraman, –Engineering Mathematics||, Volume I, 4th Edition, The National Publication Company, Chennai, 2003.
3. Sivaramakrishna Dass, C. Vijayakumari, –Engineering Mathematics||, Pearson Education India, 4th Edition 2019.
4. H. K. Dass, and Er. Rajnish Verma, –Higher Engineering Mathematics||, S. Chand Private Limited, 3rd Edition 2014.
5. B.V. Ramana, –Higher Engineering Mathematics||, Tata McGraw Hill Publishing Company, 6th Edition, New Delhi, 2008.
6. NPTEL course on "Engineering Mathematics - I", by Prof. Jitendra Kumar, IIT Kharagpur: https://onlinecourses.nptel.ac.in/noc21_ma58/preview

24CS101	PROGRAMMING IN C++ (Common to All Branches)	L	T	P	C
		3	0	3	4.5
OBJECTIVES: The Course will enable learners to: • To learn problem solving and programming fundamentals. • To gain knowledge on pointers and functions. • To apply the principles of object orientated programming. • To understand operator overloading, inheritance and polymorphism. • To use the functionalities of I/O operations, files build C++ programs using exceptions.					
UNIT I	PROGRAMMING FUNDAMENTALS				9+9
Procedural Languages vs Object Oriented Languages - Types of computer programming languages - Genesis of C++ - Program Life Cycle -Structure of C++ program - Identifiers - Variables - Keywords - Number System -Binary Number System -Octal Number System-Decimal Number System -Hexadecimal Number System - Data types - Constants - Errors – Operators- Expressions - Type Conversions - Control-Flow Statements - Conditional Statements - Iterative Statements - Unconditional Control Statements - Arrays - One-					

Dimensional Arrays - Two-Dimensional Arrays - Multi -Dimensional Arrays - Strings - String Manipulation Functions - Array of Strings.

List of Exercise/Experiments:

1. Write C++ programs for the following:
 - a. Find the sum of individual digits of a positive integer.
 - b. Compute the GCD of two numbers.
 - c. Find the roots of a number (Newton _s method)
2. Write C++ programs using arrays:
 - a. Find the maximum of an array of numbers.
 - b. Remove duplicates from an array of numbers.
 - c. Print the numbers in an array after removing even numbers.
3. Write C++ programs using strings:
 - a. Checking for palindrome.
 - b. Count the occurrences of each character in a given word.

UNIT II	POINTERS AND FUNCTIONS	9+9
----------------	-------------------------------	------------

Pointers - Pointer Variables - Pointer Operators & Expressions -Pointers with Arrays - Functions - Scope Rules -Function Arguments -return Statement - Function Variables - Storage Classes - Types of storage classes - Create Header Files - User-Defined Functions - Inline Functions - Function Overloading -Recursion - Namespaces.

List of Exercise/Experiments:

1. Generate salary slip of employees using structures and pointers. Create a structure Employee with the following members: EID, Ename, Designation, DOB, DOJ, Basic pay
Note that DOB and DOJ should be implemented using structure within structure.
2. Compute internal marks of students for five different subjects using structures and functions.

UNIT III	CLASSES AND OBJECTS	9+9
-----------------	----------------------------	------------

Concepts of Object Oriented Programming – Benefits of OOP – Simple C++ program - Classes and Objects - Member functions - Nesting of member functions - Private member functions - Memory Allocation for Objects - Static Data Members - Static Member functions - Array of Objects - Objects as function arguments - Returning objects - friend functions - Const Member functions - Constructors – Destructors.

List of Exercise/Experiments:

1. Write a program Illustrating Class Declarations, Definition, and Accessing Class Members.
2. Program to illustrate default constructor, parameterized constructor and copy constructors.

Practice Questions & Scenario Based Questions:

1. Imagine you are working as a software engineer at a tech company. Your team is developing a mathematical software library that will be used in various applications across the company. One of the features that your team lead has asked you to implement is a function that calculates the number of trailing zeros in the factorial of a number.
The team lead has emphasized the importance of encapsulation in your implementation.
2. Create a C++ class Calculator representing a simple calculator. The class should have the following attributes and methods:

Attributes: Two operands and an operation (+, -, *, /)

Methods: Perform the operation and return the result

Implement constructors to initialize the calculator with default values (0,0) and with specified values. Also, implement a destructor to perform any necessary cleanup.		
UNIT IV	OPERATOR OVERLOADING, INHERITANCE AND POLYMORPHISM	9+9
Operator Overloading - Overloading Using Friend functions - Inheritance - Types of inheritance - Virtual Base Class - Abstract Class - Constructors in Derived Classes - member class: nesting of classes. Pointer to objects - this pointer- Pointer to derived Class - Virtual functions - Pure Virtual Functions - Polymorphism. List of Exercise/Experiments: 1. Write a Program to Demonstrate the i) Operator Overloading. ii) Function Overloading. 2. Write a Program to Demonstrate Friend Function and Friend Class. 3. Program to demonstrate inline functions. 4. Program for Overriding of member functions. 5. Write C++ programs that illustrate how the following forms of inheritance are supported: a) Single inheritance b) Multiple inheritance c) Multi level inheritance d) Hierarchical inheritance. Practice Questions & Scenario Based Questions: 1. Joy is a software developer at a 3D modeling company. The company is developing a new software tool that will be used by architects and engineers to design and analyze 3D models of various structures. One of the features that her project manager has asked is to implement a function that calculates the volume of basic 3D shapes like cylinders and cuboids. The project manager has emphasized the importance of using function overloading in her implementation. 2. Imagine you are a software developer tasked with creating a utility program for a school that handles student scores. The school wants a simple program where teachers can enter the scores of students for a particular test, and the program will then provide the highest and lowest scores among them. This will help the teachers quickly identify the top performer and the student who might need extra help. Your task is to write program that satisfies the above scenario using inline function 3. Develop a software system to manage part-time worker students at a university. These students have unique attributes such as their name, student ID, hourly wage, and hours worked per week. Your goal is to create a C++ program that models this system. Ramu is a software developer at a company specializing in developing software solutions for geometric shapes. Recently, a client approached with a request to create a program to calculate the areas of rectangles and triangles.		
UNIT V	I/O, FILES AND EXCEPTIONS	9+9
C++ Streams - Unformatted I/O - Formatted Console I/O - Opening and Closing File - File modes - File pointers and their manipulations – Templates – Class Templates – Function Templates - Exception handling. List of Exercise/Experiments: 1. Program to demonstrate pure virtual function implementation. 2. Count the number of account holders whose balance is less than the minimum balance using sequential access file. 3. Write a Program to Demonstrate the Catching of all Exceptions. Practice Questions & Scenario Based Questions: 1. Develop a simple library management system. Create a base class Book representing		

a book in the library. The class should have attributes such as title, author, and ISBN (International Standard Book Number). Implement a virtual function displayDetails() to display information about the book. 2. A software developer working on a banking application. One of the requirements is to analyze account holders' balances based on a minimum balance threshold. The application should read account information from a sequential access file, where each line represents an account record in the format: account_holder_name, balance. 3. The financial company is developing a new software tool that will be used by financial analysts to perform various calculations. One of the features of that project is to implement is a function that performs division of two numbers. Write a program that takes two integer inputs, numerator and denominator, from the user. Implement error handling to check if the denominator is zero. If the denominator is zero, display the message "Division by zero is not allowed!" using an exception. If the denominator is not zero, calculate the result of the division and display it. 4. Mini project.	TOTAL: 45 (L) + 45 (P) = 90 PERIODS
OUTCOMES: Upon completion of the course, the students will be able to: CO1: Solve problems using basic constructs in C++. CO2: Implement C++ programs using pointers and functions. CO3: Apply object-oriented concepts and solve real world problems. CO4: Develop C++ programs using operator overloading and polymorphism. CO5: Implement C++ programs using Files and exceptions. CO6: Develop applications using C++ concepts	
TEXT BOOKS: 1. Herbert Schildt, –The Complete Reference C++ , 4th edition, MH, 2015. 2. E Balagurusamy, Object Oriented Programming with C++ , 4th Edition, Tata McGraw-Hill Education, 2008.	
REFERENCES: 1. Karl Beecher, Computational Thinking: A beginner's guide to problem-solving and programming , BCS Learning & Development Ltd, 2017. (Unit 1) 2. Nell Dale, Chip Weems, –Programming and Problem Solving with C++ , 5th Edition, Jones and Barklett Publishers, 2010. 3. John Hubbard, –Schaum's Outline of Programming with C++ , MH, 2016. 4. Yashavant P. Kanetkar, –Let us C++ , BPB Publications, 2020 5. ISRD Group, –Introduction to Object-oriented Programming and C++ , Tata McGraw-Hill Publishing Company Ltd., 2007. 6. D. S. Malik, –C++ Programming: From Problem Analysis to Program Design , Third Edition, Thomson Course Technology, 2007. 7. Paul Deitel, Harvey Deitel, –C++ How to Program , 10th Edition, Pearson Education Inc. 2017. 8. https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_01297200240671948837_shared/overview	
LIST OF EQUIPMENTS: 1. Standalone desktops with C/C++ compiler (or) Server with C/C++ compiler.	

24CS102	SOFTWARE DEVELOPMENT PRACTICES (Common to All Branches)	L	T	P	C
		3	0	3	4.5
OBJECTIVES: The Course will enable learners to: - To discuss the essence of agile development methods. - To set up and create a GitHub repository. - To create interactive websites using HTML - To design interactive websites using CSS. - To develop dynamic web page using Java script.					
UNIT I	AGILE SOFTWARE DEVELOPMENT AND Git and GitHub				9+9
Software Engineering Practices - Waterfall Model - Agility - Agile Process - Extreme Programming - Agile Process Models – Adaptive Software Development – Scrum – Dynamic Systems Development Method - Crystal - Feature Driven Development - Lean Software Development – Agile Modeling – Agile Unified Process – Tool set for Agile Process. Introduction to Git –Setting up a Git Repository - Recording Changes to the Repository - Viewing the Commit History - Undoing Things - Working with Remotes -Tagging - Git Aliases - Git Branching - Branches in a Nutshell - Basic Branching and Merging - Branch Management - Branching Workflows - Remote Branches - Rebasing. Introduction to GitHub - Set up and Configuration - Contribution to Projects, Maintaining a Project – Scripting GitHub.					
List of Exercise/Experiments: 1. Form a Team, Decide on a project: a) Create a repository in GitHub for the team. b) Choose and follow a Git workflow - Each team member can create a StudentName.txt file with contents about themselves and the team project - Each team member can create a branch, commit the file with a proper commit message and push the branch to remote GitHub repository. - Team members can now create a Pull request to merge the branch to master branch or main development branch. - The Pull request can have two reviewers, one peer team member and one faculty. Reviewers can give at least one comment for Pull Request updating. - Once pull request is reviewed and merged, the master or main development branch will have files created by all team members. 2. Create a web page with at least three links to different web pages. Each of the web pages is to be designed by a team member. Follow Git workflow, pull request and peer reviews.					
UNIT II	HTML				9+9
Introduction – Web Basics – Multitier Application Architecture – Cline-Side Scripting versus Server-side Scripting – HTML5 – Headings – Linking – Images – Special Characters and Horizontal Rules – Lists – Tables – Forms – Internal Linking – meta Elements - Form input Types - input and datalist Elements - Page-Structure Elements.					
List of Exercise/Experiments: 1. Create web pages using the following: - Tables and Lists - Image map					

• Forms and Form elements • Frames		
UNIT III	CSS	15
Inline Styles – Embedded Style Sheets – Conflicting Styles – Linking External Style Sheets - Positioning Elements - Backgrounds - Element Dimensions - Box Model and Text Flow - Media Types and Media Queries - Drop-Down Menus - Text Shadows - Rounded Corners – Colour – Box Shadows – Linear Gradients – Radial Gradients – Multiple Background Images – Image Borders – Animations – Transitions and Transformations - Flexible Box Layout Module - Multicolumn Layout. List of Exercise/Experiments: 1. Apply Cascading style sheets for the web pages created.		
UNIT IV	JAVASCRIPT BASICS	9+9
Introduction to Scripting – Obtaining user input – Memory Concepts – Arithmetic – Decision Making: Equality and Relational Operators - JavaScript Control Statements - Functions - Program Modules - Programmer-defined functions - Scope rules - functions - Recursion - Arrays - Declaring and Allocating Arrays - References and Reference Parameters - Passing Arrays to Functions - Multidimensional arrays. List of Exercise/Experiments: 1. Form Validation (Date, Email, User name, Password and Number validation) using JavaScript.		
UNIT V	JAVASCRIPT OBJECTS	9+9
Objects - Math, String, and Date, Boolean and Number, document Object - Using JSON to Represent objects - DOM: Objects and Collections - Event Handling. List of Exercise/Experiments: Implement Event Handling in the web pages.		
Mini Projects-Develop any one of the following web applications (not limited to one) using above technologies. - Online assessment system - Ticket reservation system - Online shopping - Student management system - Student result management system - Library management - Hospital management - Attendance management system - Examination automation system - Web based chat application		
TOTAL: 45 (L) + 45 (P) = 90 PERIODS		
OUTCOMES: Upon completion of the course, the students will be able to: CO1: Understand basic software engineering practices effectively. CO2: Apply version control using Git and GitHub, and manage code repositories proficiently. CO3: Design web applications using HTML, CSS, and JavaScript. CO4: Analyze problems and create solutions using CSS for better web page presentation		

and usability.

CO5: Develop interactive web pages using JavaScript with an event-handling mechanism.

CO6: Apply the technological changes and improve skills continuously.

TEXT BOOKS:

1. Roger S. Pressman, –Software Engineering: A Practitioner's Approach, McGraw Hill International Edition, Ninth Edition, 2020.
2. Scott Chacon, Ben Straub, –Pro GIT, Apress Publisher, 3rd Edition, 2014.
3. Deitel and Deitel and Nieto, –Internet and World Wide Web - How to Program, Pearson, 5th Edition, 2018.

REFERENCES:

1. Roman Pichler, –Agile Product Management with Scrum Creating Products that Customers Love, Pearson Education, 1st Edition, 2010.
2. Jeffrey C and Jackson, –Web Technologies A Computer Science Perspective, Pearson Education, 2011.
3. Stephen Wynkoop and John Burke, –Running a Perfect Website, QUE, 2nd Edition, 1999.
4. Chris Bates, –Web Programming - Building Intranet Applications, 3rd Edition, Wiley Publications, 2009.
5. Gopalan N.P. and Akilandeswari J., –Web Technology, Second Edition, Prentice Hall of India, 2014.
6. https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_013382690411003904735_shared/overview
7. https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_0130944214274703362099_shared/overview

LIST OF EQUIPMENTS:

1. Systems with either Netbeans or Eclipse
2. Java/JSP/ISP Webserver/Apache
3. Tomcat / MySQL / Dreamweaver or
4. Equivalent/ Eclipse, WAMP/XAMP

24CH101	ENGINEERING CHEMISTRY (Common for CSE and IT)	L	T	P	C
		3	0	2	4
OBJECTIVES: The Course will enable learners to: • Gain a comprehensive knowledge on polymers utilized in various industrial sectors. • Knowledge on the fundamental principles of energy storage devices. • Gain insights into the basic concepts and applications of chemical sensors and cheminformatics. • Identify the different types of smart materials and explore their applications in Engineering and Technology. • Assimilate the preparation, properties and applications of nanomaterials in various fields.					
UNIT I	INDUSTRIAL POLYMERS				15
Polymers: Terminology - functionality, degree of polymerization, properties - glass transition temperature and molecular weight (weight average method).					

Engineering plastics: preparation, properties and application of Polyethylene, Teflon, Polyethylene terephthalate and Epoxy resin- industrial applications of Polyethylene and Polyethylene terephthalate in packaging.

Special polymers: preparation, properties and applications of piezoelectric polymer - Polyvinylidene fluoride, electroactive polymer-Polyacetylene and biodegradable polymer - Polylactic acid.

(Theory-9)

1. Determination of the molecular weight of polymer using viscometer.
2. Determination of degradation of biodegradable polymer using photoreactor.
3. Demonstrate the applications of biodegradable plastic using 3D printing.

(Laboratory-6)

UNIT II	ENERGY STORAGE DEVICES AND GREEN FUEL	15
----------------	--	-----------

Introduction to Electrochemical cell and its terminology - electrochemical series and its applications.

Batteries: classification - construction and working principle -primary alkaline battery - secondary battery - Pb-acid battery.

Green fuel – Hydrogen - production (Photo electrocatalytic and photo catalytic water splitting), construction, working principle and applications in H₂ -O₂ fuel cells.

Batteries used in E- vehicle: Ni-metal hydride battery, Li-ion Battery - recycling of Li-ion batteries by direct cycling method; environmental effects of different energy storage devices.

(Theory-9)

1. Construction of electrochemical cell.
2. Determination of discharging state of Pb-acid battery by estimating the strength of the acid correlates with specific gravity.
3. Study of performance of a battery using battery analyzing module.

(Laboratory-6)

UNIT III	CHEMICAL SENSORS AND CHEMINFORMATICS	15
-----------------	---	-----------

Introduction: classification of chemical sensors -principle, construction and working of chemical sensors; pH sensor - Glass electrode; Breath analyzer; Industrial sensor - CO₂ sensors- Sensor for health care - Glucose sensor.

Cheminformatics: definition, scope, and significance; applications in the environmental sector - carbon footprint measurements, data analysis and interpretation.

(Theory-9)

1. Determination of the amount of given hydrochloric acid using a pH meter.
2. Calculate the carbon footprint from the provided dataset, analyze the results and draw conclusions.

(Laboratory-6)

UNIT IV	SMART MATERIALS	15
----------------	------------------------	-----------

Shape Memory Alloys: introduction - shape memory effect – functional properties of SMAs - types of SMA - Nitinol (Ni-Ti) alloys and its applications.

Chromogenic materials: introduction – types, applications in chemical and biological detection, display technologies, smart windows and light- modulating devices, biomedical and healthcare. Smart Hydrogels: Introduction - Super Absorbent Polymers (SAP) - preparation, properties and applications of polyacrylic acid and sodium polyacrylate (Theory-9) 1. Demonstrate the shape memory effect using Nitinol wire. 2. Determination of pH sensitivity of bromothymol blue. 3. Determination of absorption efficiency of hydrogel by using kinetic study. (Laboratory-6)		
UNIT V	NANO CHEMISTRY	15
Introduction - synthesis - top-down process (laser ablation, chemical vapor deposition), bottom-up process (precipitation, electrochemical deposition) - properties of nanomaterials – types – nanotubes -carbon nanotubes, applications of CNT - nanocomposites - general applications of nanomaterials in electronics, information technology, medical and healthcare, energy, environmental remediation, construction and transportation industries. (Theory-9) 1. Preparation of nano BaSO₄ by precipitation method. 2. Demonstrate the efficiency of nano adsorbents in polluted water. (Laboratory-6)		
TOTAL: 75 PERIODS		
OUTCOMES: Upon completion of the course, the students will be able to: CO1: To examine the role of polymers in different industrial sectors. CO2: To identify the suitability of batteries for various fields. CO3: To apply the fundamental principles of chemical sensors, cheminformatics and their applications across various industries. CO4: To analyze the types of smart materials used in various engineering fields. CO5: To explore the applications of nanomaterials in various fields, considering their Advantages and limitations. CO6: To integrate the concepts of chemistry for various engineering applications.		
TEXT BOOKS: 1. P. C. Jain and Monika Jain, –Engineering Chemistry, Dhanpat Rai Publishing Company Pvt. Ltd., New Delhi, 19th Edition, 2024. 2. Sivasankar B., –Engineering Chemistry, Tata McGraw-Hill Publishing Company Ltd., New Delhi, 2nd reprint, 2012.		
REFERENCES: 1. S.S. Dara and S.S. Umare, –A Textbook of Engineering Chemistry, S. Chand & company, New Delhi, 12th Edition, 2022. 2. V.R. Gowarikar, Polymer Science, New Age International Publishers, 4th edition, 2021. 3. J. C. Kuriacose and J. Rajaram, –Chemistry in Engineering and Technology, Volume		

-1 & Volume -2, Tata McGraw-Hill Education Pvt. Ltd., 2010.

- Barry A. Bunin, Brian Siesel, and J. Bajorath, –Chemoinformatics: Theory, Practice, & Products||, Springer, First Edition, 2007.
- Geoffrey A. Ozin, Andre C. Arsenault and Ludovico Cademartiri, –Nanotechnology: A Chemical Approach to Nanomaterials||, RSC publishers, 2nd Edition, 2015.
- J. Mendham, R. C. Denney, J. D. Barnes, M. J. K. Thomas and B. Sivasankar, –Vogel's Quantitative Chemical Analysis||, Pearson Education Pvt. Ltd., 6th edition, 2019.
- NPTEL course on –Polymers: concepts, properties, uses and sustainability|| Prof. Abhijit P Deshpande, IIT-Madras, https://onlinecourses.nptel.ac.in/noc20_ch41/preview
- NPTEL course on –Electrochemical Energy Storage|| Prof. Subhasish Basu Majumder, IIT Kharagpur, https://onlinecourses.nptel.ac.in/noc21_mm34/preview
- NPTEL course on –Nanotechnology, Science and Applications||, Prof. Prathap Haridoss, IIT-Madras, https://onlinecourses.nptel.ac.in/noc22_mm33/preview

LIST OF EQUIPMENTS FOR A BATCH OF 30 STUDENTS

S. No.	Description of Equipment	Quantity
1	Conductivity meter	12 Nos.
2	pH meter	12 Nos.
3	UV-Visible Spectrophotometer	2 Nos.
4	Potentiometer	12 Nos.
5	CAN Enabled BMS unit	4 Nos.
6	Photoreactor	2 Nos.

24EC101	DIGITAL PRINCIPLES AND SYSTEMS DESIGN	L	T	P	C
		3	0	2	4

OBJECTIVES:

The Course will enable learners to:

- Acquire the knowledge in Digital fundamentals and its simplification methods.
- Familiarize the design of various combinational digital circuits using logic gates.
- Realize various sequential circuits using flip flops.
- Elucidate various semiconductor memories and related technology.
- Build various logic functions using Programmable Logic Devices.

UNIT I	BOOLEAN ALGEBRA AND LOGIC GATES	15
---------------	--	-----------

Review of number systems-representation-conversions, Review of Boolean algebra theorems, sum of product and product of sum simplification, canonical forms, min term and max term, Simplification of Boolean expressions-Karnaugh map, Implementation of Boolean expressions using logic gates and universal gates.

List of Exercise/Experiments:

- 1. Implementation of Boolean expression using logic gates.

UNIT II	COMBINATIONAL LOGIC CIRCUITS	15
----------------	-------------------------------------	-----------

Design of combinational circuits - Half and Full Adders, Half and Full Subtractors, Binary Parallel Adder – Carry look ahead Adder, Magnitude Comparator, Decoder, Encoder, Priority Encoder, Mux/De-mux, Parity Generator/Checker

List of Exercise/Experiments:

- 2. Design of adders

3. Design of subtractors. 4. Design of binary adder using IC7483 5. Design of Multiplexers & Demultiplexers. 6. Design of Encoders and Decoders. 7. Implementation of a boolean function using a multiplexer		
UNIT III	SEQUENTIAL CIRCUITS	15
Flip flops – SR, JK, T, D, Master/Slave FF – operation and excitation tables, Asynchronous and Synchronous Counters Design - Shift registers, Universal Shift Register. List of Exercise/Experiments: 8. Design and implementation of 3 bit ripple counters. 9. Design and implementation of 3 bit synchronous counter 10. Design and implementation of shift registers		
UNIT IV	SYNCHRONOUS SEQUENTIAL CIRCUITS DESIGN	15
Design of clocked sequential circuits - Moore/Mealy models, state minimization, state assignment, circuit implementation.		
UNIT V	MEMORY AND PROGRAMMABLE LOGIC DEVICES	15
Basic memory structure ROM: PROM – EPROM – EEPROM –RAM – Static and dynamic RAM - Programmable Logic Devices: Programmable Logic Array (PLA) - Programmable Array Logic (PAL) - Implementation of combinational logic circuits using PLA, PAL.		
TOTAL: 45 PERIODS (THEORY) + 30 PERIODS (LAB) = 75 PERIODS		
OUTCOMES: Upon completion of the course, the students will be able to: CO1: Apply Boolean algebra to simplify and implement digital circuits. CO2: Design combinational circuits to meet specific functional requirements using logic gates. CO3: Demonstrate the operation of counters and shift registers using flip-flops in sequential circuits. CO4: Analyze synchronous sequential circuits to determine their behavior and performance characteristics. CO5: Evaluate various types of memory devices, discussing their roles and functionalities in digital systems. CO6: Construct combinational circuits using Programmable Logic Devices (PLDs) to solve complex digital design problems.		
TEXTBOOKS:		
1. M. Morris Mano and Michael D. Ciletti, Digital Design, With an Introduction to the Verilog HDL, VHDL, and System Verilog, 6th Edition, Pearson, 2018. 2. S.Salivahanan and S.Arivazhagan, Digital Circuits and Design, 5th Edition, Oxford University Press, 2018.		
REFERENCES:		

1. A.Anandkumar, Fundamental of digital circuits, 4th Edition, PHI Publication,2016.
2. WilliamKleitz, Digital Electronics-A Practical approach to VHDL, Prentice Hall International Inc, 2012.
3. CharlesH.Roth, Jr. andLarry L. Kinney, Fundamentals of Logic Design, 7th Edition, Thomson Learning, 2014.
4. Thomas L. Floyd, Digital Fundamentals, 11th Edition, Pearson Education Inc, 2017.
5. John.M Yarbrough, Digital Logic: Applications and Design, 1st Edition, Cengage India, 2006.

NPTEL Link:

<https://nptel.ac.in/courses/108105132>

LIST OF EQUIPMENT FOR A BATCH OF 30 STUDENTS

IC Trainer Kit -15 Nos

ICs each - 30 Nos

7400/ 7404 / 7486 / 7408 / 7432 / 7483 / 7473 / 7411/7474

24GE101	HERITAGE OF TAMILS (Common to All Branches)	L	T	P	C
		1	0	0	1
OBJECTIVES: The Course will enable learners to: • Recognize Tamil literature and its significance in Tamil culture. • Introduce the Tamils' rich artistic and cultural legacy. • Familiarize the different types of folk and martial arts that are unique to Tamil Nadu. • Acquaint the concept of Thinaï in Tamil literature and culture. • Comprehend the significance of Tamil in developing Indian culture.					
UNIT I	LANGUAGE AND LITERATURE				3
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.					
UNIT II	HERITAGE - ROCK ART PAINTINGS TO MODERN ART – SCULPTURE				3
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.					
UNIT III	FOLK AND MARTIAL ARTS				3
Therukoothu, Karagattam, Villu Pattu, Kaniyan Koothu, Oyillattam, Leatherpuppetry, Silambattam, Valari, Tiger dance - Sports and Games of Tamils.					
UNIT IV	THINAI CONCEPT OF TAMILS				3
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 Cholas.					
UNIT V	CONTRIBUTION OF TAMILS TO INDIAN NATIONAL MOVEMENT AND INDIAN CULTURE				3
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.
TOTAL: 15 PERIODS
OUTCOMES: Upon completion of the course, the students will be able to: CO1: State the role of Tamil literature in shaping Tamil Cultural roots. CO2: Express the cultural and religious significance of Tamil art and sculptures. CO3: Identify and describe the techniques of folk and martial arts. CO4: Classify the role of Thinaï concept in Tamil culture and literature. CO5: Compare the idea of cultural and intellectual contributions of Tamils.
TEXT BOOKS & REFERENCES:
தமிழக வரலாறு – மக்களும் பண்பாடும் – கே.கே. பிள்ளை (வெளியீடு: 1. தமிழ்நாடு பாடநூல் மற்றும் கல்வியியல் பணிகள் கழகம்).
2. கணினித் தமிழ் – முனைவர் இல. சுந்தரம். (விகடன் பிரசுரம்).
3. கீழடி – வைகை நதிக்கரையில் சங்ககால நகர நாகரிகம் (தொல்லியல் துறை வெளியீடு)
4. பொருளை – ஆற்றங்கரை நாகரிகம். (தொல்லியல் துறை வெளியீடு)
5. Social Life of Tamils (Dr.K.K. Pillay) A joint publication of TNTB & ESC and RMRL - (in print)
6. Social Life of the Tamils - The Classical Period (Dr.S .Singaravelu) (Published by: International Institute of Tamil Studies.
7. Historical Heritage of the Tamils (Dr.S.V.Subatamanian, Dr.K.D.Thirunavukkarasu) (Published by: International Institute of Tamil Studies).
8. The Contributions of the Tamils to Indian Culture (Dr. M. Valarmathi) (Published by: International Institute of Tamil Studies.)
9. Keeladi - ‘Sangam City Civilization on the banks of river Vaigai’ (Jointly Published by: Department of Archaeology & Tamil Nadu Text Book and Educational Services Corporation, Tamil Nadu)
10. Studies in the History of India with Special Reference to Tamil Nadu (Dr. K. K. Pillay) (Published by: The Author)
11. Porunai Civilization (Jointly Published by: Department of Archaeology & Tamil Nadu Text Book and Educational Services Corporation, Tamil Nadu)
12. Journey of Civilization Indus to Vaigai (R. Balakrishnan) (Published by: RMRL) - Reference Book

24MC101	STUDENT INDUCTION PROGRAM (SIP) (Common to all branches of First Year B.E./ B.Tech)	L	T	P	C
		3 Weeks			
OBJECTIVES: The Course will enable learners to: - Facilitate the adjustment of new students to the new environment, ensuring they feel comfortable and supported. - Inculcate the institution's ethos and culture in new students, helping them internalize these values. - Encourage the building of bonds among students and between students and faculty members. - Expose students to a sense of larger purpose and guide them in their journey of self- exploration. The various modules or core areas recommended for the 3-week SIP are					
Module 1	Universal Human Values I				18

Welcome and Introductions - Aspirations and Concerns - Self- Management - Health - Relationships - Society - Natural Environment - Sum Up - Self-evaluation and Closure.		
Module 2	Physical Health and Related Activities	6
Special Lecturers: Happy and Healthy lifestyle - Physical Health -Mental Health - Health and Fitness.		
Module 3	Familiarization of Department/ Branch and Innovation	8
Principal Address - Address by Head of Science and Humanities - Addresses by Respective Department HoDs - Campus Tour - CoE introduction - Introduction of Student Activity Cell (SAC).		
Module 4	Visit to a Local Area	4
Virtual tour: Government Museum - Theosophical Society - Fort St. George - Ripon Building - Kalakshetra Foundation - Anna Centenary Library - Marina Beach - St. Thomas Mount - Vivekananda House.		
Module 5	Lectures by Eminent People	10
Special Lecturers: Academics - industry - Careers - Art - Self-management.		
Module 6	Proficiency Modules	30
Basic Competencies: C Programming, Foundation in Mathematics, Interpersonal Communication.		
Module 7	Literature / Literary Activities	7
Literary Debate - Creative Writing Workshop - Literature Circle Discussions - Author Study and Presentation.		
Module 8	Creative Practices	10
Activity: Handicrafts (Painting / Drawing / Pottery / Knitting / Jewellery making, etc.)		
Module 9	Extra-Curricular Activities	10
Students Activity Cell: Activities from Coding Club - Math Club -- Language Club - Astronomy Club - ECO Club - Photography Club - Tedx Club -Yoga Club.		
Valedictory and Closing Ceremony		2
TOTAL: 105 PERIODS		
OUTCOMES: Upon completion of the course, the students will be able to: CO1: Achieve a smooth transition where students feel comfortable and confident in their new environment. CO2: Demonstrate a strong understanding and practice of the institution's ethos and culture within the campus community. CO3: Build meaningful and supportive relationships with peers and faculty members. CO4: Develop a clear sense of purpose and engage in self-exploration, leading to a deeper understanding of personal goals and aspirations.		

REFERENCES:

[https://www.aicte-](https://www.aicte-india.org/sites/default/files/Detailed%20Guide%20on%20Student%20Induction%20program.pdf)

[india.org/sites/default/files/Detailed%20Guide%20on%20Student%20Induction%20program.pdf](https://www.aicte-india.org/sites/default/files/Detailed%20Guide%20on%20Student%20Induction%20program.pdf)

24MC102	C PROGRAMMING (Common to All Branches)	
OBJECTIVES:		
The Course will enable learners to:		
- To learn problem solving and programming fundamentals in C. - To gain knowledge on Arrays and Functions. - To understand Pointers, Structures and File handling concepts.		
UNIT I	INTRODUCTION TO C PROGRAMMING	10
Introduction to C, Comments, Tokens, Character set, Identifiers- Variables, Keywords, Bits and Bytes representations- Data Types, Type Qualifiers- Symbolic Constants, One's and Two's Complement Forms, Expressions and Statements- Operators- Operator Precedence and Associativity- Pre-Processor Directives- I/O Functions- Control Structures: Decision Making Statements For Loop-While and do-while Loops		
UNIT II	ARRAYS AND FUNCTIONS	15
Array Operations-One dimensional array-Multi Dimensional Array - Practice Programs- Searching-Sorting-String Handling Functions- Function: Declaration, Definition, Function Call and Return Statements-Passing Parameters to Functions-Types of Functions-Calling Functions from Another Function and Recursive Functions. Variable Scope, Lifetime and Storage Classes-Storage Classes-Problem Solving using Recursion- Towers of Hanoi Problem, Advantages and Disadvantages of Recursion.		
UNIT III	POINTERS, STRUCTURES AND FILE HANDLING CONEPTS	15
Introduction to Pointers-Dynamic Memory Allocation-Pointers of Arrays and Strings-Pointer Arithmetic - Pointers to Functions-Introduction to User Defined Data Types-Structures, Functions and Nested Structures-Bit Fields in Structure and Union Enumerated Data Types- Introduction to File Handling-Advanced File Operations-Command Line Arguments & Error Handling.		
TOTAL HOURS: 40		
OUTCOMES:		
Upon completion of the course, the students will be able to		
Develop a C program, control the sequence of the program and give logical outputs, implement strings in C program, store different data types in the memory, write functions in C and also handle files		
REFERENCES:		
1. Programming in C. E. Balagurusamy. 9 th Edition, Tata McGraw-Hill Publishing, 2020. ISBN		
2. The C programming language. BW Kernighan, DM Ritchie. Prentice Hall, Englewood Cliffs 2015		

24HS111	INTERPERSONAL SKILLS, PSYCHOMETRIC ANALYSIS AND CAREER DEVELOPMENT (Common to All Branches)	L	T	P	C
		1	0	0	1
OBJECTIVES: The Course will enable learners to: - Evaluate and enhance language proficiency by using SMART Vox to assess communication skills and linguistic abilities. - Explore future technologies, solve real-time problems, and prepare students for campus recruitment. - Assess and develop work style, cognitive abilities, emotional intelligence, and work competencies - Explore placements, internships, higher education options, GATE and CAT exams, and career development. - Understand career milestones, assess personal skills and personality, and develop effective goal-setting strategies for successful career planning					
UNIT I	LANGUAGE PROFICIENCY EVALUATION				3
Identification of Strengths and Weaknesses - Assessing Language Skills (Diagnostic tests and interactive exercises) - Measuring Language Competence (Proficiency Levels) - Identifying Support Needs - Developing Individual Learning Plans - Enhancing Student Experience (Building Confidence) – Evaluation using SMART Vox					
UNIT II	CAREER GUIDANCE				3
Future of Engineering- Various aspects of Technology and its Applications - Future of Technologies - Branch Specific emerging technologies - Problems solving through open source - Campus recruitment process.					
UNIT III	PSYCHOMETRIC EVALUATION				15
Understanding Behavioural work style (Personality)- Testing of numerical, logical, and verbal reasoning skills (Cognitive Abilities / Aptitude) – Measure of emotional intelligence and interpersonal skills - Determination of Culture Preferences in various workplace scenarios - Evaluation of Work Competencies through targeted games and simulations.					
UNIT IV	CAREER PREPARATION				15
Present Scenario of Engineering - Placement Opportunities - Internship Opportunities - Types of Internships- Higher Education opportunities in India and Abroad - Understanding GATE and CAT Exams - Other Opportunities - Career path development plans.					
UNIT V	PROFILE BUILDING AND CODING MASTERY				15
Resume Building - Career Building for Engineering Students - Coding Orientation - Algorithmic Thinking and Advanced Problem Solving – Building Your Own App with Just Logic.					
	TOTAL 15 PERIODS				

OUTCOMES:

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

CO1: Assess and improve their English language proficiency using SMART Vox, gaining insights into their communication skills and linguistic competence

CO2: Understand future engineering trends, emerging technologies, importance of solving real-time problems, and the process of campus recruitment.

CO3: Evaluate their behavioral work style, cognitive abilities, emotional intelligence, cultural preferences, and work competencies.

CO4: Understand the current engineering landscape, placement opportunities, and higher education prospects to develop effective career path plans

CO5: Develop a clear and actionable vision for their future career path.

24GE111	IDEA LAB - I (Common to all Branches)	L	T	P	C
		0	0	1	0

The students may be grouped into 3 to 4. The device/Machine/system/component are studied by the students and a final presentation to be done by the students about the study of various devices or machinery at the end of the semester.

OBJECTIVES:

The Course will enable learners to:

- Understand the functionalities and limitation of various machines/equipment
- Demonstrate various operations that can be performed using various Machines

LIST OF EXPERIMENTS

1. Study of fundamental operations of 3D Printer and Scanner with Software.
2. Study of Laser cutting machine.
3. Study of CNC Router machine.
4. Study of Fundamentals of basic circuit design, Soldering and Desoldering.
5. Study of PCB Milling Machine.

TOTAL: 15 PERIODS

OUTCOMES:

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

CO1: Describe the working of the 3D Printer.

CO2: Explain the operation of the CNC router and laser cutting machines.

CO3: Explain the basic parts and PCB fabrication process.

CO4: Develop the ability to handle delicate electronic components carefully, minimizing damage during the soldering process.

CO5: Describe the process for converting ideas into prototypes.

LIST OF EQUIPMENT FOR A BATCH OF 30 STUDENTS

S.No.	Description of Equipment	Quantity
1	CNC Router	1 No
2	3D Printer	1 No
3	3D Scanner	1 No
4	Laser cutting Machine	1 No
5	Multimeter	5 Nos.
6	Solder Stations	5 Sets
7	Desoldering Machine	1 No
8	PCB Milling Machine	1 No
9	Variable Power Supply	1 No
10	Electronic Components like Resistors, Transistors, Diode, Inductor, Capacitor, etc.	5 Sets

SEMESTER II

24MA201	LINEAR ALGEBRA AND APPLICATIONS (Common to CSE, IT,AI ML)	L	T	P	C
		3	0	2	4
OBJECTIVES: The Course will enable learners to: - Comprehend the fundamental concepts of matrices. - Illustrate the basic notions associated with vector spaces and its properties. - utilize the Gram-Schmidt ortho normalization process. - Understand the components and implications for vector spaces by rank-nullity dimension theorem. - Calculate the eigenvalues and eigenvectors of linear transformations.					
UNIT I	MATRICES AND SYSTEM OF LINEAR EQUATIONS				15
Matrices – Row echelon form – Rank of a matrix – System of linear equations – Consistency - Gauss elimination method - Gauss Jordan method. Experiments using C language: 1. Solve the system of equations using Gauss Elimination method. 2. Solve the system of equations using Gauss Jordan method.					
UNIT II	VECTOR SPACES				15
Real and Complex fields - Vector spaces over Real and Complex fields - Subspace - Linear space - Linear independence and dependence (Statement only) - Bases and dimensions. Experiments using C language: 1. Check whether the given vectors are linearly independent or not. 2. Find the basis and dimension for given vectors.					
UNIT III	INNER PRODUCT SPACES				15
Inner product space and norms - Properties - Orthogonal, Orthonormal vectors - Gram-Schmidt ortho normalization process - Least squares approximation. Experiments using C language: 1. Find the orthogonal vectors using inner product. 2. Find the orthonormal vectors using inner product.					
UNIT IV	LINEAR TRANSFORMATION				15
Linear transformation – Range and null space – Rank and nullity – Rank nullity Dimension theorem - Matrix representation of linear transformation - Eigenvalues and eigenvectors of linear transformation. Experiments using C language: 1. Find the Rank and Nullity of a matrix. 2. Find the eigenvalues and eigenvectors of a matrix.					
UNIT V	EIGENVALUE PROBLEMS AND MATRIX DECOMPOSITION				15
Eigenvalue problems - Power method - Jacobi method - Singular value decomposition - QR decomposition. Experiments using C language: 1. Solve the system of equations using Jacobi method. 2. Find QR decomposition of a matrix.					
		TOTAL: 75 PERIODS			
OUTCOMES: Upon completion of the course, the students will be able to: CO1: Solve the system of linear equations using Gauss elimination and Gauss Jordan method.					

CO2: Analyze vector spaces to determine their bases and dimensions.
CO3: Apply Gram-Schmidt process to ortho normalize sets of vectors.
CO4: Apply rank nullity theorem to analyse linear transformations.
CO5: Compute the eigenvalues and eigenvectors using singular value decomposition.
CO6: Understand the ideas of least squares approximations and its applications.
TEXTBOOKS:
1. A.H. Friedberg, A. J. Insel, and L. Spence, –Linear Algebra , Prentice Hall of India, 5th Edition, New Delhi, 2008. 2. Steven J. Leon, –Linear Algebra with Applications , Pearson Educational International , 9th Edition, United States of America, 2015.
REFERENCES:
1. G. Strang, –Linear Algebra and its applications , Thomson (Brooks / Cole), 4th Edition, New Delhi, 2005. 2. C.F. Gerald and P.O. Wheatley, –Applied Numerical Analysis , 7th Edition, Pearson Education, New Delhi, 2004. 3. Richard Branson, –Matrix Operations , Schaum's outline series, 1989. 4. Bernard Kolman, R. David R. Hill, –Introductory Linear Algebra , Pearson Educations, New Delhi, First Reprint, 2009. 5. S. Kumaresan, –Linear Algebra - A geometric approach , Prentice Hall of India, New Delhi, Reprint, 2010. 6. NPTEL course on "Linear Algebra", by Prof. K. C. Sivakumar, IIT Madras: https://archive.nptel.ac.in/courses/111/106/111106051/###

24CS201	DATA STRUCTURES (Common to CSE, IT, CSBS and AIML)	L	T	P	C
		3	0	3	4.5
OBJECTIVES: The Course will enable learners to: • To understand the concepts of List ADT. • To learn linear data structures - stacks and queues ADTs. • To understand and apply Tree data structures. • To understand and apply Graph structures. • To analyze sorting, searching and hashing algorithms.					
UNIT I	LINEAR DATA STRUCTURES – LIST				9+9
Algorithm analysis - running time calculations - Abstract Data Types (ADTs) - List ADT - array- based implementation - linked list implementation - singly linked lists - circularly linked lists - doubly-linked lists - applications of lists - Polynomial Manipulation - All operations (Insertion, Deletion, Merge, Traversal).					
List of Exercise/Experiments: • Array implementation of List ADTs. • Linked list implementation of List ADTs.					
UNIT II	LINEAR DATA STRUCTURES – STACKS, QUEUES				9+9
Stack ADT – Stack Model - Implementations: Array and Linked list - Applications - Balancing symbols - Evaluating arithmetic expressions - Conversion of Infix to postfix expression - Queue ADT – Queue Model - Implementations: Array and Linked list - applications of queues - Priority Queues - Binary Heap - Applications of Priority Queues.					
List of Exercise/Experiments: • Array implementation of Stack and Queue ADTs.					

• Linked list implementation of Stack and Queue ADTs. • Applications of List - Polynomial manipulations • Applications of Stack - Infix to postfix conversion and expression evaluation.	
UNIT III	NON LINEAR DATA STRUCTURES – TREES
9+9	
Tree ADT - tree traversals - Binary Tree ADT - expression trees - applications of trees – binary search tree ADT- AVL Tree.	
List of Exercise/Experiments: • Implementation of Binary Trees and operations of Binary Trees. • Implementation of Binary Search Trees. • Implementation of Heaps using Priority Queues.	
UNIT IV	NON LINEAR DATA STRUCTURES - GRAPHS
9+9	
Definition - Representation of Graph - Types of graph - Breadth-first traversal - Depth-first traversal – Topological Sort – Applications of graphs – BiConnectivity – Euler circuits.	
List of Exercise/Experiments: • Graph representation and Traversal algorithms.	
UNIT V	SEARCHING, SORTING AND HASHING TECHNIQUES
9+9	
Searching- Linear Search - Binary Search - Sorting - Bubble sort - Selection sort - Insertion sort - Hashing - Hash Functions - Separate Chaining - Open Addressing - Rehashing - Extendible Hashing.	
List of Exercise/Experiments: • Implement searching and sorting algorithms.	
TOTAL: 45(L) + 45(P) = 90 PERIODS	
OUTCOMES: Upon completion of the course, the students will be able to: CO1: Analyze algorithms and abstract data types (ADTs). CO2: Evaluate fundamental data structures. CO3: Implement linked data structures and its application. CO4: Apply advanced tree data structures. CO5: Understand basic graph theory concepts. CO6: Evaluate various searching and sorting algorithms.	
TEXTBOOKS:	
1. Mark Allen Weiss, –Data Structures and Algorithm Analysis in C++ , 4th Edition, Pearson Education, 2014. 2. Sartaj Sahni, –Data Structures, Algorithms and Applications in C++ , Silicon paper publications, 2004.	
REFERENCES:	

1. Rajesh K. Shukla, –Data Structures using C and C++ , Wiley India Publications, 2009. 2. Narasimha Karumanchi, –Data Structure and Algorithmic Thinking with Python: Data Structure and Algorithmic Puzzles , CareerMonk Publications, 2020. 3. Jean-Paul Tremblay and Paul Sorenson, –An Introduction to Data Structures with Application , McGraw-Hill, 2017. 4. Mark Allen Weiss, –Data Structures and Algorithm Analysis in Java , Third Edition, Pearson Education, 2012. 5. Ellis Horowitz, Sartaj Sahni, Susan Anderson-Freed, –Fundamentals of Data Structures in C , Second Edition, University Press, 2008. 6. Ellis Horowitz, Sartaj Sahni, Dinesh P Mehta, –Fundamentals of Data Structures in C++ , Second Edition, Silicon Press, 2007. 7. https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_01350157816505139210584/overview
LIST OF EQUIPMENTS:
1. Systems with Linux/Ubuntu Operating System with gnu C++ compiler

24CS202	JAVA PROGRAMMING (Common to All Branches)	L	T	P	C
		3	0	3	4.5
OBJECTIVES: The Course will enable learners to: • To explain object oriented programming concepts and fundamentals of Java • To apply the principles of packages, interfaces and exceptions • To develop a Java application with I/O streams, threads and generic programming • To build applications using strings and collections. • To apply the JDBC concepts					
UNIT I	JAVA FUNDAMENTALS				9+9
An Overview of Java - Data Types, Variables, and Arrays - Operators - Control Statements - Class Fundamentals - Declaring objects - Methods - Constructors - this keyword - Overloading methods - Overloading constructors - Access Control - Static - Final					
List of Exercise/Experiments: 1. Develop a Java application to generate Electricity bill. You must use one super class called EB Bill and must have two sub classes namely Domestic Bill and Commercial Bill. Create a class with the following members: Consumer no., consumer name, previous month reading, current month reading, type of EB connection (i.e domestic or commercial). Compute the bill amount using the following tariff If the type of the EB connection is domestic, calculate the amount to be paid as follows: First 100 units - Rs. 1 per unit 101-200 units - Rs. 2.50 per unit 201 -500 units - Rs. 4 per unit > 501 units - Rs. 6 per unit If the type of the EB connection is commercial, calculate the amount to be paid as follows: First 100 units - Rs. 2 per unit 101-200 units - Rs. 4.50 per unit 201 -500 units - Rs. 6 per unit > 501 units - Rs. 7 per unit 2. Arrays Manipulations: (Use Methods for implementing these in a Class) a. Find kth smallest element in an unsorted array					

b. Find the sub array with given sum c. Matrix manipulations - Addition, Subtraction, Multiplication d. Remove duplicate elements in an Array e. Accept an integer value N and print the Nth digit in the integer sequence 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15 and so on till infinity. Example: The 11th digit in the sequence 12345678910111213.... is 0.		
UNIT II	INHERITANCE, INTERFACES AND EXCEPTION HANDLING	9+9
Inheritance: Inheritance basics, Using super, Method Overriding, Using Abstract Classes, Using final with Inheritance - Package and Interfaces: Packages, Packages and member access, Importing Packages, Interfaces, Static Methods in an Interface – Exception Handling: Exception- Handling Fundamentals, Exception Types, Uncaught Exceptions, Using try and catch, Multiple catch Clauses, Nested try Statements, throw, throws, finally, Java's Built-in Exceptions.		
List of Exercise/Experiments: 1. Develop a Java application to implement currency converter (Dollar to INR, EURO to INR, Yen to INR and vice versa), distance converter (meter to KM, miles to KM and vice versa), time converter (hours to minutes, seconds and vice versa) using packages. 2. Develop a Java application with Employee class with Emp_name, Emp_id, Address, Mail_id, Mobile_no as members. Inherit the classes, Programmer, Assistant Professor, Associate Professor and Professor from employee class. Add Basic Pay (BP) as the member of all the inherited classes with 97% of BP as DA, 10 % of BP as HRA, 12% of BP as PF, 0.1% of BP for staff club fund. Generate pay slips for the employees with their gross and net salary. 3. Design a Java interface for ADT Stack. Implement this interface using array and built-in classes. Provide necessary exception handling in both the implementations. 4. Write a Java Program to create an abstract class named Shape that contains two integers and an empty method named print Area(). Provide three classes named Rectangle, Triangle and Circle such that each one of the classes extends the class Shape. Each one of the classes contains the methods print Area () that prints the area of the given shape and Numberofsides() that prints the number of sides of the given shape. 5. Write a Java program to apply built-in and user defined exceptions.		
UNIT III	MULTITHREADING, I/O AND GENERIC PROGRAMMING	9+9
Multithreaded Programming: Creating a Thread, Thread Priorities, Synchronization, Interthread Communication - I/O: I/O Basics, Reading Console Input, Writing Console Output, Reading and Writing Files – Generics: Introduction, Generic class, Bounded Types, Generic Methods, Generic Interfaces, Generic Restrictions.		
List of Exercise/Experiments: 1. Write a Java program to read and copy the content of one file to other by handling all file related exceptions.		
UNIT IV	STRING HANDLING AND COLLECTIONS	9+9
Lambda Expressions - String Handling - Collections: The Collection Interfaces, The Collection Classes - Iterator - Map - Regular Expression Processing.		
List of Exercise/Experiments: 1. String Manipulation: a. Reversing a set of words and count the frequency of each letter in the string. b. Pattern Recognition - Find the number of patterns of form 1[0]1 where [0] represents		

any number of zeroes (minimum requirement is one 0) there should not be any other character except 0 in the [0] sequence in a given binary string. c. Remove all the occurrences of string S2 in string S1 and print the remaining. d. Find the longest repeating sequence in a string e. Print the number of unique string values that can be formed by rearranging the letters in the string S. 2. Write a Java program that correctly implements producer consumer problem using the concept of inter thread communication. 3. Collections: a. Write a program to perform string operations using ArrayList. Write functions for the following i. Append - add at end ii. Insert - add at particular index iii. Search iv. List all string starts with given letter b. Find the frequency of words in a given text.		
UNIT V	JDBC CONNECTIVITY	9+9
JDBC - DataSource, Configurations, Connection, Connection Pools, Driver Types, ResultSet, Prepared Statement, Named Parameter, Embedded SQL (Insert, Update, Delete, Join, union etc), ResultSet Navigation, Connection Close and Clean up. List of Exercise/Experiments: Mini Project (using JDBC)		
TOTAL: 45(L) + 45(P) = 90 PERIODS		
OUTCOMES: Upon completion of the course, the students will be able to: CO1: Solve core Java programming concepts. CO2: Utilize object-oriented programming (OOP) principles. CO3: Demonstrate competency in handling exceptions and implementing multithreading. CO4: Develop expertise in input/output (I/O) operations and file handling. CO5: Apply advanced Java programming concepts with generics and lambda expressions. CO6: Implement database connectivity using JDBC.		
TEXTBOOKS:		
1. Herbert Schildt, –Java: The Complete Reference , 11th Edition, McGraw Hill Education, 2019.		
REFERENCES:		
1. Cay S. Horstmann, Gary Cornell, –Core Java Volume – I Fundamentals , 11th Edition, Prentice Hall, 2019. 2. Paul Deitel, Harvey Deitel, Java SE 8 for programmers, 3rd Edition, Pearson, 2015. 3. Steven Holzner, Java 2 Black book, Dream tech press, 2011. 4. Timothy Budd, Understanding Object-oriented programming with Java, Third Edition, Pearson Education, 2008. 5. https://infyspringboard.onwingspan.com/web/en/app/toc/lex_29959473947367270000_shared/overview		
LIST OF EQUIPMENTS:		
1. Java and Eclipse / NetBeans IDE or Equivalent		

24PH201	PHYSICS FOR INFORMATION SCIENCE (Common to CSE, IT and AIML)		L	T	P	C
			3	0	2	4
OBJECTIVES: The Course will enable learners to: - Understand the classical free electron theory and Fermi distribution function - Relate the theory of laser with its applications in optical fibers - Solve the Schrodinger's wave equation in one dimensional and three dimensional box - Gain the basic knowledge in quantum operators and quantum gates - Comprehend the behavior of semiconductor diodes in various electron devices and nano electronic devices						
UNIT I		ELECTRICAL PROPERTIES OF MATERIALS				15
Classical free electron theory - Expression for electrical conductivity and thermal conductivity - thermal conductivity of a bad conductor- Lee's disc method -Effect of temperature on Fermi function - Density of energy states and average energy of an electron at 0 K- Effective mass of electron - Concept of hole. Semiconductors - Direct and Indirect bandgap semiconductors - Intrinsic Carrier Concentration - Bandgap Determination. (Theory -9) 1. Determination of Thermal conductivity of a bad conductor - Lee's Disc Method 2. Bandgap determination of intrinsic semiconductor (Laboratory- 6)						
UNIT II		LASER				18
Characteristics of Laser, Spatial and Temporal Coherence - Population inversion – Relation between Einstein's A and B coefficients - Components of Laser - Optical amplification (qualitative) - Semiconductor lasers: Homojunction and Heterojunction- Application: Engineering applications of lasers in data storage (qualitative), Principle of Fiber optics- Fiber optic communication system - Fiber optic sensors (pressure and displacement). (Theory 9) 1. Determination of divergence of the laser beam 2. Determination of acceptance angle and numerical aperture of an optical fiber 3. Determination of wavelength of semiconductor laser (Laboratory 9)						
UNIT III		QUANTUM THEORY				15
Introduction- Blackbody Radiation - Newton's law of cooling - Planck's quantum theory- matter waves, de-Broglie wavelength, Heisenberg's Uncertainty principle, Time independent and Time- dependent Schrödinger's wave equations, Physical significance of wave function, Particle in a one-dimensional potential box - Particle in a three-dimensional box (qualitative) - degenerate and non-degenerate energy levels- Quantum tunneling - Scanning Tunneling Microscope (STM). (Theory -9) 1. Determination of emissivity (Newton's law of cooling) 2. Determination of Planck's constant (Laboratory- 6)						
UNIT IV		BASICS OF QUANTUM COMPUTING				12
Quantum Operators: Linear vector spaces - inner product space - Hilbert space - examples Vectors and Tensors: Scalars and vectors, Dirac notations of Bra-Ket notation-						

Matrix representation of observables and states. Quantum Computing: Quantum states - classical bits - quantum bits or qubits – Entanglement and superposition - multiple qubits - Bloch sphere - quantum gates - CNOT gate - Advantages of quantum computing over classical computing. (Theory -9) 1. Truth table verification of CNOT gate through Virtual Laboratory (Laboratory-3)		
UNIT V	NANOELECTRONIC DEVICES	15
Introduction to Nano materials – synthesis by sol gel method, properties - Quantum confinement - Quantum structures: Density of energy states of quantum wells, quantum wires and quantum dots- band gap of nanomaterials - Quantum dot laser- Single electron phenomena -single electron transistor - Quantum system for information processing. (Theory -12) 1. Synthesis of Nano-powders by sol-gel method (Laboratory- 3)		
TOTAL: 75 PERIODS		
OUTCOMES: Upon completion of the course, the students will be able to: CO1: Derive electrical and thermal conductivities using classical free electron theory CO2: Calculate the electrical conductivity and bandgap in Intrinsic semiconductors CO3: Associate the basic principles of working of laser and their applications in fiber optics CO4: Calculate the energy eigen value and eigen function for a particle in a one-dimensional and three dimensional box using Schrodinger wave equations CO5: Use quantum operators to frame equations for logic gates in Quantum computing CO6: Relate the quantum properties of nanoscale materials with their applications		
TEXTBOOKS:		
1. Neil W Ashcroft and N David Mermin, Solid State Physics, Harcourt College Publishers,1976 2. M.N. Avadhanulu and P.G. Kshirsagar, A textbook of Engineering Physics, S. Chand and Company, New Delhi, 2014. 3. David J. Griffiths, Introduction to Quantum Mechanics, 2nd Edition, Pearson Prentice-Hall (2004). 4. Thomas G. Wong, Introduction to Classical and Quantum Computing, Rooted Grove (2022).		
REFERENCES:		
1. R. A. Serway and J.W. Jewett, Physics for Scientists and Engineers, Ninth Edition, Cengage Learning, 2014. 2. Marikani, Materials Science, PHI Learning Private Limited, Eastern Economy Edition, 2017. 3. R. Wolfson, Essential University Physics, Volume 1 and 2 with Mastering Physics, Global Edition,3rd Edition, Pearson 2017. 4. B.K. Pandey and S. Chaturvedi, Engineering Physics, Cengage Learning India, 2012. 5. Scott Aaronson, Quantum Computing Since Democritus, Cambridge University Press (2014). 6. Mermin, N. David, Making better sense of quantum mechanics. Reports on Progress in Physics 82.1 (2018): 012002. 7. Michael Nielsen, L. Isaac Chuang, Quantum Computation and Quantum Information, Cambridge University Press (2010).		

8. NPTEL course on –Introduction to LASER by Prof. M. R. Shenoy, IIT Delhi : https://onlinecourses.nptel.ac.in/noc24_ph45/preview
9. NPTEL course on –Introduction to Quantum Computing: Quantum Algorithms and Qiskit by Prof. Prabha Mandayam, Prof. Anupama Ray, Prof. Sheshashayee

24AM201	INTRODUCTION TO ARTIFICIAL INTELLIGENCE (Common to all Branches)		L	T	P	C
			2	0	2	3
OBJECTIVES: The Course will enable learners to: - Understand the basics and applications of Artificial Intelligence. - Apply the basics of Python programming. - Use python libraries to solve simple problems. - Understand the different types of Machine Learning algorithms. - Solve real world problems using AI/ML. - Explore the various applications in the field of Artificial Intelligence and Machine Learning.						
UNIT I		ARTIFICIAL INTELLIGENCE				6+6
Introduction - Types of AI - ANI, AGI, ASI - Narrow, General, Super AI, Examples - AI problems - Production Systems - State space Representation - Applications of AI in various industries. List of Exercise: 1. Build a simple AI model using python.						
UNIT II		BASICS OF PYTHON				6+6
Introduction to Python programming – Arithmetic Operators - values and types - variables, expressions, statements - Functions - Conditionals and Recursion -Iteration. Lists: Sequence, Mutable, Traversing, Operations, list slices, list methods - Tuples: Immutable, Tuple Assignment, Tuple as Return Values, Comparing and Sorting. List of Exercises: 1. Compute the GCD of two numbers. 2. Operations on Tuples: a) finding repeated elements, b) slice a tuple c) reverse a tuple d) replace last value of a tuple.						
UNIT III		PYTHON LIBRARIES				6+6
Introduction to Numpy - Multidimensional Ndarrays - Indexing - Properties - Constants - Data Visualization: Narray Creation - Matplotlib - Introduction to Pandas - Series - Dataframes - Visualizing the Data in Dataframes - Pandas Objects - Data Indexing and Selection – Handling missing data – Hierarchical indexing – Combining datasets – Aggregation and Grouping - Joins- Pivot Tables - String operations - Working with time series - High performance Pandas. List of Exercises: 1. Download, install and explore the features of R/Python for data analytics - Installing Anaconda - Basic Operations in Jupyter Notebook - Basic Data Handling 2. Working with Numpy arrays - Creation of numpy array using the tuple, Determine the size, shape and dimension of the array, Manipulation with array Attributes, Creation of Sub array, Perform the reshaping of the array along the row vector and column vector, Create two arrays and perform the concatenation among the arrays. 3. Working with Pandas data frames - Series, DataFrame , and Index, Implement the Data Selection Operations, Data indexing operations like: loc, iloc, and ix,						

operations of handling the missing data like None, Nan, Manipulate on the operation of Null Vaues (is null(), not null(), dropna(), fillna()). 4. Perform the Statistics operation for the data (the sum, product, median, minimum and maximum, quantiles, arg min, arg max etc.). 5. Use any data set compute the mean ,standard deviation, Percentile.	
UNIT IV	MACHINE LEARNING
6+6	
Introduction – ML Algorithms Overview – Types – Supervised – Unsupervised – Reinforcement Learning - Introduction to Neural Networks - Working of Deep Learning - Applications of DL - Ethical consideration in AI and ML. List of Exercise: 1. Apply any Machine Learning model to predict the sales in a store.	
UNIT V	CASE STUDIES
6+6	
Disease Prediction - Share Price Forecasting - Weather Prediction - Domain Specific Case Studies. List of Domain Specific Case Studies: • For CSE & allied: Sentiment analysis of product reviews using machine learning. • For ECE & allied: Smart homes using AI. • For EEE: Forecasting of Renewable energy availability during a specified period using AI. • Civil: Application of ML for crack detection on concrete structures. • Mech: Predictive Maintenance for CNC Machines Using AI and Machine Learning. List of Exercise: 1. Build a machine learning model to solve any real-world problem from your domain.	
TOTAL: 30(L) + 30(P) = 60 PERIODS	
OUTCOMES: Upon completion of the course, the students will be able to: CO1: Elaborate the basics and applications of Artificial Intelligence. CO2: Apply the basics of Python programming to solve problems. CO3: Use python libraries to solve simple ML problems. CO4: Outline the different types of Machine Learning algorithms. CO5: Use Machine Learning Algorithms to solve real world problems. CO6: Outline the recent developments in the field of Artificial Intelligence.	
TEXT BOOKS:	
1. Allen B. Downey, –Think Python: How to Think Like a Computer Scientist , 2nd edition, Updated for Python 3, Shroff/O_Reilly Publishers, 2016.	
2. Jake VanderPlas, –Python Data Science Handbook – Essential tools for working with data , O'Reilly, 2017.	
3. Steve Abrams, –Artificial Intelligence and Machine Learning for Beginners: A simple guide to understanding and Applying AI and ML , Independently published, May 14, 2024.	
REFERENCES:	
1. Vinod Chandra S S, Anand Hareendran S, Artificial Intelligence and Machine Learning, PHI Learning, 2014.	
2. Russell, S. and Norvig, P, Artificial Intelligence: A Modern Approach, Third Edition, Prentice Hall, 2010.	
3. Ethem Alpaydin, Introduction to Machine Learning, Second Edition, the MIT Press, Cambridge, Massachusetts, London, England.	
4. Stephen Marsland, Machine Learning - An Algorithmic Perspective, 2nd Edition, 2015, by Taylor & Francis Group, 2015.	
5. Tom M. Mitchell, Machine Learning, McGraw-Hill Science, ISBN: 0070428077	
6. Mayuri Mehta, Vasile Palade, Indranath Chatterjee, Explainable AI: Foundations, Methodologies and Applications, Springer, 2023.	

7. Siddhartha Bhattacharyya, Indrajit Pan, Ashish Mani, Sourav De, Elizabeth Behrman, Susanta Chakraborti, "Quantum Machine Learning", De Gruyter Frontiers in Computational Intelligence, 2020.

LIST OF EQUIPMENTS:

1. Systems with Anaconda, Jupyter Notebook, Python.

24GE201	TAMILS AND TECHNOLOGY (Common to all branches)	L	T	P	C
		1	0	0	1
OBJECTIVES: The Course will enable learners to: - Recognize the historical significance of weaving and pottery technologies in ancient Tamil civilization. - Highlight the concepts of design and construction technology during the Sangam age. - Provide an overview of manufacturing technology and its role in Tamil society. - Illustrate the agricultural and irrigation techniques employed in ancient Tamil society. - Promote scientific Tamil and Tamil computing.					
UNIT I	WEAVING AND CERAMIC TECHNOLOGY				3
Weaving Industry during Sangam Age - Ceramic technology - Black and Red Ware Potteries (BRW) - Graffiti on Potteries.					
UNIT II	DESIGN AND CONSTRUCTION TECHNOLOGY				3
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.					
UNIT III	MANUFACTURING TECHNOLOGY				3
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.					
UNIT IV	AGRICULTURE AND IRRIGATION TECHNOLOGY				3
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 - Conche diving - Ancient Knowledge of Ocean - Knowledge Specific Society.					
UNIT V	SCIENTIFIC TAMIL & TAMIL COMPUTING				3
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.					
					TOTAL: 15 PERIODS
OUTCOMES: Upon completion of the course, the students will be able to: CO1: Identify the role of weaving and ceramic technology in ancient Tamil Culture. CO2: Assess the design and construction technology ideas in the current Tamil society. CO3: Identify the different types of manufacturing technology used in Tamil society and					

their significance.

CO4: Classify agricultural and irrigation technologies in ancient Tamil society and its current relevance.

CO5: Discuss the fundamentals of scientific Tamil and Tamil computing.

REFERENCES:

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

24GE211	IDEA LAB - II (Common to all Branches)	L	T	P	C
		0	0	2	1
OBJECTIVES:					
The Course will enable learners to:					
• Develop hands-on experience and practical application of theoretical knowledge. • Develop their ability to explain the process involved.					
LIST OF EXPERIMENTS					
1. Printing of a 3D part. 2. Scanning of a 3D part. 3. Design and fabrication of press fit object using laser cutting machine. 4. Design and fabrication of 3D part using CNC Router. 5. Design and fabrication of simple PCB. 6. Soldering and desoldering of given electronic circuit.					
TOTAL: 30 PERIODS					
OUTCOMES:					
Upon the completion of this course the students will be able to					
CO1: Analyze the latest manufacturing methods in advancements and technologies related to their field.					
CO2: Understand the operations of a laser cutting machine and CNC Router.					
CO3: Analyze the process of design and fabrication of PCB and Soldering operations					
CO4: Develop technical proficiency and problem-solving abilities, making more					

competent and confident in their field. CO5: Develop themselves with the skills needed to address industry-specific problems		
LIST OF EQUIPMENT FOR A BATCH OF 30 STUDENTS		
S.No.	Description of Equipment	Quantity
1	CNC Router	1 No
2	3D Printer	1 No
3	3D Scanner	1 No
4	Laser cutting Machine	1 No
5	Multimeter	5 Nos.
6	Solder Stations	5 Sets
7	Desoldering Machine	1 No
8	PCB Milling Machine	1 No
9	Variable Power Supply	1 No
10	Electronic Components like Resistors, Transistors, Diode, Inductor, Capacitor, etc.	5 Sets

24HS211	INNOVATION AND CREATIVITY SKILLS AND DEVELOPMENT (Common to All Branches)	L	T	P	C
		1	0	0	1
OBJECTIVES: The Course will enable learners to: - Understand study plans, co-curricular activities, programming skills, recruitment test patterns, and hiring strategies through national qualifiers and hackathons. - Equip students with strategies for higher education, resume enhancement, project management, and securing internships - understand entrepreneurship fundamentals, including key differences, global hubs, business ideas, and scalability. - develop essential entrepreneurial skills such as opportunity recognition, patience, risk management, communication, persistence, and leadership - Understand life, success, self-confidence, health, scientific heritage, personal counseling, and cybercrime awareness					
UNIT I	STEPPING STONE – ENGINEERING CAREERS AND SKILL DEVELOPMENT				3
Study Plans and Resources - Identification of key resources and job opportunities - career prospects and academic growth through co-curricular activities - importance of programming/coding skills - Overview of test patterns and essential skills for popular campus recruiters - Comparison of IT Services, Dream, and Super Dream offers and their recruitment processes - National Qualifier Tests and their impact on hiring processes – Overview of Corporate contests and hackathons (e.g., TCS Codevita, HackerRank)					
UNIT II	STEPPING STONE – HIGHER EDUCATION AND CAREER DEVELOPMENT				3
Overview of higher education opportunities: GATE, GRE, GMAT, XAT, CAT, MAT - Exam formats, preparation strategies, and timelines - Resume Enhancement Strategies - Project Management - Steps to develop projects from proposal to prototype - Internship Pathways -					

Strategies for maximizing internship experiences for career advancement		
UNIT III	FUNDAMENTALS OF ENTREPRENEURSHIP: FROM IDEAS TO VENTURES	3
Introduction to Entrepreneurship – Intrapreneur vs. entrepreneur - Roles and Contributions - Global Entrepreneurship Hubs - Overview of Key Global Locations - Idea vs. Commercial Value - Transforming Ideas into Viable Business Models - Characteristics of Successful Business Ideas - Understanding Market Competition - Basics of Copyrights and Intellectual Property - Scalability in Business Ventures - Strategies for Scaling a Business		
UNIT IV	HUMAN SKILLS FOR ENTREPRENEURSHIP	3
Identifying and capitalizing on business opportunities - Case studies and anecdotes - Patience and Risk Management - The role of patience in entrepreneurial success and decision-making - Effective Communication - Techniques for clear and persuasive communication - Importance of communication in building and leading teams - Leadership qualities and their impact on entrepreneurial ventures - Analyzing success and failure stories		
UNIT V	FOUNDATIONS OF PERSONAL DEVELOPMENT AND WELL-BEING	3
Understanding Life and Success - Self-Confidence and Fear - Practical strategies for enhancing self-esteem - Adolescent Issues - Health Management - Basics of a balanced diet - Benefits of physical activity - Scientific Heritage of India - Overview of India's scientific achievements and contributions - Cyber Crime Awareness - Types and prevention strategies		
		TOTAL 15 PERIODS
OUTCOMES: Upon completion of the course, the students will be able to: CO1: Create study plans, value co-curricular activities, develop programming skills, and navigate for career advancement CO2: Understand about higher education options, resume enhancement, project management, and securing internships CO3: Learn entrepreneurship skills and strategies to develop successful business ideas.. CO4: Develop key entrepreneurial skills like opportunity recognition, risk management, and leadership through real-world examples CO5: Explore personal development, health management, scientific heritage, and cybercrime awareness		

24MC201	Yoga for Stress Management (Common to all branches)	L	T	P	C
		0	0	1	0
OBJECTIVES: The Course will enable learners to: • Understanding the different types of stress and managing stress. • Develop an understanding of practicing yoga • Learning to do asanas, including sitting, standing and lying postures					
UNIT I	Stress Management				3
Definition of Stress - Stress in Daily Life - Impact of Stress on Life - Identifying the Causes of Stress - Symptoms of Stress - Managing Stress (Habits, Tools, Training, Professional Help) - Complications of Stress Mismanagement - The Importance of Sleep for Mental Wellness -					

Connection Between Sleep and Digestion.		
UNIT II	Introduction to Yoga	3
Meaning and Definition of Yoga - Aims and Objectives of Yoga - Guidelines for Practicing Asanas - Benefits of Yoga		
UNIT III	Different Asanas	3
Methods of Performing Asanas - Pranayama - Suryanamaskar Asanas - Sitting Postures: Uttanpadasana, Paschimottanasana, Janu Sirsasana, Baddha Konasana - Shishupal Asana - Vajrasana		
UNIT IV	Standing Postures	3
Uttanasana - Trikonasana - Vrikshasana - Tadasana - Superbrain asana		
UNIT V	Lying Postures	3
Pavana Muktasana - Pada Sanchalanasana - Jhulana Lurhakanasana - Dhanurasana - Marjaryasana. Bitilasana Dictionaries - Sorkuvai Project.		
		TOTAL: 15 PERIODS
OUTCOMES: Upon completion of the course, the students will be able to: CO1: Relieve stress and achieve mental wellness. CO2: Experience the benefits of yoga CO3: Keep self and body healthy		
REFERENCES: 1. Iyengar, Bellur Krishnamukar Sundara. "Light on yoga." (1965). 2. Desikachar, Tirumalai Krishnamacharya Venkata. The heart of yoga: Developing a personal practice. Simon and Schuster, 1999. 3. Davis, Martha, Elizabeth Robbins Eshelman, and Matthew McKay. The relaxation and stress reduction workbook. New Harbinger Publications, 2008. 4. Krishnamacharya, Tirumalai, et al. "Yoga makaranda: The nectar of yoga." Swathi Soft (2013).		

24CH102	ENVIRONMENTAL SCIENCE AND SUSTAINABILITY (Common to all Branches)	L	T	P	C
		2	0	0	0
OBJECTIVES: The Course will enable learners to: • To gain knowledge of the environment and various natural resources. • To identify the Scientific and Technological solutions to pollution issues and Waste management. • To understand the significance of the conservation of biodiversity. • To recognize the needs and benefits of sustainability and its management. • To comprehend the effects of human population on the environment.					
UNIT I	NATURAL RESOURCES				7
Definition, scope and importance of environment – need for public awareness. Introduction to natural resources - types - forest resources: use and over-exploitation, deforestation and its impacts, food resources: effects of modern agriculture, organic farming, renewable energy sources - solar, wind, geothermal, tidal, OTE and biomass. field activity -tree plantation					
UNIT II	POLLUTION AND WASTE MANAGEMENT				7
Pollution - definition –causes, effects and control measures of (a) air pollution (b) water pollution(c) soil pollution (d) noise pollution (e) nuclear hazards - nuclear accidents and holocaust - role of an individual in prevention of pollution –case studies.					
Waste management- municipal solid wastes, E- waste, plastic waste.					

Field study - Solid waste management of the institution		
UNIT III	BIODIVERSITY AND ITS CONSERVATION	6
Biodiversity: types - values of biodiversity, India as a mega-diversity nation - hot-spots of biodiversity – threats to biodiversity – endangered and endemic species, extinct, rare, vulnerable species of India - conservation of biodiversity: in-situ and ex-situ method.		
Field study - Biodiversity of the institution		
UNIT IV	SUSTAINABILITY AND MANAGEMENT	5
Sustainability-concept, needs and challenges- circular economy - sustainable development goals- concept of carbon footprint, environmental impact assessment, clean development mechanism, solutions.		
Field study - Alternate energy sources and its impacts		
UNIT V	HUMAN POPULATION	5
Introduction - population growth, variation among nations, population explosion, environment and human health – endemic/epidemic/pandemic – role of information technology in environment and human health.		
Case Study - Pandemics of 21 st century		
TOTAL: 30 PERIODS		
OUTCOMES: Upon completion of the course, the students will be able to: CO1: To investigate and use conservational practices to protect natural resources. CO2: To identify the causes of pollutants and illustrate suitable methods for pollution abatement. CO3: To analyze the values of biodiversity and its conservational methods. CO4: To classify suitable sustainable development practices and apply it in day-to-day life. CO5: To assess the impacts of human population and suggest suitable solutions. CO6: To develop innovative solutions and strategies to address sustainability challenges.		
TEXTBOOKS:		
1. Anubha Kaushik and C.P. Kaushik, –Perspectives in environmental studies , New Age International Publishers, 8th edition, 2024. 2. Benny Joseph, Environmental Science and Engineering, McGraw-Hill, 1st edition, 2017. 3. Gilbert M. Masters, Introduction to Environmental Engineering and Science, Pearson Education, 3rd edition, 2014. 4. Erach Bharuch, Textbook of Environmental Studies for Undergraduate Courses, Universities Press(I) Pvt. Ltd., 3rd edition, 2021.		
REFERENCES:		
1. 1. William P. Cunningham and Mary Ann Cunningham Environmental Science: A Global Concern, McGraw Hill, 14th edition, 2017. 2. Rajagopalan, R, Environmental Studies-From Crisis to Cure, Oxford University Press, 2015. 3. G. Tyler Miller and Scott E. Spoolman, –Environmental Science, Cengage Learning India Pvt, Ltd., Delhi, 2014. 4. Allen, D. T. and Shonnard, D. R., Sustainability Engineering: Concepts, Design and Case Studies, Prentice Hall, 2012. 5. Bradley. A.S; Adebayo, A.O. and Maria, P. Engineering applications in sustainable design and development, Cengage learning, 2015. 6. 6. Environment Impact Assessment Guidelines, Notification of Government of India, 2006 and subsequent amendments, 2022		

SEMESTER III

24MA301	DISCRETE MATHEMATICS (Common to CSE, IT)	L	T	P	C
		3	1	0	4
OBJECTIVES: The Course will enable the learners to: - Describe the arguments using connectives and rules of inference. - Introduce the basic concept of counting and generating functions. - Construct recurrence relations for mathematical models. - Define the graphs and its models. - Understand the concept of group theory, Lattices and Boolean algebra.					
UNIT I	LOGIC AND PROOFS	12			
Propositional logic - Propositional equivalences - Predicates and quantifiers - Nested quantifiers - Rules of inference - Introduction to proofs - Proof methods and strategy.					
UNIT II	COMBINATORICS	12			
Mathematical induction - Strong induction and well ordering - The basics of counting - The pigeonhole principle - Permutations and combinations - Recurrence relations - Solving linear recurrence relations - Generating functions - Inclusion and exclusion principle and its applications.					
UNIT III	GRAPHS	12			
Graphs and graph models - Graph terminology and special types of graphs - Matrix representation of graphs and graph isomorphism - Connectivity - Euler and Hamilton paths.					
UNIT IV	ALGEBRAIC STRUCTURES	12			
Algebraic systems - Semi groups and monoids - Groups - Subgroups - Homomorphism_s - Normal subgroup and cosets - Lagrange_s theorem - Definitions and examples of Rings and Fields.					
UNIT V	LATTICES AND BOOLEAN ALGEBRA	12			
Partial ordering -Posets- Lattices as posets- Properties of lattices - Lattices as algebraic systems - Sub lattices - Direct product and homomorphism - Some special lattices - Boolean algebra.					
TOTAL: 60 PERIODS					
OUTCOMES: Upon completion of the course, the students will be able to: CO1: Examine the validity of the arguments. CO2: Apply various proof techniques and principles using analytic and combinatorial methods. CO3: Develop the recurrence relation for the sequence. CO4: Implement graph theory techniques to solve real time problems CO5: Apply the concepts of groups, rings, and fields in solving algebraic problems. CO6: Solve problems in Lattices and Boolean algebra.					

TEXT BOOK:

1.K. H. Rosen, "Discrete Mathematics and its Applications", 8th Edition, Tata McGraw Hill Pub. Co. Ltd., New Delhi, Special Indian Edition, 2021.

2.J. P. Tremblay, and R. Manohar. " Discrete Mathematical Structures with Applications to Computer Science", Tata McGraw Hill Pub. Co. Ltd, New Delhi, 30th Reprint, 2017.

REFERENCES:

1.R.P. Grimaldi, "Discrete and Combinatorial Mathematics: An Applied Introduction", 5th Edition, Pearson Education Asia, New Delhi, Reprint 2019.

2.S. Lipschutz, and Mark Lipson., "Discrete Mathematics", Schaum's Outlines, Tata McGraw Hill Pub. Co. Ltd., New Delhi, 4th Edition, 2021.

3.T. Koshy, "Discrete Mathematics with Applications", Elsevier Publications, 1st Edition, 2014.

24GE301	UNIVERSAL HUMAN VALUES 2: UNDERSTANDING HARMONY	L	T	P	C
		2	1	0	3
OBJECTIVES: The Course is designed to: • Development of a holistic perspective based on self-exploration about themselves (human beings), family, society and nature/existence. • Understanding (or developing clarity) of the harmony in the human being, family, society, and nature/existence • Strengthening of self-reflection. • Development of commitment and courage to act					
UNIT I	NEED, BASIC GUIDELINES, CONTENT AND PROCESS FOR VALUE EDUCATION				12
Purpose and motivation for the course - recapitulation from Universal Human Values-I - SelfExploration-what is it? – It's content and process; _Natural Acceptance' and Experiential Validation- as the process for self-exploration - Continuous Happiness and Prosperity- A Look at Basic Human Aspirations - Right understanding, Relationship and Physical Facility- the basic requirements for the fulfilment of aspirations of every human being with their correct priority - Understanding Happiness and Prosperity Correctly- A critical appraisal of the current scenario - Method to fulfil the above human aspirations: understanding and living in harmony at various levels. Practice sessions: To discuss natural acceptance in human beings as the innate acceptance of living with responsibility (living in relationship, harmony, and co-existence) rather than as arbitrariness in choice based on liking-disliking.					
UNIT II	UNDERSTANDING HARMONY IN THE HUMAN BEING – HARMONY IN MYSELF!				12
Understanding human beings as a co-existence of the sentient _I' and the material _Body' - Understanding the needs of Self (_I') and _Body' - happiness and physical facility -Understanding the Body as an instrument of _I' (I being the doer, seer and enjoyer) - Understanding the characteristics and activities of _I' and harmony in _I' - _Understanding the harmony of I with the Body: Sanyam and Health; correct appraisal of Physical needs, the meaning of Prosperity in					

detail - Programs to ensure Sanyam and Health. Practice sessions: To discuss the role others have played in making material goods available to me. Identifying from one's own life. Differentiate between prosperity and accumulation. Discuss programs for ensuring health vs dealing with the disease.

UNIT III	UNDERSTANDING HARMONY IN THE FAMILY AND SOCIETY HARMONY IN HUMAN-HUMAN RELATIONSHIP	12
-----------------	--	-----------

Understanding values in a human-human relationship - the meaning of Justice (nine universal values in relationships) and the program for its fulfillment to ensure mutual happiness -Trust and Respect as the foundational values of relationship - Understanding the meaning of Trust; Difference between intention and competence - Understanding the meaning of Respect, Difference between respect and differentiation; the other salient values in a relationship - Understanding the harmony in the society (society being an extension of the family) - Resolution, Prosperity, fearlessness (trust) and co-existence as comprehensive Human Goals - Visualizing a universal harmonious order in society- Undivided Society, Universal Order- from family to world family. Practice sessions: To reflect on relationships in family, hostel and institute as extended family, real-life examples, teacher-student relationship, the goal of education etc. Gratitude as a universal value in relationships. Discuss scenarios. Elicit examples from students' lives.

UNIT IV	UNDERSTANDING HARMONY IN NATURE AND EXISTENCE - WHOLE EXISTENCE AS COEXISTENCE	12
----------------	---	-----------

Understanding the harmony in Nature - Interconnectedness and mutual fulfillment among the four orders of nature- recyclability and self-regulation in nature - Understanding Existence as the Coexistence of mutually interacting units in all-pervasive Space - Holistic perception of harmony at all levels of existence. Practice sessions: To discuss human beings as the cause of imbalance in nature (film –Home|| can be used), pollution, depletion of resources and role of technology etc

UNIT V	IMPLICATIONS OF THE ABOVE HOLISTIC UNDERSTANDING OF HARMONY IN PROFESSIONAL ETHICS	12
---------------	---	-----------

Natural acceptance of human values - Definitiveness of Ethical Human Conduct - Basis for Humanistic Education, Humanistic Constitution and Humanistic Universal Order - Competence in professional ethics: a. Ability to utilize professional competence for augmenting universal human order b. Ability to identify the scope and characteristics of people-friendly and eco-friendly production systems, c. Ability to identify and develop appropriate technologies and management patterns for the above production systems - Case studies of typical holistic technologies, management models and production systems - Strategy for the transition from the present state to Universal Human Order: a. At the level of the individual: as socially and ecologically responsible engineers, technologists, and managers b. At the level of society: as mutually enriching institutions and organizations - Sum up. Practice Exercises and Case Studies e.g. To discuss the conduct as an engineer or scientist etc.

TOTAL: 60 PERIODS

OUTCOMES:

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

CO1: Develop self-awareness and a deeper understanding of their surroundings, including family, society, and nature

CO2: Identify and resolve inner conflicts based on natural acceptance.

CO3: Become more responsible towards life, and handle problems with sustainable solutions by considering human relationships and natural harmony.

CO4: Enhance their critical thinking and analyzing skills.

CO5: Develop a stronger commitment towards human values, relationships, and societal well-being.

CO6: Apply what they have learnt in different day-to-day settings in real life, and take the initial steps towards integrating these values into daily life.

TEXT BOOK:

1. Human Values and Professional Ethics by R R Gaur, R Sangal, G P Bagaria, Excel Books, First edition: New Delhi, 2010, ISBN: 978-81-7446-781-2 Reprint: 2010, 2011, 2013, 2016, 2018, 2019 (Second edition: New Delhi, 2019, ISBN: 978-93-87034-47-1)
2. A foundation course in Human Values and Professional Ethics RR Gaur, R Asthana, GP Bagaria, Third revised edition: New Delhi, 2023, ISBN: 978-81-957703-7-3 (Printed Copy), ISBN: 978-81-957703-6-6 (e-book), UHV Publications

REFERENCES:

1. Jeevan Vidya: Ek Parichaya, A Nagaraj, Jeevan Vidya Prakashan, Amarkantak, 1999.
2. Human Values, A.N. Tripathi, New Age Intl. Publishers, New Delhi, 2004.
3. The Story of Stuff (Book).
4. The Story of My Experiments with Truth - by Mohandas Karamchand Gandhi
5. Small is Beautiful-E. F Schumacher.
6. Slow is Beautiful-Cecile Andrews
7. Economy of Permanence - J C Kumarappa
8. Bharat Mein Angreji Raj - Pandit Sunderlal
9. Rediscovering India - by Dharampal
10. Hind Swaraj or Indian Home Rule - by Mohandas K. Gandhi

24CS301	COMPUTER ORGANIZATION AND ARCHITECTURE (Common to CSE, IT and AIML)	L	T	P	C
		3	0	0	3
OBJECTIVES: The Course will enable learners to: • Describe the basic principles and operations of digital computers. • Design arithmetic and logic unit for various fixed and floating point operations • Construct pipeline architectures for RISC processors. • Explain various memory systems & I/O interfacing • Discuss parallel processor and multi-processor architectures					
UNIT I	COMPUTER FUNDAMENTALS				9
Computer Types - Functional Units — Basic Operational Concepts - Number Representation and Arithmetic Operations - Performance Measurement - Instruction Set Architecture - Memory Locations and Addresses - Instructions and Instruction Sequencing - Addressing Modes.					
UNIT II	COMPUTER ARITHMETIC				9
Addition and Subtraction of Signed Numbers - Design of Fast Adders - Multiplication of Unsigned Numbers - Multiplication of Signed Numbers - Fast Multiplication - Integer Division - Floating-Point Numbers - Representation and Operations.					

Study Experiments: Ripple Carry Adder, Carry Look-a-head Adder, Booth's Multiplier		
UNIT III	BASIC PROCESSING UNIT AND PIPELINING	9
Basic Processing Unit: Concepts - Instruction Execution - Hardware Components - Instruction Fetch and Execution Steps -Control Signals - Hardwired Control. Pipelining: Basic Concept - Pipeline Organization- Pipelining Issues - Data Dependencies - Memory Delays - Branch Delays - Resource Limitations - Performance Evaluation -Superscalar Operation. Study Experiments: Arithmetic Logic Unit, CPU Design		
UNIT IV	I/O AND MEMORY	9
Input/Output Organization: Bus Structure - Bus Operation - Arbitration - The Memory System: Basic Concepts - Semiconductor RAM Memories - Read-only Memories - Direct Memory Access - Memory Hierarchy - Cache Memories - Performance Considerations - Virtual Memory - Memory Management Requirements - Secondary Storage. Study Experiments: Memory Design, Associative Cache Design, Direct Mapped Cache Design.		
UNIT V	PARALLEL PROCESSING AND MULTICORE COMPUTERS	9
Parallel Processing: Use of Multiple Processors - Symmetric Multiprocessors - Multithreading and Chip Multiprocessors - Clusters - Nonuniform Memory Access Computers - Vector Computation - Multicore Organization.		
TOTAL: 45 PERIODS		
OUTCOMES: Upon completion of the course, the students will be able to: CO1: Infer the basic principles and operations of digital computers. CO2: Analyze the performance of computers by identifying factors that contribute to performance. CO3: Apply arithmetic algorithms for various operations. CO4: Design hardware to solve computationally intensive problems. CO5: Compare various I/O methods and analyze memory management techniques. CO6: Demonstrate the concept of parallelism in hardware and software		
TEXT BOOKS: 1. Carl Hamacher, Zvonko Vranesic, Safwat Zaky, Computer organization, Tata McGraw Hill, Sixth edition, 2012. 2. David A. Patterson and John L. Hennessy Computer Organization and Design-The Hardware/Software Interface, 5th edition, Morgan Kaufmann, 2013.		
REFERENCES: 1. John P.Hayes, Computer Architecture and Organization, Third Edition, Tata McGraw Hill, 2012. 2. David A. Patterson and John L. Hennessy Computer Organization and Design-The Hardware/Software Interface, 6th edition, Morgan Kaufmann, 2021. 3. John L. Hennessy and David A. Patterson, Computer Architecture - A Quantitate Approach, Morgan Kaufmann / Elsevier Publishers, Fifth Edition,2012.		

24CS302	ADVANCED JAVA PROGRAMMING (Lab Integrated) (Common to All Branches)	L	T	P	C
		3	0	3	4.5

OBJECTIVES:

The Course will enable learners to:

- Gain a comprehensive understanding of the Java Collections Framework and its various interfaces and implementations.
- Learn the details of Java I/O streams and utility classes for managing dates, numbers, and currencies.

• Develop a thorough understanding of the Stream API introduced in Java 8 and its various operations. • Explore advanced object serialization and string tokenizing techniques, including pattern matching with regular expressions. • Understand advanced Stream API features and gain proficiency in using regular expressions for text processing.		
UNIT I	COLLECTIONS FRAMEWORK AND UTILITY CLASSES	9+9
Introduction to Collections Framework - Collection Interface- Methods in Collection Interface - Iterable and Iterator Interfaces - List Interface- ArrayList - LinkedList - Set Interface - HashSet- LinkedHashSet - TreeSet - Map Interface - HashMap -LinkedHashMap- TreeMap - Queue Interface -PriorityQueue - Deque Interface - Utility Classes. List of Experiments 1. Write a program that measures the time taken for insertion, deletion, and search operations on ArrayList, LinkedList, HashSet, and TreeSet for varying sizes of input data. 2. Implement a custom data structure that combines features of a list and a set. 3. Write a Java program to create a HashMap where the keys are strings, and the values are integers Add five key-value pairs to the map. Print all the keys and values in the map. Remove an entry by key. Update the value associated with a specific key. Check if the map contains a specific key and a specific value.		
UNIT II	DATE HANDLING AND SERIALIZATION	9+9
Date – Calendar – Comparable interface – Observer Interface — Serialization – Dates - Numbers, and Currency - Working with Dates - Numbers and Currencies - Object Serialization - Serializable Interface - Writing and Reading Serializable Objects -Transient Keyword- serialVersionUID. List of Experiments 1. Create a class representing a complex object with nested data structures. Serialize the object to a file, then deserialize it back and verify that the object remains intact. 2. Write a program that formats dates and currencies according to different locales. 3. Create a class hierarchy representing different types of objects (e.g., Person, Employee). Serialize instances of these classes to a file using object serialization.		
UNIT III	STREAM API AND FUNCTIONAL PROGRAMMING PARADIGMS	9+9
Overview of Stream API - Importance of Stream API in Java 8 and Beyond – Functional Programming Concepts - Creating Streams - Stream Interface Methods - Stream Operations - Intermediate Filtering (filter)-Mapping (map, flatMap)-Sorting (sorted)-Distinct (distinct) - Limit and Skip (limit, skip) - Terminal Operations -Collecting Results (collect) - Reducing and Summarizing (reduce, summaryStatistics)-Iterating (forEach) - Matching and Finding (anyMatch, allMatch, noneMatch, findFirst, findAny) -Counting (count). List of Experiments 1. Write a program that performs stream operations like filtering, mapping, and reducing. 2. Create an infinite stream generator that generates prime numbers. Implement methods to check for primality and generate the next prime number. 3. Write a program that reads a text file containing sentences. Tokenize each sentence into words, filter out stopwords, and print the remaining words.		
UNIT IV	ADVANCED STRING PROCESSING AND I/O TECHNIQUES	9+9

String Tokenizer - Parsing - Tokenizing and Formatting - Locating Data via Pattern Matching, Tokenizing - Streams - Types of Streams - The Byte-stream I/O hierarchy - Character Stream Hierarchy – Random Access File class – the java.io. Console Class - Advanced I/O - Piped Streams (PipedInputStream and PipedOutputStream) – SequenceInputStream - PushbackInputStream and PushbackReader.

List of Experiments

1. Write a program that reads a text file and tokenizes it into sentences using the StringTokenizer class.
2. Implement a java program that allows users to open a text file, navigate through it using random access, insert, delete, and modify text at specific positions within the file.
3. Implement a program that uses advanced I/O techniques like PipedInputStream, PipedOutputStream, SequenceInputStream, and PushbackInputStream.

UNIT V	ADVANCED STREAM FEATURES AND REGULAR EXPRESSIONS	9+9
---------------	---	------------

Importance and Use Cases of Advanced Stream Features - Creating Custom Streams -Stream Generators (Stream.generate, Stream.iterate) - Infinite Streams - Using Spliterators – Advanced Stream Operations - FlatMapping - Chaining Stream Operations - Stream Peeking (peek) - Advanced Filtering Techniques - Introduction to Regular Expressions - Character Classes - Quantifiers - Pattern Matching - Groups and Capturing - Regex in Java - java.util.regex Package Pattern Class - Matcher Class - String Manipulation with Regex - Splitting Strings - Replacing Text (replaceAll, replaceFirst) - Replacing with Backreferences.

List of Experiments

1. Implement custom stream generators using Stream.generate and Stream.iterate methods.
2. Write a program that demonstrates advanced stream operations like flatMapping, chaining stream operations, and peeking.
3. Develop a program that utilizes regular expressions to perform string manipulation tasks such as splitting strings, replacing text, and extracting specific patterns.

TOTAL: 45+45 = 90 PERIODS

OUTCOMES:

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

CO1: Implement various data structures by utilizing core Java features and libraries

CO2: Demonstrate proficiency in handling Java I/O operations, including file manipulation for efficient data storage and retrieval.

CO3: Apply and Analyze the Stream API for functional programming and data processing.

CO4: Implement advanced object serialization for complex data structures.

CO5: Utilize regular expressions for text parsing and string manipulation.

CO6: Build applications using advanced Java programming techniques.

TEXT BOOK:

3. Cay S. Horstmann, "Core Java Volume I--Fundamentals," 12th Edition, 2019.
4. Joshua Bloch, "Effective Java," 3rd Edition, 2018.
5. Raoul-Gabriel Urma, "Java 8 in Action: Lambdas, Streams, and Functional-Style Programming," 1st Edition, 2014.
6. Herbert Schildt, "Java: The Complete Reference," 11th Edition, 2018.

7. Alan Mycroft and Martin Odersky, "Programming in Scala," 4th Edition, 2020.
REFERENCES: 11. Bruce Eckel, "Thinking in Java," 4th Edition, 2006. 12. Herbert Schildt, "Java: A Beginner's Guide," 8th Edition, 2019. 13. Richard Warburton, "Java 8 Lambdas: Pragmatic Functional Programming," 1st Edition, 2014.
LIST OF EQUIPMENTS: JDK/Eclipse

24CS303	DATABASE MANAGEMENT SYSTEMS (Lab Integrated) (Common to CSE, IT, CSBS and AIML)	L	T	P	C
		3	0	3	4.5
COURSE OBJECTIVES: The Course will enable the learners to: - Understand the basic concepts of Data Modeling and Database Systems. - Understand SQL and effective relational database design concepts. - Learn relational algebra, calculus and normalization. - Know the fundamental concepts of transaction processing, concurrency control techniques, recovery procedure and data storage techniques. - Understand query processing, efficient data querying and advanced databases.					
UNIT I	DATABASE CONCEPTS				9+9
Concept of Database and Overview of DBMS - Characteristics of databases -Data Models Schemas and Instances - Three-Schema Architecture - Database Languages and Interfaces- Introductions to data models types- ER Model- ER Diagrams - Enhanced ER Model - reducing ER to table Applications: ER model of University Database Application – Relational Database Design by ER- and EER-to-Relational Mapping.					
List of Exercise/Experiments Case Study using real life database applications anyone from the following list a) Inventory Management for a EMart Grocery Shop b) Society Financial Management c) Cop Friendly App - Eseva d) Property Management - eMall e) Star Small and Medium Banking and Finance - Build Entity Model diagram. The diagram should align with the business and functional goals stated in the application.					
UNIT II	STRUCTURED QUERY LANGUAGE				9+9

SQL Data Definition and Data Types - Constraints - Queries - INSERT, UPDATE, and DELETE in SQL - Views - Integrity Procedures, Functions, Cursor and Triggers - Embedded SQL - Dynamic SQL.

List of Exercise/Experiments

Case Study using real life database applications anyone from the following list and do the following exercises.

- a) Inventory Management for a EMart Grocery Shop
 - b) Society Financial Management
 - c) Cop Friendly App - Eseva
 - d) Property Management - eMall
 - e) Star Small and Medium Banking and Finance
1. Data Definition Commands, Data Manipulation Commands for inserting, deleting, updating and retrieving Tables and Transaction Control statements
 2. Database Querying - Simple queries, Nested queries, Sub queries and Joins
 3. Views, Sequences, Synonyms
 4. Database Programming: Implicit and Explicit Cursors
 5. Procedures and Functions
 6. Triggers
 7. Exception Handling

UNIT III	RELATIONAL ALGEBRA, CALCULUS AND NORMALIZATION	9+9
-----------------	---	------------

Relational Algebra – Operations - Domain Relational Calculus- Tuple Relational Calculus - Fundamental operations.

Relational Database Design - Functional Dependency - Normalization (1NF, 2NF 3NF and BCNF) -Multivalued Dependency and 4NF -Joint Dependencies and 5NF - De-normalization.

List of Exercise/Experiments

1. Case Study using real life database applications anyone from the following list
 - a) Inventory Management for a EMart Grocery Shop
 - b) Society Financial Management
 - c) Cop Friendly App - Eseva
 - d) Property Management - eMall
 - e) Star Small and Medium Banking and Finance.

Apply Normalization rules in designing the tables in scope.

UNIT IV	TRANSACTIONS, CONCURRENCY CONTROL AND DATA STORAGE	9+9
----------------	---	------------

Transaction Concepts - ACID Properties - Schedules based on Recoverability, Serializability - Concurrency Control – Need for Concurrency – Locking Protocols – Two Phase Locking – Transaction Recovery -Concepts - Deferred Update - Immediate Update.

Organization of Records in Files - Unordered, Ordered - Hashing Techniques - RAID - Ordered Indexes - Multilevel Indexes - B+ tree Index Files - B tree Index Files.

List of Exercise/Experiments

Case Study using real life database applications anyone from the following list

- a) Inventory Management for a EMart Grocery Shop
- b) Society Financial Management
- c) Cop Friendly App - Eseva

d) Property Management - eMail
 e) Star Small and Medium Banking and Finance
 Ability to showcase ACID Properties with sample queries with appropriate settings for the above scenario.

UNIT V	QUERY OPTIMIZATION AND ADVANCED DATABASES	9+9
---------------	--	------------

Query Processing Overview - Algorithms for SELECT and JOIN operations - Query optimization using Heuristics.

Distributed Database Concepts - Design -Concurrency Control and Recovery - NOSQL Systems - Document-Based NOSQL Systems and MongoDB.

Explain Plan Statement - Parsing Output - Join Orders and Methods - Indexes - Standard Issues - Query Tuning - Explain Plan vs Explain Analyses.

List of Exercise/Experiments

Case Study using real life database applications anyone from the following list

- Inventory Management for a EMart Grocery Shop
- Society Financial Management
- Cop Friendly App - Eseva
- Property Management - eMail
- Star Small and Medium Banking and Finance

Build PL SQL / Stored Procedures for Complex Functionalities, ex EOD Batch Processing for calculating the EMI for Gold Loan for each eligible Customer.

TOTAL: 45 + 45 = 90 PERIODS

OUTCOMES:

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

CO1: Map ER model to Relational model to perform database design effectively.

CO2: Implement SQL and effective relational database design concepts.

CO3: Apply relational algebra, calculus and normalization techniques in database design.

CO4: Understand the concepts of transaction processing, concurrency control, recovery procedure and data storage techniques.

CO5: Evaluate and implement transaction processing, concurrency control mechanisms, and recovery procedures to maintain data integrity.

CO6: Analyze and optimize database queries and understand the features and applications of advanced and distributed database systems, including NoSQL.

TEXTBOOKS:

- Elmasri R. and S. Navathe, –Fundamentals of Database Systems||, Pearson Education, 7th Edition, 2016.
- Abraham Silberschatz, Henry F.Korth, –Database System Concepts||, Tata McGraw Hill , 7th Edition, 2021.

REFERENCES:

- Elmasri R. and S. Navathe, Database Systems: Models, Languages, Design and Application Programming, Pearson Education, 2013.Raghu Ramakrishnan, Gehrke –Database Management Systems||, MCGraw Hill, 3rd Edition 2014.
- Plunkett T., B. Macdonald, –Oracle Big Data Hand Book|| , McGraw Hill, First Edition, 2013

3. Gupta G K , –Database Management Systems , Tata McGraw Hill Education Private Limited, New Delhi, 2011
4. C. J. Date, A. Kannan, S. Swamynathan, –An Introduction to Database Systems , Eighth Edition, Pearson Education, 2015.
5. Maqsood Alam, Aalok Muley, Chaitanya Kadaru, Ashok Joshi, Oracle NoSQL Database: Real-Time Big Data Management for the Enterprise, McGraw Hill Professional, 2013.
6. Thomas Connolly, Carolyn Begg, –Database Systems: A Practical Approach to Design, Implementation and Management , Pearson, 6 th Edition, 2015.
7. Database Management System Part - 1 https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_01275806667282022456_shared/overview
8. Database Management System Part - 2 https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_0127673005629194241_shared/overview
9. Online Resources:
10. https://infyspringboard.onwingspan.com/web/en/page/home
LIST OF EQUIPMENTS
Systems with MySQL, Eclipse or NetBeans

24CS304	OPERATING SYSTEMS (Lab Integrated) (Common to CSE and IT)	L	T	P	C
		2	0	2	3
OBJECTIVES: The Course will enable learners to: • Explain the basic concepts of operating systems and process. • Discuss threads and analyse various CPU scheduling algorithms. • Describe the concept of process synchronization and deadlocks. • Analyse various memory management schemes. • Describe I/O management and file systems.					
UNIT I	INTRODUCTION TO OPERATING SYSTEMS AND PROCESSES				6+6
Introduction: Computer system organization - architecture – Resource management - Protection and Security – Virtualization - Operating System Structures: Services - User and Operating-System Interface - System Calls - System Services - Design and Implementation - Building and Booting an Operating System – Processes: Process Concept - Process Scheduling - Operations on Processes - Inter process Communication - IPC in Shared-Memory Systems - IPC in Message-Passing Systems					
List of Exercise/Experiments: 1. Basic Unix file system commands such as ls, cd, mkdir, rmdir, cp, rm, mv, more, lpr, man, grep, sed, etc.. 2. Programs using Shell Programming. 3. Implementation of Unix System Calls. 4. Implementation of IPC using message queue a. Get the input data (integer value) from a process called sender b. Use Message Queue to transfer this data from sender to receiver process c. The receiver does the prime number checking on the received data					

d. Communicate the verified/status result from receiver to sender process, this status should be displayed in the Sender process. Note: Simultaneously execute two or more processes. Don't do it as a single process		
UNIT II	THREADS AND CPU SCHEDULING	6+6
Threads & Concurrency: Overview - Multicore Programming - Multithreading Models - Thread Libraries - Implicit Threading - Threading Issues - CPU Scheduling: Basic Concepts - Scheduling Criteria - Scheduling Algorithms - Thread Scheduling - Multi-Processor Scheduling - Real-Time CPU Scheduling		
List of Exercise/Experiments: - Write a program to implement the following actions using pthreads - Create a thread in a program and called Parent thread, this parent thread creates another thread (Child thread) to print out the numbers from 1 to 20. The Parent thread waits till the child thread finishes - Create a thread in the main program, this program passes the 'count' as arguments to that thread function and this created thread function has to print your name 'count' times. - Write C programs to implement the various CPU Scheduling Algorithms.		
UNIT III	PROCESS SYNCHRONISATION AND DEADLOCKS	6+6
Process Synchronization: The critical-section problem – Peterson's Solution, Synchronization hardware, Mutex locks, Semaphores, monitors - Classic problems of synchronization: Bounded Buffer Problem - Reader's & Writer Problem, Dining Philosopher Problem. Deadlock: System model - Deadlock characterization, Methods for handling deadlocks - Deadlock prevention - Deadlock avoidance - Deadlock detection - Recovery from deadlock.		
List of Exercise/Experiments: - Process Synchronization using Semaphores. A shared data has to be accessed by two categories of processes namely A and B. Satisfy the following constraints to access the data without any data loss. - When a process A1 is accessing the database another process of the same category is permitted. - When a process B1 is accessing the database neither process A1 nor another 74 process B2 is permitted. - When a process A1 is accessing the database process B1 should not be allowed to access the database. Write appropriate code for both A and B satisfying all the above constraints using semaphores. Note: The time-stamp for accessing is approximately 10 sec. - Bankers Algorithm for Deadlock Avoidance		
UNIT IV	MEMORY MANAGEMENT	6+6
Memory Management: Contiguous Memory Allocation - Paging - Structure of the Page Table - Swapping - Virtual Memory: Demand Paging - Copy-on write - Page Replacement - Allocation of frames - Thrashing - Memory Compression		
List of Exercise/Experiments: - Analysis and Simulation of Memory Allocation and Management Techniques - First Fit - Best Fit - Worst Fit - Implementation of Page Replacement Techniques - FIFO - LRU - Optimal page replacement		
UNIT V	STORAGE MANAGEMENT	6+6
Mass Storage Structure: Overview of Mass Storage Structure- HDD scheduling - Swap Space Management, I/O systems: I/O Hardware, Application I/O interface, Kernel I/O Subsystem, File System Interface: File Concept – Access Methods – Directory Structure – Protection, File-System Implementation: File-System Structure- File-System Operations - Directory Implementation - Allocation Methods - Free-Space Management, - Case Study-Linux		

List of Exercise/Experiments:				
1. Simulation of File Allocation Techniques i. Sequential ii. Linked list iii. indexed 2. Implementation of File Organization Strategies Single level directory ii. Two level directory iii. Hierarchical level directory				
TOTAL: 60 PERIODS				
OUTCOMES: Upon completion of the course, the students will be able to: CO1: Demonstrate the basic concepts of operating systems and process. CO2: Implement process management techniques using inter-process communication. CO3: Implement the concepts of process synchronization and deadlocks. CO4: Apply various memory management schemes for the suitable scenario. CO5: Describe various I/O and file management techniques. CO6: Develop practical skills in developing system-level programming.				
TEXTBOOKS: 1. Abraham Silberschatz, Peter Baer Galvin and Greg Gagne, –Operating System Concepts II, 10th Edition, John Wiley and Sons Inc., 2018. 2. Andrew S Tanenbaum, "Modern Operating Systems", Pearson, 5th Edition, 2022 New Delhi.				
REFERENCES: 1. William Stallings, "Operating Systems: Internals and Design Principles", 7th Edition, Prentice Hall, 2018. 2. Achyut S.Godbole, Atul Kahate, –Operating Systems , McGraw Hill Education, 2016.				
LIST OF EQUIPMENTS: 1. Standalone desktops with C/C++/Java/Equivalent compiler				

24GE311	PRODUCT DEVELOPMENT LAB - 1 (Design Thinking) (Common to All Branches)	L	T	P	C
		0	0	2	1
OBJECTIVES: The Course will enable learners to: - Identify any gaps in the literature, such as open questions or conflicts in previous studies. - Identify the need for additional research and areas of prior research to prevent duplication of effort. - encourage the rapid and creative generation of diverse ideas within a limited timeframe, promoting innovation, critical thinking, and problem-solving in relation to a defined problem or research area.					
DESCRIPTION The students may be grouped into a batch of strength 3 or 4 to work under a project supervisor. The students learn to observe, empathize and ideate innovative solutions to real-world challenges. The student batches should identify a proposed work and will do literature review to develop prototype idea. Further at the end of the semester they will make a final presentation to exhibit the Identified literature gap.					
LIST OF ACTIVITIES: 1. Conduct a literature review on the identified research area.					

2. Identify the research gap from the literature review conducted. 3. Brain storming to be done to generate a large number of ideas within a specific timeframe. 4. Present the literature review conducted and identified research gap in relation to the research problem being investigated.
TOTAL: 30 PERIODS
OUTCOMES: Upon completion of the course, the students will be able to: CO1: Identify, collect, and interpret relevant scholarly literature in the chosen research area. CO2: Evaluate existing research to identify gaps and formulate precise research questions. CO3: Apply brainstorming techniques to generate innovative and diverse research ideas. CO4: Analyze reviewed literature and research gap to effectively present a research problem with clarity and rationale. CO5: Develop skills in using evidence to create and present an engaging and critical argument.

24CS311	APTITUDE AND CODING SKILLS – I (Common to All Branches)	L	T	P	C
		0	0	3	1.5
OBJECTIVES: The Course will enable learners to: - Develop vocabulary for effective communication and reading skills. - Build the logical reasoning and quantitative skills. - Develop error correction and debugging skills in programming. List of Exercises: 1. English – Phase I Vocabulary: Synonyms, Antonyms, Grammar: Subject-Verb Agreement, Tenses and Articles, Prepositions and Conjunctions, Speech and Voices, Comprehension: Inferential and Literal Comprehension, Contextual Vocabulary, Comprehension ordering 2. Logical Reasoning – Phase I Deductive Reasoning: Coding deductive logic, Directional sense, Blood relations, Objective Reasoning, Selection decision tables, Puzzles, Inductive reasoning: Coding pattern and Number series pattern recognition, Analogy and Classification pattern recognition, Abductive Reasoning: Logical word sequence, Data sufficiency 3. Quantitative Ability - Phase I Basic Mathematics: Divisibility, HCF and LCM, Numbers, decimal fractions and power, Applied Mathematics: Profit and Loss, Simple and Compound Interest, Time, Speed and Distance, Engineering Mathematics: Logarithms, Permutation and Combinations, Probability 4. Automata Fix – Phase I Logical, Compilation and Code reuse					
TOTAL: 45 PERIODS					
OUTCOMES: Upon completion of the course, the students will be able to: CO1: Develop vocabulary for effective communication skills.					

CO2: Build the logical reasoning enhance critical thinking.
CO3: Develop error correction and debugging skills in programming.
CO4: Apply programming skills to develop programs efficiently
CO5: Solve problems using quantitative skills
CO6: Develop effective reading and listening skills.

24MC301	INDIAN CONSTITUTION	L	T	P	C
		1	0	0	0
OBJECTIVES: The Course will enable learners to: • To have some knowledge about Indian Constitution. • To understand the concept of fundamental rights • To learn about Lok Sabha and Rajya Sabha • To have some knowledge about Legislative Assembly and Legislative Council • To learn about Local Self Government					
UNIT I	INTRODUCTION				
Meaning and Importance of Constitution, Preamble and Salient Features of the Constitution					
UNIT II	FUNDAMENTAL RIGHTS				
Fundamental Rights, Right to Equality, Right to Freedom, Right against exploitation, Right to freedom of religion, Cultural and Educational Rights, Right to Constitutional Remedies and Duties, Directive Principles of State Policy.					
UNIT III	LOK SABHA AND RAJYA SABHA				
Union Government - Lok Sabha and Rajya Sabha Composition, Powers, and functions: The President, The Prime Minister, and Supreme Court: Role Position and Powers/ functions.					
UNIT IV	LEGISLATIVE ASSEMBLY AND LEGISLATIVE COUNCIL				
State Government - Legislative Assembly and Legislative Council: Composition, Powers and functions: The Governor, Chief Minister and High Court: Role, Position and Powers/ functions					
UNIT V	LOCAL SELF GOVERNMENT				
Local self-Government, Panchayat Raj System in India; Election Commission; Public Service Commissions, Role, powers, and function					
OUTCOMES: At the end of this course, the students will be able to: CO1: Interpret the knowledge on Indian Constitution. CO2: Demonstrate the knowledge gained through fundamental rights concept. CO3: Relate the concept of Lok Sabha and Rajya Sabha. CO4: Illustrate the concept of Legislative Assembly and Legislative Council. CO5: Analyze the concept of Local Self Government.					
TEXT BOOK: 1. M V Pylee, An Introduction to The Constitution of India, Vikas Publishing House Pvt. Ltd., 5th Edition, 2007.					
REFERENCES: 1. Durga Das Basu, Introduction to the Constitution of India, 19th Edition Reprint 2009. 2. Sharma, Brij Kishore, –Introduction to the Constitution of India, Prentice Hall of India, 7th Edition, 2015.					

SEMESTER IV

24MA401	PROBABILITY AND STATISTICS (Common to CSE, IT)	L	T	P	C
		3	0	2	4
OBJECTIVES: The Course will enable learners to: • Provide the necessary basic concepts of random variables and introduce some standard distributions. • Comprehend the concepts of joint distributions, marginal and conditional distributions. • Test the hypothesis for small and large samples. • Introduce the concepts of analysis of variances. • Understand the concept of statistical quality control.					
UNIT I	ONE-DIMENSIONAL RANDOM VARIABLES				9+6
Basic probability, Independent events, Conditional probability (definition) - Random variable - Discrete and continuous random variables - Moments - Moment generating functions - Binomial, Poisson, Geometric, Uniform, Exponential and Normal distributions. List of Exercises/Experiments using R Programming: 1. Finding conditional probability. 2. Finding mean, variance and standard deviation.					
UNIT II	TWO-DIMENSIONAL RANDOM VARIABLES				9+6
Joint distributions - Marginal and conditional distributions - Covariance - Correlation and linear regression - Transformation of random variables. List of Exercises/Experiments using R Programming: 1. Finding marginal density functions for discrete random variables. 2. Calculating correlation and regression.					
UNIT III	TESTING OF HYPOTHESIS				9+6
Sampling distributions - Estimation of parameters - Statistical hypothesis - Large sample tests based on Normal distribution for single mean and difference of means - Tests based on t and F distributions for mean and variance - Chi-square test- Contingency table (test for independent) - Goodness of fit. List of Exercises/Experiments using R Programming: 1. Testing of hypothesis for given data using z - test. 2. Testing of hypothesis for given data using t - test.					
UNIT IV	DESIGN OF EXPERIMENTS				9+6
One way and Two-way classifications - Completely randomized design - Randomized block design - Latin square design. List of Exercises/Experiments using R Programming: 1. Perform one-way ANOVA test for the given data. 2. Perform two-way ANOVA test for the given data.					
UNIT V	STATISTICAL QUALITY CONTROL				9+6
Control charts for measurements ($\bar{X}$ and R charts) - Control charts for attributes (p, c and np charts) - Tolerance limits. List of Exercises/Experiments using R Programming: 1. Interpret the results for $\bar{X}$- Chart for variable data. 2. Interpret the results for R-Chart for variable data.					
TOTAL: 45+30=75 PERIODS					

OUTCOMES: Upon completion of the course, the students will be able to: CO1: Compute the statistical measures of standard distributions. CO2: Apply joint, marginal and conditional distributions to solve practical problems CO3: Determine the correlation and regression for two dimensional random variables CO4: Employ the concept of testing of hypothesis to solve real life problems. CO5: Apply the concept of analysis of variance for various experimental designs. CO6: Prepare the control charts for variables and attributes for analyzing the data.	
TEXT BOOKS: 1. R.A. Johnson, I. Miller and J. Freund, "Miller and Freund's Probability and Statistics for Engineers", Pearson Education, Asia, 9th Edition, 2023. 2. J.S. Milton and J.C. Arnold, "Introduction to Probability and Statistics", Tata McGraw Hill, 4th Edition, 2019.	
REFERENCES: 1. J. L. Devore, "Probability and Statistics for Engineering and the Sciences", Cengage Learning, New Delhi, 9th Edition, Reprint 2020. 2. S. M. Ross, "Introduction to Probability and Statistics for Engineers and Scientists", 6th Edition, Elsevier, 2020. 3. M. R. Spiegel, J. Schiller and R.A. Srinivasan, "Schaum's Outline of Theory and Problems of Probability and Statistics", Tata McGraw Hill, 4th Edition, 2013. 4. R. E. Walpole, R. H. Myers, S.L. Myers and K. Ye, "Probability and Statistics for Engineers and Scientists", Pearson Education, Asia, 9th Edition, Reprint 2021.	

24CS401	COMPUTER NETWORKS (Lab Integrated)	L	T	P	C
		3	0	2	4
OBJECTIVES: The Course will enable learners to: • Study the fundamental concepts of computer networks and physical layer. • Apply the knowledge of various protocols and techniques used in the data link layer. • Implement the services of network layer and network layer protocols. • Illustrate different protocols used in the transport layer. • Build applications using the application layer protocols.					
UNIT I	INTRODUCTION AND PHYSICAL LAYER				9+6
Data Communications – Network Types – Protocol Layering – Network Models (OSI, TCP/IP) Networking Devices: Hubs, Bridges, Switches – Performance Metrics – Transmission media - Guided media -Unguided media- Switching-Circuit Switching - Packet Switching.					
List of Exercise/Experiments: 1. Learn to use commands like tcpdump, netstat, ifconfig, nslookup and traceroute. 2. Capture ping and trace route PDUs using a network protocol analyzer and examine.					
UNIT II	DATA LINK LAYER				9+6

Introduction - Link-Layer Addressing- Error Detection and Correction - Wired LANs: Ethernet - Wireless LANs - Introduction - IEEE 802.11, Bluetooth List of Exercise/Experiments: 1. Simulation of an error correction code (like CRC). 2. Simulation of Flow control mechanisms.		
UNIT III	NETWORK LAYER	9+6
Network Layer Services - IPV4 Addresses - Forwarding of IP Packets - Network Layer Protocols: IP, ICMP - Unicast Routing Algorithms - Protocols - Multicasting Basics - IPV6 Addressing - IPV6 Protocol. List of Exercise/Experiments: 1. Simulation of DNS using UDP sockets. 2. Implementation of HTTP, Web Caching, FTP using socket programming.		
UNIT IV	TRANSPORT LAYER	9+6
Introduction - Transport Layer Protocols - Services - Port Numbers - User Datagram Protocol -Transmission Control Protocol - SCTP. List of Exercise/Experiments: 1. Develop a DNS client server to resolve the given host name or IP address. 2. Simulation of unicast routing protocols.		
UNIT V	APPLICATION LAYER	9+6
Application layer-WWW and HTTP - FTP - Email -Telnet -SSH - DNS - SNMP List of Exercise/Experiments: 1. Observing Packets across the network and Performance Analysis of various Routing protocols. 2. Simulation of Transport layer Protocols and analysis of congestion control techniques in the network.		
TOTAL: 45 +30 = 75 PERIODS		
OUTCOMES: Upon completion of the course, the students will be able to: CO1: Understand the fundamental concepts of computer networks. CO2: Apply the various routing protocols to solve real-world problems. CO3: Build simple applications to solve societal problems. CO4: Apply the simulation tools to implement various protocols used in the various layers. CO5: Analyze the various application layer protocols. CO6: Apply the mathematical knowledge to do performance analysis of various routing protocols.		
TEXT BOOK: 1. Data Communications and Networking, Behrouz A. Forouzan, McGraw Hill Education, 5th Ed., 2017.		
REFERENCES: 1. Computer Networking- A Top Down Approach, James F. Kurose, University of Massachusetts and Amherst Keith Ross, 8th Edition, 2021. 2. Computer Networks, Andrew S. Tanenbaum, Sixth Edition, Pearson, 2021. 3. Data Communications and Computer Networks, P.C. Gupta, Prentice-Hall of India, 2006. 4. Computer Networks: A Systems Approach, L. L. Peterson and B. S. Davie, Morgan Kaufmann, 3rd ed., 2003.		

LIST OF EQUIPMENTS:

C/Java, Ubuntu OS, NS2 simulation tool

24CS402	DESIGN AND ANALYSIS OF ALGORITHMS (Common to CSE, IT, CSBS and AIML)	L	T	P	C
		3	0	2	4
OBJECTIVES: The Course will enable learners to: • Critically analyse the efficiency of alternative algorithmic solutions for the same problem • Illustrate brute force and divide and conquer design techniques. • Explain dynamic programming for solving various problems. • Apply greedy technique and iterative improvement technique to solve optimization problems • Examine the limitations of algorithmic power and handling it in different problems.					
UNIT I	INTRODUCTION				9+6
Notion of an Algorithm - Fundamentals of Algorithmic Problem Solving -Fundamentals of the Analysis of Algorithmic Efficiency - Asymptotic Notations and their properties. Analysis Framework - Mathematical analysis for Recursive and Non-recursive algorithms List of Exercise/Experiments: 1. Perform the recursive algorithm analysis. 2. Perform the non-recursive algorithm analysis.					
UNIT II	BRUTE FORCE AND DIVIDE AND CONQUER				9+6
Brute Force - String Matching - Exhaustive Search - Knapsack Problem - Divide and Conquer Methodology – Binary Search – Merge sort – Quick sort - Multiplication of Large Integers – Closest-Pair and Convex Hull Problems - Transform and Conquer Method: Heap Sort List of Exercise/Experiments: 1. Write a program to search an element using binary search 2. Write a program to sort the elements using merge sort and find time complexity.					
UNIT III	DYNAMIC PROGRAMMING				9+6
Dynamic programming - Principle of optimality - Floyd_s algorithm - Multi stage graph - Optimal Binary Search Trees - Longest common subsequence - Matrix-chain multiplication - Travelling Salesperson Problem - Knapsack Problem and Memory functions. List of Exercise/Experiments: 1. Solve Floyd’s algorithm 2. Write a program to find the longest common subsequence					
UNIT IV	GREEDY TECHNIQUE AND ITERATIVE IMPROVEMENT				9+6
Greedy Technique – Prim_s algorithm and Kruskal’s Algorithm – Huffman Trees. The Maximum-Flow Problem - Maximum Matching in Bipartite Graphs - The Stable marriage Problem List of Exercise/Experiments: 1. Write a program to find minimum spanning tree using Prim’s algorithm 2. Implement Kruskal’s algorithm to find minimum spanning tree					
UNIT V	BACKTRACKING AND BRANCH AND BOUND				9+6
P, NP NP- Complete and NP Hard Problems. Backtracking - N-Queen problem - Subset Sum Problem. Branch and Bound - LIFO Search and FIFO search - Assignment problem - Knapsack Problem - Approximation Algorithms for NP-Hard Problems - Travelling Salesman problem List of Exercise/Experiments: 3. Write a program to implement sum of subset problem. 4. Solve knapsack problem using branch and bound technique					

TOTAL: 45+30=75 PERIODS	
OUTCOMES: Upon completion of the course, the students will be able to: CO1: Understand the different algorithm design paradigms. CO2: Design algorithms for real world problems using algorithmic design techniques. CO3: Analyse the efficiency of simple recursive and non-recursive algorithms. CO4: Analyse the algorithm's worst, best and average case behaviour in terms of time and space. CO5: Understand the approximation algorithms for solving NP Hard problems CO6: Solve the problems by selecting suitable algorithmic design techniques.	
TEXT BOOKS: 1. Anany Levitin, Introduction to the Design and Analysis of Algorithms, Third Edition, Pearson Education, 2012. 2. Ellis Horowitz, Sartaj Sahni and Sanguthevar Rajasekaran, Computer Algorithms/ C++, Second Edition, Universities Press, 2019.	
REFERENCES: 1. Thomas H.Cormen, Charles E.Leiserson, Ronald L. Rivest and Clifford Stein, Introduction to Algorithms, Third Edition, PHI Learning Private Limited, 2012. 2. S. Sridhar, Design and Analysis of Algorithms, Oxford university press, 2014. 3. http://nptel.ac.in/	
LIST OF EQUIPMENTS: 1. Standalone PC with C/C++/Java	

24IT402	WEB DEVELOPMENT FRAMEWORKS (Lab Integrated)	L	T	P	C
		3			
COURSE OBJECTIVES: The Course will enable the learners: • To understand web semantics and related tools and framework • To get hands on latest JS based web frameworks • To develop a scalable and responsive web application • To develop an industry ready application web enterprise feature					
UNIT I	ADVANCED TYPESCRIPT				9+9
Introduction to HTML5 and CSS3, Media Queries, JS, DOM, BootStrap, Variables, Loops, Operators, Scope, Hoisting, Arrays, Spread, REST, DeStructuring Introduction, Advantage of Using TS over JS, and where to Use and not to use TS - Understanding the Compiler (Transpiler), and its options, Scope of TS - Variable Scopes, Static Data Types - String, Number, Interface, Date - Union, Tuple, Undefined Data Types, Unknown vs any vs Never - Object Oriented , Arrow Funcions - Types, KeyOf, TypeOf, InstanceOf ,Narrowing, Conditional Types - Generics, Enum ,Required / Partial / Optional - Arrays, Modules ,Async Processing w Call backs ,Type Inference, Type Compatibility, Utility Type -					

Unit Testing, TSLint

List of Exercise/Experiments

- 1) Create a TS Object for Bank Account (with attributes like à customer name, account type, balance, data of creation, bank name, branch name, pan card number). Using JS Object keyword, try to perform following activities
 - List down all the entries of the bank object
 - Check the existence of a key
 - If key found, get the value for the key
- 2) Spread Operator
 - Merge Customer and Account Arrays
 - Update the Customer Object with the new values
 - Develop a function that takes an Spread Argument and calculates total balance.

UNIT II	INTRODUCTION TO REACTJS	9+9
----------------	--------------------------------	------------

Introduction to React - ES6 Features, What is React?, React Features - Setting up React Development Environment:- Node.js and npm installation, Create React App, Project structure - JSX (JavaScript XML):- What is JSX?,JSX Syntax and Rules,JSX Expressions - Components in React:- Functional Components, Class Components, Props and PropTypes - State and Lifecycle :- State and setState, Lifecycle Methods, Mounting, Updating, and Unmounting, Handling Events in React - Event Handling in React :- Synthetic Events, Event Binding, - Conditional Rendering:-If-else Statements, Ternary Operator, Logical && Operator Lists and Keys:- Rendering Lists, Keys and Reconciliation, Extracting Components.

List of Exercise/Experiments

- 1) A leading bank from APAC wants to modernize their banking services and decided to build a online multi channel mobile ecommerce platform. As part of the drive, starting building following feature set in a staggered model emplying ReactJS as front end library and associated libs from React eco system. Feature to be implemented are
 - a. User Login Page
 - b. Account Summary
 - c. Funds Transfer (within bank and outside bank)
 - d. Recurring and Fixed deposits
 - e. Letter of Credit
 - f. Salary or 3rd Party Payment
- 2) Unit 2 Scope - Project Setup, Web App Layout Completion using BootStrap or Tailwind, Login Page Implementation, Landing Page Implementation, Authentication and Authorization Implementation.

UNIT III	REACTJS COMPONENTS	9+9
-----------------	---------------------------	------------

Forms and Controlled Components :- Form Handling in React, Controlled Components, Uncontrolled Components - Basic Hooks :- useState, useRef, useEffect, - Routing in React:- Introduction to React Router, Route, Link, and Switch Components, Route Parameters, useNavigate, useParams, - REST API – Axios GET/PUT/Delete/Remove, Interceptor, Headers, Authorization Token, Promise and Observables (via rxjs)

List of Exercise/Experiments

- 1) Extend the Project developed in previous section with newly learnt concepts

a. Unit 3 Scope - Forms and Validation (React Form Validation), Integration of Backend APIs via Axios, API Security Implementation, Routes and Navigation with Private Routes, Usage of useEffect, useContext hooks		
UNIT IV	REACT PRO TOOLKIT: ERROR MANAGEMENT, ABSTRACTIONS & DATA HANDLING	9+9
Error Handling :- Error Boundaries, componentDidCatch, Error Handling Strategies Higher-Order Components (HOCs) :- What are HOCs?, Creating and Using HOCs, HOCs vs Render Props, Code Splitting and Lazy Loading, Server-Side Rendering - Data Fetching with React Query :- React Query, Introduction to React Query, Query Keys and Query Functions, Query Invalidation and Refetching. <u>List of Exercise/Experiments</u> 1) Extend the Project developed in previous section with newly learnt concepts a. Unit 4 Scope - Completion of Remaining Modules, Error Handling, HOC and AUX implementation, Lazy loaded components for improved performance Extend the Project developed in previous section with newly learnt concepts		
UNIT V	REACT UNDER THE HOOD: TESTING, CONTEXT API, AND REDUX	9+9
Testing React Components :- Introduction to Testing, Jest Framework, React Testing Library Context API :- Creating Context, Providing and Consuming Context, useContext Hook Redux Overview :- What is Redux?, Redux Principles, Single Source of Truth - Redux Actions and Reducers :- Redux Actions, Redux Reducers, Combining Reducers. <u>List of Exercise/Experiments</u> 1) Extend the Project developed in previous section with newly learnt concepts a. Unit 5 Scope – Unit Testing using JEST, Redux implementation for state management. <u>Business Use Case Implementations</u> 1) Student Management System 2) Retail Bank System 3) eCommerce System 4) Student LMS Management System		
TOTAL: 45+45=90 PERIODS		
<u>COURSE OUTCOMES:</u> After completing the course, students will have the ability to CO1. Understand and apply modern web technologies including HTML5, CSS3, JavaScript, and advanced TypeScript concepts to build dynamic web components. CO2. Develop responsive and modular front-end applications using ReactJS, including component creation, state management, and routing. CO3. Implement advanced React features like hooks (useState, useEffect, useRef), React Router, and REST API integration using Axios for dynamic content handling.		

CO4. Utilize higher-order components (HOCs), lazy loading, and server-side rendering to optimize and abstract React applications.

CO5. Perform unit testing using Jest and RTL, and manage global application state efficiently using Context API and Redux.

CO6. Design and deliver scalable and real-world enterprise web applications with complete user interface flow, security, and error handling.

TEXTBOOKS:

- 1) David Flanagan, Javascript The Definitive Guide, Paperback, 7th Edition, 2020.
- 2) David Choi ,Full-Stack React, TypeScript, and Node: Build cloud-ready web applications using React 17 with Hooks and GraphQL Paperback – Import, 18 December 2020.
- 3) Mehul Mohan, Advanced Web Development with React Paperback - 1 January 2020.

E-RESOURCES:

1. Parental Website - <https://reactjs.org/>
2. The Road to Learn React: Your journey to master plain yet pragmatic React.js by Robin Wieruch
3. Learning React: Functional Web Development with React and Redux by Alex Banks and Eve Porcello
4. Learning React by KirupaChinnathambi
5. "React Up & Running" by StoyanStefanov
6. <https://www.edureka.co/reactjs-redux-certification-training>
7. CodePen
8. CodeSandbox (Preferred)
9. Stackblitz

LIST OF EQUIPMENTS:

- NodeJS (v22.11.2)
- Github as code repository
- Visual studio code as IDE
- RTL as unit testing framework
- Responsive design w bootstrap
- ReactJS installation (v17)
- Chrome / FlreFox Browsers (latest)
- Responsive using Media Queries & Bootstrap Material & Antdesign
- Design based Apps

24GE411	PRODUCT DEVELOPMENT LAB - 2 (Functional Design) (Common to All Branches)		T	P	C
			0	2	1
OBJECTIVES:					
The Course will enable learners to:					
- To accurately capture the expectations, needs, and constraints of stakeholders to ensure that the system being developed aligns with user goals and real-world usage scenarios. - To visually represent the system's structure and functional relationships, enabling clearer communication, easier analysis, and better planning of system architecture and workflow. - To create a precise and comprehensive reference document that guides development, ensures clarity, and serves as a basis for verification, validation, and future system enhancements.					
DESCRIPTION					
The students may be grouped into a batch of strength 3 or 4 to work under a project supervisor. Further at the end of the semester they will make a final presentation to exhibit the functional design and the process to develop a product.					
LIST OF ACTIVITIES:					
- Understand and document functional requirements based on stakeholder needs. - Map out how functions interact and relate to each other using functional block diagrams or flowcharts. - Write detailed specification describing each function's role, behavior, constraints, and expected outcomes. - Check if defined functions meet the intended requirements and can be validated/tested. - Present the design concepts carried out for the identified research gap in relation to the research problem being investigated.					
TOTAL: 30 PERIODS					
OUTCOMES:					
Upon completion of the course, the students will be able to:					
CO1: Interpret stakeholder needs and document comprehensive functional requirements for the proposed system.					
CO2: Develop functional block diagrams or flowcharts to represent system interactions and functional relationships.					
CO3: Analyze functional specifications that define roles, behaviors, constraints, and performance expectations for each function.					
CO4: Evaluate the defined functional model through verification and validation techniques to ensure alignment with original requirements.					
CO5: Analyze and present functional design solutions aligned with the identified research problem and gap.					
LIST OF EQUIPMENTS FOR A BATCH OF 30 STUDENTS					
S. No.	Description of Equipment				Quantity
1	CNC Router				1 No
2	3D Printer				1 No
3	3D Scanner				1 No
4	Laser cutting Machine				1 No

5	Multimeter	
6	Solder Stations	
7	Desoldering Machine	
8	PCB Milling Machine	
9	Variable Power Supply	
10	Modelling Software(open source)	

24CS411	APTITUDE AND CODING SKILLS – II (Common to All Branches)	L	T	P	C
		0	0	3	1.5
OBJECTIVES: The Course will enable learners to: ● Develop advanced vocabulary for effective communication and reading skills. ● Build an enhanced level of logical reasoning and quantitative skills. ● To develop error correction and debugging skills in programming. ● To apply data structures and algorithms in problem solving. List of Exercises: 1. English – Phase II Vocabulary: Synonyms, Antonyms, Grammar: Subject-Verb Agreement, Tenses and Articles, Prepositions and Conjunctions, Speech and Voices, Comprehension: Inferential and Literal Comprehension, Contextual Vocabulary, Comprehension ordering 2. Logical Reasoning – Phase II Deductive Reasoning: Coding deductive logic, Directional sense, Blood relations, Objective Reasoning, Selection decision tables, Puzzles, Inductive reasoning: Coding pattern and Number series pattern recognition, Analogy and Classification pattern recognition, Abductive Reasoning: Logical word sequence, Data sufficiency 3. Quantitative Ability - Phase II Basic Mathematics: Divisibility, HCF and LCM, Numbers, decimal fractions and power, Applied Mathematics: Profit and Loss, Simple and Compound Interest, Time, Speed and Distance, Engineering Mathematics: Logarithms, Permutation and Combinations, Probability 4. Automata Fix – Phase II Logical, Compilation and Code reuse 5. Automata - Phase II Data Structure Concepts: Array and Matrices, Linked list, String processing and manipulation, Stack/Queue, Sorting and Searching Advanced Design and Analysis Techniques: Greedy Algorithms, Minimum Spanning Trees, String Matching, Divide and Conquer, Computational Geometry					
TOTAL: 45 PERIODS					
OUTCOMES: Upon completion of the course, the students will be able to: CO1: Develop advanced vocabulary for effective communication skills. CO2: Build an enhanced level of logical reasoning and quantitative skills.					

CO3: Develop error correction and debugging skills in programming.
CO4: Apply data structures and algorithms in problem solving.
CO5: Develop advanced vocabulary for effective reading skills
CO6: Apply advanced algorithm design techniques to develop programs

SEMESTER - V

24CS501	THEORY OF COMPUTATION	L	T	P	C
		3	0	0	3
OBJECTIVES: The Course will enable learners to: • Learn about automata construction • Learn equivalence of automata and regular expressions • Design a context free grammar for any given language • Construct Pushdown Automata for Context Free Grammar. • Understand the concepts of Turing Machine and undecidable problems					
UNIT I	FINITE AUTOMATA	9			
Finite Automata (FA) - Deterministic Finite Automata (DFA) - Non-deterministic Finite Automata (NFA) - Equivalence between NFA and DFA - Finite Automata with Epsilon transitions - Equivalence of NFA and DFA- Equivalence of NFAs with and without ϵ -moves - Conversion of NFA into DFA					
UNIT II	REGULAR EXPRESSIONS and LANGUAGES	9			
Definition - Operators of Regular Expression - Algebraic Laws for Regular Expressions- Equivalence of FA and Regular Expressions - Minimization of Finite Automata - Pumping Lemma for Regular Languages - Closure Properties of Regular Languages.					
UNIT III	CONTEXT FREE GRAMMAR AND NORMAL FORMS	9			
Chomsky's Hierarchy of Languages; Context-Free Grammar and Languages: Context-Free Gram- mar (CFG) - Parse trees - Ambiguity in grammars and languages; Normal Forms for Context Free Grammars: Elimination of Useless Symbols, epsilon and unit productions - Chomsky Normal Form (CNF) - Greibach Normal Form (GNF).					
UNIT IV	PUSHDOWN AUTOMATA	9			
Pushdown Automata (PDA): Definition of the Pushdown automaton - The languages of a PDA - Equivalence of PDAs and CFGs - Deterministic Pushdown automata; Pumping Lemma for CFL - Closure Properties of CFL.					
UNIT V	TURING MACHINE AND UNDECIDABILITY	9			
Turing Machines and its types - Programming Techniques for Turing Machine - Turing machines as a computer of integer functions - Non-Recursive Enumerable Language - Undecidable Problem with Regular Expressions - Undecidable Problems about Turing Machine - Universal Turing Machine- Post's Correspondence Problem and modified PCP.					
TOTAL: 45 PERIODS					

OUTCOMES:

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

CO1: Understand the fundamentals of finite automata and their applications in solving problems.

CO2: Analyze the behavior of formal languages and regular expressions to solve computational problems.

CO3: Understand the concept of context-free grammars (CFG) and their applications to construct language.

CO4: Construct Pushdown Automata for Context Free Grammar.

CO5: Evaluate the computational power and limitations of Turing machines and their relevance to modern computing systems.

CO6: Identify undecidable problems and analyze their implications for engineering practice and theoretical computer science.

TEXT BOOK:

1. Introduction to Automata Theory, Languages, and Computation, 3rd Edition, John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman, Pearson Education, 2014.

REFERENCES:

1. Introduction to Languages and The Theory of Computation, 4th Edition, John C Martin, TMH, 2010.
2. Introduction to Computer Theory, Daniel I.A. Cohen, John Wiley.
3. Introduction to the Theory of Computation, Michael Sipser, 3rd edition, Cengage Learning.
4. Introduction to Formal languages Automata Theory and Computation Kamala Krithivasan, Rama R, Pearson.

24CS502	DISTRIBUTED AND CLOUD COMPUTING	L	T	P	C
		2	0	2	3
OBJECTIVES: ✓ To articulate the concepts and models underlying distributed computing ✓ To maintain consistency and perform efficient coordination in distributed systems through the use of logical clocks, global states, and snapshot recording algorithms. ✓ To learn different distributed mutual exclusion algorithms. ✓ To develop the ability to understand the cloud infrastructure and virtualization that help in the development of cloud. ✓ To explain the high-level automation and orchestration systems that manage the virtualized infrastructure.					
UNIT I	INTRODUCTION	6 + 6			
Definition - Relation to computer system components - Message-passing systems versus shared memory systems - Primitives for distributed communication - Synchronous versus asynchronous executions. A model of distributed computations: A distributed program - A model of distributed executions - Models of communication networks - Global state of a distributed system.					
List of Exercise/Experiments: 1. Implement a simple distributed program that communicates between two nodes using Java's RMI (Remote Method Invocation) API.					

2. Develop a model of a distributed program using Java's concurrency and synchronization primitives.		
UNIT II	LOGICAL TIME, GLOBAL STATE, AND SNAPSHOT ALGORITHMS	6 + 6
Logical time - Scalar Time - Vector Time. Global state and snapshot recording algorithms: System model - Snapshot algorithms for FIFO channels and non-FIFO channels. List of Exercise/Experiments: 1. Develop a program in Java that implements vector clocks to synchronize the order of events between nodes in a distributed system. 2. Implement a snapshot algorithm for recording the global state of the distributed system using vector clocks, for both FIFO and non-FIFO channels. Test the algorithm by recording snapshots at various points in the system's execution and analyzing the resulting global state.		
UNIT III	DISTRIBUTED MUTUAL EXCLUSION ALGORITHMS	6 + 6
Introduction - Lamport's algorithm - Ricart-Agrawala algorithm - Quorum-based mutual exclusion algorithms - Maekawa's algorithm - Suzuki-Kasami's broadcast algorithm. List of Exercise/Experiments: 1. Implement Lamport's algorithm for mutual exclusion in a distributed system using Java's RMI API. 2. Develop a program in Java that implements Maekawa's algorithm for mutual exclusion in a distributed system. 3. Implement Suzuki-Kasami's broadcast algorithm in Java to achieve reliable message delivery in a distributed system.		
UNIT IV	CLOUD INFRASTRUCTURE AND VIRTUALIZATION	6 + 6
Data Center Infrastructure and Equipment - Virtual Machines - Containers - Virtual Networks - Virtual Storage. List of Exercise/Experiments: 1. Set up a virtualized data center using a hypervisor like VMware or VirtualBox and create multiple virtual machines (VMs) on it. Configure the VMs with different operating systems, resources, and network configurations, and test their connectivity and performance. 2. Deploy a containerized application on a virtual machine using Docker or Kubernetes.		
UNIT V	AUTOMATION AND ORCHESTRATION	6 + 6
Automation - Orchestration: Automated Replication and Parallelism - The MapReduce Paradigm: The MapReduce Programming Paradigm - Splitting Input - Parallelism and Data size - Data access and Data Transmission - Apache Hadoop - Parts of Hadoop - HDFS Components - Block Replication and Fault Tolerance - HDFS and MapReduce - Microservices. List of Exercise/Experiments: 1. Set up and configure a single-node Hadoop cluster. 2. Run the word count program in Hadoop. 3. Deploy a microservices architecture using a container orchestration tool like Kubernetes or Docker Swarm.		
TOTAL: 30(L) + 30(P) = 60 PERIODS		
OUTCOMES: Upon completion of the course, the students will be able to: CO1: Articulate the main concepts and models underlying distributed computing. CO2: Learn how to maintain consistency and perform efficient coordination in distributed systems through the use of logical clocks, global states, and snapshot recording algorithms. CO3: Learn different distributed mutual exclusion algorithms		

CO4: Develop the ability to understand the cloud infrastructure and virtualization that help in the development of cloud.	
CO5: Explain the high-level automation and orchestration systems that manage the virtualized infrastructure.	
TEXT BOOKS:	
1. Ajay D. Kshemkalyani, Mukesh Singhal, "Distributed Computing: Principles, Algorithms, and Systems", Cambridge University Press, 2011. (Unit 1, 2, 3) 2. Douglass E. Comer, "The Cloud Computing Book: The future of computing explained", CRC Press, 2023. (Unit 4, 5)	
REFERENCES:	
1. Arshdeep Bahga, Vijay Madisetti, "Cloud Computing: A Hands-on Approach", Universities Press Private Limited, 2014. 2. Rajkumar Buyya, Christian Vecchiola, S. ThamaraiSelvi, "Mastering Cloud Computing", Tata Mcgraw Hill, 2024. 3. Kai Hwang, Geoffrey C. Fox, Jack G. Dongarra, "Distributed and Cloud Computing, From Parallel Processing to the Internet of Things", Morgan Kaufmann Publishers, 2017. 4. Hagit Attiya, Jennifer Welch, "Distributed Computing: Fundamentals, Simulations and Advanced Topics", John Wiley & Sons, Inc., 2004.	
LIST OF EQUIPMENTS:	
Java, VMWare/VirtualBox, Hadoop, AWS/GCP/Azure/Any cloud platform, Eclipse/Equivalent IDE.	

24CS503	MACHINE LEARNING	L	T	P	C
		3	0	2	4
OBJECTIVES: • To discuss the basics of Machine Learning and model evaluation. • To study dimensionality reduction techniques. • To understand the various classification algorithms. • To elaborate on unsupervised learning techniques. • To discuss the basics of neural networks and various types of learning.					
UNIT I	INTRODUCTION	9+6			
Machine Learning - Types - Applications - Preparing to Model - Activities - Data - Exploring structure of Data - Data Quality and Remediation - Data Pre-processing - Modelling and Evaluation: Selecting a Model -Training a Model - Model representation and Interpretability - Evaluating Performance of a Model - Improving Performance.					
Lab Programs: 1. Implementation of Candidate Elimination algorithm 2. Implementation of ML model evaluation techniques (R-Squared/Adjusted R-Squared/Mean Absolute Error/Mean Squared Error) 3. Implementation of ML model evaluation techniques (Confusion Matrix/F1 Score/AUC-ROC Curve)					
UNIT II	FEATURE ENGINEERING AND DIMENSIONALITY REDUCTION	9+6			
Feature Engineering - Feature Transformation - Feature Subset Selection - Principle Component Analysis - Feature Embedding - Factor Analysis - Singular value decomposition and Matrix Factorization - Multidimensional scaling - Linear Discriminant Analysis - Canonical Correlation Analysis - Isomap - Locally linear Embedding - Laplacian Eigenmaps.					

Lab Programs:		
1. Write python code to identify feature co-relations (PCA) 2. Interpret Canonical Covariates with Heatmap 3. Feature Engineering is the way of extracting features from data and transforming them into formats that are suitable for Machine Learning algorithms. Implement python code for Feature Selection/ Feature Transformation/ Feature Extraction. 4. Mini Project - Feature Subset Selection		
UNIT III	SUPERVISED LEARNING	9+6
Linear Regression -Relation between two variables - Steps - Evaluation - Logistic Regression - Decision Tree - Algorithms - Construction - Classification using Decision Tree - Issues - Rule - based Classification - Pruning the Rule Set - Support Vector Machines - Linear SVM - Optimal Hyperplane - Radial Basis Functions - Naïve Bayes Classifier - Bayesian Belief Networks.		
Lab Programs:		
1. Implement the non-parametric Locally Weighted Regression algorithm in order to fit data points. Select the appropriate data set for your experiment and draw graphs. 2. Implement and demonstrate the working of the decision tree-based ID3 algorithm 3. Build a Simple Support Vector Machines using a data set		
UNIT IV	UNSUPERVISED LEARNING	9+6
Clustering - Types - Applications - Partitioning Methods - K-means Algorithm - K-Medoids - Hierarchical methods - Density based methods DBSCAN - Finding patterns using Association Rules - Hidden Markov Model.		
Lab Programs:		
1. Implement a k-Nearest Neighbour algorithm to classify the iris data set. Print both correct and wrong predictions 2. Implement market basket analysis using association rules 3. Mini Project using Clustering analysis.		
UNIT V	NEURAL NETWORKS AND TYPES OF LEARNING	9+6
Biological Neuron - Artificial Neuron - Types of Activation function - Implementations of ANN - Architectures of Neural Networks - Learning Process in ANN - Back propagation - Deep Learning - Representation Learning - Active Learning - Instance based Learning - Association Rule Learning - Ensemble Learning Algorithm - Regularization Algorithm-Reinforcement Learning - Elements- Model-based- Temporal Difference Learning.		
Lab Programs:		
1. Build an ANN by implementing the Single-layer Perceptron. Test it using appropriate data sets. 2. Implement Multi-layer Perceptron and test the same using appropriate data sets. 3. Build a RBF Network to calculate the fitness function with five neurons. 4. Mini Project - Face recognition,		
TOTAL: 45(L) + 30(P) = 75 PERIODS		
OUTCOMES:		
At the end of this course, the students will be able to:		
CO1: Explain the basics of Machine Learning and model evaluation.		
CO2: Study dimensionality reduction techniques.		
CO3: Understand and implement various classification algorithms.		
CO4: Understand and implement various unsupervised learning techniques.		
CO5: Build Neural Networks and understand the different types of learning.		
TEXT BOOKS:		
1. Saikat Dutt, Subramanian Chandramouli, Amit Kumar Das, "Machine Learning", Pearson,2019. (Unit 1 - chap 1,2,3/ Unit 2 - Chap 4 / Unit 4 - 9 / Unit 5 - Chap 10, 11) 2. Ethem Alpaydin, "Introduction to Machine Learning, Adaptive Computation and Machine Learning Series", Third Edition, MIT Press, 2014. (Unit 2 - Chap 6 / Unit 4 - chap 8.2.3/ Unit 5 - Chap 18)		
REFERENCES:		

1. Anuradha Srinivasaraghavan,Vincy Joseph, “Machine Learning”, First Edition, Wiley, 2019.(Unit 3 - Chap 7,8,9,10,11 / Unit 4 - 13, 11.4, 11.5,12)
2. Peter Harrington, “Machine Learning in Action”, Manning Publications, 2012.
3. Stephen Marsland, “Machine Learning - An Algorithmic Perspective”, Second Edition,
4. Chapman and Hall/CRC Machine Learning and Pattern Recognition Series, 2014.
5. Tom M Mitchell, “Machine Learning”, First Edition, McGraw Hill Education, 2013.
6. Christoph Molnar, “Interpretable Machine Learning - A Guide for Making Black Box Models Explainable”, Creative Commons License, 2020.
7. NPTEL Courses:
 - a. Introduction to Machine Learning -
https://onlinecourses.nptel.ac.in/noc23_cs18/preview

LIST OF EQUIPMENTS:

Systems with Anaconda, Jupyter Notebook, Python, Pytorch, scikit-learn, Tensorflow, Colab

24CS511	ADVANCED APTITUDE AND CODING SKILLS - I	L	T	P	C
		0	0	3	1.5

OBJECTIVES:

- To develop vocabulary for effective communication and reading skills.
- To build the logical reasoning and quantitative skills.
- To develop error correction and debugging skills in programming.

LIST OF EXERCISES:

1. English – Phase I Advanced

Vocabulary: Synonyms, Antonyms, Grammar: Subject-Verb Agreement, Tenses and Articles, Prepositions and Conjunctions, Speech and Voices, Comprehension: Inferential and Literal Comprehension, Contextual Vocabulary, Comprehension ordering

2. Logical Reasoning – Phase I Advanced

Deductive Reasoning: Coding deductive logic, Directional sense, Blood relations, Objective Reasoning, Selection decision tables, Puzzles, Inductive reasoning: Coding pattern and Number series pattern recognition, Analogy and Classification pattern recognition, Abductive Reasoning: Logical word sequence, Data sufficiency

3. Quantitative Ability - Phase I Advanced

Basic Mathematics: Divisibility, HCF and LCM, Numbers, decimal fractions and power, Applied Mathematics: Profit and Loss, Simple and Compound Interest, Time, Speed and Distance, Engineering Mathematics: Logarithms, Permutation and Combinations, Probability

4. Automata Fix – Phase I

Logical, Compilation and Code reuse

TOTAL: 45 PERIODS

OUTCOMES:

At the end of this course, the students will be able to:

CO1: Develop vocabulary for effective communication and reading skills.

CO2: Build the logical reasoning and quantitative skills.

CO3: Develop error correction and debugging skills in programming.

24GE511	PRODUCT DEVELOPMENT LAB – 3 (Simulation and Analysis Phase) (Common to All Branches)	L	T	P	C
		0	0	2	1
OBJECTIVES: The Course will enable learners to: • Use the innovative design methodology to articulate the product concepts. • Conduct basic tests to extract the qualitative and quantitative performance factors. • Perform simulations, experiments, or case studies to test the reliability and relevance of the proposed design.					
DESCRIPTION The students may be grouped into a batch of strength 3 or 4 to work under a project supervisor. Further at the end of the semester, they will make a final presentation to exhibit the methodology and implementation of design process.					
LIST OF ACTIVITIES: 1. Prepare the design concepts for the identified prototype. 2. Perform simulations, experiments, or case studies to test the reliability and relevance of the proposed design. 3. Present the methodology to be followed and the steps carried out in the implementation of design process.					
TOTAL: 30 PERIODS					
OUTCOMES: Upon completion of the course, the students will be able to: CO1: Analyze critically the design process and outcomes to identify areas for improvement and future research directions. CO2: Develop preliminary prototypes or models to demonstrate design concepts and evaluate their effectiveness. CO3: Conduct iterative testing and evaluation of designs, incorporating feedback to improve outcomes. CO4: Apply industry-standard tools, technologies, and frameworks to execute the design process effectively. CO5: Explain design concepts through well-structured oral presentations, reports, and visual aids tailored to diverse audiences.					
LIST OF EQUIPMENT					
S. No.	Description of Equipment				Quantity
1	CNC Router				1 No
2	3D Printer				1 No
3	3D Scanner				1 No
4	Laser cutting Machine				1 No
5	Multimeter				5 Nos
6	Solder Stations				5 Sets

7	Desoldering Machine	1 No
8	PCB Milling Machine	1 No
9	Variable Power Supply	1 No
10	Electronic Components like Resistors, Transistors, Diode, Inductor, Capacitor, etc.	5 Sets
11	Open source simulation software	1

SEMESTER - VI

24CS601	COMPILER DESIGN (Lab Integrated)	L	T	P	C
		3	0	2	4

OBJECTIVES:
The Course will enable learners to:

- Study the different phases of compiler
- Understand the techniques for tokenization and parsing
- Understand the conversion of source program into an intermediate representation
- Learn the different techniques used for run time environment and code generation
- Analyze various code optimization techniques

UNIT I	INTRODUCTION TO COMPILERS AND LEXICAL ANALYZERS	9 + 6
---------------	--	--------------

Introduction - Translators - Compilation and Interpretation - Language processors - The Phases of Compiler - Lexical Analysis - Role of Lexical Analyzer - Input Buffering - Specification of Tokens - Recognition of Tokens - Finite Automata - Regular Expressions to Automata NFA, DFA - Minimizing DFA - Language for Specifying Lexical Analyzers - Lex tool.

List of Exercise/Experiments:

1. Develop a lexical analyzer to recognize a few patterns in C. (Ex. identifiers, constants, comments, operators etc.). Create a symbol table, while recognizing identifiers.
2. Implement a Lexical Analyzer using Lex Tool.

UNIT II	SYNTAX ANALYSIS	9 + 6
----------------	------------------------	--------------

Role of the Parser - Context-free grammars - Derivation Trees - Ambiguity in Grammars and Languages - Writing a grammar - Types of parsing - Top-Down Parsing - Predictive parser - LL(1) Parser - Bottom-Up Parsing - Shift Reduce Parser - LR Parsers - SLR, CLR, LALR Parser - Error Handling and Recovery in Syntax Analyzer - YACC tool - Design of a syntax Analyzer for a Sample Language.

List of Exercise/Experiments:

1. Generate YACC specification for a few syntactic categories.
 - a. Program to recognize a valid arithmetic expression that uses operator +, -, * and /.
 - b. Program to recognize a valid variable which starts with a letter followed by any number of letters or digits.
2. Design Predictive Parser for the given language

UNIT III	SYNTAX DIRECTED TRANSLATION & INTERMEDIATE CODE GENERATION	9 + 6
-----------------	---	--------------

Syntax Directed Definitions - Evaluation Orders for Syntax Directed Definitions - Application of Syntax Directed Translation - Intermediate Languages - Syntax Tree - Three Address Code - Implementation of Three address code - Types and Declarations, Translation of Expressions, Type Checking, Back patching.

List of Exercise/Experiments:

1. Implement an Arithmetic Calculator using LEX and YACC
2. Generate three address code for a simple program using LEX and YACC.

UNIT IV	RUN-TIME ENVIRONMENT AND CODE GENERATION	9 + 6
----------------	---	--------------

Runtime Environments - source language issues - Storage organization - Storage Allocation Strategies: Static, Stack and Heap allocation - Parameter Passing - Symbol Tables - Dynamic Storage Allocation - Issues in the Design of a code generator - Design of a simple Code Generator - Register allocation and assignment.

List of Exercise/Experiments:

1. Implement back-end of the compiler for which the three address code is given as input and the 8086 assembly language code is produced as output.

UNIT V	CODE OPTIMIZATION	9 + 6
---------------	--------------------------	--------------

Principle sources of optimization - Peep hole Optimization - DAG construction - Basic blocks and flow graph - Optimization in Basic blocks - Data flow analysis - Recent trends in Compiler Design

List of Exercise/Experiments:

1. Implement simple code optimization techniques (Constant folding, Strength reduction and Algebraic transformation)

TOTAL: 45(L) +30(P) = 75 PERIODS

OUTCOMES:

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

CO1: Understand the structure and role of lexical analyzers in the compilation process. **CO2:** Design and implement different parsers and the generators like YACC for automating parser construction.

CO3: Understand syntax-directed definitions and their applications in intermediate code generation.

CO4: Analyze the different techniques used for efficient assembly code generation.

CO5: Design efficient code by implementing different code optimization techniques

CO6: Apply DAG technique to optimize the basic blocks used for data flow analysis

TEXT BOOK:

1. Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman, "Compilers: Principles, Techniques and Tools", Second Edition, Pearson Education Limited, 2014.

REFERENCES:

1. Randy Allen, Ken Kennedy, "Optimizing Compilers for Modern Architectures: A Dependence-based Approach", Morgan Kaufmann Publishers, 2002.
2. Steven S. Muchnick, "Advanced Compiler Design and Implementation", Morgan Kaufmann Publishers - Elsevier Science, India, Indian Reprint, 2003.
3. Keith D Cooper and Linda Torczon, "Engineering a Compiler", Morgan Kaufmann Publishers, Elsevier Science, 2004.
4. V. Raghavan, "Principles of Compiler Design", Tata McGraw Hill Education Publishers, 2010.
5. Allen I. Holub, "Compiler Design in C", Prentice-Hall Software Series, 1993.

LIST OF EQUIPMENTS:

C/C++

24CS602	CRYPTOGRAPHY AND CYBER SECURITY (Lab Integrated)	L	T	P	C
		3	0	2	4
OBJECTIVES: - To understand the fundamentals of network security and security architecture. - To learn the different symmetric key cryptographic algorithms. - To study the various asymmetric key cryptographic algorithms and techniques. - To know the importance of message authentication and integrity. - To learn the various cyber-crimes and cyber security.					
UNIT I	INTRODUCTION TO SECURITY	9+6			
Computer Security Concepts - The OSI Security Architecture - Security Attacks - Security Services and Mechanisms - A Model for Network Security - Classical encryption techniques: Substitution techniques, Transposition techniques, Steganography - Foundations of modern cryptography: Perfect security					
List of Exercise/Experiments: 1. Perform encryption, decryption using the following substitution techniques (i) Ceaser cipher, (ii) playfair cipher iii) Hill Cipher iv) Vigenere cipher 2. Perform encryption and decryption using following transposition techniques (i) Rail fence ii) row & Column Transformation					
UNIT II	SYMMETRIC CIPHERS	9+6			
SYMMETRIC KEY CIPHERS: SDES - Block Ciphers - DES, Strength of DES - Differential and linear cryptanalysis - Block cipher design principles - Block cipher mode of operation - Evaluation criteria for AES - Pseudorandom Number Generators - RC4 - Key distribution.					
List of Exercise/Experiments: 1. Apply DES algorithm for practical applications. 2. Apply AES algorithm for practical applications.					
UNIT III	ASYMMETRIC CRYPTOGRAPHY	9+6			
MATHEMATICS OF ASYMMETRIC KEY CRYPTOGRAPHY: Primes - Primality Testing - Factorization - Euler's totient function, Fermat's and Euler's Theorem - Chinese Remainder Theorem - Exponentiation and logarithm ASYMMETRIC KEY CIPHERS: RSA cryptosystem - Key distribution - Key management - Diffie Hellman key exchange					
List of Exercise/Experiments: 1. Implement RSA Algorithm using HTML and JavaScript. 2. Implement the Diffie-Hellman Key Exchange algorithm for a given problem. 3. Calculate the message digest of a text using the SHA-1 algorithm.					
UNIT IV	INTEGRITY AND AUTHENTICATION ALGORITHMS	9+6			
Hash function - Security of hash function: HMAC, CMAC - SHA - Digital signature and authentication protocols - DSS - Schnorr Digital Signature Scheme - ElGamal cryptosystem - Entity Authentication: Biometrics, Passwords, Challenge Response protocols - Authentication applications - Kerberos MUTUAL TRUST: Key management and distribution - Symmetric key distribution using symmetric and asymmetric encryption - Distribution of public keys					
List of Exercise/Experiments: 1. Implement the SIGNATURE SCHEME - Digital Signature Standard. 2. Demonstrate intrusion detection system (ids) using any tool eg. Snort or any other s/w					
UNIT V	CYBER CRIMES AND CYBER SECURITY	9+6			

Cyber Crime and Information Security - classifications of Cyber Crimes - Tools and Methods - Password Cracking, Keyloggers, Spywares, SQL Injection - Network Access Control - Cloud Security - Web Security - Wireless Security

List of Exercise/Experiments:

1. Automated Attack and Penetration Tools
 - i) Exploring N-Stalker, a Vulnerability Assessment Tool
2. Defeating Malware
 - i) Building Trojans ii) Rootkit Hunter

TOTAL: 45(L) + 30(P) = 75 PERIODS

OUTCOMES:

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

CO1: Understand the fundamentals of networks security, security architecture, threats and vulnerabilities

CO2: Apply the different cryptographic operations of symmetric cryptographic algorithms

CO3: Apply the different cryptographic operations of public key cryptography

CO4: Apply the various Authentication schemes to simulate different applications

CO5: Understand various cyber-crimes and cyber security.

CO6: Apply the Penetration Tools.

TEXT BOOK:

1. William Stallings, Cryptography and Network Security: Principles and Practice, Pearson education 8th Edition, 2023.
2. Nina Godbole, Sunit Belapure, "Cyber Security: Understanding Cyber crimes, Computer Forensics and Legal Perspectives", First Edition, Wiley India, 2011.

REFERENCES:

1. C K Shyamala, N Harini and Dr. T R Padmanabhan: Cryptography and Network Security, Wiley India Pvt.Ltd 1st Edition, 2011
2. Behrouz A. Ferouzan, Debdeep Mukhopadhyay, "Cryptography and Network Security", 3rd Edition, Tata Mc Graw Hill, 2015.
3. Wade Trappe, Lawrence C. Washington: Introduction to Cryptography with Coding Theory, 3rd Edition, 2020.
4. Charles Pfleeger, Shari Pfleeger, Jonathan Margulies, "Security in Computing", Fifth Edition, Prentice Hall, New Delhi, 2015.

LIST OF EQUIPMENTS:

C/C++/JAVA

24CS603	OBJECT ORIENTED SOFTWARE ENGINEERING (Lab Integrated)	L	T	P	C
		2	0	2	3
OBJECTIVES: The Course will enable learners to: • Explain software engineering principles and activities involved in building large software programs. • Describe the process of requirements gathering, analysis and unified modelling • Illustrate the object oriented design process. • Analyse various traditional and object oriented testing methods • Apply estimation techniques, schedule project activities and compute pricing.					

UNIT I	PRODUCT AND PROCESS	6+6
The Nature of Software - Defining the Discipline - The Software Process - Process models - Prescriptive Process Models - Product and Process - Agility and Process - What is an Agile Process? - Scrum - Other Agile Frameworks - Kanban - DevOps List of Exercise/Experiments: 1. Identify a software system that needs to be developed. 2. Document the Software Requirements Specification (SRS) for the identified system.		
UNIT II	REQUIREMENTS AND UNIFIED MODELING	6+6
Requirements Engineering - Establishing the Groundwork: Nonfunctional Requirements - Requirements Gathering - Developing Use Cases - Negotiating and Validating Requirements. Introduction to Unified Modeling Language - Static and Dynamic Modelling -UML Diagrams: Use-Case Diagram - Class Diagrams - Sequence Diagrams- Activity Diagrams-State chart Machine- Package Diagram-Component and Deployment Diagrams. List of Exercise/Experiments: 1. Identify use cases and develop the Use Case model. 2. Identify the conceptual classes and develop a Domain Model and also derive a Class Diagram from that.		
UNIT III	OBJECT ORIENTED ANALYSIS AND DESIGN	6+6
Object oriented Analysis process - Business object Analysis - Use-case driven OOA - Business process modelling. Design Concepts - Design Process - Design Model: Design Principles and Design Elements. Architectural Styles - Designing class-based components - Conducting Component Level Design - User Interface Analysis and Design - Pattern-Based Software Design. List of Exercise/Experiments: 1. Using the identified scenarios, find the interaction between objects and represent them using UML Sequence and Collaboration Diagrams 2. Draw relevant State Chart and Activity Diagrams for the same system		
UNIT IV	SOFTWARE TESTING	6+6
Component Level: A Strategic Approach to Software Testing - White-Box Testing - Black Box Testing - Object Oriented Testing Integration Level: Integration Testing - AI and Regression Testing - Integration Testing in the OO Context Specialized Testing for Mobility: Web application Testing - Web Testing Strategies - Security Testing - Performance Testing - Real time Testing - Testing AI Systems - Testing Virtual Environments. List of Exercise/Experiments: 1. Implement the system as per the detailed design 2. Test the software system for all the scenarios identified as per the use case diagram		
UNIT V	SOFTWARE PROJECT MANAGAMENT	6+6
Software Metrics and Analytics: Software Measurement - Product Metrics. Creating a Viable Software Plan: The Project Planning Process -Estimation Techniques: LOC, FP Based Estimation - Project Scheduling. Risk Management: Risk Identification - Risk Projection - The RMMM Plan. CASE TOOLS List of Exercise/Experiments: 1. Improve the reusability and maintainability of the software system by applying appropriate design patterns. 2. Implement the modified system and test it for various scenarios		
TOTAL: 30(L) + 30(P) = 60 PERIODS		
OUTCOMES: Upon completion of the course, the students will be able to: CO1: Summarize software engineering principles and activities involved in building large software programs		

CO2: Describe the different phases of software development.
CO3: Explain the basics of OOAD and develop software using object oriented design.
CO4: Illustrate the different stages of the design process with a case study.
CO5: Develop miniprojects using the application of object oriented analysis and design.
CO6: Apply different testing strategies to develop efficient projects.

TEXT BOOKS:

1. Roger S. Pressman, "Software Engineering: A Practitioner's Approach", McGraw Hill International Edition, Ninth Edition, 2020.
2. Ali Bahrami, "Object Oriented Systems Development", McGraw Hill International Edition, 2017.

REFERENCES:

1. Micheal Blalh and James Rumbaugh, Object Oriented Modeling and Design with UML, 2nd edition Pearson 2013.
2. Ian Sommerville, "Software Engineering", Tenth Edition, Pearson Education, 2016.
3. Ivar Jacobson, Harold Bud Lawson, Pan-Wei Ng, Paul E. McMahon, Michael Goedicke, "The Essentials of Modern Software Engineering", Morgan & Claypool Publishers, 2019.
4. Booch, G, Jacobson I, Rumbaugh J, "The Unified Modeling Language User Guide", Addison Wesley, 2008.
5. Martin Fowler, "UML Distilled: A Brief Guide to the Standard Object Modeling Language", 3rd edition, Addison Wesley, 2003.

24CS604	DESIGN THINKING (Lab Integrated)	L	T	P	C
		1	0	2	2
OBJECTIVES: The Course will enable learners to: • Familiarize design thinking and its phases. • Perform immersion activity in empathize phase of design thinking. • Create problem statements in the define phase of design thinking. • Ideate and find solutions to the problem defined. • Develop a prototype and perform testing. • Analyze to work in a multidisciplinary environment.					
UNIT I	INTRODUCTION				3+6
Introduction to design thinking - Importance of design thinking for business - Phases of design thinking - Experiential activity - Case study. List of Exercises: 1. An Individual activity 2. Reflection -The Marshmallow Challenge 3. The Wallet Challenge -Team Activity					
UNIT II	EMPATHIZE PHASE				3+6
Empathize phase - Steps involved - Immersion activity- Questionnaire - Empathy map for case study List of Exercises: 1. Creating Empathy maps 2. Creating Customer Journey maps					
UNIT III	DEFINE PHASE				3+6

Creation of personas in define phase - steps in problem statement creation - problem statement definition - Examples - Key problem statements.

List of Exercises:

1. Persona Creation & User Research
2. Develop & present a 3-Minute Pitch

UNIT IV

IDEATION PHASE

3+6

Ideation phase steps - Ideation games - Ideate to find solutions - Doodling - Storytelling in presenting ideas and prototypes.

List of Exercises:

1. Thirty circle - Story Telling
2. Framing the Design Challenge with mind mapping
3. Six Thinking Hats

UNIT V

PROTOTYPE AND TESTING

3+6

Importance of prototype in design thinking - Lean Startup Method for Prototype Development- Design of the prototype test - Visualization and Presentation techniques-Story Board - Value proposition statement - Kano Model - Design thinking in functional work **List of Exercises:**

1. Make a paper prototype for user testing (mock-up model)
2. Develop & present a 3-Minute Pitch (Sample Pitches)

TOTAL: 15(L)+30(P) = 45 PERIODS

OUTCOMES:

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

- CO1: Understand the phases of design thinking process.
- CO2: Conduct an immersion activity to create an empathy map
- CO3: Define the key problems of the personas created.
- CO4: Apply the ideation phase steps to present the prototype ideas
- CO5: Create a prototype with value propositions and test the prototype
- CO6: Investigate techniques to identify new directions in design thinking.

TEXT BOOKS:

1. Christian Müller-Roterberg, "Handbook of Design Thinking", Kindle Direct Publishing, November 2018.
2. Dan Senor and Saul Singer, "Start-Up Nation", Grand Central Publishing, Twelfth Edition, 2009.

REFERENCES:

1. Nir Eyal and Ryan Hoover, "Hooked: How to Build Habit-Forming Products", Library Congress, 2014.
2. Corral, Luis & Fronza, Ilenia, "Design Thinking and Agile Practices for Software Engineering: An Opportunity for Innovation", 2018.

REQUIREMENTS:

1. Newspaper sheets
2. Ice cream sticks
3. Tape
4. Glue
5. Scissors
6. Colored papers
7. Pens and Markers
8. String/thread
9. Chart

24CS611	ADVANCED APTITUDE AND CODING SKILLS - II	L 0	T 0	P 3	C 1.5
OBJECTIVES: - To develop advanced vocabulary for effective communication and reading skills. - To build an enhanced level of logical reasoning and quantitative skills. - To develop error correction and debugging skills in programming. - To apply data structures and algorithms in problem solving.					
LIST OF EXERCISES:					
1. English - Phase II Advanced Vocabulary: Synonyms, Antonyms, Grammar: Subject-Verb Agreement, Tenses and Articles, Prepositions and Conjunctions, Speech and Voices, Comprehension: Inferential and Literal Comprehension, Contextual Vocabulary, Comprehension ordering 2. Logical Reasoning - Phase II Advanced Deductive Reasoning: Coding deductive logic, Directional sense, Blood relations, Objective Reasoning, Selection decision tables, Puzzles, Inductive reasoning: Coding pattern and Number series pattern recognition, Analogy and Classification pattern recognition, Abductive Reasoning: Logical word sequence, Data sufficiency 3. Quantitative Ability - Phase II Advanced Basic Mathematics: Divisibility, HCF and LCM, Numbers, decimal fractions and power, Applied Mathematics: Profit and Loss, Simple and Compound Interest, Time, Speed and Distance, Engineering Mathematics: Logarithms, Permutation and Combinations, Probability 4. Automata Fix - Phase II Logical, Compilation and Code reuse 5. Automata - Phase II Data Structure Concepts: Array and Matrices, Linked list, String processing and manipulation, Stack/Queue, Sorting and Searching Advanced Design and Analysis Techniques: Greedy Algorithms, Minimum Spanning Trees, String Matching, Divide and Conquer, Computational Geometry					
TOTAL: 45 PERIODS					
OUTCOMES: At the end of this course, the students will be able to: CO1: Develop advanced vocabulary for effective communication and reading skills. CO2: Build an enhanced level of logical reasoning and quantitative skills. CO3: Develop error correction and debugging skills in programming. CO4: Apply data structures and algorithms in problem solving.					

24GE611	PRODUCT DEVELOPMENT LAB – 4 (Prototype and Validation Phase) (Common to All Branches)	L	T	P	C
		0	0	2	1
OBJECTIVES: The Course will enable learners to: - Analyze the real time problems during project/product development in engineering perspective. - Develop comprehensive report on the engineering facts applied to a specific problem. - Evaluate the effectiveness of the product or a system through the knowledge acquired. - Synthesize the business opportunities for a new product with novel design.					
DESCRIPTION The students may be grouped into a batch of strength 3 or 4 to work under a project supervisor. The student batches should present their work done, process involved at the end of the semester and submit a report.					
LIST OF ACTIVITIES: - Develop a prototype based on the proposed design. - Demonstration of the project/product and submission of report.					
TOTAL: 30 PERIODS					
OUTCOMES: Upon completion of the course, the students will be able to: CO1: Understand and explain the real time problems. CO2: Analyze the methods to develop solution to the systems. CO3: Evaluate the performance and effectiveness of the existing systems. CO4: Summarize and prepare reports for the experimental determinations. CO5: Classify, compare and analyze business opportunities for a new product.					
LIST OF EQUIPMENT					
S. No.	Description of Equipment				Quantity
1	CNC Router				1 No
2	3D Printer				1 No
3	3D Scanner				1 No
4	Laser cutting Machine				1 No
5	Multimeter				5 Nos
6	Solder Stations				5 Sets
7	Desoldering Machine				1 No
8	PCB Milling Machine				1 No
9	Variable Power Supply				1 No
10	Electronic Components like Resistors, Transistors, Diode, Inductor, Capacitor, etc.				5 Sets

SEMESTER - VII

22CS701	DATA ANALYTICS (Lab Integrated)	L	T	P	C
		3	0	2	4
OBJECTIVES: - To explain the fundamentals of big data and data analytics - To discuss the Hadoop framework - To explain about exploratory data analysis and data manipulation tools and use it for developing applications - To analyse and interpret streaming data - To discuss various applications of data analytics					
UNIT I	INTRODUCTION				9+6
Evolution of Big Data- Definition of Big Data-Challenges with Big Data- Traditional Business Intelligence (BI) versus Big Data- Introduction to big data analytics- Classification of Analytics- Analytics Tools- Importance of big data analytics - Technologies used in Big Data Environment. List of Exercise/Experiments: - Download, install and explore the features of R/Python for data analytics. - Working with Numpy arrays and Pandas data Frames					
UNIT II	HADOOP FRAMEWORK				9+6
Introducing hadoop - Why hadoop, Why not RDBMS - RDBMS Vs Hadoop -History of Hadoop - Hadoop overview - Use case of Hadoop - HDFS (Hadoop Distributed File System) - Processing data with Hadoop and Map Reduce Programming - Managing resources and applications with Hadoop YARN (Yet Another Resource Negotiator). List of Exercise/Experiments: - Implement a MapReduce program to implement Matrix Multiplication - Implement a Map Reduce program that mines weather data and displays appropriate messages indicating the weather conditions of the day.					
UNIT III	EXPLORATORY DATA ANALYSIS				9+6
EDA fundamentals - Understanding data science - Significance of EDA - Making sense of data - Comparing EDA with classical and Bayesian analysis - Software tools for EDA -Data transformation techniques - Introduction to NoSQL - MongoDB: RDBMS Vs MongoDB - Data Types - Query Language - Hive - Hive Architecture - Data Types - File Formats - Hive Query Language (HQL) - RC File Implementation - User Defined Functions. List of Exercise/Experiments: - Statistical and Probability measures - Frequency distributions, Mean, Mode, Standard Deviation, Variability, Normal curves, Correlation and scatter plots, Correlation coefficient, Regression. - Use the standard benchmark data set for performing the following: - Univariate Analysis: Frequency, Mean, Median, Mode, Variance, Standard Deviation, Skewness and Kurtosis. - Bivariate Analysis: Linear and logistic regression modelling. - Multiple Regression Analysis - Compare the results of the above analysis for the two data sets.					
UNIT IV	MINING DATA STREAMS				9+6

The data stream model - stream queries-sampling data in a stream-general streaming problem filtering streams-analysis of filtering- dealing with infinite streams- Counting Distance Elements in a Stream - Estimating Moments - Counting Ones in Window - Decaying Windows - Real time Analytics Platform (RTAP)

List of Exercise/Experiments:

1. Apply and explore various plotting functions on any data set using Matplotlib and Seaborn
2. Implement the following algorithms on real time stream data sets.
 - a.) Support Vector Machine
 - b.) Decision tree classifier
 - c.) Clustering Algorithms

UNIT V	APPLICATIONS	9+6
---------------	---------------------	------------

Application: Sales and Marketing - Industry Specific Data Mining - microRNA Data Analysis Case Study - Credit Scoring Case Study - Data Mining Non tabular Data.

List of Exercise/Experiments:

1. Write a program to construct a Bayesian network considering medical data. Use this model to demonstrate the diagnosis of heart patients using the standard Heart Disease Data Set. You can use Java/Python ML library classes/API
2. Mini Project: The project should contain the following components
 - Realtime dataset
 - Data preparation & Transformation
 - Handling missing Data
 - Data Storage
 - Algorithm for data analytics

Data visualization: Charts, Heatmap, Crosstab, Treemap

TOTAL: 45(L) + 30(P) = 75 PERIODS

OUTCOMES:

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

CO1: Explain the fundamentals of big data and data analytics and illustrate it

CO2: Discuss the Hadoop framework

CO3: Develop applications using exploratory data analysis and data manipulation tools

CO4: Analyze and interpret streaming data

CO5: Illustrate various applications of data analytics

CO6: Build applications for various domains.

TEXT BOOKS:

1. Subhashini Chellappan, Seema Acharya, "Big Data and Analytics", 2nd edition, Wiley Publications, 2019.
2. Suresh Kumar Mukhiya and Usman Ahmed, "Hands-on Exploratory Data Analysis with Python", Packt publishing, March 2020.
3. Jure Leskovek, Anand Rajaraman and Jeffrey Ullman, "Mining of Massive Datasets. v2.1", Cambridge University Press, 2019.
4. Glenn J. Myatt, Wayne P. Johnson, Making Sense of Data II: A Practical Guide To Data Visualization, Advanced Data Mining Methods, and Applications, Wiley 2009.

REFERENCES:

1. Nelli, F., Python Data Analytics: with Pandas, NumPy and Matplotlib, Apress, 2018.
2. Bart Baesens, "Analytics in a Big Data World: The Essential Guide to Data Science and its Applications", John Wiley & Sons, 2014.
3. Min Chen, Shiwen Mao, Yin Zhang, Victor CM Leung, Big Data: Related Technologies, Challenges and Future Prospects, Springer, 2014.
4. Michael Minelli, Michele Chambers, Ambiga Dhiraj, "Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends", John Wiley & Sons, 2013.
5. Marcello Trovati, Richard Hill, Ashiq Anjum, Shao Ying Zhu, "Big Data Analytics and cloud computing - Theory, Algorithms and Applications", Springer International Publishing, 2016.

LIST OF EQUIPMENTS:

R / Python

22CS702	WIRELESS AND MOBILE COMMUNICATION	L	T	P	C
		3	0	0	3

OBJECTIVES:

- To introduce the basic concepts and fundamentals of wireless communication systems and their applications
- To provide knowledge on cellular mobile communication concepts such as frequency reuse, handoff, and interference management
- To develop understanding of wireless channel behavior including path loss, fading, multipath propagation, and shadowing effects
- To enable comparison and application of multiple access techniques such as FDMA, TDMA, CDMA, and OFDMA
- To familiarize students with the architecture and functioning of modern wireless systems including GSM, UMTS, LTE, 5G, Wi-Fi, and Bluetooth

UNIT I	INTRODUCTION TO WIRELESS SYSTEMS	9
Overview of wireless communication -Applications of wireless systems-Types of wireless communication systems -Cellular communication, Satellite communication, Wireless LAN (Wi-Fi) -Electromagnetic spectrum and frequency bands -Basic wireless transmission concepts - Signal attenuation and basic propagation loss		
UNIT II	CELLULAR MOBILE COMMUNICATION	9
Cellular system concept and architecture-Frequency reuse and channel allocation-Handoff Strategies- Interference in cellular systems- Capacity improvement techniques-cell splitting, sectoring.		
UNIT III	WIRELESS CHANNEL AND PROPAGATION	9
Wireless channel characteristics-Radio wave propagation mechanisms-Reflection, Diffraction, Scattering-Free space propagation model-Path loss and large-scale propagation models-Shadowing effects-Multipath propagation-Delay spread-Fading (small-scale and large-scale)		
UNIT IV	MULTIPLE ACCESS TECHNIQUES FOR WIRELESS COMMUNICATIONS	9
Introduction to multiple access techniques-Channel sharing in wireless systems-Frequency Division Multiple Access (FDMA)- Time Division Multiple Access (TDMA)- Code Division Multiple Access (CDMA)- Orthogonal Frequency Division Multiple Access (OFDMA) basics- Comparison of multiple access techniques.		

UNIT V	WIRELESS NETWORKS AND MODERN SYSTEMS	9
GSM- Mobile Services, System architecture, Radio interface, Protocols, Localization and calling, Handover, Security, New data services- UMTS- UMTS releases and standardization, UMTS System Architecture, UMTS radio interface, Core Network, Handover-3G, 4G (LTE), and 5G overview-Wi-Fi (IEEE 802.11), Bluetooth.		
TOTAL: 45 PERIODS		
OUTCOMES: Upon completion of the course, the students will be able to: CO1: Describe the fundamentals of wireless communication systems and their applications. CO2: Analyze cellular mobile communication concepts including frequency reuse, handoff, and interference CO3: Interpret wireless channel characteristics such as path loss, fading, multipath propagation, and shadowing. CO4: Differentiate multiple access techniques like FDMA, TDMA, CDMA, and OFDMA used in wireless systems. CO5: Describe the architecture and operation of modern wireless systems such as GSM, UMTS, LTE, 5G, Wi-Fi, and Bluetooth CO6: Illustrate the role of modern wireless technologies in mobile communication and IoT applications.		
TEXT BOOKS: 1. Theodore S. Rappaport, Wireless Communications: Principles and Practice, Updated 2nd Edition, Pearson Education, 2024. 2. Jochen H. Schiller, Mobile Communications, 2nd Edition, Pearson Education, 2003.		
REFERENCES: 1. William Stallings, Wireless Communications and Networks, 2nd Edition, Pearson Education, 2005. 2. Andrea Goldsmith, Wireless Communications, Cambridge University Press, 2005		

**PROFESSIONAL ELECTIVES
CYBER SECURITY**

24CS901	ETHICAL HACKING (Lab Integrated)	L	T	P	C
		2	0	2	3
OBJECTIVES: The Course will enable learners to: • To understand Information Security, Cyber threats, attacks, web security. • To know about different modes of hacking tools and phases of penetration tests and Methodologies. • To Gain the knowledge of the use and availability of tools to support an ethical hack • To Gain the knowledge of interpreting the results of a controlled attack					
UNIT I	Fundamentals of Ethical Hacking				6+6
Introduction to Cyber Security-Overview of Cyber threats - Data and Network Security Attacks - Threats: MAC spoofing - Access control Network protocol and services-Hacking terms - Ethical Hacking overview -Modes of Ethical Hacking - Ethics and Legality.					
List of Exercise/Experiments: 1. Setup a honey pot and monitor the honey pot on network 2. Write a script or code to demonstrate SQL injection attacks 3. Write a code to demonstrate DoS attacks					
UNIT II	Hacking Methodology Reconnaissance				6+6
Foot printing: Reconnaissance - Footprinting theory - Penetration test - Phases of Penetration test - Methods of Footprinting - Network Information gathering process - Terminologies of Foot printing -Footprinting through search engine directives - Whois tool -NetCraft - Extract Information from DNS - Foot printing from Email servers - Shodan - Dig - MetaGooFil - Social Engineering.					
List of Exercise/Experiments: 1. Performing footprinting using Google Hacking, website information, information about an archived website, to extract contents of a website, to trace any received email, to fetch DNS information. 2. Create a social networking website login page using phishing techniques					
UNIT III	Scanning and Enumeration				6+6
Scanning: Concept of Nmap -Port scanning with Nmap - Subnet - Scanning IPs with Nmap Pings and Ping sweeps - Port - Three way handshake - NmapSyn scanning - Nmap TCP Scan - Nmap UDP Scan - Bypass of IPS and IDS - Nmap Script Engine Enumeration: Service Fingerprinting - Vulnerability Scanners - Basic Banner Grabbing - Common Network services - SMTP - DNS - RPCBIND Enumeration - SMB - NetBIOS					
List of Exercise/Experiments: 1. Implement Passive scanning, active scanning, session hijacking, cookies extraction using Burp suit tool 2. Use port scanning. network scanning tools,IDS tool, sniffing tool and generate reports.					
UNIT IV	System and Network Vulnerability				6+6

Metasploit - Penetration testing with framework Metasploit - Scan services to identify vulnerabilities - Scan FTP services - Scan HTTP services - Exploitation - Post exploitation techniques - Meterpreter - Rootkit - Backdoor - Password hashes - Privilege Escalation - Scanning vulnerable services with Nessus

List of Exercise/Experiments:

1. Penetration Testing using Metasploit and metasploitable
2. Creating a simple keylogger in python
3. Creating a virus
4. Creating a trojan.
5. Install rootkits and study variety of options

UNIT V Software Vulnerability (OWASP 10)

6+6

Fundamentals of OWASP Zed Attack Proxy (ZAP) - Web app vulnerability scan - Code Injection Attacks - Broken Authentication - Sensitive Data Exposure - XML External Entities - Broken Access Control - Security misconfiguration - Website pen testing - Cross Site Scripting (XSS) - Insecure Deserialization - Using Components with known vulnerabilities - Insufficient logging and monitoring.

List of Exercise/Experiments:

1. Install jcrypt tool (or any other equivalent) and demonstrate Asymmetric, Symmetric Crypto algorithm, Hash and Digital/PKI signatures studied in theory Network Security And Management

TOTAL:30(T) + 30(P) = 60 PERIODS

OUTCOMES:

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

CO1: Identify cybersecurity threats and network vulnerabilities to enhance data and system security.

CO2: Perform Penetration Testing using tools Metasploit and Nmap to evaluate system defenses.

CO3: Apply ethical hacking techniques reconnaissance, scanning, and enumeration to evaluate security posture.

CO4: Detect and exploit vulnerabilities in networks and systems while adhering to ethical standards.

CO5: Analyze Software Vulnerabilities and mitigate risks.

CO6: Develop and implement countermeasures against attacks such as SQL injection, DoS, and malware.

TEXTBOOKS:

1. McClure, S., Scambray, J. and Kurtz, G., 2012. Hacking Exposed 7 Network Security Secrets and Solutions. New York: McGraw-Hill.
2. Engebretson, P., 2013. The Basics Of Hacking And Penetration Testing. Amsterdam: Syngress, an imprint of Elsevier.

REFERENCES:

1. Zaid Sabih, Learn Ethical Hacking from Scratch, 2018, PACKT publishing, ISBN: 978-1-78862-205-9
2. Harsh Bothra, Hacking be a hacker with ethics, Khanna Publishing, 2016, ISBN: 978-03-86173-05-8

LIST OF SOFTWARE:

1. Kali Linux -Metasploit Framework (MSF) & Nmap
2. WireShark, Jcrypt Tool
3. Burp suite

24CS902	SOCIAL NETWORK SECURITY (Lab Integrated)	L	T	P	C
		2	0	2	3
OBJECTIVES: The Course will enable learners to: • Learn the Concepts of Social Network Security • Understand the methods of Social Network Anonymization • Learn the techniques for security and privacy in social networks • Learn the security challenges in social networks • Explore the Tools to learn about the social network security implementation					
UNIT I	INTRODUCTION TO SOCIAL NETWORK SECURITY				6+6
Introduction - Social Networking Applications - Social media Websites - Social Network Representation -Building Social Authority -Privacy and Information sharing - Controlling Application privacy - Cybercrime - Information Leakage - False information - Content Management in Social Networks					
List of Exercise/Experiments 1. Explore a Social network analysis tools to learn about the users and networks 2. Learn a program / tool to illustrate information leakage					
UNIT II	SOCIAL NETWORK ANONYMIZATION				6+6
Social Networks - Privacy in Social Networks - Social Network Representation - Social Network Analysis - Data Anonymization - Challenges in Anonymization - Privacy preservation - Social Network Anonymization Factors - Anonymization Algorithms - Link Anonymization techniques -Background Knowledge Attacks - Anonymity in Modern Social Networks					
List of Exercise/Experiments 1. Experiment a link anonymization technique 2. Explore ARX anonymization tool					
UNIT III	ANALYZING AND SECURING SOCIAL NETWORKS				6+6
Supporting Technologies - Aspects of Analyzing and Securing Social Networks - Techniques and Tools for Social Network Analytics - Social Network Analytics and Privacy Considerations - Access Control and Inference for Social Networks - Social Media Integration and Analytics Systems - Social Media Application Systems - Secure Social Media Systems - Secure Social Media Directions.					
List of Exercise/Experiments 1. Implement a program for network access control to illustrate malware attacks 2. Create a simple social network application to show authentication mechanisms 3. Create an application for the following scenario: “Social networking users are presented with two apparently similar emails or websites. They must first identify the differences between them and then decide which one is a scam attempting to steal their information or money.”					
UNIT IV	SECURITY CHALLENGES IN SOCIAL NETWORKS				6+6

Identity manipulation - Threats from third party applications - Trust in Social Networking Sites - Viruses, Phishing Attacks and Malwares-Tracking users - Privacy of Data - Identity Federation Challenges -Social media threats - Location disclosure - Spoofing - Profile cloning - Fake product sale - Cyber bullying - Prevention Strategies

List of Exercise/Experiments

1. Implement a program in python to estimate trust of social network users group
2. Write a SQL injection program in python/JAVA to handle session hijacking
3. Create an application using any social network platform to demonstrate profile cloning concept.

UNIT V | SOCIAL NETWORK SECURITY TOOLS

6+6

Analysis Tools for Social Media - AutoMap - Gephi - ORA Lite - ORA Pro - Wolfram Alpha - Social Media Data Collection -Blog Trackers -Crowd Tangle - MalTego - Pulse - SCRAAWL - Fact and Image Trackers - Google Fact Check Tools - Bot Mitigation - BotSlayer - Social Cyber Security

List of Exercise/Experiments

1. Perform fact checking of social networking content using google fact checking tools
2. Explore a tool that helps protect websites from bot traffic and bot attacks.
3. Create a fake news tracker program to collect, detect and help visualize fake news data from any social network

TOTAL:30(T) + 30(P) = 60 PERIODS

OUTCOMES:

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

- CO1:** Understand social network security concepts and applications.
CO2: Apply data anonymization techniques to enhance privacy in social networks.
CO3: Analyze and secure social networks using relevant technologies and tools.
CO4: Identify and address security challenges inherent in social networks.
CO5: Utilize security tools for monitoring and protecting social network activities.
CO6: Demonstrate ethical considerations in social network security practices.

TEXTBOOKS:

1. Brij B. Gupta, Somya Ranjan Sahoo, "Online Social Networks Security-Principles, Algorithm, Applications, and Perspectives", First Edition, 2021.
2. Bhavani Thuraisingham, SatyenAbrol, Raymond Heatherly, Murat Kantarcioglu, Vaibhav Khadilkar, Latifur Khan, "Analyzing and Securing Social Networks", First Edition, 2020.

REFERENCES:

1. B. K. Tripathy, Kiran Baktha, "Security, Privacy, and Anonymization in Social Networks: Emerging Research and Opportunities", IGI Global Publication, 2019.
2. Michael Cross, "Social Media Security, Leveraging Social Networking While Mitigating Risk", Elsevier Publication, First Edition, 2013.
3. El-Sayed M. El-Alfy; Mohamed Eltoweissy; Errin W. Fulp; Wojciech Mazurczyk, "Nature-Inspired Cyber Security and Resiliency: Fundamentals, Techniques and Applications", IET Publication, 2019.
4. Yaniv Altshuler, "Security and Privacy in Social Networks", Springer, 2013.

LIST OF SOFTWARE:

Software/ Tools Required:

Python
 ARX anonymization tool
 BotSlayer
 GOOGLE FACT CHECK TOOLS

24CS903	BLOCKCHAIN TECHNOLOGY	L	T	P	C
		3	0	0	3
OBJECTIVES: The Course will enable learners to: • To understand block chain system’s fundamental components, how they fit together and examine a decentralization using block chain. • To explain how Crypto currency works. • To explain the components of Ethereum and Programming Languages for Ethereum • To study the basics of Web3 and Hyper ledger. • To give an insight of alternative block chains and its emerging trends.					
UNIT I	INTRODUCTION TO BLOCKCHAIN				9
History of Blockchain - Types of Blockchain - Consensus - Decentralization using Blockchain - Blockchain and Full Ecosystem Decentralization - Platforms for Decentralization - Symmetric Cryptography - Mathematics - Asymmetric Cryptography - public and private keys - Elliptic curve cryptography - Discrete logarithm problem in ECC					
UNIT II	INTRODUCTION TO CRYPTOCURRENCY				9
Bitcoin - Digital Keys and Addresses - Transactions - Mining - Bitcoin Networks and Payments Wallets - innovation in Bitcoin - Alternative Coins - Theoretical Foundations - Bitcoin.					
UNIT III	ETHEREUM				9
The Ethereum Network - Components of Ethereum Ecosystem - Ethereum Programming Languages: Runtime Byte Code - Blocks and Blockchain - Fee Schedule - Supporting Protocols - Solidity Language.					
UNIT IV	WEB3 AND HYPERLEDGER				9
Introduction to Web3 - Contract Deployment - POST Requests - Development frameworks Hyperledger as a protocol - The Reference Architecture - Hyperledger Fabric - Distributed Ledger.					
UNIT V	ALTERNATIVE BLOCKCHAINS AND NEXT EMERGING TRENDS				9
Kadena - Ripple- Rootstock - Quorum - Tendermint - Scalability - Privacy - Other Challenges - Blockchain Research. Case Study - Install IPFS locally on our machine, initialize your node, view the nodes in network.					
TOTAL:45 PERIODS					
OUTCOMES: At the end of this course, the students will be able to: CO1: Understand the fundamentals and history of blockchain technology. CO2: Analyze different types of blockchain and their consensus mechanisms. CO3: Apply cryptographic principles to blockchain systems. CO4: Explore cryptocurrency concepts, including Bitcoin and alternative coins. CO5: Evaluate Ethereum and its programming languages for decentralized applications. CO6: Investigate other blockchain platforms like Hyperledger and emerging trends.					

TEXTBOOKS:	
1. Imran Bashir, "Mastering Blockchain: Distributed Ledger Technology, decentralization, and smart contracts explained", 2nd Edition, Packt Publishing Ltd, March 2018.	
2. BellajBadr, Richard Horrocks, Xun (Brian) Wu, "Blockchain By Example: A developer's guide to creating decentralized applications using Bitcoin, Ethereum, and Hyperledger", Packt Publishing Limited, 2018.	
REFERENCES:	
1. Andreas M. Antonopoulos, "Mastering Bitcoin: Unlocking Digital Cryptocurrencies", O'Reilly Media Inc, 2015	
2. Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller and Steven Goldfeder, "Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction", Princeton University Press, 2016.	

24CS904	CLOUD AND DATA SECURITY	L	T	P	C
		3	0	0	3
OBJECTIVES: The Course will enable learners to: - Learn the basics of cloud security, including the shared responsibility model and identity management. - Set up a secure cloud infrastructure with features like virtual private clouds and security groups. - Develop skills for identifying and managing security incidents in the cloud, adhering to best practices. - Safeguard application data at rest and in transit using encryption. - Understand the features of Database Security and Security in Data Warehouses.					
UNIT I	INTRODUCTION TO SECURITY IN CLOUD				9
Introduction to Security, Security in the Cloud, Security design principles, Shared responsibility model, Activity: Shared Responsibility Model, Identity and Access Management (IAM) fundamentals, Authenticating and Authorizing with IAM, Examples of authorizing with IAM, Additional authentication and access management services, Using Organizations.					
UNIT II	SECURING INFRASTRUCTURE				9
Structure of a three-tier web application, virtual private cloud (VPC), Setting up public and private subnets and internet protocols, Security groups, Network access control lists (ACLs), Load balancers, Protecting compute resources- Cloud service models: IaaS, PaaS, SaaS.					
UNIT III	INCIDENT RESPONSE AND RISK MANAGEMENT				9
Identifying an incident, Services that support the discovery and recognition phase, AWS Config and AWS Lambda, Services that support the resolution and recovery phase, Best practices for handling an incident.					
UNIT IV	SECURING CLOUD: DATA SECURITY				9
Overview of Data Security in Cloud Computing- Common Risks with Cloud Data Security- Data Encryption: Applications and Limits- Cloud Data Security: Sensitive Data Categorization- Authentication and Identity- Data Categorization and the Use of Data Labels- Cloud Data Storage.					
UNIT V	DATABASE SECURITY				9
Database Security: Recent Advances in Access Control, Access Control Models for XML, Database Issues in Trust Management and Trust Negotiation, Security in Data Warehouses and OLAP Systems.					
TOTAL:45 PERIODS					

OUTCOMES:

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

CO1: Understand security principles in cloud computing.

CO2: Implement infrastructure security measures in cloud environments.

CO3: Demonstrate incident response and risk management techniques in cloud systems.

CO4: Apply data security practices to protect cloud-based information.

CO5: Evaluate database security mechanisms for cloud-based data management.

CO6: Integrate ethical considerations into cloud security practices.

TEXTBOOKS:

1. Tim Mather, Subra Kumaraswamy, Shahed Latif, "Cloud Security and Privacy, An Enterprise Perspective on Risks and Compliance", Oreilly Media 2009.
2. Vic (J.R.) Winkler, "Securing the Cloud, Cloud Computer Security Techniques and Tactics", Syngress, April 2011.

REFERENCES:

1. Michael Gertz, Sushil Jajodia, "Handbook on Database security: Applications and Trends", Springer, 2010.
2. John R. Vacca, "Cloud Computing Security", CRC Press, 2016.
3. Giulio D'Agostino, "Data Security in Cloud Computing, Volume I", Momentum Press, 2019.

24CS905	ENTERPRISE CYBER SECURITY	L	T	P	C
		3	0	0	3
OBJECTIVES: The Course will enable learners to: • To learn the fundamentals of cryptography. • To learn the key management techniques and authentication approaches. • To explore the network and transport layer security techniques. • To understand the application layer security standards. • To learn the real time security practices.					
UNIT I	INTRODUCTION TO CYBERSECURITY				9
Cyber Security - Need of Cybersecurity in Organizations - CIA Triad- Confidentiality, Integrity, Availability; Reason for Cyber Crime -Need for Cyber Security - History of Cyber Crime; Cybercriminals - Classification of Cybercrimes- A Global Perspective on Cyber Crimes; Cyber Laws - The Indian IT Act - Cybercrime and Punishment.					
UNIT II	NETWORK SECURITY BASICS				9
Network Security Concepts- Basics of Networks- Common Types of Network Attacks- Introduction to Firewalls- Types of Firewalls- IDS/IPS- Virtual Private Networks (VPN's)- Secure configuration and management of network devices. Case Study: Install Kali Linux on Virtual box.					
UNIT III	SECURE COMMUNICATION PROTOCOLS				9
Encryption Principles- Cryptography, Cryptanalysis, Feistel Cipher Structure. Block Encryption algorithms: DES, triple DES, and AES. Transport-Level Security: Secure Sockets Layer (SSL), Transport Layer Security TLS). Electronic Mail Security- Pretty Good Privacy (PGP), S/MIME. Securing wireless networks: WPA, WPA2, WPA3.					
UNIT IV	INTRUSION DETECTION AND PREVENTION SYSTEMS				9

IDPS- Need of Intrusion Detection Systems in Cyber Security- Types of IDPS: Network-based and Host-based. Configuring and Managing IDPS for threat detection using Honeypots. Case Study: Setup a honey pot and monitor the honey pot on network.

UNIT V	WEB APPLICATION SECURITY	9
---------------	---------------------------------	----------

Introduction to Web Application Vulnerabilities - Cross Site Scripting (XSS) - SQL injection- Denial of Service (DoS)- Web Application Testing - Types of Penetration Tests- OWASP and OWASP Top.

TOTAL: 45 PERIODS

OUTCOMES:

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

- CO1:** Understanding the core concepts and importance of cybersecurity in organizational settings.
- CO2:** Acquire the knowledge common network attacks and deploy appropriate security measures.
- CO3:** Implement encryption and secure communication protocols for data integrity and confidentiality.
- CO4:** Deploy and manage Intrusion Detection and Prevention Systems for threat detection.
- CO5:** Identify and mitigate common web application vulnerabilities.
- CO6:** Conduct penetration tests to evaluate the security posture of web applications.

TEXTBOOKS:

1. Anand Shinde, "Introduction to Cyber Security Guide to the World of Cyber Security", Notion Press, 2021.
2. Network Security Essentials (Applications and Standards) by William Stallings Pearson Education, 2018.

REFERENCES:

1. William Stallings, "Cryptography and Network Security - Principles and Practice", Seventh Edition, Pearson Education, 2017.
2. Ravi Das and Greg Johnson, "Testing and Securing Web Applications", 2021.
3. Andrew Hoffman, Web Application Security: Exploitation and Countermeasures for Modern Web Applications, O'Reilly Media, Inc, 2020.

24CS940	CYBER SECURITY ESSENTIALS	L	T	P	C
		3	0	0	3
OBJECTIVES: The Course will enable learners to: • The difference between threat, risk, attack and vulnerability. • How threats materialize into attacks. • Where to find information about threats, vulnerabilities and attacks. • Typical threats, attacks and exploits and the motivations behind them.					
UNIT I	INTRODUCTION TO CYBER SECURITY				9
Introduction -Computer Security - Threats -Harm - Vulnerabilities - Controls - Authentication -Access Control and Cryptography - Web–User Side - Browser Attacks - Web Attacks Targeting Users - Obtaining User or Website Data - Email Attacks					
UNIT II	SECURITY IN OPERATING SYSTEM & NETWORKS				9

Security in Operating Systems - Security in the Design of Operating Systems - Rootkit - Networking basics (home network and large-scale business networks), Networking protocols, Security of protocols, sample application hosted on-premises.

UNIT III DIGITAL SECURITY

9

Basics of digital security, protecting personal computers and devices, protecting devices from Virus and Malware, Identity, Authentication and Authorization, need for strong credentials, keeping credentials secure, protecting servers using physical and logical security, World Wide Web (www), the Internet and the HTTP protocol, security of browser to web server interaction

UNIT IV PRIVACY IN CYBERSPACE

9

Introduction to cyber-attacks, application security (design, development and testing), operations security, monitoring, identifying threats and remediating them, Principles of data security - Confidentiality, Integrity and Availability, Data Privacy, Data breaches, preventing attacks and breaches with security controls, Compliance standards, Computer Ethics.

UNIT V MANAGEMENT AND INCIDENTS

9

Security Planning - Business Continuity Planning - Handling Incidents - Risk Analysis - Dealing with Disaster - Emerging Technologies - The Internet of Things - Economics - Electronic Voting - Cyber Warfare- Cyberspace and the Law - International Laws - Cyber crime - Cyber Warfare and Home Land Security

TOTAL:45 PERIODS

OUTCOMES:

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

CO1: Understand basic concepts of computer security, identify threats and vulnerabilities, and implement basic security measures.

CO2: Comprehend security mechanisms in operating systems and networks, including rootkits and network protocols, and secure network servers.

CO3: Protect personal computers and devices from malware, understand the importance of strong credentials, and secure web interactions.

CO4: Understand cyber-attacks, data security principles, data privacy, and compliance standards.

CO5: Develop security planning strategies, handle security incidents, understand emerging technologies, and comprehend legal aspects of cyberspace.

CO6: Analyze and evaluate real-world case studies to apply concepts of computer security, operating system and network security, digital security, data privacy, and incident management.

TEXTBOOKS:

1. Charles P. Pleegeer Shari Lawrence Pleegeer Jonathan Margulies, Security in Computing, 5th Edition , Pearson Education , 2015
2. Brooks, Charles J., Christopher Grow, Philip Craig, and Donald Short. Cybersecurity essentials. John Wiley & Sons, 2018.

REFERENCES:

1. George K.Kostopoulous, Cyber Space and Cyber Security, CRC Press, 2013.
2. Martti Lehto, Pekka Neittaanmäki, Cyber Security: Analytics, Technology and Automation edited, Springer International Publishing Switzerland 2015
3. Nelson Phillips and Eninger Steuart, –Computer Forensics and Investigations||, Cengage Learning, New Delhi, 2009.

LIST OF SOFTWARE:

List of Software: Windows
Defender Firewall Wireshark
Event Viewer
Microsoft Baseline Analyzer Linux
Distribution
Nmap, ZenMap
Windows local Security Policy

24CS941	AI IN CYBER SECURITY	L	T	P	C
		3	0	0	3
OBJECTIVES: The Course will enable learners to: • Learn the Concepts of AI in Cybersecurity. • Understand the methods of Cybersecurity Arsenal • Learn the techniques for security threats with AI • Protect the sensitive information and assets. • Explore the Tools and Applications to learn about the AI in Cybersecurity.					
UNIT I	INTRODUCTION TO AI IN CYBERSECURITY				9
Introduction - Applying AI in cyber security - Evolution from expert systems to data mining and AI- The different forms of automated learning - Characteristics of algorithm training and optimization - Beginning with AI via Jupyter Notebooks - Introducing AI in the context of cyber security.					
UNIT II	AI FOR CYBER SECURITY ARSENAL				9
Classification - Regression - Dimensionality reduction - Clustering - Speech recognition - Video anomaly detection - Natural language processing (NLP) - Large-scale image processing - Social media analysis.					
UNIT III	CYBER SECURITY THREATS WITH AI				9
Detection of Spam with Perceptron - Image spam detection with support vector machines - Phishing detection with logistic regression and decision trees- Spam detection with Naive Bayes - Spam detection adopting NLP					
UNIT IV	PROTECTING SENSITIVE INFORMATION AND ASSETS				9
Authentication abuse prevention - Account reputation scoring - User authentication with keystroke recognition -Biometric authentication with facial recognition					
UNIT V	APPLICATIONS				9
Machine learning (ML) algorithms for fraud detection - bagging and boosting techniques - analyze data with IBM Watson and Jupyter Notebook - resort to statistical metrics for results evaluation.					
TOTAL:45 PERIODS					

OUTCOMES:

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

- CO1:** Apply AI to enhance cybersecurity using Jupyter Notebooks for algorithm development and optimization.
- CO2:** Implement AI for cyber security arsenal.
- CO3:** Implement Spam Detection using ML and NLP Models
- CO4:** Protect and handle sensitive information and assets.
- CO5:** Implement AI-driven applications specifically tailored for fraud detection purposes.
- CO6:** Evaluate the effectiveness of AI-based solutions in enhancing cybersecurity measures.

TEXTBOOKS:

1. Alessandro Parisi .(2019). Hands-On Artificial Intelligence for Cybersecurity: Implement smart AI systems for preventing cyber attacks and detecting threats and network anomalies, Packt Publication

REFERENCES:

1. Mark Stamp Corrado Aaron Visaggio Francesco Mercaaldo Fabio Di Troia Editors , “ Artificial Intelligence for Cybersecurity” , Springer
2. Leslie F. Sikos Editor “AI in Cybersecurity”, Springer

LIST OF SOFTWARE:

1. **Software Required:** Python
2. **Software Tools Required:**
 - Pytorch
 - Scikit Learn
 - TensorFlow
 - Py TorchOpenNN

24IT903	REST APPLICATION DEVELOPMENT USING SPRING BOOT AND JPA (Lab Integrated)	L	T	P	C
		2	0	2	3
OBJECTIVES: The Course will enable learners to: • To provide comprehensive knowledge of RESTful APIs and the HTTP methods used in the Spring Boot framework. • To cover advanced querying techniques using JPA, including LIKE queries, and to manage CRUD operations using JPQL. • To explore various relational mappings in JPA, such as one-to-one and one-to-many associations, and their practical implementations. • To implement and manage Spring AOP applications using annotation-based configurations for method interception and post-execution operations. • To build production-grade Spring Boot applications with integrated security using JWT, detailed API documentation with SwaggerUI and OpenUI, and effective logging practices.					
UNIT I	Server-Side Programming and REST APIs				6+6

Introduction to server-side programming and client-server architecture. Fundamentals of RESTful web services and API design. Overview of HTTP methods, status codes, and data formats such as JSON. Introduction to the Spring Framework and its features. Basics of dependency injection and Spring Boot application development.

List of Exercise/Experiments:

1. Create a basic application using Spring Framework. (Create a simple Spring project. Define a bean Retrieve bean from container).
2. Develop a RESTful API for retrieving a welcome message, emphasizing the basics of data exchange between client and server.
3. Implement a RESTful API to acknowledge the user's favorite color choice, highlighting property value injection principles.
4. Create a Spring Boot application that retrieves and displays application information, demonstrating the usage of the @Value annotation to inject property values from the application configuration file.
5. Construct a RESTful API for student details retrieval, illustrating the utilization of @JsonIgnore annotation, focusing on advanced JSON property handling and data access control.

UNIT II	Database operations	6+6
----------------	----------------------------	------------

Database Operations & Spring MVC: MYSQL integration, mapping Java classes to relational database, repository interface, retrieving data, request body mapping.

Spring MVC & REST Integration: Spring MVC architecture, building REST services using Spring, HTTP methods (GET, POST, PUT, DELETE), error handling, logging with Log4J.

List of Exercise/Experiments:

1. Develop a web application for managing patient details using RESTful APIs, implementing POST and GET operations.
2. Create a web application for managing product details using RESTful APIs, enabling POST and GET operations.
3. Build an application for managing employee details using RESTful APIs, supporting POST, PUT, and DELETE operations.
4. Performing CRUD operations using Spring MVC and RESTful services.

UNIT III	Java Persistence API	6+6
-----------------	-----------------------------	------------

Pagination & Sorting using JPA, @Transient Annotation, Queries using JPA, Starts and Ends with query using JPA, JPQL with @Query Annotation, custom JPQL queries.

List of Exercise/Experiments

1. Develop a web application for pagination and sorting of children details using RESTful APIs, implementing POST and GET operations.
2. Create a web application for managing Person details using JPA methods via RESTful APIs, enabling POST and GET operations.
3. Retrieve person details using JPQL with conditions for names starting or ending with specific patterns.
4. Build a web application for managing Person details using custom JPQL queries via RESTful APIs, supporting POST and GET operations.

UNIT IV	JPA Mapping	6+6
JPA Mapping of One-to-One Associations - fetching entities using queries - Loading optimization technique - Two-way One-to-One Relationship Mapping with JPA - single entity instance associated with multiple instances - Adding Data with One-to-One and One-to-Many Associations using JPA. List of Exercise/Experiments: 1. Develop a Spring Boot application with "Person" and "Address" entities, where each person has exactly one address. Utilize Spring JPA to establish a one-to-one mapping between these entities. 2. Create a Spring Boot application with "Author" and "Book" entities, where each author can have multiple books, and each book belongs to only one author. Use Spring JPA to establish a one-to-many bidirectional mapping between these entities. 3. Build a Spring Boot application with "Employee" and "Address" entities, ensuring that each employee has exactly one address, and each address belongs to only one employee. Establish a one-to-one mapping between these entities using Spring JPA and utilize the Criteria API to retrieve employee details efficiently		
UNIT V	Spring Boot Security, JWT Authentication, AOP and API Development Tools	6+6
API Documentation and Logging-Using Swagger UI and OpenAPI with Spring Boot to design and test APIs. Logging in Spring Boot, changing log levels, logging requests and responses, and managing logging configurations. Aspect-Oriented Programming (AOP)-Basic concepts of Aspect-Oriented Programming, handling method parameters, post-execution processing, and managing return values. API Security-Introduction to secure APIs using JSON Web Token. List of Exercise/Experiments: 1. Implement Logging using Log4j. 2. Implement Spring Security using JWT and OAuth2 3. Develop a web application for managing Employee and Payroll details via RESTful APIs. Utilize Spring JPA to establish a one-to-one mapping between Employee and Payroll entities. Demonstrate the usage of Swagger for API documentation and interaction. 4. Develop a Spring Boot application focused on handling person details and integrate comprehensive logging capabilities to track application activities effectively. 5. Explore the implementation of Aspect-Oriented Programming (AOP) in a Spring application to enhance the behavior of a service method and demonstrate its impact on application functionality.		
TOTAL: 60 PERIODS		
OUTCOMES: Upon completion of the course, the students will be able to: CO1: Create simple applications using RESTful APIs and effectively manage HTTP		

methods within the Spring Boot framework.

CO2: Apply database connectivity with JPA, utilizing advanced queries to interact with the database.

CO3: Build applications using Spring Boot and perform CRUD operations efficiently using JPQL

CO4: Demonstrate the implementation of various relational mappings in JPA, including one- to-one and one-to-many associations

CO5: Develop real-time applications that integrate user interfaces and utilize Spring AOP for method interception and advice handling.

CO6: Apply security measures to REST APIs using Spring Security and JWT to protect sensitive data and ensure secure communication between clients and servers.

TEXTBOOKS:

1. Raja CSP Raman, Ludovic Dewayly, "Building RESTful Web Services with Spring 5", Packt Publishing, 2018.
2. Leonard Richardson, Sam Ruby "RESTful Web Services" O'Reilly Media, 2008.
3. Ludovic Dewayly, "Building a RESTful Web Service with Spring: A hands-on guide to building an enterprise-grade, scalable RESTful web service using the Spring Framework", Packt Publishing, 2015
4. Raja CSP Raman, Ludovic Dewayly, "Building RESTful Web Services with Spring 5 - Second Edition
5. Edition: Leverage the power of Spring 5.0, Java SE 9, and Spring Boot 2.0", Packt Publishing, 2018

REFERENCES:

1. Ranga Karanam, "Master Java Web Services and REST API with Spring Boot", Packt Publishing, 2018.
2. Balaji Varanasi, Sudha Belida, "Spring REST", Apress, 2015.
3. Greg L. Turnquist, "Learning Spring Boot 2.0", Packt Publishing, 2021
4. Sourabh Sharma, "Modern API Development with Spring and Spring Boot", Packt Publishing, 2021

LIST OF SOFTWARES

1. VS Code, JRE, MySQL

CLOUD COMPUTING

24CS911	CLOUD FOUNDATIONS (Lab Integrated)	L	T	P	C
		2	0	2	3
OBJECTIVES: • To describe the different ways a user can interact with Cloud. • To discover the different compute options in Cloud and implement a variety of structured and unstructured storage models. • To confer the different application managed service options in the cloud and outline how security is administered in Cloud. • To demonstrate how to build secure networks in the cloud and identify cloud automation and management tools.					

- To determine a variety of managed big data services in the cloud.

UNIT I	INTRODUCTION TO CLOUD	6+6
---------------	------------------------------	------------

Introduction to Cloud Computing - Cloud Versus Traditional Architecture - IaaS, PaaS, and SaaS - Cloud Architecture - The Console - Understanding projects - Billing - Install and configure Cloud SDK - Use Cloud Shell - APIs - Cloud Console Mobile App.

List of Exercise/Experiments:

1. Install and configure cloud SDK.
2. Connect to computing resources hosted on Cloud via Cloud Shell.

UNIT II	COMPUTE AND STORAGE	6+6
----------------	----------------------------	------------

Compute options in the cloud - Exploring IaaS with Compute Engine - Configuring elastic apps with autoscaling - Exploring PaaS - Event driven programs - Containerizing and orchestrating apps - Storage options in the cloud - Structured and unstructured storage in the cloud - Unstructured storage using Cloud Storage - SQL managed services - NoSQL managed services.

List of Exercise/Experiments:

1. Create virtual machine instances of various machine types using the Cloud Console and the command line. Connect an NGINX web server to your virtual machine.
2. Create a small App Engine application that displays a short message.

UNIT III	APIs AND SECURITY IN THE CLOUD	6+6
-----------------	---------------------------------------	------------

The purpose of APIs - API Services - Managed message services - Introduction to security in the cloud - The shared security model - Encryption options - Authentication and authorization with Cloud IAM - Identify Best Practices for Authorization using Cloud IAM.

List of Exercise/Experiments:

1. Publish messages with managed message service using the Python client library.
2. Create a storage bucket, upload objects to it, create folders and subfolders in it, and make objects publicly accessible using the Cloud command line.

UNIT IV	NETWORKING, AUTOMATION AND MANGAEMENT TOOLS	6+6
----------------	--	------------

Introduction to networking in the cloud - Defining a Virtual Private Cloud - Public and private IP address basics - Cloud network architecture - Routes and firewall rules in the cloud - Multiple VPC networks - Building hybrid clouds using VPNs - Different options for load balancing - Introduction to Infrastructure as Code - Terraform - Monitoring and management tools.

List of Exercise/Experiments:

1. Create several VPC networks and VM instances and test connectivity across networks.
2. Create two managed instance groups in the same region. Then, configure and test an Internal Load Balancer with the instances groups as the backends.

UNIT V	BIG DATA AND MACHINE LEARNING SERVICES	6+6
---------------	---	------------

Introduction to big data managed services in the cloud - Leverage big data operations
 - Build Extract, Transform, and Load pipelines - Enterprise Data Warehouse Services -
 Introduction to machine learning in the cloud - Building bespoke machine learning
 models with AI Platform - Pre-trained machine learning APIs.

List of Exercise/Experiments:

1. Create a cluster, run a simple Apache Spark job in the cluster, then modify the number of workers in the cluster.
2. Set up your Python development environment, get the relevant SDK for Python, and run an example pipeline using the Cloud Console.

TOTAL: 30(T) + 30(P) = 60 PERIODS

OUTCOMES:

At the end of this course, the students will be able to:

CO1: Explain the basic concepts of cloud computing and the different cloud service models.

CO2: Deploy and manage various cloud compute and storage options.

CO3: Implement security measures in the cloud, including Cloud IAM for authentication and authorization.

CO4: Design and manage cloud networks and use automation tools like Terraform.

CO5: Use cloud monitoring tools to optimize cloud resource performance.

CO6: Leverage cloud-based big data and machine learning services to build data pipelines and models.

REFERENCES:

1. <https://cloud.google.com/docs>
2. <https://www.cloudskillsboost.google/paths/36>
3. <https://nptel.ac.in/courses/106105223>
4. Anthony J. Sequeira, "AWS Certified Cloud Practitioner (CLF-C01) Cert Guide", First Edition, Pearson Education, 2020.
5. [AWS Documentation \(amazon.com\)](https://aws.amazon.com/documentation/)
6. [AWS Skill Builder](https://aws.amazon.com/skillbuilder/)
7. AWS Academy Cloud Foundations Course - https://www.awsacademy.com/vforcesite/LMS_Login

LIST OF EQUIPMENTS:

GCP / CloudSkillBoost Platform/AWS Console /AWS Academy Learner Lab.

24CS912	VIRTUALIZATION	L	T	P	C
		3	0	0	3
OBJECTIVES:					
To explain the fundamental concepts of virtualization					

- To analyze the role of hypervisors in hardware virtualization
- To apply the understanding of CPU, memory (MMU), and I/O virtualization techniques
- To assess security considerations of virtualized environments
- To discuss strategies for protecting VMs and data centers

UNIT I	INTRODUCTION	9
---------------	---------------------	----------

Virtualization - Virtual Machines - Hypervisors - Type-1 and Type-2 Hypervisors - Multiplexing and Emulation - Approaches to Virtualization and Paravirtualization - Benefits of Using Virtual Machines. Working with Virtual Machines.

UNIT II	HARDWARE VIRTUALIZATION	9
----------------	--------------------------------	----------

The Popek/Goldberg Theorem - Virtualization without Architectural Support: Full Virtualization - Paravirtualization - Designs Options for Type-1 Hypervisors. Hypervisors: Describing a Hypervisor - Role of Hypervisor - VMWare ESX - Citrix Hypervisor - Microsoft Hyper-V.

UNIT III	TYPES OF VIRTUALIZATIONS	10
-----------------	---------------------------------	-----------

CPU Virtualization with VT-x: Design requirements - The VT-x Architecture - KVM. MMU Virtualization: Extended Paging - Virtualizing Memory in KVM. I/O Virtualization: Benefits of I/O Interposition - Physical I/O - Virtual I/O Without Hardware Support- Virtual I/O with Hardware Support. Virtualization Support in ARM Processors.

UNIT IV	VIRTUALIZATION SECURITY	9
----------------	--------------------------------	----------

Fundamentals of Virtualization Security: Virtualization Architecture - Threats to a Virtualized Environment. Securing Hypervisors: Hypervisor Configuration and Security. Designing Virtual Networks for Security: Comparing Virtual and Physical Networks - Virtual Network Security Considerations - Configuring Virtual Switches for Security.

UNIT V	VIRTUALIZATION AND AVAILABILITY	8
---------------	--	----------

Availability - Protecting a Virtual Machine - Protecting Multiple Virtual Machines - Protecting Datacenters - Deploying Applications in a Virtual Environment - Recent Trends in Virtualization.

TOTAL: 45 PERIODS

OUTCOMES:

At the end of this course, the students will be able to:

CO1: Understand the basics of virtualization and its benefits.

CO2: Assess the significance of hypervisors in hardware virtualization, examining their roles and implications for system efficiency and performance

CO3: Utilize knowledge of virtualization technologies to solve practical problems and implement effective solutions

CO4: Analyze security threats and design secure virtual networks

CO5: Discuss strategies to improve availability in virtual environment and for protecting VMs and data centers

CO6: Integrate virtualization concepts and technologies to design, implement, and

manage secure, efficient, and resilient virtualized environments.

TEXTBOOKS:

1. Edouard Bugnion, Jason Nieh, Dan Tsafir, "Hardware and Software Support for Virtualization", Morgan & Claypool Publishers, 2017.
2. Matthew Portnoy, "Virtualization Essentials", Third Edition, Sybex - John Wiley & Sons, 2023.
3. Dave Shackelford, "Virtualization Security: Protecting Virtualized Environments", Sybex - John Wiley & Sons, 2012.

REFERENCES:

1. Nelson Ruest, Danielle Ruest, Virtualization, A beginners guide, 2009, McGrawHill.
2. Nadeau, Tim Cerng, Je Buller, Chuck Enstall, Richard Ruiz, Mastering Microsoft Virtualization, Wiley Publication, 2010.
3. William Von Hagen, Professional Xen Virtualization, Wiley Publication, 2008.

24CS913	DATA ENGINEERING IN CLOUD	L	T	I	C
		3	0	0	3
OBJECTIVES: • To grasp the fundamentals of data engineering, emphasizing cloud-based data access. • To construct robust and secure data pipelines using AWS services. • To select and implement appropriate data storage solutions while prioritizing pipeline security. • To utilize cloud tools for handling extensive data for machine learning purposes. • To efficiently analyze, visualize, and automate data pipelines to streamline operations.					
UNIT I	INTRODUCTION				8
Introduction to data Engineering - The Data Engineering Life Cycle - Data Engineering and Data Science - Data-Driven Organizations: Data-driven decisions - The data pipeline - The role of the data engineer in data-driven organizations - Modern data strategies - The Elements of Data: The five Vs of data - volume, velocity, variety, veracity, and value. Demo: Accessing and Analyzing Data by Using Amazon S3.					
UNIT II	SECURE AND SCALABLE DATA PIPELINES				10
The evolution of data architectures - Modern data architecture on AWS - Modern data architecture pipeline: Ingestion and storage - Processing and consumption - Streaming analytics pipeline - Security of analytics workloads - Scaling - Creating a scalable infrastructure and components. ETL and ELT comparison - Data wrangling.					
UNIT III	STORING AND ORGANIZING DATA				9
Comparing batch and stream ingestion - Batch ingestion processing - Purpose-built ingestion tools - AWS Glue for batch ingestion processing - Kinesis for stream processing - Scaling considerations for batch processing and stream processing -					

Storage in the modern data architecture - Data lake storage - Data warehouse storage - Purpose-built databases - Storage in support of the pipeline - Securing storage.

UNIT IV	PROCESSING BIG DATA AND DATA FOR ML	10
----------------	--	-----------

Big data processing concepts - Apache Hadoop - Apache Spark - Amazon EMR - Managing your Amazon EMR clusters - Apache Hudi - The ML lifecycle - Collecting data - Applying labels to training data with known targets - Preprocessing data - Feature engineering - Developing a model - Deploying a model - ML infrastructure on AWS - SageMaker - Amazon CodeWhisperer - AI/ML services on AWS.

UNIT V	DATA ANALYSIS AND VISUALIZATION	8
---------------	--	----------

Analyzing and Visualizing Data: Considering factors that influence tool selection - Comparing AWS tools and services - Selecting tools for a gaming analytics use case. Automating the Pipeline: Automating infrastructure deployment - CI/CD - Automating with Step Functions.

TOTAL: 45 PERIODS

OUTCOMES:

At the end of this course, the students will be able to:

CO1: Understand data engineering, pipelines & access data in the cloud.

CO2: Build secure & scalable data pipelines using AWS services.

CO3: Choose the right data storage & secure your data pipelines.

CO4: Process big data for machine learning with cloud tools.

CO5: Analyze & visualize data and automate data pipelines.

CO6: Apply data engineering techniques and cloud tools to design comprehensive data solutions that meet business needs effectively.

REFERENCES:

1. Joe Reis, Matt Housley, "Fundamentals of Data Engineering", O'Reilly Media, 2012.
2. Wes McKinney, "Python for Data Analysis", 2nd Edition, O'Reilly Media, 2017.
3. Martin Kleppman, "Designing Data-Intensive Applications", O'Reilly Media, 2017.
4. AWS Documentation (amazon.com) - <https://docs.aws.amazon.com/>
5. AWS Skill Builder - <https://skillbuilder.aws/>
6. AWS Academy Data Engineering Course - https://www.awsacademy.com/vforcesite/LMS_Login

24CS914	DEVOPS (Lab Integrated)	L	T	P	C
		2	0	2	3

OBJECTIVES:

- Bridge the gap between development and operations for faster, more reliable software releases.
- Automate software delivery with CI/CD pipelines.
- Package and deploy apps efficiently using Docker containers.
- Automate infrastructure with Infrastructure as Code (IaC).
- Monitor and troubleshoot applications in production.

UNIT I	INTRODUCTION TO DEVOPS	6+6
---------------	-------------------------------	------------

Software Development Methodologies - Operations Methodologies - Systems Methodologies - Development, Release, and Deployment Concepts - Infrastructure Concepts. What is DevOps? - DevOps importance and benefits -DevOps principles and practices - 7 C's of DevOps lifecycle for business agility - DevOps and continuous testing. How to choose right DevOps tools? - Challenges with DevOps implementation.

List of Exercise/Experiments:

1. Install Git and Jenkins on your system and verify the installation using version commands.
2. Install Docker and run a sample container to confirm successful setup.

UNIT II	VERSION CONTROL WITH GIT	6+6
----------------	---------------------------------	------------

Introduction to Git version control system - Git commands for basic operations (clone, commit, push, pull) - Branching and merging strategies - Collaboration using Git workflows.

List of Exercise/Experiments:

1. Initialize a local repository using Git, add a sample file, commit changes, and view commit history.
2. Create a remote repository, push local code to it, and clone the repository to another directory/system.

UNIT III	CONTINUOUS INTEGRATION AND DELIVERY (CI/CD)	6+6
-----------------	--	------------

Introduction to CI/CD pipelines - Benefits of CI/CD for faster deployments - Setting up a CI/CD pipeline with Jenkins - Automating builds, tests, and deployments.

List of Exercise/Experiments:

1. Using Git and Jenkins, implement a CI/CD pipeline to fetch code from a repository, build it, and deploy it on a local HTTP server.
2. Extend the above pipeline by creating a Jenkins job that executes a custom script during the build stage and verifies the output in the console.

UNIT IV	CONTAINERIZATION WITH DOCKER	6+6
----------------	-------------------------------------	------------

Introduction to containerization and its benefits - Understanding Docker concepts: images, containers, registries - Building and managing Docker containers - Docker Compose for multi-container applications - Introduction to container orchestration with Docker Swarm or Kubernetes.

List of Exercise/Experiments:

1. Containerize the deployed application using Docker and run it as a containerized service.
2. Deploy and manage the containerized application using Kubernetes, ensuring the application is accessible.

UNIT V	INFRASTRUCTURE AS CODE (IAC) AND MONITORING	6+6
---------------	--	------------

Introduction to Infrastructure as Code (IaC) - Benefits of using IaC for repeatable infrastructure provisioning - Learning IaC with Terraform - Setting up infrastructure configurations with Terraform - Introduction to monitoring and logging tools for applications - Alerting and troubleshooting techniques.

List of Exercise/Experiments:

1. Automate infrastructure provisioning for the application using Terraform and integrate it into the CI/CD pipeline.
2. Build a complete DevOps pipeline integrating Git, Jenkins, Docker, Kubernetes, and Terraform to automate code integration, deployment, and infrastructure setup.

TOTAL: 30(T) + 30(P) = 60 PERIODS

OUTCOMES:

At the end of this course, the students will be able to:

CO1: Understand the core principles and philosophies of DevOps.

CO2: Implement version control systems for code management and collaboration.

CO3: Automate software delivery pipelines using CI/CD tools.

CO4: Utilize containerization technologies for packaging and deploying applications.

CO5: Configure infrastructure as code (IaC) for repeatable deployments.

CO6: Monitor and maintain applications in a production environment.

TEXT BOOKS:

1. Deepak Gaikwad, Viral Thakkar, "DevOps Tools: from Practitioner's Point of View", Wiley, 2019.
2. Jennifer Davis, Ryn Daniels, "Effective DevOps", O'Reilly Media, 2016.

REFERENCES:

1. Gene Kim, Jez Humble, Patrick Debois, "The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations", IT Revolution Press, 2016.
2. Jez Humble, Gene Kim, "Continuous Delivery: Reliable Software Releases Through Build, Test, and Deployment Automation", Addison-Wesley, 2010.
3. Yevgeniy Brikman, "Terraform: Up & Running: Writing Infrastructure as Code", O'Reilly Media, 2019.
4. Joseph Muli, "Beginning DevOps with Docker", Packt Publishing, 2018.

24CS942	CLOUD ARCHITECTING	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners:

- To make architectural decisions based on AWS architectural principles and best practices.
- To describe the features and benefits of Amazon EC2 instances, and compare and contrast managed and unmanaged database services.
- To create a secure and scalable AWS network environment with VPC, and configure IAM for improved security and efficiency.

- To use AWS services to make infrastructure scalable, reliable, and highly available.
- To use AWS managed services to enable greater flexibility and resiliency in an infrastructure.

UNIT I	INTRODUCING CLOUD ARCHITECTING AND STORAGE LAYER	9
---------------	---	----------

Cloud architecting - The AWS Well-Architected Framework - AWS global infrastructure - Amazon S3 - Amazon S3 Versioning - Storing data in Amazon S3 - Moving data to and from Amazon S3 - Amazon S3 Transfer Acceleration - Choosing Regions for your architecture.

UNIT II	COMPUTE LAYER AND DATABASE LAYER	9
----------------	---	----------

Adding compute with Amazon EC2 - Choosing an Amazon Machine Image (AMI) to launch an Amazon EC2 instance - Selecting an Amazon EC2 instance type - Using user data to configure an EC2 instance - Adding storage to an Amazon EC2 instance - Amazon EC2 pricing options - Amazon EC2 considerations - Database layer considerations - Amazon Relational Database Service (Amazon RDS) - Amazon DynamoDB - Database security controls - Migrating data into AWS databases.

UNIT III	CREATING AND CONNECTING NETWORKS	9
-----------------	---	----------

Creating an AWS networking environment - Connecting your AWS networking environment to the internet - Securing your AWS networking environment - Connecting your remote network with AWS Site-to-Site VPN - Connecting your remote network with AWS Direct Connect - Connecting virtual private clouds (VPCs) in AWS with VPC peering - Scaling your VPC network with AWS Transit Gateway - AWS Transit Gateway - Connecting your VPC to supported AWS services. Securing User and Application Access: Account users and AWS Identity and Access Management (IAM) - Organizing users - Federating users - Multiple accounts.

UNIT IV	RESILIENT CLOUD ARCHITECTURE	9
----------------	-------------------------------------	----------

Scaling your compute resources - Scaling your databases - Designing an environment that's highly available - Monitoring - Reasons to automate - Automating your infrastructure - Automating deployments - AWS Elastic Beanstalk - Overview of caching - Edge caching - Caching web sessions - Caching databases.

UNIT V	BUILDING DECOUPLED ARCHITECTURES, MICROSERVICES AND SERVERLESS ARCHITECTURE	9
---------------	--	----------

Decoupling your architecture - Decoupling with Amazon Simple Queue Service (Amazon SQS) - Decoupling with Amazon Simple Notification Service (Amazon SNS) - Sending messages between cloud applications and on-premises with Amazon MQ. Introducing microservices - Building microservice applications with AWS container services - Introducing serverless architectures - Building serverless architectures with AWS Lambda - Extending serverless architectures with Amazon API Gateway - Orchestrating microservices with AWS Step Functions - Disaster planning strategies - Disaster recover patterns.

TOTAL: 45 PERIODS

OUTCOMES:

Upon Completion of this course, the students will be able to:

- CO1:** Explain cloud architecture principles and AWS storage solutions.
- CO2:** Deploy and manage AWS compute and database resources securely.
- CO3:** Design and configure secure AWS networks using VPC and IAM.
- CO4:** Implement scalable and resilient AWS architectures with high availability.
- CO5:** Build decoupled and serverless applications using AWS services like

Lambda.

CO6: Develop disaster recovery strategies for AWS environments.

REFERENCES:

1. AWS Certified Solutions Architect Official Study Guide by Joe Baron, Hisham Baz, Tim Bixler
2. Architecting the Cloud by Michael Kavis.
3. [AWS Documentation \(amazon.com\)](https://aws.amazon.com/documentation/)
4. [AWS Skill Builder](https://aws.amazon.com/skillbuilder/)
5. AWS Academy Cloud Architecting Course - [| 24CS935 | AGENTIC AI IN CLOUD | L | T | P | C |
|---|-----------------------------------|----------|---|---|---|
| | | 3 | 0 | 0 | 3 |
| Objectives:

The objective of learning this Agentic AI course is to: <ul style="list-style-type: none">• Learn the basics, types, applications, and core principles of agentic systems.• Understand reflection, planning, tool usage, and practical agent frameworks.• Develop skills to build and manage multi-agent systems using structured approaches like CWD.• Evaluate and implement multi-agent orchestration frameworks.• Architect secure, protocol-driven agentic ecosystems in the cloud. | | | | | |
| UNIT I | INTRODUCTION TO AGENTIC AI | 9 | | | |
| Generative AI - Types of Generative AI Models - Applications of Generative AI - Challenges of Generative AI - Principles of Agentic Systems - Understanding Self-governance, Agency, Autonomy - Intelligent Systems and characteristics - Architecture of Agentic Systems - Understanding Multi Agent Systems. | | | | | |
| UNIT II | GENERATIVE AI BASED AGENTS | 9 | | | |
| Reflection and Introspection in Agents - Importance of reflection in agents - Introspection in intelligent agents - Implementing reflective capabilities - Use cases and examples - Enabling tool use and planning in agents - Understanding the concept of tool use in agents - Planning algorithms for agents - Integrating tool use and planning - Exploring practical implementations. | | | | | |
| UNIT III | AGENTIC AI DESIGN | 9 | | | |
| Exploring Coordinator, Worker and Delegator \(CWD\) Approach - Understanding the CWD Model - Designing agents with role assignments - Communication and collaboration between agents - Implementing CWD approach in generative AI systems - Focused system prompts and instructions for agents - State spaces and environment modeling - Agent memory architecture and context management - Sequential and parallel processing in agentic workflows. | | | | | |
| UNIT IV | AGENTIC AI FRAMEWORKS | 9 | | | |
| Strands Agents - LangChain and LangGraph - CrewAI - AutoGen - LlamaIndex - | | | | | |](https://www.awsacademy.com/vforcesite/LMS>Login</div><div data-bbox=)

Comparing agentic AI frameworks - Considerations in choosing an agentic AI framework.

UNIT V	AGENTIC AI PLATFORMS, PROTOCOLS & TOOLS	9
---------------	--	----------

Types of agentic AI platforms - Amazon Bedrock Agents - Amazon Bedrock AgentCore - Protocols - Agent-to-agent protocols - Selecting agentic protocols - Model Context Protocol (MCP) - Tools - Tool categories - Protocol-based tools - Framework-native tools - Meta-tools - Tool integration strategy - Security best practices for tool integration.

TOTAL: 45 PERIODS

OUTCOMES:

At the end of this course, the students will be able to:

CO1: Describe the fundamentals of Generative AI and Agentic AI concepts

CO2: Identify different types of generative models and their applications.

CO3: Apply reflection, planning, and tool usage in designing intelligent agents.

CO4: Design agentic systems using architectures like multi-agent systems and CWD approach.

CO5: Analyze and select appropriate Agentic AI frameworks.

CO6: Design and deploy secure cloud-native agentic platforms.

TEXTBOOKS:

1. **Biswas, A. and Talukdar, W.,** Building Agentic AI Systems: Create Intelligent, Autonomous AI Agents that Can Reason, Plan, and Adapt, 1st Edition, 2025.
2. **Ozdemir, S.,** Building Agentic AI: Workflows, Fine-Tuning, Optimization, and Deployment, 1st Edition, 2025.

REFERENCES:

1. [AWS Prescriptive Guidance - Agentic AI frameworks, platforms, protocols, and tools on AWS](#)
2. [Agentic AI: Theories and Practices | Springer Nature Link](#)
3. Yoav Shoham & Kevin Leyton-Brown, Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations
4. Russell, S., & Norvig, P. (2021). Artificial Intelligence: A Modern Approach (4th Edition). Pearson.
<https://www.ibm.com/think/>

24IT903	REST APPLICATION DEVELOPMENT USING SPRING BOOT AND JPA (Lab Integrated)	L	T	P	C
		2	0	2	3

OBJECTIVES:

The Course will enable learners to:

- To provide comprehensive knowledge of RESTful APIs and the HTTP methods used in the Spring Boot framework.
- To cover advanced querying techniques using JPA, including LIKE queries, and to manage CRUD operations using JPQL.
- To explore various relational mappings in JPA, such as one-to-one and one-to-many associations, and their practical implementations.
- To implement and manage Spring AOP applications using annotation-based configurations for method interception and

post-execution operations.

- To build production-grade Spring Boot applications with integrated security using JWT, detailed API documentation with SwaggerUI and OpenUI, and effective logging practices.

UNIT I

Server-Side Programming and REST APIs

6+6

Introduction to server-side programming and client-server architecture. Fundamentals of RESTful web services and API design. Overview of HTTP methods, status codes, and data formats such as JSON. Introduction to the Spring Framework and its features. Basics of dependency injection and Spring Boot application development.

List of Exercise/Experiments:

1. Create a basic application using Spring Framework. (Create a simple Spring project. Define a bean Retrieve bean from container).
2. Develop a RESTful API for retrieving a welcome message, emphasizing the basics of data exchange between client and server.
3. Implement a RESTful API to acknowledge the user's favorite color choice, highlighting property value injection principles.
4. Create a Spring Boot application that retrieves and displays application information, demonstrating the usage of the @Value annotation to inject property values from the application configuration file.
5. Construct a RESTful API for student details retrieval, illustrating the utilization of @JsonIgnore annotation, focusing on advanced JSON property handling and data access control.

UNIT II

Database operations

6+6

Database Operations & Spring MVC: MYSQL integration, mapping Java classes to relational database, repository interface, retrieving data, request body mapping.

Spring MVC & REST Integration: Spring MVC architecture, building REST services using Spring, HTTP methods (GET, POST, PUT, DELETE), error handling, logging with Log4J.

List of Exercise/Experiments:

1. Develop a web application for managing patient details using RESTful APIs, implementing POST and GET operations.
2. Create a web application for managing product details using RESTful APIs, enabling POST and GET operations.
3. Build an application for managing employee details using RESTful APIs, supporting POST, PUT, and DELETE operations.
4. Performing CRUD operations using Spring MVC and RESTful services.

UNIT III

Java Persistence API

6+6

Pagination & Sorting using JPA, @Transient Annotation, Queries using JPA, Starts and Ends with query using JPA, JPQL with @Query Annotation, custom JPQL queries.

List of Exercise/Experiments

1. Develop a web application for pagination and sorting of children details using RESTful APIs, implementing POST and GET operations.

2. Create a web application for managing Person details using JPA methods via RESTful APIs, enabling POST and GET operations. 3. Retrieve person details using JPQL with conditions for names starting or ending with specific patterns. 4. Build a web application for managing Person details using custom JPQL queries via RESTful APIs, supporting POST and GET operations.		
UNIT IV	JPA Mapping	6+6
JPA Mapping of One-to-One Associations - fetching entities using queries - Loading optimization technique - Two-way One-to-One Relationship Mapping with JPA - single entity instance associated with multiple instances - Adding Data with One-to-One and One-to-Many Associations using JPA. List of Exercise/Experiments: 1. Develop a Spring Boot application with "Person" and "Address" entities, where each person has exactly one address. Utilize Spring JPA to establish a one-to-one mapping between these entities. 2. Create a Spring Boot application with "Author" and "Book" entities, where each author can have multiple books, and each book belongs to only one author. Use Spring JPA to establish a one-to-many bidirectional mapping between these entities. 3. Build a Spring Boot application with "Employee" and "Address" entities, ensuring that each employee has exactly one address, and each address belongs to only one employee. Establish a one-to-one mapping between these entities using Spring JPA and utilize the Criteria API to retrieve employee details efficiently		
UNIT V	Spring Boot Security, JWT Authentication, AOP and API Development Tools	6+6
API Documentation and Logging-Using Swagger UI and OpenAPI with Spring Boot to design and test APIs. Logging in Spring Boot, changing log levels, logging requests and responses, and managing logging configurations. Aspect-Oriented Programming (AOP)-Basic concepts of Aspect-Oriented Programming, handling method parameters, post-execution processing, and managing return values. API Security-Introduction to secure APIs using JSON Web Token. List of Exercise/Experiments: 1. Implement Logging using Log4j. 2. Implement Spring Security using JWT and OAuth2 3. Develop a web application for managing Employee and Payroll details via RESTful APIs. Utilize Spring JPA to establish a one-to-one mapping between Employee and Payroll entities. Demonstrate the usage of Swagger for API documentation and interaction. 4. Develop a Spring Boot application focused on handling person details and integrate comprehensive logging capabilities to track application activities effectively. 5. Explore the implementation of Aspect-Oriented Programming (AOP) in a Spring application to enhance the behavior of a service method and demonstrate its impact on application functionality.		
TOTAL: 60 PERIODS		

OUTCOMES:

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

CO1: Create simple applications using RESTful APIs and effectively manage HTTP methods within the Spring Boot framework.

CO2: Apply database connectivity with JPA, utilizing advanced queries to interact with the database.

CO3: Build applications using Spring Boot and perform CRUD operations efficiently using JPQL

CO4: Demonstrate the implementation of various relational mappings in JPA, including one- to-one and one-to-many associations

CO5: Develop real-time applications that integrate user interfaces and utilize Spring AOP for method interception and advice handling.

CO6: Apply security measures to REST APIs using Spring Security and JWT to protect sensitive data and ensure secure communication between clients and servers.

TEXTBOOKS:

1. Raja CSP Raman, Ludovic Dewayilly, "Building RESTful Web Services with Spring 5", Packt Publishing, 2018.
2. Leonard Richardson, Sam Ruby "RESTful Web Services" O'Reilly Media, 2008.
3. Ludovic Dewayilly, "Building a RESTful Web Service with Spring: A hands-on guide to building an enterprise-grade, scalable RESTful web service using the Spring Framework", Packt Publishing, 2015
4. Raja CSP Raman, Ludovic Dewayilly, "Building RESTful Web Services with Spring 5 - Second Edition
5. Edition: Leverage the power of Spring 5.0, Java SE 9, and Spring Boot 2.0", Packt Publishing, 2018

REFERENCES:

1. Ranga Karanam, "Master Java Web Services and REST API with Spring Boot", Packt Publishing, 2018.
2. Balaji Varanasi, Sudha Belida, "Spring REST", Apress, 2015.
3. Greg L. Turnquist, "Learning Spring Boot 2.0", Packt Publishing, 2021
4. Sourabh Sharma, "Modern API Development with Spring and Spring Boot", Packt Publishing, 2021

LIST OF SOFTWARES

1. VS Code, JRE, MySQL

FULL STACK TECHNOLOGY

24CS921	UI/UX DESIGN (Lab Integrated)	L	T	P	C
		2	0	2	3

OBJECTIVES:

The Course will enable learners:

- Explain the principles of User Interface (UI) in order to do design with intention.
- Define the User experience (UX) and the psychology behind user decision making.
- Discuss about UX process and user Psychology.
- Apply technology for designing web applications with multimedia effects.
- Create a wireframe and prototype.

UNIT I	INTRODUCTION TO UI	6+6
Introduction to UI - Designing Behavior: Designing with Intention - Conditioning and Addiction - Timing Matters - Gamification - Social/Viral Structure-Trust - Hidden versus Visible. Basic Visual Design Principles: Visual Weight - Contrast - Depth and Size - Color- Layout: Page Framework - Footers - Navigation -Images, and Headlines - Forms - Input Types - Labels and Instructions - Primary and Secondary Buttons - Adaptive and Responsive Design - Touch versus Mouse. List of Exercise/Experiments: 1. Design UI for a Game website. 2. Design one-page UI for a website.		
UNIT II	USER OBSERVATION AND EXPERIENCE	6+6
User Research - Subjective Research - Objective Research - Sample size - Three Basic Types of Questions. Observe a user: Watch How They Choose - Interviews - Surveys - Card Sorting - Creating User Profiles - Bad profile - Useful profile. List of Exercise/Experiments: 1. Design UI for a mobile. 2. Explore the Look and Feel of the new Project developed in Ex1.		
UNIT III	INTRODUCTION TO UX	6+6
Introduction about UX - Five Main Ingredients of UX - Three “Whats” of user Perspective - Pyramid of UX Impact - UX Is a Process - UX - Not an Event or Task. Behaviour Basics: Psychology versus Culture - User Psychology - Experience - Conscious vs Subconscious Experience - Emotions - Gain and Loss - Motivations. List of Exercise/Experiments: 1. Design a mascot for an imaginary brand. 2. Create a Sample Pattern Library for a product (Mood board, Fonts, Colors based on UI principles).		
UNIT IV	WEB INTERFACE DESIGN	6+6
Designing Web Interfaces - Drag and Drop, Direct Selection, Contextual Tools, Overlays, Inlays and Virtual Pages, Process Flow - Using Motion for UX - Design Pattern: Z-Pattern - F- Pattern - Visual Hierarchy - Lookup patterns - Feedback patterns. List of Exercise/Experiments: 1. Design a mock-up website for a service sector company. 2. Create a brainstorming feature for proposed product.		
UNIT V	WIREFRAMING, PROTOTYPING AND TESTING	6+6
Sketching Principles - Sketching Red Routes - Responsive Design - Wireframing - Creating Wire flows - Building a Prototype - Building High-Fidelity Mock-ups - Designing Efficiently with Tools - Interaction Patterns - Conducting Usability Tests - Other Evaluative User Research Methods - Synthesizing Test Findings - Prototype Iteration List of Exercise/Experiments: 1. Sketch, design with popular tool and build a prototype and perform usability testing and identify improvements. 2. Design a mobile mock-up website for an online store.		
TOTAL: 30(T) + 30(P) = 60 PERIODS		
OUTCOMES: Upon completion of the course, the students will be able to: CO1: Create visually appealing and functional interfaces that enhance user interaction. CO2: Ensure products are intuitive, accessible, and meet user needs. CO3: Build and test design concepts to optimize user experience. CO4: Evaluate and refine designs based on user feedback. CO5: Structure content effectively for intuitive navigation.		

CO6: Design engaging interactions that improve usability.
TEXT BOOKS: 1. Joel Marsh, "UX for Beginners", O'Reilly Media, Inc., 1st Edition 2015. 2. Xia Jiajia, "UI UX Design", O'Reilly, Artpower International, 2016. 3. Jenifer Tidwell, Charles Brewer, Aynne Valencia, "Designing Interface" 3rd Edition, O'Reilly 2020
REFERENCES: 1. Jenifer Tidwell, Charles Brewer, Aynne Valencia, "Designing Interface" 3rd Edition, O'Reilly 2020.
SOFTWARE REQUIREMENTS: Javascript, Applets, Equivalent Frontend tools, MySQL, Figma or equivalent.

24CS903	BLOCKCHAIN TECHNOLOGY	L	T	P	C
		3	0	0	3
OBJECTIVES: The Course will enable learners to:					
- To understand block chain system’s fundamental components, how they fit together and examine a decentralization using block chain. - To explain how Crypto currency works. - To explain the components of Ethereum and Programming Languages for Ethereum					
- To study the basics of Web3 and Hyper ledger. - To give an insight of alternative block chains and its emerging trends.					
UNIT I	INTRODUCTION TO BLOCKCHAIN				9
History of Blockchain - Types of Blockchain - Consensus - Decentralization using Blockchain - Blockchain and Full Ecosystem Decentralization - Platforms for Decentralization - Symmetric Cryptography - Mathematics - Asymmetric Cryptography - public and private keys - Elliptic curve cryptography - Discrete logarithm problem in ECC					
UNIT II	INTRODUCTION TO CRYPTOCURRENCY				9
Bitcoin - Digital Keys and Addresses - Transactions - Mining - Bitcoin Networks and Payments Wallets - innovation in Bitcoin - Alternative Coins - Theoretical Foundations - Bitcoin.					
UNIT III	ETHEREUM				9
The Ethereum Network - Components of Ethereum Ecosystem - Ethereum Programming Languages: Runtime Byte Code - Blocks and Blockchain - Fee Schedule - Supporting Protocols - Solidity Language.					
UNIT IV	WEB3 AND HYPERLEDGER				9
Introduction to Web3 - Contract Deployment - POST Requests - Development frameworks Hyperledger as a protocol - The Reference Architecture - Hyperledger Fabric - Distributed Ledger.					
UNIT V	ALTERNATIVE BLOCKCHAINS AND NEXT EMERGING TRENDS				9
Kadena - Ripple- Rootstock - Quorum - Tendermint - Scalability - Privacy - Other Challenges - Blockchain Research. Case Study - Install IPFS locally on our machine, initialize your node, view the nodes in network.					
TOTAL:45 PERIODS					

OUTCOMES:
At the end of this course, the students will be able to:
CO1: Understand the fundamentals and history of blockchain technology.
CO2: Analyze different types of blockchain and their consensus mechanisms.
CO3: Apply cryptographic principles to blockchain systems.
CO4: Explore cryptocurrency concepts, including Bitcoin and alternative coins.
CO5: Evaluate Ethereum and its programming languages for decentralized applications.
CO6: Investigate other blockchain platforms like Hyperledger and emerging trends.

TEXTBOOKS:
1. Imran Bashir, "Mastering Blockchain: Distributed Ledger Technology, decentralization, and smart contracts explained", 2nd Edition, Packt Publishing Ltd, March 2018.
2. BellajBadr, Richard Horrocks, Xun (Brian) Wu, "Blockchain By Example: A developer's guide to creating decentralized applications using Bitcoin, Ethereum, and Hyperledger", Packt Publishing Limited, 2018.

REFERENCES:
1. Andreas M. Antonopoulos, "Mastering Bitcoin: Unlocking Digital Cryptocurrencies", O'Reilly Media Inc, 2015
2. Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller and Steven Goldfeder, "Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction", Princeton University Press, 2016.

24CS914	DEVOPS	L	T	P	C
	(Lab Integrated)	2	0	2	3

OBJECTIVES:
• Bridge the gap between development and operations for faster, more reliable software releases. • Automate software delivery with CI/CD pipelines. • Package and deploy apps efficiently using Docker containers. • Automate infrastructure with Infrastructure as Code (IaC). • Monitor and troubleshoot applications in production.

UNIT I	INTRODUCTION TO DEVOPS	6+6
---------------	-------------------------------	------------

Software Development Methodologies - Operations Methodologies - Systems Methodologies - Development, Release, and Deployment Concepts - Infrastructure Concepts. What is DevOps? - DevOps importance and benefits -DevOps principles and practices - 7 C's of DevOps lifecycle for business agility - DevOps and continuous testing. How to choose right DevOps tools? - Challenges with DevOps implementation.

List of Exercise/Experiments:

1. Install Git and Jenkins on your system and verify the installation using version commands.
2. Install Docker and run a sample container to confirm successful setup.

UNIT II	VERSION CONTROL WITH GIT	6+6
----------------	---------------------------------	------------

Introduction to Git version control system - Git commands for basic operations (clone, commit, push, pull) - Branching and merging strategies - Collaboration using Git workflows.

List of Exercise/Experiments:

1. Initialize a local repository using Git, add a sample file, commit changes, and view commit history.
2. Create a remote repository, push local code to it, and clone the repository to another directory/system.

UNIT III	CONTINUOUS INTEGRATION AND DELIVERY (CI/CD)	6+6
-----------------	--	------------

Introduction to CI/CD pipelines - Benefits of CI/CD for faster deployments - Setting up a CI/CD pipeline with Jenkins - Automating builds, tests, and deployments.

List of Exercise/Experiments:

1. Using Git and Jenkins, implement a CI/CD pipeline to fetch code from a repository, build it, and deploy it on a local HTTP server.
2. Extend the above pipeline by creating a Jenkins job that executes a custom script during the build stage and verifies the output in the console.

UNIT IV	CONTAINERIZATION WITH DOCKER	6+6
----------------	-------------------------------------	------------

Introduction to containerization and its benefits - Understanding Docker concepts: images, containers, registries - Building and managing Docker containers - Docker Compose for multi-container applications - Introduction to container orchestration with Docker Swarm or Kubernetes.

List of Exercise/Experiments:

1. Containerize the deployed application using Docker and run it as a containerized service.
2. Deploy and manage the containerized application using Kubernetes, ensuring the application is accessible.

UNIT V	INFRASTRUCTURE AS CODE (IAC) AND MONITORING	6+6
---------------	--	------------

Introduction to Infrastructure as Code (IaC) - Benefits of using IaC for repeatable infrastructure provisioning - Learning IaC with Terraform - Setting up infrastructure configurations with Terraform - Introduction to monitoring and logging tools for applications - Alerting and troubleshooting techniques.

List of Exercise/Experiments:

1. Automate infrastructure provisioning for the application using Terraform and integrate it into the CI/CD pipeline.
2. Build a complete DevOps pipeline integrating Git, Jenkins, Docker, Kubernetes, and Terraform to automate code integration, deployment, and infrastructure setup.

TOTAL: 30(T) + 30(P) = 60 PERIODS

OUTCOMES:

At the end of this course, the students will be able to:

CO1: Understand the core principles and philosophies of DevOps.

CO2: Implement version control systems for code management and collaboration.

CO3: Automate software delivery pipelines using CI/CD tools.

CO4: Utilize containerization technologies for packaging and deploying applications.

CO5: Configure infrastructure as code (IaC) for repeatable deployments.

CO6: Monitor and maintain applications in a production environment.

TEXT BOOKS:	
1. Deepak Gaikwad, Viral Thakkar, "DevOps Tools: from Practitioner's Point of View", Wiley, 2019.	
2. Jennifer Davis, Ryn Daniels, "Effective DevOps", O'Reilly Media, 2016.	
REFERENCES:	
1. Gene Kim, Jez Humble, Patrick Debois, "The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations", IT Revolution Press, 2016.	
2. Jez Humble, Gene Kim, "Continuous Delivery: Reliable Software Releases Through Build, Test, and Deployment Automation", Addison-Wesley, 2010.	
3. Yevgeniy Brikman, "Terraform: Up & Running: Writing Infrastructure as Code", O'Reilly Media, 2019.	
4. Joseph Muli, "Beginning DevOps with Docker", Packt Publishing, 2018.	

24CS923	USABILITY DESIGN OF SOFTWARE APPLICATION	L	T	P	C
		3	0	0	3
OBJECTIVES:					
The Course will enable learners:					
• To create a learning system through which management students can enhance their innovation. • To acquaint themselves with the special challenges of starting new ventures. • To use IPR as an effective tool to protect their innovations and intangible assets from exploitation. • To develop a learning system that enables creative thinking skills.					
UNIT I	INTRODUCTION TO USER CENTERED DESIGN AND ASPECT				9
Basics of User Centered Design- Product Appreciation Assignment - Evaluating the product from user centered design aspects such as functionality, ease of use, ergonomics, and aesthetics. Case Study: Identify a website or an App to redesign, with justification.					
UNIT II	HEURISTIC EVALUATION AND PROJECT DESIGN LIFECYCLE				9
10 Heuristic Principles, Examples Heuristic Evaluation: Group Assignment initiation (Website and App) Evaluation for key tasks of the app or website for heuristic principles, severity, recommendations. Case Study: Analyze the mobile app or website throughout the design life cycle.					
UNIT III	PROJECT DESIGN LIFECYCLE				9
Redesign project through the design lifecycle - Discovery - Define - Design - Implement (Design Prototype) - Usability Testing. Case Study: Identify personas and scenarios for the app or website					
UNIT IV	WEB INTERFACE DESIGN				9
Understanding users, their goals, context of use, and environment of use. Research Techniques: Contextual Enquiry, User Interviews, Competitive Analysis for UX. Scenarios and Persona Technique -Overview of Design Thinking Technique - Discovery and brainstorming.					
UNIT V	DEVELOPMENT AND PROTOTYPING				9
Concept Development - Task flow detailing for the Project - Prototyping Techniques - Paper, Electronic, and Prototyping Tools. Case Study: Prototype development with iterations and justification.					

OUTCOMES:

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

- CO1:** Sensitize the students to the fundamentals of User Centred Design and User Experiences their relevance and contribution to businesses.
- CO2:** Familiarize them to the facets of User Experience (UX) Design, particularly as applied to the digital artefacts.
- CO3:** Appreciation of user research, solution conceptualization and validation as interwoven activities in the design and development lifecycle.
- CO4:** Acquire the ability to constructively engage with the Design professionals they would work with in the future.
- CO5:** Analyze and identify the methods to offer a better UI experience for the applications Gain expertise in redesigning an existing Application or website for better user experience.
- CO6:** Develop the capability to integrate user feedback and usability testing results into iterative design processes, ensuring continuous improvement and alignment with user needs.

TEXT BOOKS:

1. Jennifer Preece, Helen Sharp, Yvonne Rogers, "Interaction Design: Beyond Human-Computer Interaction", 2015, 4th Edition, Wiley publications.
2. Bruce Ferwerda, Mark Graus, Marko Tkalcic, Panagiotis Germanakos, A Human-centered Perspective of Intelligent Personalized Environments and Systems, 2024, Springer.
3. Steve Krug, "Don't Make Me Think, Revisited: A Common Sense Approach to Web Usability" 2013, 3rd Edition, New Riders.

REFERENCES:

1. Alan Cooper and Robert Riemann, "About Face The Essentials of Interaction Design", 2014, 4th Edition, Wiley Publications.
2. Elizabeth Goodman, Mike Kuniavsky, Andrea Moed, "Observing the User Experience - A Practitioner's Guide to User Research", 2012, Second Edition, Morgan Kaufmann Publications.
3. David Benyon, "Designing Interactive Systems- A Comprehensive Guide to UX and Interaction Design", 2013, Pearson Publications.

24CS944	AI INTEGRATION IN SOFTWARE DEVELOPMENT AND OPERATIONS	L	T	P	C
		3	0	0	3
Objectives: • To understand fundamentals of AI, ML, LLMs, and their applications in software engineering. • To apply AI techniques in the Software Development Life Cycle for planning, requirements, and project management. • To utilize Generative AI in software design, development, and CI/CD pipeline optimization. • To evaluate AI-driven methods for software testing and quality assurance. • To explore AI applications in continuous delivery, deployment, operations, and database management.					

UNIT I	INTRODUCTION	9
Artificial Intelligence - DevOps - Understanding Machine Learning - Construct of AI and ML - AI Techniques and Algorithms - AI Techniques and Algorithms - Large Language Models (LLM)		
UNIT II	SOFTWARE DEVELOPMENT AND AI AUGMENTATION	9
Use of AI in SDLC - Overview of the Software Delivery Lifecycle and AI Infusion - Challenges in the Adoption of AI - Planning and Requirements Management in Projects - Integrating AI and LLMs in the Planning Phase - Agile Estimation Techniques Infused with AI - Requirements Gathering Overview		
UNIT III	GENERATIVE AI IN SOFTWARE DESIGN AND ARCHITECTURE	9
Overview of Software Design - Software Design Components - AI Infusion in Software Build and Development - Overview of the Software Build and Development Process - The Software Build Process - CI and CD Pipeline Creation and Optimization		
UNIT IV	AI IN SOFTWARE TESTING	9
The Evolution of Software Testing - AI in Software Testing - Key Benefits of AI in Software Testing - Challenges in Implementing AI in Software Testing - V-Model Evolvment with AI Testing - AI-Driven Testing Tools - Challenges with AI-Driven Testing Implementation.		
UNIT V	AI IN CONTINUOUS DELIVERY	9
The Role of AI in Continuous Delivery - AI in Deployment Automation - AI-Powered Release Management - AI Technologies Enhancing Test and Production Environments - AI in FinOps (Financial Operations) - AI in Network Management - AI in Access Management and Identity Governance - AI in Access Management and Identity Governance - AI for Database Management		
TOTAL: 45 PERIODS		
OUTCOMES: CO1: Explain the fundamentals of AI, ML, LLMs, and their applications in software engineering. CO2: Apply AI techniques in software planning, requirements engineering, and project management. CO3: Utilize Generative AI tools for software design, development, and CI/CD pipeline optimization. CO4: Analyze and apply AI-driven testing methods for defect detection and quality assurance. CO5: Examine AI applications in deployment automation, continuous delivery, and operational management. CO6: Evaluate AI-enabled solutions for FinOps, network management, access governance, and database management.		
TEXTBOOKS: 1. Kaiser, A. K., & Meda, V., AI Integration in Software Development and Operations: Transformation Through AI Infusion in DevOps, Testing, and SRE, 1st ed., Apress, Berkeley, 2024.		
REFERENCES: 1. Ian Sommerville, Software Engineering, 10th Ed., Pearson 2. Hui Liu and Tianpei Xia, Artificial Intelligence Techniques for Software Engineering, Springer, 2021		

24IT903	REST APPLICATION DEVELOPMENT USING SPRING BOOT AND JPA (Lab Integrated)	L	T	P	C
		2	0	2	3
OBJECTIVES: The Course will enable learners to: - To provide comprehensive knowledge of RESTful APIs and the HTTP methods used in the Spring Boot framework. - To cover advanced querying techniques using JPA, including LIKE queries, and to manage CRUD operations using JPQL. - To explore various relational mappings in JPA, such as one-to-one and one-to-many associations, and their practical implementations. - To implement and manage Spring AOP applications using annotation-based configurations for method interception and post-execution operations. - To build production-grade Spring Boot applications with integrated security using JWT, detailed API documentation with SwaggerUI and OpenUI, and effective logging practices.					
UNIT I	Server-Side Programming and REST APIs				6+6
Introduction to server-side programming and client-server architecture. Fundamentals of RESTful web services and API design. Overview of HTTP methods, status codes, and data formats such as JSON. Introduction to the Spring Framework and its features. Basics of dependency injection and Spring Boot application development.					
List of Exercise/Experiments: - Create a basic application using Spring Framework. (Create a simple Spring project. Define a bean Retrieve bean from container). - Develop a RESTful API for retrieving a welcome message, emphasizing the basics of data exchange between client and server. - Implement a RESTful API to acknowledge the user's favorite color choice, highlighting property value injection principles. - Create a Spring Boot application that retrieves and displays application information, demonstrating the usage of the @Value annotation to inject property values from the application configuration file. - Construct a RESTful API for student details retrieval, illustrating the utilization of @JsonIgnore annotation, focusing on advanced JSON property handling and data access control.					
UNIT II	Database operations				6+6

Database Operations & Spring MVC: MYSQL integration, mapping Java classes to relational database, repository interface, retrieving data, request body mapping.

Spring MVC & REST Integration: Spring MVC architecture, building REST services using Spring, HTTP methods (GET, POST, PUT, DELETE), error handling, logging with Log4J.

List of Exercise/Experiments:

1. Develop a web application for managing patient details using RESTful APIs, implementing POST and GET operations.
2. Create a web application for managing product details using RESTful APIs, enabling POST and GET operations.
3. Build an application for managing employee details using RESTful APIs, supporting POST, PUT, and DELETE operations.
4. Performing CRUD operations using Spring MVC and RESTful services.

UNIT III | Java Persistence API

6+6

Pagination & Sorting using JPA, @Transient Annotation, Queries using JPA, Starts and Ends with query using JPA, JPQL with @Query Annotation, custom JPQL queries.

List of Exercise/Experiments

1. Develop a web application for pagination and sorting of children details using RESTful APIs, implementing POST and GET operations.
2. Create a web application for managing Person details using JPA methods via RESTful APIs, enabling POST and GET operations.
3. Retrieve person details using JPQL with conditions for names starting or ending with specific patterns.
4. Build a web application for managing Person details using custom JPQL queries via RESTful APIs, supporting POST and GET operations.

UNIT IV | JPA Mapping

6+6

JPA Mapping of One-to-One Associations - fetching entities using queries - Loading optimization technique - Two-way One-to-One Relationship Mapping with JPA - single entity instance associated with multiple instances - Adding Data with One-to-One and One-to-Many Associations using JPA.

List of Exercise/Experiments:

1. Develop a Spring Boot application with "Person" and "Address" entities, where each person has exactly one address. Utilize Spring JPA to establish a one-to-one mapping between these entities.
2. Create a Spring Boot application with "Author" and "Book" entities, where each author can have multiple books, and each book belongs to only one author. Use Spring JPA to establish a one-to-many bidirectional mapping between these entities.
3. Build a Spring Boot application with "Employee" and "Address" entities, ensuring that each employee has exactly one address, and each address belongs to only one employee. Establish a one-to-one mapping between these entities using Spring JPA and utilize the Criteria API to retrieve employee details efficiently

UNIT V | Spring Boot Security, JWT Authentication, AOP and API Development Tools

6+6

API Documentation and Logging-Using Swagger UI and OpenAPI with Spring Boot to design and test APIs. Logging in Spring Boot, changing log levels, logging requests and responses, and managing logging configurations. Aspect-Oriented Programming (AOP)-Basic concepts of Aspect-Oriented Programming, handling method parameters, post-execution processing, and managing return values. API Security-Introduction to secure APIs using JSON Web Token.

List of Exercise/Experiments:

1. Implement Logging using Log4j.
2. Implement Spring Security using JWT and OAuth2
3. Develop a web application for managing Employee and Payroll details via RESTful APIs. Utilize Spring JPA to establish a one-to-one mapping between Employee and Payroll entities. Demonstrate the usage of Swagger for API documentation and interaction.
4. Develop a Spring Boot application focused on handling person details and integrate comprehensive logging capabilities to track application activities effectively.
5. Explore the implementation of Aspect-Oriented Programming (AOP) in a Spring application to enhance the behavior of a service method and demonstrate its impact on application functionality.

TOTAL: 60 PERIODS

OUTCOMES:

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

CO1: Create simple applications using RESTful APIs and effectively manage HTTP methods within the Spring Boot framework.

CO2: Apply database connectivity with JPA, utilizing advanced queries to interact with the database.

CO3: Build applications using Spring Boot and perform CRUD operations efficiently using JPQL

CO4: Demonstrate the implementation of various relational mappings in JPA, including one- to-one and one-to-many associations

CO5: Develop real-time applications that integrate user interfaces and utilize Spring AOP for method interception and advice handling.

CO6: Apply security measures to REST APIs using Spring Security and JWT to protect sensitive data and ensure secure communication between clients and servers.

TEXTBOOKS:

1. Raja CSP Raman, Ludovic Dewayilly, "Building RESTful Web Services with Spring 5", Packt Publishing, 2018.
2. Leonard Richardson, Sam Ruby "RESTful Web Services" O'Reilly Media, 2008.
3. Ludovic Dewayilly, "Building a RESTful Web Service with Spring: A hands-on guide to building an enterprise-grade, scalable RESTful web service using the Spring Framework", Packt Publishing, 2015
4. Raja CSP Raman, Ludovic Dewayilly, "Building RESTful Web Services with Spring 5 - Second Edition
5. Edition: Leverage the power of Spring 5.0, Java SE 9, and Spring Boot 2.0", Packt Publishing, 2018

REFERENCES:	
1. Ranga Karanam, "Master Java Web Services and REST API with Spring Boot", Packt Publishing, 2018. 2. Balaji Varanasi, Sudha Belida, "Spring REST", Apress, 2015. 3. Greg L. Turnquist, "Learning Spring Boot 2.0", Packt Publishing, 2021 4. Sourabh Sharma, "Modern API Development with Spring and Spring Boot", Packt Publishing, 2021	
LIST OF SOFTWARES	
1. VS Code, JRE, MySQL	

FOUNDATIONS OF DEEP LEARNING		L	T	P	C
		3	0	0	3
OBJECTIVES:					
- To outline the basics of deep neural networks. - To discuss advanced deep learning models. - To discuss CNN and RNN architectures of deep neural networks. - To elaborate autoencoders in neural networks. - To discuss the deep generative models.					
UNIT I	DEEP NETWORKS				9
Challenges motivating deep learning - Deep feedforward networks - Learning XOR - Gradient based learning - Hidden Units - Architecture Design - Back Propagation - Regularization - Parameter Norm Penalties - Constrained Optimization - Under-Constrained Problems - Dataset Augmentation - Noise Robustness - Semi-Supervised Learning - Multi-Task Learning - Early Stopping - Parameter Tying and Sharing - Bagging and Other Ensemble methods - Dropout - Adversarial Training.					
UNIT II	OPTIMIZATION FOR TRAINING DEEP MODELS				9
Pure optimization - Challenges - Basic Algorithms - Parameter initialization Strategies - Algorithms with Adaptive Learning Rates - Approximate Second-Order methods - Optimization Strategies and Meta Algorithms.					
UNIT III	CONVOLUTIONAL AND RECURRENT NEURAL NETWORKS				9
Convolution Operation - motivation - Pooling - Infinitely Strong prior - Variants - Structured Output - Data Types - Efficient Convolutional Algorithms - Random or Unsupervised features - Neuroscientific Basis - Deep Learning - Sequence Modelling - Computational Graphs - RNN - Bidirectional RNN - Encoder-Decoder - Sequence to Sequence RNN - Deep Recurrent Networks - Recursive Neural Networks -- Long Term Dependencies; Leaky Units - Strategies for multiple time scales - LSTM and Gated RNNs - Optimization for Long Term Dependencies.					
UNIT IV	AUTOENCODERS				9
Autoencoders: Undercomplete autoencoders - Regularized autoencoders - Power, Layer Size and Depth - Stochastic encoders and decoders - Denoising Autoencoders - Learning with autoencoders - contractive Autoencoders - Applications of autoencoders.					
UNIT V	DEEP GENERATIVE MODELS				9
Boltzmann Machine - Restricted Boltzmann Machine - Deep Belief Networks - Deep Boltzmann Machines - Boltzmann Machines for Real-Valued Data - Convolutional Boltzmann Machines - Boltzmann Machine for Structured or Sequential Outputs - Directed Generative Nets - Evaluating Generative Models.					
TOTAL: 45 PERIODS					

OUTCOMES:

At the end of this course, the students will be able to:

CO1: Outline the basics of deep neural networks.

CO2: Develop advanced deep learning models.

CO3: Implement CNN and RNN architectures of deep neural networks.

CO4: Interpret autoencoders in neural networks.

CO5: Apply deep generative models to solve real world problems.

CO6: Build deep learning models and evaluate them.

TEXT BOOKS:

1. Ian Goodfellow, Yoshua Bengio, Aaron Courville, "Deep Learning", MIT Press, 2016.

REFERENCES:

1. Charu C. Aggarwal, "Neural Networks and Deep Learning: A Textbook", Springer International Publishing, 2018.
2. Yoav Goldberg, "Neural Network Methods for Natural Language Processing", Synthesis Lectures on Human Language Technologies, Morgan & Claypool publishers, 2017.
3. Francois Chollet, "Deep Learning with Python", Manning Publications Co, 2018.
4. Josh Patterson, Adam Gibson, "Deep Learning: A Practitioner's Approach", O'Reilly Media, 2017.
5. Navin Kumar Manaswi, "Deep Learning with Applications Using Python", Apress, 2018.

NPTEL:

6. Deep Learning - https://onlinecourses.nptel.ac.in/noc24_ee04/preview
Deep Learning - IIT Ropar - https://onlinecourses.nptel.ac.in/noc24_cs59/preview

24CS922	SOFTWARE TESTING AND AUTOMATION	L	T	P	C
		3	0	0	3
OBJECTIVES: The Course will enable learners to: • Introduce the basics and necessity of software testing. • Provide various testing techniques along with concepts of software bugs and its impact. • Develop and validate a test plan. • Build a testing team required. • Understand the need for and challenges in test automation and to develop testing scripts.					
UNIT I	FOUNDATIONS OF SOFTWARE TESTING				9
Why do we test Software? Stages of Testing, Software Testing Principles, Software Testing Life Cycle, Program Correctness and Verification, Reliability versus Safety, Failures, Errors and Faults (Defects), Case Study: Test plan for testing an e-commerce web/mobile application (Amazon).					
UNIT II	TEST PLANNING				9
The Goal of Test Planning, High Level Expectations, Intergroup Responsibilities, Test Phases, Test Strategy, Resource Requirements, Tester Assignments, Test Schedule, Test Cases, Bug Reporting, Metrics and Statistics.					
UNIT III	TEST DESIGN AND EXECUTION				9

Test Objective and Requirement Identification, Test Design Factors, Modeling a Test Design Process, Model and Results, Boundary Value Testing, Equivalence Class Testing, Path Testing, Data Flow Testing, Test Design Metrics and Effectiveness, Test Procedures. Case Study: Execute the test cases on the client-server or desktop application to identify any defect.		
UNIT IV	ADVANCED TESTING CONCEPTS	9
Performance Testing: Load Testing, Stress Testing, Volume Testing, Fail-Over Testing, Recovery Testing, Configuration Testing, Compatibility Testing, Usability Testing, Testing the Documentation, Security testing, Testing in the Agile Environment, Testing Web and Mobile Applications.		
UNIT V	TEST AUTOMATION AND TOOLS	9
Software Test Automation - Skill Needed for Automation - Scope of Automation - Design and Architecture for Automation - Requirements for a Test Tool - Challenges in Automation - Test Metrics and Measurements - Project, Progress and Productivity Metrics - Maintenance of Documents During Testing. Case Study: Install and learn popular software testing tools like Selenium, WinRunner, LoadRunner, Performance Tester etc.		
TOTAL: 45 PERIODS		
COURSE OUTCOMES:		
Upon completion of the course, the students will be able to:		
CO1: Explore the basic concepts of software testing and the need for software testing. CO2: Design Test planning and different activities involved in test planning. CO3: Design effective test cases that can uncover critical defects in the application. CO4: Carry out advanced types of testing. CO5: Automate the software testing using Selenium and TestNG. CO6: Analyze and evaluate the results of testing to ensure software quality, and implement continuous improvement processes based on testing outcomes.		
TEXT BOOKS:		
1. Yogesh Singh, "Software Testing", Cambridge University Press, 2012. 2. Ron Patton, Software testing, 2nd Edition, 2006, Sams Publishing. 3. Srinivasan Desikan and Gopalaswamy Ramesh, "Software Testing: Principles and Practices", Pearson Education, 2006. 4. Dorothy Graham, Mark Fewster, "Experiences of Test Automation: Case Studies of Software Test Automation", Pearson Education, 2012. 5. Unmesh Gundecha, Satya Avasarala, "Selenium WebDriver 3 Practical Guide" - Second Edition 2018. 6. Paul C. Jorgensen, "Software Testing", A Craftsman Approach, CRC Press, Fourth Edition, 2013.		
REFERENCES:		
1. Glenford J. Myers, Corey Sandler, Tom Badgett, The Art of Software Testing, 3rd Edition, 2012, John Wiley& Sons, Inc. 2. Mark Fewster, Software Test Automation, 1999, Addison-Wesley. 3. Ali Mili, Fairouz Chier, "Software Testing: Concepts and Operations", Wiley, 2015.		

EMERGING TECHNOLOGIES

24CS929	MULTIMEDIA COMPUTING	L	T	P	C
		3	0	0	3
OBJECTIVES: The course should enable the students to: • This course aims to further develop students’ competency in producing dynamic and creative graphic solutions for multimedia productions. • To provide the foundation knowledge of multimedia computing, e.g. media characteristics, compression standards, multimedia representation, data formats, multimedia technology development.					
UNIT I	FUNDAMENTAL CONCEPTS IN TEXT AND IMAGE				9
Multimedia and hypermedia, World Wide Web, overview of multimedia software tools. Graphics and image data representation graphics/image data types, file formats, Color in image and video: color science, color models in images, color models in video.					
UNIT II	FUNDAMENTAL CONCEPTS IN VIDEO AND DIGITAL AUDIO				9
Types of video signals, analog video, digital video, digitization of sound, MIDI, quantization and transmission of audio.					
UNIT III	MULTIMEDIA DATA COMPRESSION				9
Lossy compression algorithm: Quantization, Transform Coding, Wavelet-Based Coding. MPEG Video Coding: Embedded Zerotree of Wavelet Coefficients Set Partitioning in Hierarchical Trees (SPIHT)..					
UNIT IV	BASIC VIDEO COMPRESSION TECHNIQUES				9
Introduction to video compression, video compression based on motion compensation, search for motion vectors, MPEG, Basic Audio Compression Techniques.					
UNIT V	MULTIMEDIA NETWORKS				9
Basics of Multimedia Networks, Multimedia Network Communications and Applications: Quality of Multimedia Data Transmission, Multimedia over IP, Multimedia over ATM Networks, Transport of MPEG-4, Media-on-Demand (MOD)..					
TOTAL: 45 PERIODS					
OUTCOMES: At the end of this course, the students will be able to: CO1: Understand the characteristics of different media; understand the representations of different multimedia data. CO2: Understand different data formats; be able to take into considerations in multimedia system designs. CO3: Understand the characteristics of human’s visual system; understand the characteristics of human’s audio system; be able to take into considerations in multimedia techniques design and implementation. CO4: Understand different compression principles and techniques. CO5: Understand different multimedia compression standards; be able to design and develop multimedia systems according to the requirements of multimedia applications.					

TEXT BOOKS:

1. Fundamentals of Multimedia by Ze-Nian Li and Mark S. Drew PHI Learning, 2004
2. Professional Adobe Flex 3, Joseph Balderson, Peter Ent, et al, Wrox

Publications, Wiley India, 2009.

REFERENCES:

1. Digital Multimedia, Nigel chapman and jenny chapman, Wiley-Dreamtech
2. Macromedia Flash MX Professional 2004 Unleashed, Pearson.
3. Multimedia and communications Technology, Steve Heath, Elsevier (Focal Press).
4. Multimedia Applications, Steinmetz, Nahrstedt, Springer.
5. Multimedia Basics by Weixel Thomson
6. Multimedia Technology and Applications, David Hilman , Galgotia

24CS930	GPU COMPUTING	L	T	P	C
		3	0	0	3
OBJECTIVES: The Course will enable learners to: - To understand the basics of GPU Architectures and CUDA Programming. - To learn synchronization using CUDA. - To discuss memories and its impact on performance. - To understand the various parallel algorithms on GPU. - To learn the basics of OPENCL.					
UNIT I	GPU ARCHITECTURES AND CUDA PROGRAMMING				9
Heterogeneous Parallel Computing - Architecture of a modern GPU - Parallel Programming languages and models - GPU Computing - Introduction to Data Parallelism and CUDA C: Data Parallelism - CUDA Program Structure - A vector additional Kernel - Device Global Memory and Data Transfer - Kernel functions and Threading.					
UNIT II	MULTI-DIMENSIONAL DATA & SYNCHRONIZATION				9
CUDA Thread Organization - Mapping Threads to Multi-Dimensional Data - Synchronization and Transparent Scalability - Assigning resources to Blocks - Querying Device Properties - Thread Scheduling and Latency Tolerance.					
UNIT III	CUDA MEMORIES & PERFORMANCE				9
CUDA Memories - Memory Access Efficiency - CUDA Device Memory Types - Reducing global Memory Traffic - Performance Considerations - Warps and Thread Execution - Global Memory Bandwidth - Dynamic Partitioning of Execution Resources - Instruction Mix and Thread Granularity.					
UNIT IV	ALGORITHMS ON GPU				9
Parallel Patterns: Convolution - Prefix Sum - Sparse Matrix - Vector Multiplication.					
UNIT V	OPENCL BASICS				9
Introduction - OpenCL Platform Model - Execution Model - Programming model - Memory Model - OpenCL Runtime.					
TOTAL: 45 PERIODS					

OUTCOMES: Upon completion of the course, the students will be able to: CO1: Explain the basics of GPU Architectures and implement simple CUDA Programs.	
CO2: Discuss synchronization using CUDA. CO3: Elaborate CUDA memories and its impact on performance. CO4: Design various parallel algorithms on GPU. CO5: Solve simple problems using parallel algorithms. CO6: Apply OpenCL to solve programs and improve performance.	
TEXT BOOKS:- 1. David Kirk and Wen-mei Hwu, Programming Massively Parallel Processors - A hands-on Approach, Morgan Kaufmann, Second Edition, 2013. 2. Benedict Gaster, Lee Howes, David R. Kaeli, "Heterogeneous Computing with OpenCL", Third Edition, Morgan Kaufman, 2012.	
REFERENCES: 1. David Kaeli, Perhaad Mistry, Dana Schaa, Dong Ping Zhang, "Heterogeneous Computing with OpenCL 2.0", Third Edition, Morgan Kaufman, 2015. 2. John L. Hennessy and David A. Patterson, "Computer Architecture - A Quantitative Approach", Sixth Edition, Morgan Kaufman, 2017. 3. NPTEL Courses: a. GPU Architectures And Programming - https://onlinecourses.nptel.ac.in/noc23_cs61/preview	

24CS931	INTRODUCTION TO AUGMENTED AND VIRTUAL REALITY	L	T	P	C
		3	0	0	3
OBJECTIVES: The Course will enable learners to: • Get exposure on Augmented Reality. • Introduce Virtual Reality and input and output devices. • Acquire knowledge on computing architectures and modelling. • Explore Virtual Reality programming and human factors. • Learn various applications of Virtual Reality.					
UNIT I	AUGMENTED REALITY				9
Introduction to Augmented Reality-Computer vision for AR-Interaction-Modelling and Annotation-Navigation-Wearable devices.					
UNIT II	INTRODUCTION TO VIRTUAL REALITY AND INPUT AND OUTPUT DEVICES				9
Introduction: The three I's of Virtual Reality - Early commercial VR technology - The five classic components of a VR system. Input devices: Three-Dimensional position trackers - tracker performance parameters - ultrasonic trackers - optical trackers - Navigation and manipulation interfaces - gesture interfaces. Output devices: graphics displays - large-volume displays - sound displays.					
UNIT III	COMPUTING ARCHITECTURES AND MODELING OF A VR SYSTEM				9

Computing architectures for VR: The rendering pipeline - The graphics rendering pipeline - The haptics rendering pipeline - PC graphics architecture - PC graphics accelerators - Graphics benchmarks - Distributed VR architectures - Multipipeline synchronization - Colocated rendering pipelines. Modeling: geometric modeling - kinematics modeling - physical and behavior modelling		
UNIT IV	VR PROGRAMMING AND HUMAN FACTORS	9
Toolkits and scene graphs - World Toolkit - Model geometry and appearance - The WTK scene graph - Sensors and action functions - WTK networking - Java 3D - Model geometry and appearance - Java 3D scene graph - Sensors and behaviors - Java 3D networking - WTK and Java 3D performance comparison - Human factors in VR: Methodology and terminology - user performance studies - VR health and safety issues - VR and society		
UNIT V	APPLICATIONS OF VR	9
Medical Application of VR - Virtual anatomy-Triage and diagnostic - Surgery - VR in education - VR and the Arts - Entertainment applications of VR - military VR applications - Army use of VR - VR applications in the Navy - Air force use of VR - Applications of VR in Robotics - Robot programming - Robot teleoperation		
TOTAL: 45 PERIODS		
OUTCOMES: Upon completion of the course, the students will be able to: CO1: Explain the Augmented Reality. CO2: Explore different input and output devices used in Virtual Reality system. CO3: Model the VR system. CO4: Learn about Google Toolkit's and Scene Graph. CO5: Apply VR in various fields. CO6: Use modern tools to design Applications.		
TEXT BOOKS: 1. Dieter Schmalstieg, Tobias Hollerer, "Augmented Reality: Principles & Practice", Addison Wesley, 2016. 2. Grigore C. Burdea, Philippe Coiffet, "Virtual reality technology", Wiley, Second Edition, 2017.		
REFERENCE BOOKS: 1. Sherman, William R & Craig, Alan B, "Understanding Virtual reality", Elsevier India Private Limited, Noida, 2018. 2. Charles Palmer, John Williamson, "Virtual Reality Blueprints: Create compelling VR experiences for mobile", Packt Publisher, 2018.		
SOFTWARE REQUIREMENTS: Unity, Maya/3DS MAX/Blender.		

24CS932	DIGITAL MARKETING		L	T	P	C
			3	0	0	3

OBJECTIVES:
The Course will enable learners to:

- Examine and explore the role and importance of digital marketing in today's rapidly changing business environment.
- Focus on how digital marketing can be utilized by organizations and how its effectiveness can be measured.

UNIT I	INTRODUCTION TO ONLINE MARKET	9
Online Market space- Digital Marketing Strategy- Components - Opportunities for building Brand Website - Planning and Creation - Content Marketing		
UNIT II	SEARCH ENGINE OPTIMIZATION	9
Search Engine optimization - Keyword Strategy- SEO Strategy - SEO success factors -On- Page Techniques - Off-Page Techniques. Search Engine Marketing- How Search Engine works- SEM components- PPC advertising -Display Advertisement		
UNIT III	E- MAIL MARKETING	9
E- Mail Marketing - Types of E- Mail Marketing - Email Automation - Lead Generation - Integrating Email with Social Media and Mobile- Measuring and maximizing email campaign effectiveness. Mobile Marketing- Mobile Inventory/channels- Location based; Context based; Coupons and offers, Mobile Apps, Mobile Commerce, SMS Campaigns- Profiling and target in		
UNIT IV	SOCIAL MEDIA MARKETING	9
Social Media Marketing - Social Media Channels- Leveraging social media for brand conversations and buzz. Successful /benchmark social media campaigns. Engagement Marketing- Building Customer relationships - Creating Loyalty drivers - Influencer Marketing.		
UNIT V	DIGITAL TRANSFORMATION	9
Digital Transformation & Channel Attribution - Analytics - Ad-words, Email, Mobile, Social Media, Web Analytics - Changing your strategy based on analysis- Recent trends in Digital marketing.		
TOTAL: 45 PERIODS		

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

CO1: Examine and explore the role and importance of digital marketing in today's rapidly changing business environment.

CO2: Focuses on how digital marketing can be utilized by organizations and how its effectiveness can be measured.

CO3: Explain the key elements of a digital marketing strategy

CO4: Study how the effectiveness of a digital marketing campaign can be measured

CO5: Demonstrate advanced practical skills in common digital marketing tools such as SEO, SEM, social media and Blogs

CO6: Explain the recent trends in digital marketing.

TEXT BOOKS:
1. Puneet Singh Bhatia, "Fundamentals of Digital Marketing", Pearson Education; First edition, 2017.
2. Vandana Ahuja, "Digital Marketing", Oxford University Press, 2015
REFERENCE BOOKS:
1. Philip Kotler, "Marketing 4.0: Moving from Traditional to Digital", Wiley; 1st edition, 2017
2. Ryan, D., "Understanding Digital Marketing: Marketing Strategies for Engaging the Digital Generation", Kogan Page Limited., 2014
3. Barker, Barker, Bormann and Neher, Social Media Marketing: A Strategic Approach, 2E South-Western, Cengage Learning. 2017.
4. Pulizzi, J., "Beginner's Guide to Digital Marketing", McGraw Hill Education, 2019.
SOFTWARE REQUIREMENTS:
Unity, Maya/3DS MAX/Blender.

24CS933	QUANTUM COMPUTING	L	T	P	C
		3	0	0	3
OBJECTIVES: The Course will enable learners to: - Analyse the behaviour of basic quantum algorithms - To discuss simple quantum algorithms and information channels in the quantum circuit model - Apply the Quantum Algorithms in Superdense coding and quantum Teleportation - Analyse the Algorithms with Superpolynomial Speed-up - Illustrate a simple quantum error-correcting code					
UNIT I	FOUNDATION				9
Overview of traditional computing - Church-Turing thesis - circuit model of computation - reversible computation - quantum physics - quantum physics and computation - Dirac notation and Hilbert Spaces - dual vectors - operators - the spectral theorem - functions of operators - tensor products - Schmidt decomposition theorem					
UNIT II	QUBITS AND QUANTUM MODEL OF COMPUTATION				9
State of a quantum system - time evolution of a closed system - composite systems - measurement - mixed states and general quantum operations - quantum circuit model - quantum gates - universal sets of quantum gates - unitary transformations - quantum circuits					
UNIT III	QUANTUM ALGORITHMS – I				9
Superdense coding - quantum teleportation - applications of teleportation - probabilistic versus quantum algorithms - phase kick-back - the Deutsch algorithm - the Deutsch- Jozsa algorithm - Simon's algorithm - Quantum phase estimation and quantum Fourier Transform - eigenvalue estimation					
UNIT IV	QUANTUM ALGORITHMS – II				9
Order-finding problem - eigenvalue estimation approach to order finding - Shor's algorithm for order finding - finding discrete logarithms - hidden subgroups - Grover's quantum search algorithm - amplitude amplification - quantum amplitude estimation - quantum counting - searching without knowing the success probability 101					

UNIT V	QUANTUM COMPUTATIONAL COMPLEXITY AND ERROR CORRECTION	9
Computational complexity - black-box model - lower bounds for searching - general black-box lower bounds - polynomial method - block sensitivity - adversary methods - classical error correction - classical three-bit code - fault tolerance - quantum error correction - three- and nine-qubit quantum codes - fault-tolerant quantum computation		
TOTAL: 45 PERIODS		
OUTCOMES: Upon completion of the course, the students will be able to: CO1: Analyse the behaviour of basic quantum algorithms CO2: Discuss simple quantum algorithms and information channels in the quantum circuit model CO3: Apply the Quantum Algorithms in Superdense coding and quantum Teleportation CO4: Analyse the Algorithms with Super polynomial Speed-up CO5: Illustrate a simple quantum error-correcting code CO6: Elaborate various quantum algorithms.		
TEXT BOOKS: 1. P. Kaye, R. Laflamme, and M. Mosca, "An introduction to Quantum Computing", Oxford University Press, 2007.		
REFERENCES: 1. E. Rieffel and W. Polak "Quantum Computing A Gentle Introduction", The MIT Press Cambridge, 2011. 2. Jack D. Hidary "Quantum Computing: An Applied Approach" Springer, 2019. 3. V. Sahni, "Quantum Computing", Tata McGraw-Hill Publishing Company, 2007. 4. Michael A. Nielsen and Issac L. Chuang, "Quantum Computation and Quantum Information", Tenth Edition, Cambridge University Press, 2010		

24CS934	INTELLIGENT ROBOTS	L	T	P	C
		3	0	0	3
OBJECTIVES: • To understand the basics of Intelligent Robots. • To discuss the Autonomous capabilities and Software architecture. • To elaborate the Reactive Functionality of intelligent Robots. • To use the various sensors in building Intelligent Robots. • To illustrate the Deliberative Functionality of intelligent Robots.					
UNIT I	INTRODUCTION				9
Overview- Definition - Components -Three Modalities - Need for Intelligent Robots - History of AI Robotics - Industrial Manipulators - Mobile Robots - Drones - Cognitive Systems.					
UNIT II	AUTOMATION AND AUTONOMY				9
Autonomous Capabilities - Bounded Rationality - Automation and Autonomy - Programming Style - Hardware Design - Types of Functional Failures - Autonomous Capabilities. Types of Software Architectures - Operational Architectures - Components of a Telesystem - Human Supervisory Control.					
UNIT III	REACTIVE FUNCTIONALITY				9

Behaviours : Agency and Marr's Computational Theory - Animal Behaviours - Schema Theory. Perception: Action-Perception cycle - Functions. Behaviour Coordination - Function - Cooperating Methods - Competing Methods - Sequences.		
UNIT IV	SENSORS AND SENSING	9
Locomotion: Mechanical, Biomimetic, Legged Locomotion - Action Selection - Sensors and Sensing Model - Choosing - Range Sensing: Stereo - Depth from X - Sonar or Ultrasonics.		
UNIT V	DELIBERATIVE FUNCTIONALITY	9
Deliberation - Strips - Navigation - Spatial Memory - Types of Path Planning - Configuration Space - Metric Path Planning - Motion Planning - Localization - Feature based Localization - Iconic Localization - Static vs Dynamic Environments - Simultaneous Localization and Mapping - Terrain Identification and Mapping - Scale and Traversability - Exploration - Multirobot Systems and AI - Human-Robot Interaction and areas of AI.		
TOTAL: 45 PERIODS		
OUTCOMES: Upon completion of the course, the students will be able to: CO1: Implement the basics of Intelligent Robots. CO2: Design and implement Autonomous capabilities in Robotics systems. CO3: Elaborate the Reactive Functionality of intelligent Robots. CO4: Use the various sensors in building Intelligent Robots. CO5: Illustrate the Deliberative Functionality of intelligent Robots. CO6: Analyse the various applications of AI Robotics.		
TEXT BOOKS: 1. Robin R. Murphy, "Introduction to AI Robotics", MIT Press, Second Edition, 2019.		
REFERENCES: 1. Francis X. Govers, "Artificial Intelligence for Robotics: Build Intelligent Robots that Perform Human Tasks Using AI Techniques", Packt Publishing, 2018. 2. Sebastian Thrun, Wolfram Burgard, and Dieter Fox, "Probabilistic Robotics", MIT Press, 2005. 3. Yoon Seok Pyo, Han Cheol Cho, Ryu Woon Jung, and Tae Hoon Lim, "ROS Robot Programming", ROBOTIS Co., Ltd, 2017.		

24CS936	PRINCIPLES OF 3D PRINTING AND DESIGN	L	T	P	C
		3	0	0	3
OBJECTIVES: The Course will enable learners to: • To learn the basics of 3D Modelling and viewing. • To demonstrate the principles of 3D processing technique and application. • To design and illustrate the working principle of inkjet technology. • To acquire the knowledge of laser technology and Printing machines. • To apply the knowledge in manufacturing, healthcare and medical applications.					
UNIT I	INTRODUCTION				9
Introduction; Design considerations - Material, Size, Resolution, Process; Modelling and viewing - 3D; Scanning; Model preparation - Digital; Slicing; Software; File formats.					
UNIT II	PRINCIPLE				9

Processes - Extrusion, Wire, Granular, Lamination, Photo polymerisation; Materials - Paper, Plastics, Metals, Ceramics, Glass, Wood, Fiber, Sand, Biological Tissues, Hydrogels, Graphene; Material Selection - Processes, applications, limitations.		
UNIT III	INKJET TECHNOLOGY	9
Printer - Working Principle, Positioning System, Print head, Print bed, Frames, Motion control; Print head Considerations - Continuous Inkjet, Thermal Inkjet, Piezoelectric Drop-On-Demand; Material Formulation for jetting; Liquid based fabrication - Continuous jet, Multijet; Powder based fabrication - Colourjet.		
UNIT IV	LASER TECHNOLOGY	9
Light Sources - Types, Characteristics; Optics - Deflection, Modulation; Material feeding and flow - Liquid, powder; Printing machines - Types, Working Principle, Build Platform, Print bed Movement, Support structures.		
UNIT V	INDUSTRIAL APPLICATIONS	9
Product Models, manufacturing - Printed electronics, Biopolymers, Packaging, Healthcare, Food, Medical, Biotechnology, Displays; Future trends;		
TOTAL: 45 PERIODS		
OUTCOMES: Upon completion of the course, the students will be able to: CO1: Outline and examine the basic concepts of 3D printing technology. CO2: Develop 3D printing workflow. CO3: Examine and categorize the concepts and working principles of 3D printing using inkjet technique. CO4: Explain the inkjet technology and its applications. CO5: Describe the working principles of 3D printing using laser technique. CO6: Apply various method for designing and modeling for industrial applications.		
TEXT BOOKS: 1. Christopher Barnatt, 3D Printing: The Next Industrial Revolution, CreateSpace Independent Publishing Platform, 2013. 2. Ian M. Hutchings, Graham D. Martin, Inkjet Technology for Digital Fabrication, John Wiley & Sons, 2013.		
REFERENCES: 1. Chua, C.K., Leong K.F. and Lim C.S., Rapid prototyping: Principles and applications, second edition, World Scientific Publishers, 2010. 2. Ibrahim Zeid, Mastering CAD CAM Tata McGraw-Hill Publishing Co., 2007. 3. Joan Horvath, Mastering 3D Printing, APress, 2014.		

24CS937	HIGH PERFORMANCE COMPUTING	L	T	P	C
		3	0	0	3
OBJECTIVES: • To understand the basics of Modern processors. • To analyze the various optimization techniques for writing parallel high-performance applications. • To discuss the basics of Parallel computers. • To learn shared memory parallel programming using OpenMP. • To understand the distributed memory parallel programming with MPI.					
UNIT I	MODERN PROCESSORS				9
Stored Program Computer Architecture - General purpose cache- based microprocessor- Performance based metrics and benchmarks- Moore's Law- Pipelining- Superscalarity - SIMD- Memory Hierarchies - Cache- mapping- prefetch- Multicore processors- Mutithreaded processors- Vector Processors- Design Principles- Maximum performance estimates- Programming for vector architecture.					
UNIT II	OPTIMIZATION TECHNIQUES				9
Basic optimization techniques for serial code : scalar profiling - function and line based runtime profiling- hardware performance counters- common sense optimizations- elimination of common subexpressions- avoiding branches - using SIMD instruction sets- the role of compilers - C++ optimizations - data access optimization: balance analysis and light speed estimates- storage order- Case study: jacobi algorithm and dense matrix transpose.					
UNIT III	PARALLEL COMPUTERS				9
Taxonomy of parallel computing paradigms - Shared memory computers - Distributed-memory computers- Hierarchical systems- Networks - Basics of parallelization - Need to parallelize - Parallelism - Parallel Scalability- Factors that limit parallel execution- Scalability metrics- Simple scalability laws- parallel efficiency - serial performance Vs Strong scalability- Load balance.					
UNIT IV	SHARED MEMORY PARALLEL PROGRAMMING WITH OpenMp				9
Introduction to Open MP - parallel execution - data scoping- OpenMp work sharing for loops- synchronization - reductions - loop scheduling - tasking - case study: OpenMp-parallel jacobi algorithm - Efficient OpenMP programming: Profiling OpenMP programs - Performance pitfalls.					
UNIT V	DISTRIBUTED MEMORY PARALLEL PROGRAMMING WITHMPI				9
Message passing - introduction to MPI - example - messages and point-to-point communication - collective communication - nonblocking point-to-point communication- virtual topologies - MPI parallelization of Jacobi solver- MPI implementation - performance properties - Basic MPI/OpenMP programming models.					
TOTAL: 45 PERIODS					

OUTCOMES:

At the end of this course, the students will be able to:

CO1: Understand the basics of Modern processors.

CO2: Analyze the various optimization techniques for writing parallel high performance applications.

CO3: Discuss the basics of Parallel computers.

CO4: Learn shared memory parallel programming using OpenMP.

CO5: Understand the distributed memory parallel programming with MPI.

CO6: Use the tools to implement parallel programming.

HONOURS – CYBER SECURITY

24CS906	DIGITAL AND MOBILE FORENSICS	L	T	P	C
		3	0	0	3
OBJECTIVES: The Course will enable learners to: • Learn how to acquire digital forensic evidence. • Learn how to investigate different digital artifacts and write reports • Understand network forensics processes and procedures • Understand mobile forensics processes and procedures. • Able to analyze SIM cards and analyze mobile file systems.					
UNIT I	ACQUIRING DIGITAL FORENSICS EVIDENCE				9
Types of Computer-Based Investigations - The Forensic Analysis Process-Acquisition of Evidence - Computer Systems.					
UNIT II	DIGITAL FORENSICS INVESTIGATION& REPORTING				9
Computer Investigation Process-Windows Artifact Analysis-RAM Memory Forensic Analysis-Email Forensics-Investigation Techniques-Internet Artifacts.					
UNIT III	NETWORKING FORENSICS				9
Characteristics in the network world-Identifying threats to the enterprise-Data breach surveys-Defining network forensics-Differentiating between computer forensics and network forensics-Digital footprints-Collecting network traffic using tcp dump-Collecting network traffic using Wireshark-Collecting network logs-Acquiring memory using FTK Imager-Tapping into network traffic-Packet sniffing and analysis using Wireshark-Packet sniffing and analysis using Network Miner					
UNIT IV	MOBILE FORENSICS FUNDAMENTALS				9
Mobile Devices vs. Computer Devices in the World of Forensics-Living in the Cloud: The Place to Hide and Store Mobile Data-Preparing, Protecting, and Seizing Digital Device Evidence					
UNIT V	ANALYSING MOBILE INTERNALS				9
Analyzing SIM Cards - Advanced Android Analysis - Advanced iOS Analysis-Case Study: Use Andriller or equivalent to extract data from Android					
TOTAL: 45 PERIODS					

OUTCOMES:

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

CO1: Understand the process of acquiring digital forensics evidence.

CO2: Apply digital forensics investigation techniques and report findings.

CO3: Analyze network forensics data to identify threats and breaches.

CO4: Explore fundamentals of mobile forensics and evidence preparation.

CO5: Investigate mobile device internals for forensic analysis.

CO6: Demonstrate ethical considerations in digital forensic practices.

TEXTBOOKS:

1. William Oettinger, "Learn Computer Forensics: A beginner's guide to searching, analyzing, and securing digital evidence", Packt Publishing, 1st Edition, 2020
2. Samir Datt, "Learning Network Forensics", Packt Publishing, 1st Edition, 2016
3. Lee Reiber, "Mobile Forensic Investigations: A Guide to Evidence Collection, Analysis, and Presentation", McGraw Hill, 2nd Edition, 2018

REFERENCES:

1. Rohit Tamma, Oleg Skulkin, Heather Mahalik, Satish Bommisetty, "Practical Mobile Forensics", Packt Publishing, 3rd Edition, 2018
2. Gerard Johansen, "Digital Forensics and Incident Response: Incident response tools and techniques for effective cyber threat response", Packt Publishing, 3rd Edition, 2022

24CS907	VULNERABILITY ANALYSIS & PENETRATION TESTING	L	T	P	C
		3	0	0	3
OBJECTIVES: The Course will enable learners to: • To learn the tools that can be used to perform information gathering. • To identify operating systems, server applications to widen the attack surface and perform vulnerability assessment activity and exploitation phase. • To learn how vulnerability assessment can be carried out by means of automatic tools or manual investigation. • To learn the web application attacks starting from information gathering to exploitation phases. • To learn how to metasploit and meterpreter are used to automate the attacks and penetration testing techniques.					
UNIT I	INTRODUCTION				9
Introduction- Vulnerability Assessment- Understanding the Risks Posed by Vulnerabilities Detecting Vulnerabilities via Security Technologies- Categories of Penetration Testing - Types of Penetration Test - Structure of Penetration Testing Reports - Information Gathering Techniques - Active, Passive and Sources of Information Gathering - Approaches and Tools - Traceroutes, Neotrace, Whatweb, Netcraft, Xcode Exploit Scanner and NSlookup. Host discovery - Scanning for open ports and services - Types of Port.					
UNIT II	NETWORK VULNERABILITY ASSESSMENT				9
Project Scoping-Assessing Vulnerability assessment timeline-NVAT-Prioritizing risks and threats Assessment Methodology-Top down and Bottom up Examination-Case study with assessment report- Case Study: Web Based Email Attacks.					
UNIT III	MOBILE APPLICATION SECURITY				9

Types of Mobile Application Key challenges in Mobile Application and its impact Need for mobile application penetration testing Mobile application penetration testing methodology Android and ios Vulnerabilities - OWASP mobile security risk - Exploiting WM - BlackBerry Vulnerabilities - Vulnerability Landscape for Symbian - Exploit Prevention - Handheld Exploitation	
UNIT IV	WIRELESS NETWORK VULNERABILITY ANALYSIS
WLAN and its inherent insecurities Bypassing WLAN Authentication uncovering hidden SSIDs MAC Filters Bypassing open and shared authentication - Attacking the client coffee latte attack Deauthenticating the client cracking WEP with the hirt attack AP-less WPA cracking - Advanced WLAN Attacks Wireless eavesdropping using MITM session hijacking over wireless - WLAN Penetration Test Methodology	
UNIT V	PENETRATION TESTING
Introduction to Kali and Backtrack-Linux tools - Attack Machine- Phases of penetration test- reconnaissance extracting information from DNS-scanning-pings and ping sweeps-port scanning- NMap-Vulnerability scanning.	
TOTAL: 45 PERIODS	
OUTCOMES: Upon completion of the course, the students will be able to: CO1: Understand vulnerability assessment principles and methods. CO2: Analyze network vulnerabilities and prioritize risks. CO3: Evaluate mobile application security challenges and methodologies. CO4: Assess wireless network vulnerabilities and conduct penetration testing. CO5: Apply penetration testing methodologies using appropriate tools. CO6: Demonstrate ethical considerations in penetration testing practices.	
TEXTBOOKS:	
1. Rafay Baloch, Ethical Hacking and Penetration Testing Guide, CRC Press, 2015. 2. Dr. Patrick Engebretson, The Basics of Hacking and Penetration Testing Ethical Hacking and Penetration Testing made easy, Syngress publications, Elsevier, 2013.	
REFERENCES:	
1. Steve Manzuik, Andre Gold, Chris Gatford, "Network Security Assessment from Vulnerability to Patch", Syngress Publishing Incorporation, 2007. 2. Mastering Modern Web Penetration Testing By Prakhar Prasad, October 2016. 3. Kali Linux 2: Windows Penetration Testing, By Wolf Halton, Bo Weaver, June 2016.	

24CS908	ENGINEERING SECURE SOFTWARE SYSTEMS	L	T	P	C
		3	0	0	3
OBJECTIVES: The Course will enable learners to: • To Know the importance and need for software security. • To Know about various attacks. • To Learn about secure software design. • To Understand risk management in secure software development. • To Know the working of tools related to software security.					
UNIT I	NEED OF SOFTWARE SECURITY AND LOW-LEVEL ATTACKS				9

Software Assurance and Software Security - Threats to software security - Sources of software insecurity - Benefits of Detecting Software Security - Properties of Secure Software - Memory Based Attacks: Low-Level Attacks Against Heap and Stack - Defense Against Memory-Based Attacks		
UNIT II	SECURE SOFTWARE DESIGN	9
Requirements Engineering for secure software- SQUARE process Model- Tools- SQUARE Final Results- Requirement Elicitation and Prioritization- The Critical Role of Architecture and Design- Issues and Challenges - Software Characterization - Threat Analysis - Architectural Vulnerability Assessment.		
UNIT III	SECURITY RISK MANAGEMENT	9
Risk Management Life Cycle - Risk Profiling - Risk Exposure Factors - Risk Evaluation and Mitigation - Risk Assessment Techniques - Threat and Vulnerability Management.		
UNIT IV	SECURITY TESTING	9
Software Security Testing- Contrasting Software Testing and Software Security Testing- Functional Testing- Risk-Based Testing-Secure Software Development Life Cycle- Unit Testing, Testing Libraries and Executable Files, Integration Testing, System Testing.		
UNIT V	SECURE PROJECT MANAGEMENT	9
Governance and security - Adopting an enterprise software security framework - Security and project management - Maturity of Practice-Case Study: Implement the SQL Injection attack and Buffer Overflow attack.		
TOTAL: 45 PERIODS		
OUTCOMES: Upon completion of the course, the students will be able to:		
CO1: Analyze low-level memory attacks and implement corresponding defenses. CO2: Implement requirements engineering and architectural vulnerability assessment. CO3: Evaluate and mitigate risks throughout the software development lifecycle. CO4: Implement various testing techniques to ensure software security CO5: Apply enterprise security frameworks in project governance. CO6: Analyze case studies to understand real-world security threats in project management.		
TEXTBOOKS:		
1. Julia H. Allen, "Software Security Engineering", Pearson Education, 2009. 2. Evan Wheeler, "Security Risk Management: Building an Information Security Risk Management Program from the Ground Up", First edition, Syngress Publishing, 2011.		
REFERENCES:		

1. Rajib Mall," Fundamentals Of Software Engineering", 5th Edition, PHI Learning, 2018.
2. Jon Erickson,"Hacking:The Art of Exploitation",2nd Edition, No Starch Press, 2008.
3. Chris Wysopal, Lucas Nelson, Dino Dai Zovi, and Elfriede Dustin, "The Art of Software Security Testing: Identifying Software Security Flaws (Symantec Press)", Addison-Wesley Professional, 2006.
4. Mike Shema,"Hacking Web Apps: Detecting and Preventing Web Application Security Problems", First Edition, Syngress Publishing,2012.
5. Bryan Sullivan and Vincent Liu,"Web Application Security, A Beginner's Guide",Kindle Edition, McGraw Hill,2012.
6. Lee Allen,"Advanced Penetration Testing for Highly-Secured Environments: The Ultimate Security Guide(Open Source:Community Experience Distilled)", Kindle Edition, Packt Publishing,2012.

24CS909	NETWORK DESIGN AND PROGRAMMING	L	T	P	C
		3	0	0	3
OBJECTIVES: The Course will enable learners to: • To understand the basic networking principles. • To explore various networking devices and protocols required for network design and management. • To Gain knowledge in logical and physical designs for scalable LAN and WAN networks • To study two novel networking technologies: SDN and DTN. • To learn network programming in UNIX C.					
UNIT I	NETWORKING PRINCIPLES				9
Advanced multiplexing - Code Division Multiplexing, DWDM and OFDM - Shared media networks - Collision detection and collision avoidance, Hidden and Exposed Terminals - Switched networks - Datagrams, Virtual circuits, Cell switching and Label switching - Wireless Networks - Infrastructure based, ad hoc and hybrid - End to end semantics - Connectionless, Connection oriented, Wireless Scenarios -Applications, Quality of Service - End to end level and network level solutions.					
UNIT II	PHYSICAL NETWORK DESIGN				9
LAN cabling topologies - Ethernet Switches - High speed and Gigabit and 10Gbps - Building cabling topologies and Campus cabling topologies - Routers, Firewalls and L3 switches -Remote Access Technologies and Devices - Modems and DSLs - SLIP and PPP - WAN Design and Enterprise Networks - Core networks, distribution networks and access networks					
UNIT III	LOGICAL DESIGN AND MANAGEMENT				9
IPv4 and IPv6 Dynamic Addressing -Hierarchical routing - VLSMand CIDR - Transition from IPv4 to IPv6 - NAT and DHCP - Static and Dynamic routes - RIP, OSPF and BGP - VPN -RMON and SNMP					
UNIT IV	INNOVATIVE NETWORKS				9

Software Defined Networks - Evolution of switches and control planes - Centralized and distributed data and control planes - OpenFlow and SDN Controllers - Network Function Virtualization - Needs of the Data Centres - SDN solutions for data centres - Delay Tolerant Networks - Overlay architecture - Bundle Protocol - Opportunistic routing and Epidemic routing		
UNIT V	NETWORK PROGRAMMING IN UNIX C	9
Socket address structures - Byte ordering and byte manipulation functions - Elementary TCP sockets - socket, connect, bind, listen, accept and close functions - TCP client and server - Elementary UDP sockets -recvfrom and sendto functions, connect function with UDP - Raw sockets - Client-server design alternatives - Iterative and Concurrent servers.		
TOTAL: 45 PERIODS		
OUTCOMES: Upon completion of the course, the students will be able to: CO1: Understand advanced multiplexing methods like DWDM and OFDM. CO2: Evaluate network protocols for efficient data transmission. CO3: Develop logical and physical designs for scalable LAN and WAN networks. CO4: Apply strategies for transitioning from IPv4 to IPv6. CO5: Investigate Software Defined Networks and Delay Tolerant Networks. CO6: Gain proficiency in network programming using socket APIs in C.		
TEXTBOOKS:		
1. Larry Peterson and Bruce Davie, "Computer Networks: A Systems Approach", 5th edition, Morgan Kauffman, 2011 2. ParitoshPuri, M.P.Singh,"A survey paper on routing in delay tolerant networks", International Conference on Information and Computer Networks (ISCON), 2013.		
REFERENCES:		
1. Paul Goransson, Chuck Black, "Software Defined Networks: A Comprehensive Approach", Morgan Kauffman, 2016. 2. W.Richard Stevens, Bill Fenner and Andrew M Rudoff, "Unix Network Programming: The Sockets Networking API: Volume 1 ", 3rd Edition, Addison Wesley, 2003. 3. Ying Dar Lin, Ren-Hung Hwang and Fred Baker, "Computer Networks: An Open Source Approach", McGraw Hill, 2011.		

24CS910	FAULT TOLERANT COMPUTING	L	T	P	C
		3	0	0	3
OBJECTIVES: The Course will enable learners to: • To create understanding of the fundamental concepts of fault tolerance systems • To learn basic techniques for achieving fault tolerance in hardware • To have in-depth understanding in software fault tolerance systems • To gain knowledge in design & testing of fault tolerance systems • To develop skills in modelling and evaluating fault tolerant architectures in Real time systems					
UNIT I	INTRODUCTION	9			

Faults, Errors and Failures - Levels of Fault tolerance - Dependability measures - Dependability evaluation - Fault Tolerant techniques - Hardware redundancy - Information redundancy - Software redundancy- Time redundancy -Software Testing.		
UNIT II	HARDWARE FAULT TOLERANCE	9
The Rate of Hardware Failures - Failure Rate, Reliability, and Mean Time to Failure - Canonical and Resilient Structures - Poisson Processes - Markov Models Fault-Tolerance Processor-Level Techniques - Byzantine failures.		
UNIT III	SOFTWARE FAULT TOLERANCE	9
Single-Version Fault Tolerance - N Version programming - Recovery Block Approach - Exception-Handling - Software Reliability Models - Check pointing - Optimal Checkpointing - Checkpointing in Distributed Systems, Shared-Memory Systems and Real-Time Systems.		
UNIT IV	DESIGN DIVERSITY & TESTING	9
Fault tolerant Control and coordination algorithms design - F-T system abstractions and functions- Pitfalls- Practical application- Modeling and analysing F-T Distributed systems - Software fault insertion testing- Fault manager- Categorization of Software faults, Errors, and failures- SIFT methodology and Test plans		
UNIT V	FAULT TOLERANCE IN REAL TIME SYSTEMS	9
Time- Space tradeoff - Fault tolerant scheduling algorithms - Fault tolerant ATM switches - Fault tolerant Routing and sparing Techniques - Yield and reliability enhancement for VLSI/WSI array processors. Case studies: Non-stop systems, Stratus systems, Cassini command and data sub system, IBM G5, Itanium		
TOTAL: 45 PERIODS		
OUTCOMES: Upon completion of the course, the students will be able to: CO1: Understand the need for fault tolerance systems. CO2: Evaluate hardware fault tolerance techniques and their reliability metrics. CO3: Apply software redundancy and fault tolerance methods in program design. CO4: Develop fault-tolerant algorithms and architectures for dependable systems. CO5: Design and implement fault injection testing methodologies for software reliability. CO6: Implement fault-tolerant algorithms for real-time applications and analyze their tradeoffs.		
TEXTBOOKS:		
1. E.Dubrova, "Fault-Tolerant Design", Springer, 2013. 2. I. Korenand, M.Krishna, "Fault Tolerant Systems", Morgan Kaufmann, 2 nd Edition, November 2020.		
REFERENCES:		
1. Kjetil Norvag, "An Introduction to fault tolerant systems", IDI Technical report, July 2000. 2. Olga Goloubeva , Maurizio Rebaudengo , Matteo Sonza Reorda , Massimo Violante, "Software-Implemented Hardware Fault Tolerance", Springer, 2006.		

24CS925	WEB APPLICATION SECURITY	L	T	P	C
		3	0	0	3
OBJECTIVES: The Course will enable learners to: - Understand the fundamentals of web application security - Focus on wide aspects of secure development and deployment of web applications - Learn how to build secure APIs - Learn the basics of vulnerability assessment and penetration testing - Get an insight about Hacking techniques and Tools					
UNIT I	FUNDAMENTALS OF WEB APPLICATION SECURITY				9
The history of Software Security-Recognizing Web Application Security Threats, Web Application Security, Authentication and Authorization, Secure Socket layer, Transport layer Security, Session Management-Input Validation.					
UNIT II	SECURE WEB APPLICATIONS AND APIs				9
Introduction to Web & API Security- Security Testing - Security Incident Response Planning - Secure Development Frameworks - Authentication and Session Management - API Security Fundamentals - Advanced API Security Mechanisms - Securing Service-to-Service APIs - Securing Microservice APIs - Securing Incoming Requests - Case Study: Vulnerability Identification using OWASP ZAP.					
UNIT III	VULNERABILITY ASSESSMENT AND PENETRATION TESTING				9
Vulnerability Assessment Lifecycle, Vulnerability Assessment Tools: Cloud-based vulnerability scanners, Host-based vulnerability scanners, Network-based vulnerability scanners, Database- based vulnerability scanners, Types of Penetration Tests: External Testing, Web Application Testing, Internal Penetration Testing, SSID or Wireless Testing, Mobile Application Testing.					
UNIT IV	HACKING TECHNIQUES AND TOOLS				9
Social Engineering, Injection, Cross-Site Scripting (XSS), Broken Authentication and Session Management, Cross-Site Request Forgery, Security Misconfiguration, Insecure Cryptographic Storage, Failure to Restrict URL Access, Tools: Comodo, OpenVAS, Nexpose, Nikto, Burp Suite, etc. Case Study: Attack the website using Social Engineering method.					
UNIT V	SECURITY FOR AI-ENABLED APPLICATIONS				9
Introduction to AI Security - AI Attack Surface - Data Security in AI - Common AI Threats - Securing AI Applications - API and Model Security - Testing and Monitoring AI Systems - Responsible and Ethical AI - Use Case: Secure Test Data Generation System (Scenario: AI generates test data for an application)					
TOTAL: 45 PERIODS					

COURSE OUTCOMES:

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

CO1: Explain the basic concepts of web application security and the need for it.

CO2: Develop a thorough understanding of the process for securely designing, implementing, testing, and deploying web applications.

CO3: Acquire the skill to design and develop Secure Web Applications that use Secure APIs.

CO4: Develop the ability to grasp the importance of conducting vulnerability assessments and penetration testing.

CO5: Acquire the skill to think like a hacker and to use hackers tool sets.

CO6: Develop the ability to implement and manage web application security measures effectively, ensuring ongoing protection against evolving threats.

TEXT BOOKS:

1. Andrew Hoffman, Web Application Security: Exploitation and Countermeasures for Modern Web Applications, First Edition, 2020, O'Reilly Media, Inc.
2. Bryan Sullivan, Vincent Liu, Web Application Security: A Beginners Guide, 2012, The McGraw-Hill Companies. **(Unit I, II, IV)**
3. Neil Madden, API Security in Action, 2020, Manning Publications Co., NY, USA. **(Unit II)**
4. Rishabh Bhardwaj, "Vulnerability Assessment and Penetration Testing (VAPT): Detailed guide with highlighted threats, risk exposure, and remediations", 1st Edition, BPB Publications, 2025 **(Unit III)**

REFERENCES:

1. Michael Cross, Developer's Guide to Web Application Security, 2007, Syngress Publishing, Inc.
2. Ravi Das and Greg Johnson, Testing and Securing Web Applications, 2021, Taylor & Francis Group, LLC.
3. Prabath Siriwardena, Advanced API Security, 2020, Apress Media LLC, USA.
4. Malcom McDonald, Web Security for Developers, 2020, No Starch Press, Inc.
5. Allen Harper, Shon Harris, Jonathan Ness, Chris Eagle, Gideon Lenkey, and Terron Williams Grey Hat Hacking: The Ethical Hacker's Handbook, Third Edition, 2011, The McGraw-Hill Companies.

HONOURS – CLOUD COMPUTING

24CS915	MACHINE LEARNING FOR NLP IN CLOUD	L	T	P	C
		3	0	0	3
OBJECTIVES: • Illustrate how to apply the ML pipeline to NLP. • Implement text extraction to obtain data from web pages. • Build a solution that uses AWS services to transcribe and translate text from multimedia. • Build a solution using a combination of algorithms and Amazon Machine Learning (Amazon ML) services. • Identify use cases to use generative AI and LLMs. • Use LLMs with AWS generative AI services.					
UNIT I	INTRODUCTION TO NLP	8			
NLP - Business Problems Solved by NLP - NLP Roles - NLP and ML - Common NLP tasks - Apply ML to NLP problem - Evolution of NLP architectures.					
UNIT II	PROCESSING TEXT FOR NLP	10			
Text processing overview - Getting text - Extracting Text from Webpages and Images - Text preprocessing - Vectorizing text - Encoding and Vectorizing Text - Advanced processing - Storing and visualizing unstructured data - Implement Sentiment Analysis: Identifying the steps for text processing - Examining the algorithms for sentiment analysis.					
UNIT III	INFORMATION EXTRACTION	9			
Information extraction overview - Types of information extraction - Implementing information extraction - Working with Entities - Topic Modeling - Identifying the approach - Implementing Topic Modeling with Amazon Comprehend, Neural Topic Model (NTM).					
UNIT IV	TRANSLATING LANGUAGES	9			
Working with language issues - Detecting and translating languages - Transcribing and vocalizing text with AWS services - Implementing a Multilingual Solution.					
UNIT V	GENERATIVE AI	9			
Generative AI - Amazon Bedrock Overview - Introducing foundations models and large language models - Transformer architecture - LLMs configuration parameters - Introducing prompt engineering - Use LLMs to Perform NLP Tasks - Adapting LLMs - Application Integration.					
TOTAL: 45 PERIODS					
OUTCOMES: At the end of this course, the students will be able to: CO1: Understand basic concepts and applications of NLP. CO2: Apply text preprocessing and vectorization techniques. CO3: Analyze and perform sentiment analysis on text data. CO4: Apply information extraction and topic modeling methods. CO5: Implement language translation and speech-based NLP tasks. CO6: Use Generative AI and LLMs for NLP applications.					

REFERENCES:

1. Mona M, Premkumar Rangarajan, Natural Language Processing with AWS AI Services, Packt Publications, 2021.
2. Saket S R Mengle, Maximo Gurmendez, Mastering Machine Learning on AWS: Advanced machine learning in Python using SageMaker, Apache Spark, and TensorFlow, Packt Publications, 2019.
3. AWS Documentation (amazon.com)
4. AWS Skill Builder
5. AWS Academy Machine Learning for Natural Language Processing Course - https://www.awsacademy.com/vforcesite/LMS_Login

24CS916	MICROSERVICE ARCHITECTURE	L	T	P	C
		3	0	0	3
OBJECTIVES: The Course will enable learners to: • Learn the Concepts of Micro Service Architecture • Understand the inter process communication in a microservice architecture • Design business logic in a microservice architecture • Develop business logic with event sourcing • Implement queries in a microservice architecture					
UNIT I	INTRODUCTION TO MICROSERVICE ARCHITECTURE				9
Introduction - Microservice - Microservice Architecture - Comparing Microservice Architecture with SOA - Benefits and Drawbacks of Microservice Architecture - Microservice Architecture Pattern Language - Beyond microservices -Process and Organization - Microservice Architecture Style - Application Microservice Architecture.					
UNIT II	INTERPROCESS COMMUNICATION AND TRANSACTION MANAGEMENT				9
Overview of Inter process Communication - Synchronous Remote Procedure Invocation Pattern - Asynchronous Messaging Pattern - Asynchronous Messaging to Improve Availability - Transaction Management in a Microservice Architecture - Coordinating sagas - Handling the Lock of Isolation - Order Service and Order Saga.					
UNIT III	DESIGN AND DEVELOPMENT OF BUSINESS LOGIC				9
Business logic organization patterns - Designing a domain model using the DDD aggregate pattern - Publishing domain events - Kitchen Service business logic - Order Service business logic - Developing business logic with event sourcing - Implementing an event store - Using sagas and event sourcing together.					
UNIT IV	QUERY IMPLEMENTATION, EXTERNAL API PATTERNS AND MICROSERVICE TESTING				9
Querying using the API composition pattern - Using the CQRS pattern - Designing CQRS views - Implementing a CQRS view with AWS DynamoDB - External API design issues - The API gateway pattern - Implementing an API gateway - Testing strategies for microservice architectures - Writing unit tests for a service - Writing integration tests - Developing component tests - Writing end-to-end tests.					

UNIT V	MICROSERVICES DEVELOPMENT AND DEPLOYMENT	9
Developing secure services - Designing configurable services - Designing observable services - Developing services using the Microservice chassis pattern - Deploying Microservices - Language-specific packaging format pattern - Service as a virtual machine pattern - Service as a container pattern - Serverless deployment - Refactoring to microservices.		
TOTAL : 45 PERIODS		
OUTCOMES: Upon completion of the course, the students will be able to: CO1: Develop microservice applications. CO2: Deploy microservices in different platforms CO3: Create integration and component testing for microservices CO4: Design External API patterns for microservices CO5: Develop inter process communication mechanisms in microservice architecture CO6: Create transaction management in microservice.		
TEXT BOOKS:		
1. Chris Richardson, "Microservices Pattern with Examples in JAVA", Manning Publications, 1 st edition, 2019.		
REFERENCES:		
1. Irakli Nadareishvili, Ronnie Mitra, Matt McLarty & Mike Amundsen, "Microservice Architecture ALIGNING PRINCIPLES, PRACTICES, AND CULTURE," 1 st edition, O'Reilly Publication, 2016. 2. Eberhard Wolff, "Microservices Flexible Software Architectures," 1 st edition, Addison- Wesley, 2016. 3. John Kernal, Illary Huaylupo Sanchez, "Micro Service in Action", 2 nd edition, Manning Publication, 2018. 4. Susan J Fowler, "Production-Ready Microservices", O'Reilly Publication, 2016. 5. Sam Newman, "Building Microservices: Designing Fine-Grained Systems", 2 nd edition, O'Reilly Publication, 2021.		

24CS917	SOFTWARE DEFINED NETWORKS	L	T	P	C
		3	0	0	3
OBJECTIVES: • To understand the need for SDN and its data plane operations • To understand the functions of control plane • To comprehend the migration of networking functions to SDN environment • To explore various techniques of network function virtualization • To comprehend the concepts behind network virtualization					
UNIT I	SDN: INTRODUCTION				9
Evolving Network Requirements - The SDN Approach - SDN architecture - SDN Data Plane, Control plane and Application Plane.					
UNIT II	SDN DATA PLANE AND CONTROL PLANE				9
Data Plane functions and protocols - OpenFlow Protocol - Flow Table - Control Plane Functions - Southbound Interface, Northbound Interface - SDN Controllers - Ryu, OpenDaylight, ONOS - Distributed Controllers.					
UNIT III	SDN APPLICATIONS				9

SDN Application Plane Architecture - Network Services Abstraction Layer - Traffic Engineering - Measurement and Monitoring - Security - Data Center Networking.		
UNIT IV	NETWORK FUNCTION VIRTUALIZATION	9
Network Virtualization - Virtual LANs - OpenFlow VLAN Support - NFV Concepts - Benefits and Requirements - Reference Architecture.		
UNIT V	NFV FUNCTIONALITY	9
NFV Infrastructure - Virtualized Network Functions - NFV Management and Orchestration - NFV Use cases - SDN and NFV.		
TOTAL: 45 PERIODS		
OUTCOMES: At the end of this course, the students will be able to: CO1: Describe the motivation behind SDN CO2: Identify the functions of the data plane and control plane CO3: Design and develop network applications using SDN CO4: Orchestrate network services using NFV CO5: Explain various use cases of SDN and NFV CO6: Evaluate the impact of SDN and NFV on modern networking infrastructures and their potential for future innovations.		
TEXTBOOKS: 1. William Stallings, "Foundations of Modern Networking: SDN, NFV, QoE, IoT and Cloud", Pearson Education, 1st Edition, 2015.		
REFERENCES: 1. Ken Gray, Thomas D. Nadeau, "Network Function Virtualization", Morgan Kaufman, 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. Paul Goransson, Chuck Black Timothy Culver, "Software Defined Networks: A Comprehensive Approach", 2nd Edition, Morgan Kaufmann Press, 2016. 5. Oswald Coker, Siamak Azodolmolky, "Software-Defined Networking with OpenFlow", 2nd Edition, O'Reilly Media, 2017.		

24CS918	STORAGE TECHNOLOGIES	L	T	P	C
		3	0	0	3
OBJECTIVES:					
● Characterize the functionalities of logical and physical components of storage ● Describe various storage networking technologies ● Identify different storage virtualization technologies ● Discuss the different backup and recovery strategies ● Understand common storage management activities and solutions					
UNIT I	STORAGE SYSTEMS				9

Introduction to Information Storage: Digital data and its types, Information storage, Key characteristics of data center and Evolution of computing platforms. Information Lifecycle Management. Third Platform Technologies: Cloud computing and its essential characteristics, Cloud services and cloud deployment models, Big data analytics, Social networking and mobile computing, Characteristics of third platform infrastructure and Imperatives for third platform transformation. Data Center Environment: Building blocks of a data center, Compute systems and compute virtualization and Software-defined data center.		
UNIT II	INTELLIGENT STORAGE SYSTEMS AND RAID	8
Components of an intelligent storage system, Components, addressing, and performance of hard disk drives and solid-state drives, RAID, Types of intelligent storage systems, Scale-up and scale-out storage Architecture. Block-Based Storage System, File-Based Storage System, Object-Based and Unified Storage.		
UNIT III	STORAGE NETWORKING TECHNOLOGIES AND VIRTUALIZATION	10
Fibre Channel SAN: Software-defined networking, FC SAN components and architecture, FC SAN topologies, link aggregation, and zoning, Virtualization in FC SAN environment. Internet Protocol SAN: iSCSI protocol, network components, and connectivity, Link aggregation, switch aggregation, and VLAN, FCIP protocol, connectivity, and configuration. Fibre Channel over Ethernet SAN: Components of FCoE SAN, FCoE SAN connectivity, Converged Enhanced Ethernet, FCoE architecture.		
UNIT IV	BACKUP, ARCHIVE AND REPLICATION	10
Introduction to Business Continuity, Backup architecture, Backup targets and methods, Data deduplication, Cloud-based and mobile device backup, Data archive, Uses of replication and its characteristics, Compute based, storage-based, and network-based replication, Data migration, Disaster Recovery as a Service (DRaaS).		
UNIT V	SECURING STORAGE INFRASTRUCTURE	8
Information security goals, Storage security domains, Threats to a storage infrastructure, Security controls to protect a storage infrastructure, Governance, risk, and compliance, Storage infrastructure management functions, Storage infrastructure management processes.		
TOTAL: 45 PERIODS		
OUTCOMES: At the end of this course, the students will be able to: CO1: Demonstrate the fundamentals of information storage management and various models of Cloud infrastructure services and deployment CO2: Illustrate the usage of advanced intelligent storage systems and RAID CO3: Interpret various storage networking architectures - SAN, including storage subsystems and virtualization CO4: Examine the different role in providing disaster recovery and remote replication technologies CO5: Infer the security needs and security measures to be employed in information storage management CO6: Apply integrated knowledge of storage technologies to design effective storage solutions aligned with organizational requirements.		

TEXTBOOKS:

1. EMC Corporation, Information Storage and Management, Wiley, India.

2. Jon Tate, Pall Beck, Hector Hugo Ibarra, Shanmuganathan Kumaravel and Libor Miklas, Introduction to Storage Area Networks, Ninth Edition, IBM - Redbooks, December 2017.

3. Ulf Troppens, Rainer Erkens, Wolfgang Mueller-Friedt, Rainer Wolafka, Nils Haustein, Storage Networks Explained, Second Edition, Wiley, 2009.

24CS919

CLOUD SECURITY FOUNDATIONS

L

T

P

C

3

0

0

3

OBJECTIVES:

The Course will enable learners to:

Learn the basics of cloud security, including the shared responsibility model and identity management.

Set up a secure cloud infrastructure with features like virtual private clouds and security groups.

Safeguard application data at rest and in transit using encryption and Amazon S3 protection features.

Learn to capture and analyze log data using AWS services like CloudTrail and CloudWatch.

Develop skills for identifying and managing security incidents in the cloud, adhering to best practices.

UNIT I

SECURITY IN CLOUD

9

Introduction to Security, Security in the Cloud, Security design principles, Shared responsibility model, Activity: Shared Responsibility Model, Identity and Access Management (IAM) fundamentals, Authenticating and Authorizing with IAM, Examples of authorizing with IAM, Additional authentication and access management services, Using Organizations.

UNIT II

SECURING INFRASTRUCTURE

9

Structure of a three-tier web application, virtual private cloud (VPC), Setting up public and private subnets and internet protocols, Security groups, Network access control lists (ACLs), Load balancers, Protecting compute resources.

UNIT III

PROTECTING APPLICATION DATA

9

Basics on Data Protection, Protect data at rest, Amazon S3 protection features, Protection through encryption, Protect data in transit, protect data in Amazon S3, additional data protection services.

UNIT IV

LOGGING AND MONITORING

9

Importance of logging and monitoring, Capture and collect, Reading a Log File, AWS services with built-in logs, Monitor and report, CloudTrail and Amazon CloudWatch, methods for logging and monitoring, additional AWS services for logging and monitoring, AWS Security Hub.

UNIT V

RESPONDING AND MANAGING AN INCIDENT

9

Identifying an incident, Services that support the discovery and recognition phase, AWS Config and AWS Lambda, Services that support the resolution and recovery phase, Best practices for handling an incident.

TOTAL: 45 PERIODS

OUTCOMES:	
Upon completion of the course, the students will be able to:	
CO1: Learn essential cloud security principles and identity management fundamentals.	
CO2: Design and implement secure cloud infrastructure components for effective resource protection.	
CO3: Explore methods for securing application data stored in the cloud, including encryption techniques and data protection features.	
CO4: Gain proficiency in implementing logging and monitoring practices to detect and respond to security events efficiently in cloud environments.	
CO5: Acquire skills to identify and manage security incidents in the cloud, utilizing appropriate tools and techniques for incident detection, analysis, and resolution.	
CO6: Apply comprehensive cloud security principles and practices to design and manage secure cloud environments.	
TEXT BOOKS:	
1. Tim Mather, Subra Kumaraswamy, Shahed Latif, "Cloud Security and Privacy, An Enterprise Perspective on Risks and Compliance", Oreilly Media 2009. 2. Vic (J.R.) Winkler, "Securing the Cloud, Cloud Computer Security Techniques and Tactics", Syngress, April 2011.	
REFERENCES:	
1. Rajkumar Buyya, James Broberg, Andrzej, "Cloud Computing: Principles and Paradigms", Wiley India Publications 2011. 2. Arshdeep Bahga and Vijay Madisetti, "Cloud Computing -A Hands on Approach", Universities Press (India) Pvt Ltd. 2014.	

24CS920	CLOUD SERVICES MANAGEMENT	L	T	P	C
		3	0	0	3
OBJECTIVES:					
✓ Introduce Cloud Service Management terminology, definition & concepts ✓ Compare and contrast cloud service management with traditional IT service management ✓ Identify strategies to reduce risk and eliminate issues associated with adoption of cloud services ✓ Select appropriate structures for designing, deploying and running cloud-based services in a business environment ✓ Illustrate the benefits and drive the adoption of cloud-based services to solve real world problems					
UNIT I	CLOUD SERVICE MANAGEMENT FUNDAMENTALS				9

Cloud Ecosystem, The Essential Characteristics, Basics of Information Technology Service Management and Cloud Service Management, Service Perspectives, Cloud Service Models, Cloud Service Deployment Models.

UNIT II	CLOUD SERVICES STRATEGY	9
----------------	--------------------------------	----------

Cloud Strategy Fundamentals, Cloud Strategy Management Framework, Cloud Policy, Key Driver for Adoption, Risk Management, IT Capacity and Utilization, Demand and Capacity matching, Demand Queueing, Change Management, Cloud Service Architecture.

UNIT III	CLOUD SERVICE MANAGEMENT	9
-----------------	---------------------------------	----------

Cloud Service Reference Model, Cloud Service LifeCycle, Basics of Cloud Service Design, Dealing with Legacy Systems and Services, Benchmarking of Cloud Services, Cloud Service Capacity Planning, Cloud Service Deployment and Migration, Cloud Marketplace, Cloud Service Operations Management.

UNIT IV	CLOUD SERVICE ECONOMICS	9
----------------	--------------------------------	----------

Pricing models for Cloud Services, Freemium, Pay Per Reservation, Pay per User, Subscription based Charging, Procurement of Cloud-based Services, Capex vs Opex Shift, Cloud service Charging, Cloud Cost Models.

UNIT V	CLOUD SERVICE GOVERNANCE & VALUE	9
---------------	---	----------

IT Governance Definition, Cloud Governance Definition, Cloud Governance Framework, Cloud Governance Structure, Cloud Governance Considerations, Cloud Service Model Risk Matrix, Understanding Value of Cloud Services, Measuring the value of Cloud Services, Balanced Scorecard, Total Cost of Ownership.

TOTAL: 45 PERIODS

OUTCOMES:

At the end of this course, the students will be able to:

- CO1:** Exhibit cloud-design skills to build and automate business solutions using cloud technologies.
- CO2:** Possess Strong theoretical foundation leading to excellence and excitement towards adoption of cloud-based services
- CO3:** Solve the real world problems using Cloud services and technologies
- CO4:** Develop and deploy services on the cloud and set up a cloud environment
- CO5:** Explain security challenges in the cloud environment
- CO6:** Evaluate the impact of cloud services management strategies on organizational efficiency and scalability.

TEXT BOOKS:

1. Cloud Service Management and Governance: Smart Service Management in Cloud Era by Enamul Haque, Enel Publications, 2023.
2. Cloud Computing: Concepts, Technology & Architecture by Thomas Erl, Ricardo Puttini, Zaigham Mohammad 2013.
3. Cloud Computing Design Patterns by Thomas Erl, Robert Cope, Amin Naserpour.

REFERENCES:

1. Economics of Cloud Computing by Praveen Ayyappa, LAP Lambert Academic Publishing.
2. Mastering Cloud Computing Foundations and Applications Programming Rajkumar Buyya, Christian Vechhiola, S. Thamarai Selvi.

HONOURS – FULL STACK TECHNOLOGY

24CS916	MICROSERVICE ARCHITECTURE	L	T	P	C
		3	0	0	3
OBJECTIVES: The Course will enable learners to: - Learn the Concepts of Micro Service Architecture - Understand the inter process communication in a microservice architecture - Design business logic in a microservice architecture - Develop business logic with event sourcing - Implement queries in a microservice architecture					
UNIT I	INTRODUCTION TO MICROSERVICE ARCHITECTURE	9			
Introduction - Microservice - Microservice Architecture - Comparing Microservice Architecture with SOA - Benefits and Drawbacks of Microservice Architecture - Microservice Architecture Pattern Language - Beyond microservices -Process and Organization - Microservice Architecture Style - Application Microservice Architecture.					
UNIT II	INTERPROCESS COMMUNICATION AND TRANSACTION MANAGEMENT	9			
Overview of Inter process Communication - Synchronous Remote Procedure Invocation Pattern - Asynchronous Messaging Pattern - Asynchronous Messaging to Improve Availability - Transaction Management in a Microservice Architecture - Coordinating sagas - Handling the Lock of Isolation - Order Service and Order Saga.					
UNIT III	DESIGN AND DEVELOPMENT OF BUSINESS LOGIC	9			
Business logic organization patterns - Designing a domain model using the DDD aggregate pattern - Publishing domain events - Kitchen Service business logic - Order Service business logic - Developing business logic with event sourcing - Implementing an event store - Using sagas and event sourcing together.					
UNIT IV	QUERY IMPLEMENTATION, EXTERNAL API PATTERNS AND MICROSERVICE TESTING	9			
Querying using the API composition pattern - Using the CQRS pattern - Designing CQRS views - Implementing a CQRS view with AWS DynamoDB - External API design issues - The API gateway pattern - Implementing an API gateway - Testing strategies for microservice architectures - Writing unit tests for a service - Writing integration tests - Developing component tests - Writing end-to-end tests.					
UNIT V	MICROSERVICES DEVELOPMENT AND DEPLOYMENT	9			

Developing secure services - Designing configurable services - Designing observable services - Developing services using the Microservice chassis pattern - Deploying Microservices - Language-specific packaging format pattern - Service as a virtual machine pattern - Service as a container pattern - Serverless deployment - Refactoring to microservices.
TOTAL : 45 PERIODS
OUTCOMES: Upon completion of the course, the students will be able to: CO1: Develop microservice applications. CO2: Deploy microservices in different platforms CO3: Create integration and component testing for microservices CO4: Design External API patterns for microservices CO5: Develop inter process communication mechanisms in microservice architecture CO6: Create transaction management in microservice.
TEXT BOOKS: 1. Chris Richardson, "Microservices Pattern with Examples in JAVA", Manning Publications, 1 st edition, 2019.
REFERENCES: 1. Irakli Nadareishvili, Ronnie Mitra, Matt McLarty & Mike Amundsen, "Microservice Architecture ALIGNING PRINCIPLES, PRACTICES, AND CULTURE," 1 st edition, O'Reilly Publication, 2016. 2. Eberhard Wolff, "Microservices Flexible Software Architectures," 1 st edition, Addison- Wesley, 2016. 3. John Kernal, Illary Huaylupo Sanchez, "Micro Service in Action", 2 nd edition, Manning Publication, 2018. 4. Susan J Fowler, "Production-Ready Microservices", O'Reilly Publication, 2016. 5. Sam Newman, "Building Microservices: Designing Fine-Grained Systems", 2 nd edition, O'Reilly Publication, 2021.

24CS924	MERN STACK DEVELOPMENT (Lab Integrated)	L	T	P	C
		2	0	2	3
OBJECTIVES: The Course will enable learners: ▮ Design applications using Node .JS ▮ Create architecture involving Express and graphQL ▮ Develop applications using mongoDB ▮ Apply the concepts of React Components and State ▮ Build web applications using React Router, Forms and Bootstrap					
UNIT I	INTRODUCTION TO MERN and NODE JS				6+6

Introduction - MERN Components - Node JS: Introduction to Node JS, Setting up Node.js, Node.js Modules - HTTP Servers and Clients - Request Handling - Database connectivity - Data Storage and Retrieval - Dynamic Client/Server Interaction with Socket.IO

List of Exercise/Experiments:

1. Create your own modules and return Current date and time.
2. Create the HTTP server using createServer() method that listens to server ports and gives a response back to the client.

UNIT II	EXPRESS	6+6
----------------	----------------	------------

Express - Routing - Request Matching - Route parameters - Route Lookup - Handler Function Request Object - Response Object - Middleware - REST API - GraphQL - About API - List API List API Integration - Custom Scalar Types - Create API Integration - Query Variables - Input Validations - Displaying Errors

List of Exercise/Experiments:

1. Create an application using Express.js to print Hello world on the Homepage.
2. Build a Simple Node.js/Express server that handles GET and POST request and returns data in JSON format.

UNIT III	MongoDB	6+6
-----------------	----------------	------------

MongoDB Basics - CRUD Operations - NODE.js driver - Schema Initialization - Reading from MongoDB - Writing to MongoDB - UI Server - Multiple Environments - Proxy-based Architecture - ESLint - ESLint for Front End - React PropTypes - Back End Modules - Front End Modules and Webpack - Transform and bundle - Libraries Bundle - Module Replacement Debugging - Define plugin - Product Optimization.

List of Exercise/Experiments:

1. Build an application to perform Basic CRUD operation in MongoDB using Node/Express.
2. Building the MongoDB database for the My To-do List app.

UNIT IV	REACT COMPONENTS AND STATE	6+6
----------------	-----------------------------------	------------

React Components- Issue Tracker - React Classes - Composing Components - Passing Data
- Dynamic Composition - React State - Hooks - Event handling - Stateless Components Designing Components.

List of Exercise/Experiments:

1. Write a program to create a simple calculator Application using React JS
2. Build a simple React application that displays the list of items and allows the user to add new items to the list.

UNIT V	REACT ROUTER, FORMS AND BOOTSTRAP	6+6
---------------	--	------------

React Router - Simple Routing - Route Parameters - Query Parameters - Links - Programmatic Navigation - Nested Routes - React Forms - Controlled Components - Specialized Input Components - Update API - Delete API - React Bootstrap - Buttons - Navigation - Panels - Tables - Forms - Grid - Inline Forms - Horizontal Forms - Validation Alerts - Toasts - Modals.

List of Exercise/Experiments:

1. Create a Simple Login form using React JS.
2. Build an application for an E-Commerce platform.
3. Build a full-stack MERN app that allows the user to register, login, and create a list of items that are stored in a MongoDB database. The app should also display the list of items using React components.

TOTAL: 30(T) + 30(P) = 60 PERIODS

OUTCOMES:

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

CO1: Develop applications using Node.js for backend functionality and server-side logic.

CO2: Handle data queries with GraphQL on an Express server for efficient API management.

CO3: Build applications using MongoDB to perform CRUD operations and manage databases.

CO4: Apply dynamic composition and event handling techniques.

CO5: Implement React forms and Bootstrap for creating responsive, user-friendly interfaces.

CO6: Design and develop full-stack applications with the MERN stack.

TEXT BOOKS:

1. Vasan Subramanian, Pro MERN Stack - Full stack web app development, 2nd Edition, Apress, 2019
2. David Herron, Node.js Web Development - Fourth Edition, Packt Publishing, 2018.

REFERENCES:

1. Adam Freeman, Essential TypeScript, Apress, 2019.
2. Shama Hoque, Full-Stack React Projects, 2nd edition, Apress, 2022
3. Karl Seguin, "The Little Mongo DB Book",
<https://github.com/karlseguin/the-little-mongodb-book>.
4. <https://aws.amazon.com/education/awseducate/>
5. <http://packaging.ubuntu.com/html/packaging-new-software.html>
6. https://www.tutorialspoint.com/nodejs/nodejs_express_framework.htm

LIST OF EQUIPMENTS:

Node, Express, MongoDB, React

24CS925	WEB APPLICATION SECURITY	L	T	P	C
		3	0	0	3
OBJECTIVES: The Course will enable learners to: • Understand the fundamentals of web application security • Focus on wide aspects of secure development and deployment of web applications • Learn how to build secure APIs • Learn the basics of vulnerability assessment and penetration testing • Get an insight about Hacking techniques and Tools					
UNIT I	FUNDAMENTALS OF WEB APPLICATION SECURITY				9
The history of Software Security-Recognizing Web Application Security Threats, Web Application Security, Authentication and Authorization, Secure Socket layer, Transport layer Security, Session Management-Input Validation.					
UNIT II	SECURE WEB APPLICATIONS AND APIs				9
Introduction to Web & API Security- Security Testing - Security Incident Response Planning - Secure Development Frameworks - Authentication and Session Management - API Security Fundamentals - Advanced API Security Mechanisms - Securing Service-to-Service APIs - Securing Microservice APIs - Securing Incoming Requests - Case Study: Vulnerability Identification using OWASP ZAP.					
UNIT III	VULNERABILITY ASSESSMENT AND PENETRATION TESTING				9
Vulnerability Assessment Lifecycle, Vulnerability Assessment Tools: Cloud-based vulnerability scanners, Host-based vulnerability scanners, Network-based vulnerability scanners, Database- based vulnerability scanners, Types of Penetration Tests: External Testing, Web Application Testing, Internal Penetration Testing, SSID or Wireless Testing, Mobile Application Testing.					
UNIT IV	HACKING TECHNIQUES AND TOOLS				9
Social Engineering, Injection, Cross-Site Scripting (XSS), Broken Authentication and Session Management, Cross-Site Request Forgery, Security Misconfiguration, Insecure Cryptographic Storage, Failure to Restrict URL Access, Tools: Comodo, OpenVAS, Nexpose, Nikto, Burp Suite, etc. Case Study: Attack the website using Social Engineering method.					
UNIT V	SECURITY FOR AI-ENABLED APPLICATIONS				9
Introduction to AI Security - AI Attack Surface - Data Security in AI - Common AI Threats - Securing AI Applications - API and Model Security - Testing and Monitoring AI Systems - Responsible and Ethical AI - Use Case: Secure Test Data Generation System (Scenario: AI generates test data for an application)					
TOTAL: 45 PERIODS					

COURSE OUTCOMES:

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

CO1: Explain the basic concepts of web application security and the need for it.

CO2: Develop a thorough understanding of the process for securely designing, implementing, testing, and deploying web applications.

CO3: Acquire the skill to design and develop Secure Web Applications that use Secure APIs.

CO4: Develop the ability to grasp the importance of conducting vulnerability assessments and penetration testing.

CO5: Acquire the skill to think like a hacker and to use hackers tool sets.

CO6: Develop the ability to implement and manage web application security measures effectively, ensuring ongoing protection against evolving threats.

TEXT BOOKS:

1. Andrew Hoffman, Web Application Security: Exploitation and Countermeasures for Modern Web Applications, First Edition, 2020, O'Reilly Media, Inc.
2. Bryan Sullivan, Vincent Liu, Web Application Security: A Beginners Guide, 2012, The McGraw-Hill Companies.
3. Neil Madden, API Security in Action, 2020, Manning Publications Co., NY, USA.
4. Rishabh Bhardwaj, " Vulnerability Assessment and Penetration Testing (VAPT): Detailed guide with highlighted threats, risk exposure, and remediations", 1st Edition, BPB Publications, 2025

REFERENCES:

1. Michael Cross, Developer's Guide to Web Application Security, 2007, Syngress Publishing, Inc.
2. Ravi Das and Greg Johnson, Testing and Securing Web Applications, 2021, Taylor & Francis Group, LLC.
3. Prabath Siriwardena, Advanced API Security, 2020, Apress Media LLC, USA.
4. Malcom McDonald, Web Security for Developers, 2020, No Starch Press, Inc.
5. Allen Harper, Shon Harris, Jonathan Ness, Chris Eagle, Gideon Lenkey, and Terron Williams Grey Hat Hacking: The Ethical Hacker's Handbook, Third Edition, 2011, The McGraw-Hill Companies.

24CS927	MOBILE ARCHITECTURE AND APPLICATION DEVELOPMENT (Lab Integrated)	L	T	P	C
		3	0	0	3
OBJECTIVES: The Course will enable learners to: - Understand Android SDK - Understand and develop Android applications - Learn the concepts of iOS application development - Develop Hybrid platform application - Develop cross-platform applications with event handling					
UNIT I	MOBILE ARCHITECTURE AND BASICS IN APP DEVELOPMENT				6+6
Mobile Device Operating Systems – Special Constraints & Requirements – Commercial Mobile Operating Systems – Software Development Kit: iOS, Android, BlackBerry, Mobile App Architecture: Android - iOS - Hybrid - Cross Platform - Factors of Mobile App Architecture - Windows Phone – MCommerce – Structure – Pros & Cons – Mobile Payment System – Security Issues.					
List of Exercise/Experiments 1. evelop an application that uses GUI components, Font, Layout Managers and event listeners. 2. evelop a native application that uses GPS location information.					
UNIT II	ANDROID APPLICATION				6+6
Introduction to Android: The Android Platform - Android SDK, Eclipse Installation, Android Installation, Building First Android application - Life cycle of android application - Android User Interface Design Essentials - Android APIs - Deploying android applications.					
List of Exercise/Experiments 1. Create an android application that converts the user input text to voice. 2. Develop a Mobile application for simple and day to day needs.					
UNIT III	IOS APPLICATION				6+6
Introduction - Performance metrics - Core optimizations - concurrent programming - Application life cycle - User Interface - Network -Security - Testing - Tools - Instrumentation and Analytics					
List of Exercise/Experiments 1. Implement an iOS app allowing users to type or paste text and have it spoken aloud using Apple's speech synthesis. 2. Develop an ios application for simple and day to day needs.					
UNIT IV	HYBRID APP ENVIRONMENT				6+6
Hybrid Web App, Benefits of Hybrid App, Criteria for creating Native App, Tools for creating Hybrid App, Cons of Hybrid App, Popular Hybrid App Development Frameworks, Ionic, Apache Cordova					

List of Exercise/Experiments		
1. 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. 2. Design and develop an android application using Apache Cordova to find and display the current location of the user.		
UNIT V	CROSS-PLATFORM APP DEVELOPMENT	6+6
What is Cross-platform App, Benefits of Cross-platform App, Criteria for creating Cross- platform App, Tools for creating Cross-platform App, Cons of Cross-platform App, Popular Cross-platform App Development Frameworks, Flutter, Xamarin, React-Native, Basics of React Native, Native Components, JSX, State, Props		
List of Exercise/Experiments		
1. Design and develop a cross platform application for day to day task (to-do) management. 2. Build a cross platform application for a simple expense manager which allows entering expenses and income on each day and displays category wise weekly income and expenses.		
TOTAL: 30(L)+30(P) = 60 PERIODS		
COURSE OUTCOMES:		
Upon completion of the course, the students will be able to:		
CO1: Identify various concepts of mobile architecture. CO2: Create, test and debug Android application by setting up Android development CO3: Develop iOS application with performance metrics CO4: Develop hybrid applications with basic event handling. CO5: Implement cross platform applications with basic GUI and event handling. CO6: Deploy applications to the Android Market place for distribution.		
TEXT BOOKS:		
1. Gaurav Vaish, "High Performance iOS Apps", O'Reilly Media, 1st ed. 2016. 2. Lauren Darcey and Shane Conder, "Android Wireless Application Development", Pearson Education, 2nd ed.2012. 3. Prasanth Kumar Pattnaik, Rajib Mall, "Fundamentals of Mobile Computing", PHI Learning Pvt. Ltd, New Delhi-2012.		
REFERENCES:		
1. Pawan Lingras, Matt Triff, Rucha Lingras, "Building Cross-Platform Mobile and Web Apps for Engineers and Scientists: An Active Learning Approach", Cengage Learning, 2016. 2. Mahesh Panhale, Beginning Hybrid Mobile Application Development, Apress,2016. 3. Android Application development James C. Sheusi Cengage learning 2017. 4. Erik Hellman, "Android Programming - Pushing the Limits", 1st Edition, Wiley India Pvt Ltd, 2014.		
LIST OF SOFTWARES:		
1. Software required: Android Studio, macOS. 2. Software Tools required: Android 11 SDK, Xcode.		

24CS943	INTRODUCTION TO AGENTIC AI	L	T	P	C
		3	0	0	3
Objectives:					
The objective of learning this Agentic AI course is to:					
• Learn the basics, types, applications, and core principles of agentic systems. • Understand reflection, planning, tool usage, and practical agent frameworks. • Develop skills to build and manage multi-agent systems using structured approaches like CWD. • Gain knowledge of LLMs, prompt engineering, and key workflows. • Learn about ethical AI, trust mechanisms, and real-world applications.					
UNIT I	INTRODUCTION TO AGENTIC AI				9
Generative AI - Types of Generative AI Models - Applications of Generative AI - Challenges of Generative AI - Principles of Agentic Systems - Understanding Self governance, Agency, Autonomy - Intelligent Systems and characteristics - Architecture of Agentic Systems - Understanding Multi Agent Systems.					
UNIT II	GENERATIVE AI BASED AGENTS				9
Reflection and Introspection in Agents - Importance of reflection in agents - Introspection in intelligent agents - Implementing reflective capabilities - Use cases and examples - Enabling tool use and planning in agents - Understanding the concept of tool use in agents - Planning algorithms for agents - Integrating tool use and planning - Exploring practical implementations - CrewAI example -AutoGen example- LangGraph example.					
UNIT III	AGENTIC AI DESIGN				9
Exploring Coordinator, Worker and Delegator (CWD) Approach - Understanding the CWD Model - Designing agents with role assignments - Communication and collaboration between agents - Implementing CWD approach in generative AI systems - Focused system prompts and instructions for agents - State spaces and environment modeling - Agent memory architecture and context management - Sequential and parallel processing in agentic workflows.					
UNIT IV	LANGUAGE MODELS				9
Basics of Large Language Models (LLM) - LLM Tasks - Prompt Engineering - Special LLM Features - LLM Workflows - Case Studies: Text-to-SQL workflow, A Simple Summary Prompt, From RAG to Agents, A End-to-End SDR, Agents Complying with Policies Plus Synthetic Data Generation.					
UNIT V	TRUST AND APPLICATIONS OF AGENTIC AI				9
Importance of trust in AI - Techniques for establishing trust - Implementing transparency and explainability - handling uncertainty and biases - Common use cases and Applications - Creative and Artistic applications - Natural Language Processing and Conversational Agents - Robotics and autonomous systems - Decision support and optimization.					
TOTAL: 45 PERIODS					

OUTCOMES:

At the end of this course, the students will be able to:

CO1: Describe the fundamentals of Generative AI and Agentic AI concepts

CO2: Identify different types of generative models and their applications.

CO3: Apply reflection, planning, and tool usage in designing intelligent agents.

CO4: Design agentic systems using architectures like multi-agent systems and CWD approach.

CO5: Utilize Large Language Models (LLMs) and prompt engineering for real-world tasks.

CO6: Analyze trust, ethics, and real-world applications of Agentic AI systems.

TEXTBOOKS:

1. Biswas, A. and Talukdar, W., Building Agentic AI Systems: Create Intelligent, Autonomous AI Agents that Can Reason, Plan, and Adapt, 1st Edition, 2025.

2. Ozdemir, S., Building Agentic AI: Workflows, Fine-Tuning, Optimization, and Deployment, 1st Edition, 2025.

REFERENCES:

1. Agentic AI Theories and Practices <https://link.springer.com/book/10.1007/978-3-031-90026-6> Yoav Shoham & Kevin Leyton-Brown, Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations

2. Russell, S., & Norvig, P. (2021). Artificial Intelligence: A Modern Approach (4th Edition). Pearson. <https://www.ibm.com/think/>

24CS926	GENERATIVE AI FUNDAMENTALS	L	T	P	C
		3	0	0	3
OBJECTIVES: The Course will enable learners to: - Understand the basic concepts of Generative AI. - Build Generative AI systems to generate images. - Understand the concept used in Generative AI Models. - Use various Generative AI models. - Compare and use the various Large Language Models. - Understand the basics of Prompt Engineering.					
UNIT I	INTRODUCTION	9			
Generative Models - Image transformation - Challenges - Deep Neural Networks - Perceptron - back propagation - CNN - RNN - Optimizer.					
UNIT II	IMAGE GENERATION	9			
Creating encodings of images - variational objective - Inverse Autoregressive flow - Importing CIFAR - Creating the network from TensorFlow 2.					
UNIT III	GENERATIVE ADVERSARIAL NETWORKS	9			
Generative Adversarial Networks - Vanilla GAN - Improved GANs - Progressive GAN - Challenges - Paired style transfer - Unpaired style transfer - Deepfakes - Modes of operation - key feature set - High level flow - Replacement - Re-enactment.					
UNIT IV	LARGE LANGUAGE MODELS	9			
Overview of LLMs - Transformers - GPT - Types of LLMs - Key concepts - other Transformers - T5 - Generative Pre-Training Models - Multi-modal Models - DALL.E 2					
UNIT V	PROMPT ENGINEERING	9			
Basics - In-Context Learning - In-Context Prompting - Techniques - Image Prompting - Prompt Hijacking - Challenges.					
TOTAL: 45 PERIODS					

OUTCOMES:

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

- CO1:** Elaborate the basic concepts of Generative AI.
- CO2:** Build Generative AI systems to generate images.
- CO3:** Apply the concepts used in Generative AI Models.
- CO4:** Use various Generative AI models.
- CO5:** Compare and use the various Large Language Models.
- CO6:** Analyze the basics of Prompt Engineering.

TEXT BOOKS:

1. Ben Auffarth, Generative AI with LangChain, Packt Publishing, 2023.
2. Amit Bahree, Generative AI in Action, Manning Publication, First Edition, 2023.

REFERENCES:

1. David Foster, Generative Deep Learning, 2nd Edition, O'Reilly Media, 2023.
2. Numa Dhamani and Maggie Engler, Introduction to Generative AI, Manning Publication, First Edition, 2024.
3. Valentina Alto, Modern Generative AI with ChatGPT and OpenAI Models, Packt publications, 2024.

HONOR – DATA SCIENCE

24AM901	FOUNDATIONS OF DATA SCIENCE (Lab Integrated)	L	T	P	C
		2	0	2	3
OBJECTIVES: The Course will enable learners to: • Learn the fundamentals of data science. • Describe and Understand data for analysis. • Describe and Analyze relationships between variables. • Apply various data pre-processing strategies. • Apply data wrangling on data for further processing.					
UNIT I	INTRODUCTION				6+6
Data Science: Benefits and uses - facets of data - Data Science Process: Overview - Defining research goals - Retrieving data - data preparation - Exploratory Data analysis - build the model - presenting findings and building applications - Data Mining - Data Warehousing - Basic statistical descriptions of Data.					
List of Exercise/Experiments 1. Explore a dataset and describe it using basic statistics and represent it using simple graphs. 2. Group data and find averages or counts for each group. 3. Build a simple machine learning model from a dataset and show how well it works.					
UNIT II	ARTIFICIAL NEURAL NETWORKS				6+6
Types of Data - Types of Variables -Describing Data with Tables and Graphs - Describing Data with Averages - Describing Variability - Normal Distributions and Standard (z) Scores.					
List of Exercise/Experiments 1. Use graphs to find patterns or trends in a dataset. 2. Compare two or more columns to find relationships in a dataset. 3. Find the average, spread, and z-scores for numbers in a dataset. 4. Find which values are close to the average and which are far using z-scores.					

UNIT III	FUZZY SYSTEMS	6+6
Correlation -Scatter plots -correlation coefficient for quantitative data - computational formula for correlation coefficient - Regression -regression line - least squares regression line - Standard error of estimate - interpretation of r^2 - multiple regression equations -regression towards the mean.		
List of Exercise/Experiments		
1.	Use a scatter plot to see how two variables are related. Calculate the correlation coefficient to measure the strength of the relationship.	
2.	Find the line of best fit (regression line) for two variables. Calculate r^2 to see how well the line fits the data.	
3.	Use a Multiple Linear Regression model to predict a target variable from multiple inputs, and plot the predicted vs. actual values to evaluate model performance.	
4.	Create a dashboard or summary using charts and key numbers.	
UNIT IV	GENETIC ALGORITHMS	6+6
Data Preprocessing - Purpose - Tools - Levels - Cleaning up the Table - Unpacking - Restructuring - Reformulating the Table - Data Cleaning: - Missing Values - Outliers - Errors - Data Fusion vs Data Integration - Directions - Adding attributes - data objects - Entity identification - Data Reduction vs data redundancy - Types - Performing Numerosity Data reduction - Sampling - Principal Component Analysis - Data Transformation - Normalization and Standardization - Discretization - Smoothing - Aggregation - Binning.		
List of Exercise/Experiments		
1.	Clean and preprocess a dataset by handling missing values, scaling features, treating outliers, and creating dummy variables.	
2.	Train and tune a model using cross-validation to understand and manage the bias-variance trade-off.	
3.	Perform Outlier Analysis on any data set..	
UNIT V	HYBRID SOFT COMPUTING AND SWARM INTELLIGENCE ALGORITHMS	6+6
Processing Uni-dimensional Data: Creating Vectors - Inspecting the Data Distribution with Histograms-Aggregating Numerical Data - Arithmetic Operators - Indexing vectors- Multidimensional data: Creating matrices - Reshaping - visualization (2d,3d) - Heterogeneous Data: Data frames - Aggregation - Transformation - Indexing - Accessing Database: Filtering - Ordering - Removing Duplicates - Grouping and Aggregating - Joining - Handling with Many Files.		
List of Exercise/Experiments		
1.	Create and reshape matrices, apply matrix operations, and visualize data using 2D heatmaps and 3D surface plots.	
2.	Build a DataFrame, perform aggregation and transformation, filter and sort data, remove duplicates, group, and join tables.	
3.	Simulate SQL-style operations like filtering, ordering, deduplication, and grouping in a DataFrame.	
TOTAL: 30(T) + 30(P) = 60 PERIODS		

OUTCOMES:

Upon completion of the course, the students will be able to: CO1:

Explain the fundamentals of data science.

CO2: Illustrate the basics of data for analysis.

CO3: Identify relationships between variables.

CO4: Implement various data pre-processing strategies.

CO5: Implement data wrangling for further processing of data.

CO6: Apply Data preprocessing on real-time data sets.

TEXT BOOKS:

1. Robert S. Witte and John S. Witte, "Statistics", Eleventh Edition, Wiley Publications, 2017. (Units 2, 3)
2. Jacqueline Kazil and Katharine Jarmul, Data Wrangling with Python: Tips and Tools to Make Your Life Easier, O'Reilly Media, 1st Edition, 2016..

REFERENCES:

1. David Cielen, Arno D. B. Meysman, and Mohamed Ali, "Introducing Data Science", Manning Publications, 2016. (Unit 1).
2. Roy Jafari, Hands-On Data Preprocessing in Python: Learn how to effectively prepare data for successful data analytics, Packt Publications, First Edition, 2022. (Unit 4)
3. Marek Gagolewski, Minimalist Data Wrangling with Python, Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License (CC BY-NC-ND 4.0), 2022. (Unit 5)
4. Hui Lin, Ming Li, Practitioner's Guide to Data Science, 1st Edition, Chapman and Hall/CRC, 2023.
5. Jiawei Han, Micheline Kamber, Jian Pei, "Data Mining: Concepts and Techniques", 3rd Edition, Morgan Kaufmann, 2012.
6. Suresh Kumar Mukhiya, Usman Ahmed, Hands-On Exploratory Data Analysis with Python: Perform EDA techniques to understand, summarize, and investigate your data, Packt Publications, 2020.
7. Dr. Tirthajyoti Sarkar, Shubhadeep Roychowdhury, Data Wrangling with Python: Creating actionable data from raw sources, Packt Publications, First Edition, 2019.
8. Robert H. Shumway, David S. Stoffer, Time Series Analysis and Its Applications With R Examples, Fourth Edition, 2016, Springer.
9. Trevor Hastie, Robert Tibshirani, Jerome Friedman, The Elements Of Statistical Learning: Data Mining, Inference, and Prediction, Springer, Second Edition, 2017.
10. Aurélien Géron, Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems, O'Reilly, Third Edition, 2022.
11. Jake VanderPlas, "Python Data Science Handbook - Essential tools for working with data", O'Reilly, 2017.
12. Ashwin Pajankar, Aditya Joshi, Hands-on Machine Learning with Python: Implement Neural Network Solutions with Scikit-learn and PyTorch, Apress, 2022.
13. NPTEL Courses:

a. Data Science for Engineers -

https://onlinecourses.nptel.ac.in/noc23_cs17/preview

b. Python for Data Science -

https://onlinecourses.nptel.ac.in/noc23_cs21/preview

LIST OF EQUIPMENTS:

24AI602	DATA VISUALIZATION (Lab Integrated)	L	T	P	C
		2	0	2	3
OBJECTIVES: The Course will enable learners to: - To outline exploratory data analysis and the phases involved in data analysis. - To discuss various statistical techniques for data analysis. - To demonstrate the basics of feature engineering on different types of data. - To perform data analysis and apply visualization techniques. - To apply the methods of time series analysis. - To formulate dashboards using different datasets by applying data engineering and feature extraction techniques.					
UNIT I	EXPLORATORY DATA ANALYSIS				6+6
EDA fundamentals - Understanding data science - Significance of EDA - Making sense of data - Comparing EDA with classical and Bayesian analysis - Software tools for EDA. Visual Aids For EDA- Data transformation techniques-merging database, reshaping and pivoting, Transformation techniques - Descriptive Statistics-types of kurtosis, quartiles, Grouping Datasets-data aggregation, group wise transformation.					
List of Exercise/Experiments 1. Install the following Data Mining and data Analysis tool: Weka, KNIME, Tableau Public. 2. Perform exploratory data analysis (EDA) on with datasets like email data set. Export all your emails as a dataset, import them inside a pandas data frame, visualize them and get different insights from the data.					
UNIT II	FEATURE ENGINEERING				6+6
Text Data - Visual Data - Feature-based Time-Series Analysis - Data Streams - Feature Selection and Evaluation.					
List of Exercise/Experiments 1. Implement document embeddings for fake news identification. 2. Implement feature based representations of time series 3. Implement feature selection algorithm for data streams					
UNIT III	VISUALIZING DATA				6+6
The Seven Stages of Visualizing Data, Processing-load and displaying data - functions, sketching and scripting, Mapping - Location, Data, two sided data ranges, smooth interpolation of values over time - Visualization of numeric data and non-numeric data.					
List of Exercise/Experiments 1. Perform text mining on a set of documents and visualize the most important words in a visualization such as word cloud. 2. Perform Data Analysis and representation on a Map using various Map data sets with Mouse Rollover effect, user interaction, etc.. 3. Build cartographic visualization for multiple datasets involving various countries of the world; states and districts in India etc.					
UNIT IV	TIME SERIES ANALYSIS				6+6

Overview of time series analysis - showing data as an area, drawing tabs, handling mouse input, Connections and Correlations - Preprocessing-introducing regular expression, sophisticated sorting, Scatterplot Maps-deployment issues.

List of Exercise/Experiments

1. Perform Time Series Analysis with datasets like Open Power System Data.
2. Build a time-series model on a given dataset and evaluate its accuracy.

UNIT V	TREES, HIERARCHIES, AND RECURSION	6+6
---------------	--	------------

Treemaps - treemap library, directory structure, maintaining context, file item, folder item, Networks and Graphs-approaching network problems-advanced graph example, Acquiring data, Parsing data.

List of Exercise/Experiments

1. Use a case study on a data set and apply the various visualization techniques and present an analysis report.
2. Mini-Project:- Create a Dashboard for a dataset with a visualization tool.

TOTAL: 30+30 = 75 PERIODS

OUTCOMES:

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

CO1: Outline exploratory data analysis and the phases involved in data analysis.

CO2: Demonstrate various statistical techniques for data analysis.

CO3: Present the basics of feature engineering on different types of data.

CO4: Perform data analysis and apply visualization techniques.

CO5: Apply the methods of time series analysis.

CO6: Develop dashboards using different datasets by applying data engineering and feature extraction techniques.

TEXT BOOKS:

1. Suresh Kumar Mukhiya and Usman Ahmed, "Hands-on Exploratory Data Analysis with Python", Packt Publishing, First Edition, March 2020.
2. Guozhu Dong, Huan Liu, "Feature Engineering for Machine Learning and Data Analytics", First Publication, CRC Press, First edition, 2018.
3. Ben Fry, "Visualizing Data", O'reilly Publications, First Edition, 2007.

REFERENCES:

1. Danyel Fisher & Miriah Meyer, "Making Data Visual: A Practical Guide To Using Visualization For Insight", O'reilly publications, 2018.
2. Claus O. Wilke, "Fundamentals of Data Visualization", O'reilly publications, 2019.
3. EMC Education Services, "Data Science and Big data analytics: Discovering, Analyzing, Visualizing and Presenting Data", Wiley Publishers, 2015.
4. Tamara Munzner, "Visualization Analysis and Design", A K Peters/CRC Press; 1st edition, 2014.
5. Matthew O. Ward, Georges Grinstein, Daniel Keim, "Interactive Data Visualization: Foundations, Techniques, and Applications", 2nd Edition, CRC press, 2015.

LIST OF EQUIPMENTS:

1. Systems with Python/R, Tableau Public / PowerBI

24CS926	GENERATIVE AI FUNDAMENTALS	L	T	P	C
		3	0	0	3
OBJECTIVES: The Course will enable learners to: • Understand the basic concepts of Generative AI. • Build Generative AI systems to generate images. • Understand the concept used in Generative AI Models. • Use various Generative AI models. • Compare and use the various Large Language Models. • Understand the basics of Prompt Engineering.					
UNIT I	INTRODUCTION	9			
Generative Models - Image transformation - Challenges - Deep Neural Networks - Perceptron - back propagation - CNN - RNN - Optimizer.					
UNIT II	IMAGE GENERATION	9			
Creating encodings of images - variational objective - Inverse Autoregressive flow - Importing CIFAR - Creating the network from TensorFlow 2.					
UNIT III	GENERATIVE ADVERSARIAL NETWORKS	9			
Generative Adversarial Networks - Vanilla GAN - Improved GANs - Progressive GAN - Challenges - Paired style transfer - Unpaired style transfer - Deepfakes - Modes of operation - key feature set - High level flow - Replacement - Re-enactment.					
UNIT IV	LARGE LANGUAGE MODELS	9			
Overview of LLMs - Transformers - GPT - Types of LLMs - Key concepts - other Transformers - T5 - Generative Pre-Training Models - Multi-modal Models - DALL.E 2					
UNIT V	PROMPT ENGINEERING	9			
Basics - In-Context Learning - In-Context Prompting - Techniques - Image Prompting - Prompt Hijacking - Challenges.					
TOTAL: 45 PERIODS					
OUTCOMES: Upon completion of the course, the students will be able to: CO1: Elaborate the basic concepts of Generative AI. CO2: Build Generative AI systems to generate images. CO3: Apply the concepts used in Generative AI Models. CO4: Use various Generative AI models. CO5: Compare and use the various Large Language Models. CO6: Analyze the basics of Prompt Engineering.					
TEXT BOOKS: 1. Ben Auffarth, Generative AI with LangChain, Packt Publishing, 2023. 2. Amit Bahree, Generative AI in Action, Manning Publication, First Edition, 2023.					
REFERENCES: 1. David Foster, Generative Deep Learning, 2nd Edition, O'Reilly Media, 2023. 2. Numa Dhamani and Maggie Engler, Introduction to Generative AI, Manning Publication, First Edition, 2024. 3. Valentina Alto, Modern Generative AI with ChatGPT and OpenAI Models, Packt publications, 2024.					

24AM902	TEXT AND SPEECH ANALYTICS	L	T	P	C
		3	0	0	3
OBJECTIVES: • To introduce the tools and techniques for performing text and speech analytics in diverse contexts. • To understand the tools and technologies involved in developing text and speech applications. • To demonstrate the use of computing for building applications in text and speech processing. • To use information Retrieval Techniques to build and evaluate text processing systems. • To apply advanced speech recognition methodologies in practical applications.					
UNIT I	TEXT PROCESSING	9			
Speech and Language Processing - Regular Expression - Text normalization - Edit Distance - Lemmatization - Stemming - N-gram Language Models - Vector Semantics and Embeddings.					
UNIT II	TEXT CLASSIFICATION	9			
Text Classification Tasks - Language Model - Neural Language Models - RNNs as Language Models - Transformers and Large Language Models.					
UNIT III	QUESTION ANSWERING AND DIALOGUE SYSTEMS	9			
Information Retrieval - Dense Vectors - Neural IR for Question Answering - Evaluating Retrieval-based Question Answering - Frame-based Dialogue Systems - Dialogue Acts and Dialogue State - Chatbots - Dialogue System Design.					
UNIT IV	TEXT TO SPEECH SYNTHESIS	9			
Automatic Speech Recognition Task - Feature Extraction for ASR: Log Mel Spectrum - Speech Recognition Architecture - CTC - ASR Evaluation: Word Error Rate - TTS - Speech Tasks.					
UNIT V	SPEECH RECOGNITION	9			
LPC for speech recognition - Hidden Markov Model (HMM) - Training procedure for HMM- subword unit model based on HMM - Language models for large vocabulary speech recognition - Overall recognition system based on subword units - Context dependent subword units- Semantic post processor for speech recognition.					
TOTAL: 45 PERIODS					
OUTCOMES: At the end of this course, the students will be able to: CO1: Apply the fundamental techniques in text processing for various NLP tasks. CO2: Implement advanced language models and improve text classification accuracy. CO3: Designing text processing systems using state-of-the-art techniques. CO4: Design, implement, and evaluate ASR and TTS systems. CO5: Apply advanced speech recognition methodologies in practical applications. CO6: Use information Retrieval Techniques to build and evaluate text processing systems.					
TEXT BOOKS: 1. Jurafsky, D. and J. H. Martin, Speech and language processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition Pearson Publication, Third Edition, 2022. 2. Lawrence Rabiner, Biing-Hwang Juang and B.Yegnanarayana, “Fundamentals					

of Speech Recognition”, Pearson Education, 2009.

REFERENCES:

1. John Atkinson-Abutridy, Text Analytics: An Introduction to the Science and Applications of Unstructured Information Analysis, CRC Press, 2022.
2. Jim Schwoebel, NeuroLex, Introduction to Voice Computing in Python, 2018
3. Lawrence R. Rabiner, Ronald W. Schafe, Theory and Applications of Digital Speech Processing, First Edition, Pearson, 2010.
4. 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.

24AM905	IMAGE AND VIDEO ANALYTICS	L	T	P	C
		3	0	0	3
OBJECTIVES: The Course will enable learners to: • To understand the basics of image processing techniques for computer vision and video analysis. • To illustrate the techniques used for image pre-processing. • To discuss the various image Segmentation techniques. • To understand the various Object recognition mechanisms. • To elaborate on the motion analysis techniques for video analytics.					
UNIT I	INTRODUCTION				9
Computer Vision - Image representation and image analysis tasks - Image representations - digitization - properties - color images - Data structures for Image Analysis - Levels of image data representation - Traditional and Hierarchical image data structures.					
UNIT II	IMAGE PRE-PROCESSING				9
Pixel brightness transformations - Geometric transformations - Local pre-processing - Image smoothing - Edge detectors - Zero-crossings of the second derivative - Scale in image processing - Canny edge detection - Parametric edge models - Edges in multi-spectral images - Local pre-processing in the frequency domain - Line detection by local pre-processing operators - Detection of corners (interest points) - Detection of maximally stable extremal regions - Image restoration.					
UNIT III	SEGMENTATION				9
Thresholding - Edge-based segmentation - Region-based segmentation - Matching - Evaluation issues in segmentation - Mean shift segmentation - Active contour models.					
UNIT IV	OBJECT RECOGNITION				9
Knowledge representation - Statistical pattern recognition - Neural nets - Syntactic pattern recognition - Recognition as graph matching - Optimization techniques in recognition - Fuzzy systems - Boosting in pattern recognition - Random forests - Image understanding control strategies.					
UNIT V	MOTION ANALYSIS				9
Differential motion analysis methods - Optical flow - Analysis based on correspondence of interest points - Detection of specific motion patterns - Video tracking - Motion models to aid tracking.					
TOTAL: 45 PERIODS					

OUTCOMES: Upon completion of the course, the students will be able to:	
CO1: Understand the basics of image processing techniques for computer vision and video analysis.	
CO2: Illustrate the techniques used for image pre-processing.	
CO3: Analyze the various image Segmentation techniques.	
CO4: Understand the various Object recognition mechanisms.	
CO5: Elaborate on the motion analysis techniques for video analytics.	
CO6: Apply image processing techniques in real-world applications.	
TEXT BOOKS:	
1. Milan Sonka, Vaclav Hlavac, Roger Boyle, "Image Processing, Analysis, and MachineVision", 4nd edition, Thomson Learning, 2013.	
REFERENCES:	
1. Richard Szeliski, "Computer Vision: Algorithms and Applications", Springer Verlag London Limited, 2011.	
2. Caifeng Shan, Fatih Porikli, Tao Xiang, Shaogang Gong, "Video Analytics for BusinessIntelligence", Springer, 2012.	
3. D. A. Forsyth, J. Ponce, "Computer Vision: A Modern Approach", Pearson Education, 2003.	
4. E. R. Davies, "Computer & Machine Vision", Fourth Edition, Academic Press, 2012.	

24AM906	STREAM PROCESSING AND ANALYTICS	L	T	P	C
		3	0	0	3
OBJECTIVES: The Course will enable learners to: • To outline the framework for real time stream processing. • To learn various algorithms for data streaming. • To identify frequent item sets by mining from data streams. • To introduce approaches to evaluate stream learning algorithms. • To use tools for distributed data flow management. • To design solutions to stream processing problems					
UNIT I	INTRODUCTION TO DATA STREAMS				9
Data Stream Models - Bounds of Random variables - Poisson Process - Maintaining Simple Statistics from Data Streams - Sliding Window and computing statistics over sliding windows - Data Synopsis - Sampling - Histograms - Wavelets - DFT - Change Detection: Tracking Drifting Concepts - Monitoring the Learning Process.					
UNIT II	STREAMING ALGORITHMS				9
Clustering Examples: Basic Concepts - Partitioning Clustering - Hierarchical Clustering - Micro Clustering - Grid Clustering - Clustering Variables - The Very Fast Decision Tree Algorithm (VFDT) - The Base Algorithm, Analysis of the VFDT Algorithm, Extensions to the Basic Algorithm: Processing Continuous Attributes, Functional Tree Leaves, Concept Drift.					
UNIT III	FREQUENT PATTERN MINING				9
Introduction - Heavy Hitters - Mining Frequent Itemsets from Data Streams - Landmark Windows - Mining Recent Frequent Itemsets - Frequent Itemsets at Multiple Time Granularities - Sequence Pattern Mining - Reservoir Sampling for Sequential Pattern Mining over data stream.					
UNIT IV	EVALUATING STREAMING ALGORITHMS				9

Learning from Data Streams - Evaluation Issues - Design of Evaluation Experiments
 - Evaluation Metrics - Comparative Assessment - Evaluation Methodology in Non-Stationary Environments.

UNIT V	DATA FLOW MANAGEMENT
---------------	-----------------------------

	9
--	----------

Distributed Data Flows - Apache Kafka - Apache Flume - Processing Streaming Data - Storing Streaming Data - Delivering Streaming Metrics.

TOTAL: 45 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to: CO1:

To outline the framework for real time stream processing.

CO2: To learn various algorithms for data streaming.

CO3: To identify frequent item sets by mining from data streams. **CO4:**

To introduce approaches to evaluate stream learning algorithms. **CO5:**

To use tools for distributed data flow management.

CO6: To design solutions to stream processing problems

TEXT BOOKS:

1. Joao Gama, "Knowledge Discovery from Data Streams", CRC Press, 2010.
2. Byron Ellis, Real-Time Analytics: Techniques to Analyze and Visualize Streaming Data, First Edition, WILEY Big Data Series, 2014

REFERENCES:

1. Andrew Psaltis, Streaming Data: Paul Lewis, First Edition, Manning Publication, 2017.
2. Bugra Gedik, Deepak S. Turaga, Henrique C. M. Andrade, Fundamentals of Stream Processing: Application Design, Systems, and Analytics, Cambridge University Press, 2014.
3. Charu C. Aggarwal, "Data Streams: Models and Algorithms", Kluwer Academic Publishers, 2007.
4. David Luckham, "The Power of Events: An Introduction to Complex Event Processing in Distributed Enterprise Systems", Addison Wesley, 2002.

LIST OF EQUIPMENTS:

Apache Flink, Python API via PyFlink, Apache Kafka, SQLite, TimescaleDB , Grafana

HONOR - ARTIFICIAL INTELLIGENCE

24CS938	SOFT COMPUTING (Lab Integrated)	L	T	P	C
		2	0	2	3
OBJECTIVES: The Course will enable learners to: • Learn the basic concepts of Soft Computing. • Understand artificial neural networks. • Explain fuzzy systems. • Explain Genetic Algorithms. • Discuss the various Hybrid algorithms and various Swarm Intelligence algorithms.					
UNIT I	INTRODUCTION				6+6
Neural Networks - Application Scope of Neural Networks - Fuzzy Logic - Genetic Algorithm - Hybrid Systems - Soft Computing - Artificial Neural Network - Evolution of Neural Networks - Basic Models of ANN - Weights - Bias - Threshold - Learning Rate - Momentum Factor - Vigilance Parameter- McCulloch-Pitts Neuron - Linear Separability - Hebb Network.					
List of Exercise/Experiments 1. Write a program to implement Hebb's rule. 2. Implement McCulloch-Pitts model using Simple Neural Network.					
UNIT II	ARTIFICIAL NEURAL NETWORKS				6+6
Perceptron Networks - Adaptive Linear Neuron - Multiple Adaptive Linear Neurons - Back-Propagation Network - Radial Basis Function Network - Pattern Association – Auto associative and Hetero associative Memory Networks - Bidirectional Associative Memory (BAM) - Hopfield Networks - Fixed Weight Competitive Nets - Kohonen Self-Organizing Feature Maps.					
List of Exercise/Experiments 1. Implement Kohonen self-Organizing feature maps 2. Write a program for solving linearly separable problem using Perceptron Model					
UNIT III	FUZZY SYSTEMS				6+6
Fuzzy Logic - Classical Sets (Crisp Sets) - Fuzzy Sets - Fuzzy Relation - Features of the Membership Functions - Fuzzification - Methods of Membership Value Assignments - Defuzzification - Lambda-Cuts for Fuzzy Sets (Alpha-Cuts) - Lambda-Cuts for Fuzzy Relations - Defuzzification Methods - Fuzzy Reasoning - Fuzzy Inference Systems.					
List of Exercise/Experiments 1. Implement Union, Intersection, Complement and Difference operations on fuzzy sets. Also create fuzzy relation by Cartesian product of any two fuzzy sets and perform max-min composition on any two fuzzy relations. 2. Implementation of fuzzy relations (Max-Min Composition)					
UNIT IV	GENETIC ALGORITHMS				6+6

Biological Background - Traditional Optimization and Search Techniques- Genetic Algorithm and Search Space- - Simple GA - General Genetic Algorithm - Operators - Stopping Condition - Constraints - Problem Solving - The Schema Theorem- Classification - Holland Classifier Systems- Genetic Programming - Advantages and Limitations- Applications.

List of Exercise/Experiments

1. Implement travelling salesperson problem (tsp) using genetic algorithms.
2. Implement two classes city and fitness using genetic algorithm.

UNIT V

HYBRID SOFT COMPUTING AND SWARM INTELLIGENCE ALGORITHMS

6+6

Neuro-Fuzzy Hybrid Systems - Genetic Neuro-Hybrid Systems - Genetic Fuzzy Hybrid and Fuzzy Genetic Hybrid Systems - Simplified Fuzzy ARTMAP - Swarm Intelligence Algorithms - Ant Colony Optimization - Artificial Bee Colony - Particle Swarm Optimization - Firefly Algorithm.

List of Exercise/Experiments

1. To design and implement fuzzy logic for a washing machine system.
2. Case study on hybrid system. To study the designing of Neuro-Fuzzy Systems

Mini Project:

1. Apply Swarm Intelligence Algorithms for any one of the following applications:
 - a. Disease diagnosis
 - b. Image Processing
 - c. Business Intelligence
 - d. Cyber Security etc.
2. Case study on Hybrid Systems.
3. To study the designing of Neuro Fuzzy systems.
4. To design and implement fuzzy logic for a washing machine system.

TOTAL: 30(T) + 30(P) = 60 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to: CO1:

Understand the basic concepts of Soft Computing

CO2: Artificial neural networks and its applications.

CO3: Fuzzy logic and its applications.

CO4: Solving problems using Genetic algorithms.

CO5: Applications of Soft computing to solve problems in varieties of application domains.

CO6: Use various algorithms in Soft computing to solve real-world problems

TEXT BOOKS:

1. S. N. Sivanandam , S. N. Deepa, "Principles of Soft Computing", Wiley India Pvt. Ltd., 2nd Edition, 2019.
2. Adam Slovik, "Swarm Intelligence Algorithms: Modification and Applications", Taylor & Francis, First Edition, 2020.

REFERENCES:

1. Jyh-Shing Roger Jang, Chuen-Tsai Sun, Eiji Mizutani, Neuro-Fuzzy and Soft Computing, Prentice-Hall of India, 2002.
2. Kwang H. Lee, First course on Fuzzy Theory and Applications, Springer, 2005.
3. N.P. Padhy, S. P. Simon, "Soft Computing with MATLAB Programming", Oxford University Press, 2015.
4. S. Rajasekaran, G. A. Vijayalakshmi Pai, "Neural Networks, Fuzzy Logic and Genetic Algorithm, Synthesis and Applications", PHI Learning Pvt. Ltd., 2017.
5. NPTEL Courses:
 - a. Introduction To Soft Computing -
https://onlinecourses.nptel.ac.in/noc23_cs40/preview

24AM933	KNOWLEDGE ENGINEERING	L	T	P	C
		3	0	0	3
OBJECTIVES: • To understand the basics of Knowledge Engineering. • To discuss reasoning under uncertainty. • To design and develop ontologies. • To apply reasoning with ontologies and rules. • To understand learning and rule learning.					
UNIT I	INTRODUCTION				9
Knowledge, Representation and Reasoning - Need for Logic - First order logic - Syntax - Semantics - Pragmatics- Implicit and Explicit Belief - Expressing Knowledge - Resolution - Propositional case - Horn Logic - Horn clauses - Procedural Control of Reasoning.					
UNIT II	REASONING UNDER UNCERTAINTY				9
Introduction - Abductive reasoning - Probabilistic reasoning: Enumerative Probabilities - Subjective Bayesian view - Belief Functions - Baconian Probability - Fuzzy Probability - Uncertainty methods - Evidence-based reasoning - Intelligent Agent - Mixed-Initiative Reasoning - Knowledge Engineering - Evidence-based reasoning task: Intelligent Analysis.					
UNIT III	ONTOLOGIES – DESIGN AND DEVELOPMENT				9
Concepts and Instances - Generalization Hierarchies - Object Features - Defining Features -Representation - Transitivity - Inheritance - Concepts as Feature Values - Ontology Matching. Design and Development Methodologies - Steps in Ontology Development- Domain Understanding and Concept Elicitation - Modelling-based Ontology Specification.					
UNIT IV	REASONING WITH ONTOLOGIES AND RULES				9

Production System Architecture - Complex Ontology-based Concepts - Reduction and Synthesis rules and the Inference Engine - Evidence-based hypothesis analysis - Rule and Ontology Matching - Partially Learned Knowledge - Reasoning with Partially Learned Knowledge - Rules in Production Systems - Object-Oriented Representation - Structured Descriptions.		
UNIT V	LEARNING AND RULE LEARNING	9
Machine Learning - Concepts - Generalization and Specialization Rules - Types - Inductive concept learning from Examples - Learning with an Incomplete Representation Language - Formal definition of Generalization. Modelling, Learning and Problem Solving - Rule learning and Refinement - Overview.		
TOTAL: 45 PERIODS		
OUTCOMES: At the end of this course, the students will be able to: CO1: Understand the basics of Knowledge Engineering. CO2: Discuss reasoning under uncertainty. CO3: Design and develop ontologies. CO4: Apply reasoning with ontologies and rules. CO5: Understand learning and rule learning. CO6: Apply learning concepts to solve problems.		
TEXT BOOKS: 1. Ronald J. Brachman, Hector J. Levesque: Knowledge Representation and Reasoning, Morgan Kaufmann, 2004. 2. Gheorghe Tecuci, Dorin Marcu, Mihai Boicu, David A. Schum, Knowledge Engineering Building Cognitive Assistants for Evidence-based Reasoning, Cambridge University Press, First Edition, 2016. (Unit 3 - Chapter 5, 6, Unit 4 - 7, Unit 5 - Chapter 8, 9)		
REFERENCES: 1. Ela Kumar, Knowledge Engineering, I K International Publisher House, 2018. 2. John F. Sowa: Knowledge Representation: Logical, Philosophical, and Computational Foundations, Brooks/Cole, Thomson Learning, 2000. 3. King (2009), Knowledge Management and Organizational Learning, Springer 4. Jay Liebowitz, Knowledge Management Learning from Knowledge Engineering, 1st Edition, 2001.		

	REINFORCEMENT LEARNING	L	T	P	C
		3	0	0	3
OBJECTIVES:					
• To outline the concepts of Reinforcement Learning • To solve problems using Monte Carlo Decision Process and Dynamic Programming. • To implement problems using temporal difference learning. • To apply functional approximation in reinforcement learning. • To implement Deep Reinforcement Learning. • To solve real-world problems using Reinforcement Learning.					

UNIT I	INTRODUCTION TO REINFORCEMENT LEARNING	9
Introduction - Elements of RL, History of RL- Limitation and Scope - Examples - Multi-arm Bandits - k-armed Bandit Problem - Action-Value Methods - Incremental Implementation - Nonstationary Problem - Optimistic Initial Values - Upper Confidence Bound Action Selection - Gradient Bandit Algorithms - Contextual Bandits.		
UNIT II	MONTE DECISION PROCESS AND DYNAMIC PROGRAMMING	9
Finite Markov Decision Processes - The Agent Environment interface - Goals and Rewards - Returns and Episodes - Episodic and Continuing Tasks - Policies and Value Functions - Optimality and Approximation - Dynamic Programming - Policy Evaluation - Policy Improvement, Iteration - Value Iteration - Asynchronous DP- Efficiency of DP - Monte Carlo Prediction - Monte Carlo Estimation of Action Values - Monte Carlo Control - Off - policy Monte Carlo Prediction - Off-Policy Monte Carlo Controls.		
UNIT III	TEMPORAL DIFFERENCE LEARNING	9
Temporal-Difference prediction - Advantages of TD Prediction Methods -Optimality of TD(0) - Sarsa: On-policy TD Control - Q Learning: Off-Policy TD Control - Expected Sarsa - n-step TD Prediction - n-step Sarsa - n-step Off-Policy Learning.		
UNIT IV	FUNCTION APPROXIMATION METHODS	9
On-Policy Prediction with Approximation: Value-function Approximation - The Prediction Objective - Stochastic-gradient and Semi-gradient Methods - Linear Methods - Feature Construction for Linear Methods - Eligibility Traces: The λ -return - TD(λ).		
UNIT V	DEEP REINFORCEMENT LEARNING	9
Deep Q-Learning - Deep Q-Networks - Rainbow DQN - Policy Gradient Methods - Policy Approximation and its advantages - Policy Gradient Theorem - REINFORCE: Monte Carlo Policy Gradient - Actor-Critic Methods - Hierarchical RL - Multi-Agent RL.		
TOTAL: 45 PERIODS		
OUTCOMES: At the end of this course, the students will be able to: CO1: Outline the concepts of Reinforcement Learning. CO2: Solve problems using Dynamic Programming and Monte Carlo Decision Process. CO3: Implement the concept of Temporal difference Learning (TML) to solve real world problems. CO4: Apply functional approximation in reinforcement learning. CO5: Implement Deep Reinforcement Learning to solve real world problems. CO6: Solve real-world problems using Reinforcement Learning		
TEXT BOOKS: 1. Sutton R. S. and Barto A. G., "Reinforcement Learning: An Introduction", MIT Press, Second Edition, 2020. 2. Phil Winder, "Reinforcement Learning: Industrial Applications of Intelligent Agents". Oreilly, 2021.		

REFERENCES:

1. Nimish Sanghi, "Deep Reinforcement Learning with Python with PyTorch, TensorFlow and OpenAI Gym", Apress, First Edition, 2021.
2. Maxim Lapan, "Deep Reinforcement Learning Hands-On", Packt, Second Edition, 2020.
3. Miguel Morales, "Deep Reinforcement Learning ", Miguel Morales, Manning Publication, 2020.
4. Kevin Murphy, "Machine Learning - A Probabilistic Perspective" , MIT press, 2012.
5. Christopher Bishop, "Pattern Recognition and Machine Learning", Springer, 2006.

24AM932	RECOMMENDER SYSTEMS	L	T	P	C
		3	0	0	3
OBJECTIVES:					
The Course will enable learners to:					
• To understand the foundations of the recommender system. • To learn about collaborative filtering. • To discuss content-based recommendation systems. • To elaborate on the evaluation paradigms for a recommendation system. • To make students design and implement a recommender system.					
UNIT I	INTRODUCTION TO RECOMMENDER SYSTEMS				9
Introduction - Basic Models of Recommender Systems - Domain-Specific Challenges in Recommender Systems - Cold-Start Problem - Attack-Resistant Recommender Systems - Group - Multi-Criteria - Active-Learning - Privacy - Application Domains.					
UNIT II	COLLABORATIVE FILTERING				9
Neighborhood-Based Collaborative Filtering - Key Properties - Predicting Ratings - Clustering -Dimensionality Reduction - A Regression Modeling - Graph Models - Model-based Collaborative Filtering - Decision and Regression Trees - Rule-Based Collaborative Filtering - Naive Bayes Collaborative Filtering - Latent Factor Models.					
UNIT III	CONTENT-BASED RECOMMENDATION				9
Basic Components of Content-Based Systems - Preprocessing and Feature Extraction - Learning User Profiles and Filtering - Content-Based Versus Collaborative Recommendations - Using Content-Based Models for Collaborative Filtering.					
UNIT IV	DESIGN EVALUATION				9
Evaluating Paradigms - General Goals of Evaluation Design-Design Issues in Offline Recommender Evaluation-Accuracy Metrics in Offline Evaluation-Limitations of Evaluation Measures.					
UNIT V	TYPES OF RECOMMENDATION SYSTEMS				9
Content-based Recommender Systems - Basic Components - Constraint-based Recommender Systems - Context-sensitive Recommender Systems - Social and Trust-Centric Recommender Systems.					
TOTAL: 45 PERIODS					

OUTCOMES: Upon completion of the course, the students will be able to: CO1: Elaborate the foundations of the recommender system. CO2: Use collaborative filtering to design recommendation systems. CO3: Discuss content-based recommendation systems. CO4: Elaborate on the evaluation paradigms for a recommendation system. CO5: Use appropriate type of recommendation systems to solve real-world problems. CO6: Design, implement and evaluate a recommendation algorithm.	
TEXT BOOKS: 1. Charu C. Aggarwal, Recommender Systems: The Textbook, Springer, 2016. 2. Jannach D., Zanker M., Felfering A., Friedrich G., Recommender Systems: An Introduction, Cambridge University Press, First Edition, 2011.	
REFERENCES: 1. Jure Leskovec, Anand Rajaraman, Jeffrey David Ullman, Mining of massive datasets, 3rd edition, Cambridge University Press, 2020. 2. Ricci, F., Rokach, L. and Shapira, B., Introduction to recommender systems handbook. In Recommender systems handbook, Springer, 2011. 3. Manouselis N., Drachsler H., Verbert K., Duval E., Recommender Systems For Learning, Springer, First Edition, 2013.	

24CS939	NATURAL LANGUAGE PROCESSING	L	T	P	C
		3	0	0	3
OBJECTIVES: The Course will enable learners to: - To learn the fundamentals of natural language processing - To discuss word level analysis.					
- To discuss the different language models. - To understand the significance of syntactic and semantic analysis. - To learn discourse algorithms and various lexical resources.					
UNIT I	INTRODUCTION				
Natural Language Processing - Ambiguities in NLP - Regular Expressions - Words - Corpora - Text Normalization, Minimum Edit Distance.					
UNIT II	WORD LEVEL ANALYSIS	9			
Morphological Analysis - Morphological Parsing - Unsmoothed N-grams, Evaluating N-grams, Smoothing, Interpolation and Backoff - Word Classes, Part-of-Speech Tagging, Rule-based - HMM - Transformation -based tagging.					
UNIT III	LANGUAGE MODELS	9			
Markov Chains - Hidden Markov Model - Forward Algorithm - Decoding: Viterbi Algorithm - Training HMMs - Maximum Entropy Models - Maximum Entropy Markov Models.					
UNIT IV	SYNTACTIC AND SEMANTIC ANALYSIS	10			
Context-Free Grammars - Grammar rules - Treebanks - Normal Forms for grammar - Finite-state - CFG - Dependency Grammar - Parsing with CFG - Search - Ambiguity - Syntax-Driven Semantic analysis - Semantic Augmentations - Semantic attachments - Unification based approaches to Semantic Analysis - Semantic Attachments - Integrating Semantic Analysis to Early Parser - WordNet.					
UNIT V	APPLICATIONS OF NLP	8			

Information Extraction - Question Answering and Summarization - Dialogue and Conversational Agent - Machine Translation.

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Elaborate the fundamentals of natural language processing.

CO2: Perform word level analysis in NLP.

CO3: Illustrate different ML models for NLP.

CO4: Analyze the syntax and semantics using various methods.

CO5: Analyze text at the word level.

CO6: Apply NLP to solve real-world problems.

TEXTBOOKS:

1. Daniel Jurafsky, James H. Martin, "Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech", Pearson Publication, Second Edition, 2019.

REFERENCES:

1. Steven Bird, Ewan Klein and Edward Loper, "Natural Language Processing with Python", First Edition, O'Reilly Media, 2009.
2. Breck Baldwin, "Language Processing with Java and LingPipe Cookbook", Atlantic Publisher, 2015.
3. Richard M Reese, "Natural Language Processing with Java", O'Reilly Media, 2015.
4. Nitin Indurkha and Fred J. Damerau, "Handbook of Natural Language Processing", Second Edition, Chapman and Hall/CRC Press, 2010.
5. Tanveer Siddiqui, U.S. Tiwary, "Natural Language Processing and Information Retrieval", Oxford University Press, 2008.

24CS926	GENERATIVE AI FUNDAMENTALS	L	T	P	C
		3	0	0	3
OBJECTIVES: The Course will enable learners to: • Understand the basic concepts of Generative AI. • Build Generative AI systems to generate images. • Understand the concept used in Generative AI Models. • Use various Generative AI models. • Compare and use the various Large Language Models. • Understand the basics of Prompt Engineering.					
UNIT I	INTRODUCTION	9			
Generative Models - Image transformation - Challenges - Deep Neural Networks - Perceptron - back propagation - CNN - RNN - Optimizer.					
UNIT II	IMAGE GENERATION	9			
Creating encodings of images - variational objective - Inverse Autoregressive flow - Importing CIFAR - Creating the network from TensorFlow 2.					
UNIT III	GENERATIVE ADVERSARIAL NETWORKS	9			
Generative Adversarial Networks - Vanilla GAN - Improved GANs - Progressive GAN - Challenges - Paired style transfer - Unpaired style transfer - Deepfakes - Modes of operation - key feature set - High level flow - Replacement - Re-enactment.					
UNIT IV	LARGE LANGUAGE MODELS	9			
Overview of LLMs - Transformers - GPT - Types of LLMs - Key concepts - other Transformers - T5 - Generative Pre-Training Models - Multi-modal Models - DALL.E 2					
UNIT V	PROMPT ENGINEERING	9			
Basics - In-Context Learning - In-Context Prompting - Techniques - Image Prompting - Prompt Hijacking - Challenges.					
TOTAL: 45 PERIODS					
OUTCOMES: Upon completion of the course, the students will be able to: CO1: Elaborate the basic concepts of Generative AI. CO2: Build Generative AI systems to generate images. CO3: Apply the concepts used in Generative AI Models. CO4: Use various Generative AI models. CO5: Compare and use the various Large Language Models. CO6: Analyze the basics of Prompt Engineering.					
TEXT BOOKS: 3. Ben Auffarth, Generative AI with LangChain, Packt Publishing, 2023. 4. Amit Bahree, Generative AI in Action, Manning Publication, First Edition, 2023.					
REFERENCES: 4. David Foster, Generative Deep Learning, 2nd Edition, O'Reilly Media, 2023. 5. Numa Dhamani and Maggie Engler, Introduction to Generative AI, Manning Publication, First Edition, 2024. 6. Valentina Alto, Modern Generative AI with ChatGPT and OpenAI Models, Packt publications, 2024.					

OPEN ELECTIVES OFFERED TO OTHER DEPARTMENTS

S. No.	Course Code	Course Title	Contact Periods	L	T	P	C
1	24CS001	Blockchain Technology	3	3	0	0	3
2	24CS002	Cloud Architecting	3	3	0	0	3
3	24CS003	Software Testing and Automation	3	3	0	0	3
4	24CS004	Digital and Mobile Forensics	3	3	0	0	3

24CS001	BLOCKCHAIN TECHNOLOGY	L	T	P	C
		3	0	0	3
OBJECTIVES: The Course will enable learners to:					
- To understand block chain system's fundamental components, how they fit together and examine a decentralization using block chain. - To explain how Crypto currency works. - To explain the components of Ethereum and Programming Languages for Ethereum - To study the basics of Web3 and Hyper ledger. - To give an insight of alternative block chains and its emerging trends.					
UNIT I	INTRODUCTION TO BLOCKCHAIN				9
History of Blockchain - Types of Blockchain - Consensus - Decentralization using Blockchain - Blockchain and Full Ecosystem Decentralization - Platforms for Decentralization - Symmetric Cryptography - Mathematics - Asymmetric Cryptography - public and private keys - Elliptic curve cryptography - Discrete logarithm problem in ECC					
UNIT II	INTRODUCTION TO CRYPTOCURRENCY				9
Bitcoin - Digital Keys and Addresses - Transactions - Mining - Bitcoin Networks and Payments Wallets - innovation in Bitcoin - Alternative Coins - Theoretical Foundations - Bitcoin.					
UNIT III	ETHEREUM				9
The Ethereum Network - Components of Ethereum Ecosystem - Ethereum Programming Languages: Runtime Byte Code - Blocks and Blockchain - Fee Schedule - Supporting Protocols - Solidity Language.					
UNIT IV	WEB3 AND HYPERLEDGER				9
Introduction to Web3 - Contract Deployment - POST Requests - Development frameworks Hyperledger as a protocol - The Reference Architecture - Hyperledger Fabric - Distributed Ledger.					
UNIT V	ALTERNATIVE BLOCKCHAINS AND NEXT EMERGING TRENDS				9
Kadena - Ripple- Rootstock - Quorum - Tendermint - Scalability - Privacy - Other Challenges - Blockchain Research. Case Study - Install IPFS locally on our machine, initialize your node, view the nodes in network.					
TOTAL: 45 PERIODS					
OUTCOMES: At the end of this course, the students will be able to: CO1: Understand the fundamentals and history of blockchain technology. CO2: Analyze different types of blockchain and their consensus mechanisms. CO3: Apply cryptographic principles to blockchain systems.					

- CO4:** Explore cryptocurrency concepts, including Bitcoin and alternative coins.
- CO5:** Evaluate Ethereum and its programming languages for decentralized applications.
- CO6:** Investigate other blockchain platforms like Hyperledger and emerging trends.

TEXTBOOKS:

1. Imran Bashir, "Mastering Blockchain: Distributed Ledger Technology, decentralization, and smart contracts explained", 2nd Edition, Packt Publishing Ltd, March 2018.
2. BellajBadr, Richard Horrocks, Xun (Brian) Wu, "Blockchain By Example: A developer's guide to creating decentralized applications using Bitcoin, Ethereum, and Hyperledger", Packt Publishing Limited, 2018.

REFERENCES:

1. Andreas M. Antonopoulos, "Mastering Bitcoin: Unlocking Digital Cryptocurrencies", O'Reilly Media Inc, 2015
2. Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller and Steven Goldfeder, "Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction", Princeton University Press, 2016.

24CS002	CLOUD ARCHITECTING	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners:

- To make architectural decisions based on AWS architectural principles and best practices.
- To describe the features and benefits of Amazon EC2 instances, and compare and contrast managed and unmanaged database services.
- To create a secure and scalable AWS network environment with VPC, and configure IAM for improved security and efficiency.
- To use AWS services to make infrastructure scalable, reliable, and highly available.
- To use AWS managed services to enable greater flexibility and resiliency in an infrastructure.

UNIT I	INTRODUCING CLOUD ARCHITECTING AND STORAGE LAYER	9
--------	--	---

Cloud architecting - The AWS Well-Architected Framework - AWS global infrastructure - Amazon S3 - Amazon S3 Versioning - Storing data in Amazon S3 - Moving data to and from Amazon S3 - Amazon S3 Transfer Acceleration - Choosing Regions for your architecture.

UNIT II	COMPUTE LAYER AND DATABASE LAYER	9
---------	----------------------------------	---

Adding compute with Amazon EC2 - Choosing an Amazon Machine Image (AMI) to launch an Amazon EC2 instance - Selecting an Amazon EC2 instance type - Using user data to configure an EC2 instance - Adding storage to an Amazon EC2 instance - Amazon EC2 pricing options - Amazon EC2 considerations - Database layer considerations - Amazon Relational Database Service (Amazon RDS) - Amazon DynamoDB - Database security controls - Migrating data into AWS

databases.		
UNIT III	CREATING AND CONNECTING NETWORKS	9
Creating an AWS networking environment - Connecting your AWS networking environment to the internet - Securing your AWS networking environment - Connecting your remote network with AWS Site-to-Site VPN - Connecting your remote network with AWS Direct Connect - Connecting virtual private clouds (VPCs) in AWS with VPC peering - Scaling your VPC network with AWS Transit Gateway - AWS Transit Gateway - Connecting your VPC to supported AWS services. Securing User and Application Access: Account users and AWS Identity and Access Management (IAM) - Organizing users - Federating users - Multiple accounts.		
UNIT IV	RESILIENT CLOUD ARCHITECTURE	9
Scaling your compute resources - Scaling your databases - Designing an environment that's highly available - Monitoring - Reasons to automate - Automating your infrastructure - Automating deployments - AWS Elastic Beanstalk - Overview of caching - Edge caching - Caching web sessions - Caching databases.		
UNIT V	BUILDING DECOUPLED ARCHITECTURES, MICROSERVICES AND SERVERLESS ARCHITECTURE	9
Decoupling your architecture - Decoupling with Amazon Simple Queue Service (Amazon SQS) - Decoupling with Amazon Simple Notification Service (Amazon SNS) - Sending messages between cloud applications and on-premises with Amazon MQ. Introducing microservices - Building microservice applications with AWS container services - Introducing serverless architectures - Building serverless architectures with AWS Lambda - Extending serverless architectures with Amazon API Gateway - Orchestrating microservices with AWS Step Functions - Disaster planning strategies - Disaster recover patterns.		
TOTAL: 45 PERIODS		
OUTCOMES: Upon Completion of this course, the students will be able to: CO1: Explain cloud architecture principles and AWS storage solutions. CO2: Deploy and manage AWS compute and database resources securely. CO3: Design and configure secure AWS networks using VPC and IAM. CO4: Implement scalable and resilient AWS architectures with high availability. CO5: Build decoupled and serverless applications using AWS services like Lambda. CO6: Develop disaster recovery strategies for AWS environments.		
REFERENCES: 1. AWS Certified Solutions Architect Official Study Guide by Joe Baron, Hisham Baz, Tim Bixler 2. Architecting the Cloud by Michael Kavis. 3. AWS Documentation (amazon.com) 4. AWS Skill Builder 5. AWS Academy Cloud Architecting Course - https://www.awsacademy.com/vforcesite/LMS_Login		

24CS003	SOFTWARE TESTING AND AUTOMATION	L	T	P	C
		3	0	0	3
OBJECTIVES: The Course will enable learners to: • Introduce the basics and necessity of software testing. • Provide various testing techniques along with concepts of software bugs and its impact. • Develop and validate a test plan. • Build a testing team required. • Understand the need for and challenges in test automation and to develop testing scripts.					
UNIT I	FOUNDATIONS OF SOFTWARE TESTING	9			
Why do we test Software? Stages of Testing, Software Testing Principles, Software Testing Life Cycle, Program Correctness and Verification, Reliability versus Safety, Failures, Errors and Faults (Defects), Case Study: Test plan for testing an e-commerce web/mobile application (Amazon).					
UNIT II	TEST PLANNING	9			
The Goal of Test Planning, High Level Expectations, Intergroup Responsibilities, Test Phases, Test Strategy, Resource Requirements, Tester Assignments, Test Schedule, Test Cases, Bug Reporting, Metrics and Statistics.					
UNIT III	TEST DESIGN AND EXECUTION	9			
Test Objective and Requirement Identification, Test Design Factors, Modeling a Test Design Process, Model and Results, Boundary Value Testing, Equivalence Class Testing, Path Testing, Data Flow Testing, Test Design Metrics and Effectiveness, Test Procedures. Case Study: Execute the test cases on the client-server or desktop application to identify any defect.					
UNIT IV	ADVANCED TESTING CONCEPTS	9			
Performance Testing: Load Testing, Stress Testing, Volume Testing, Fail-Over Testing, Recovery Testing, Configuration Testing, Compatibility Testing, Usability Testing, Testing the Documentation, Security testing, Testing in the Agile Environment, Testing Web and Mobile Applications.					
UNIT V	TEST AUTOMATION AND TOOLS	9			
Software Test Automation - Skill Needed for Automation - Scope of Automation - Design and Architecture for Automation - Requirements for a Test Tool - Challenges in Automation - Test Metrics and Measurements - Project, Progress and Productivity Metrics - Maintenance of Documents During Testing. Case Study: Install and learn popular software testing tools like Selenium, WinRunner, LoadRunner, Performance Tester etc.					
TOTAL: 45 PERIODS					

COURSE OUTCOMES:

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

- CO1:** Explore the basic concepts of software testing and the need for software testing.
- CO2:** Design Test planning and different activities involved in test planning.
- CO3:** Design effective test cases that can uncover critical defects in the application.
- CO4:** Carry out advanced types of testing.
- CO5:** Automate the software testing using Selenium and TestNG.
- CO6:** Analyze and evaluate the results of testing to ensure software quality, and implement continuous improvement processes based on testing outcomes.

TEXT BOOKS:

1. Yogesh Singh, "Software Testing", Cambridge University Press, 2012.
2. Ron Patton, Software testing, 2nd Edition, 2006, Sams Publishing.
3. Srinivasan Desikan and Gopalaswamy Ramesh, "Software Testing: Principles and Practices", Pearson Education, 2006.
4. Dorothy Graham, Mark Fewster, "Experiences of Test Automation: Case Studies of Software Test Automation", Pearson Education, 2012.
5. Unmesh Gundecha, Satya Avasarala, "Selenium WebDriver 3 Practical Guide" - Second Edition 2018.
6. Paul C. Jorgensen, "Software Testing", A Craftsman Approach, CRC Press, Fourth Edition, 2013.

REFERENCES:

1. Glenford J. Myers, Corey Sandler, Tom Badgett, The Art of Software Testing, 3rd Edition, 2012, John Wiley & Sons, Inc.
2. Mark Fewster, Software Test Automation, 1999, Addison-Wesley.
3. Ali Mili, Fairouz Chier, "Software Testing: Concepts and Operations", Wiley, 2015.

24CS004	DIGITAL AND MOBILE FORENSICS	L	T	P	C
		3	0	0	3
OBJECTIVES: The Course will enable learners to: • Learn how to acquire digital forensic evidence. • Learn how to investigate different digital artifacts and write reports • Understand network forensics processes and procedures • Understand mobile forensics processes and procedures. • Able to analyze SIM cards and analyze mobile file systems.					
UNIT I	ACQUIRING DIGITAL FORENSICS EVIDENCE				9
Types of Computer-Based Investigations - The Forensic Analysis Process-Acquisition of Evidence - Computer Systems.					
UNIT II	DIGITAL FORENSICS INVESTIGATION& REPORTING				9
Computer Investigation Process-Windows Artifact Analysis-RAM Memory Forensic Analysis-Email Forensics-Investigation Techniques-Internet Artifacts.					
UNIT III	NETWORKING FORENSICS				9
Characteristics in the network world-Identifying threats to the enterprise-Data breach surveys-Defining network forensics-Differentiating between computer forensics and network forensics-Digital footprints-Collecting network traffic using tcp dump-Collecting network traffic using Wireshark-Collecting network logs-Acquiring memory using FTK Imager-Tapping into network traffic-Packet sniffing and analysis using Wireshark-Packet sniffing and analysis using Network Miner					
UNIT IV	MOBILE FORENSICS FUNDAMENTALS				9
Mobile Devices vs. Computer Devices in the World of Forensics-Living in the Cloud: The Place to Hide and Store Mobile Data-Preparing, Protecting, and Seizing Digital Device Evidence					
UNIT V	ANALYSING MOBILE INTERNALS				9
Analyzing SIM Cards - Advanced Android Analysis - Advanced iOS Analysis-Case Study: Use Andriller or equivalent to extract data from Android					
TOTAL: 45 PERIODS					
OUTCOMES: Upon completion of the course, the students will be able to: CO1: Understand the process of acquiring digital forensics evidence. CO2: Apply digital forensics investigation techniques and report findings. CO3: Analyze network forensics data to identify threats and breaches. CO4: Explore fundamentals of mobile forensics and evidence preparation. CO5: Investigate mobile device internals for forensic analysis. CO6: Demonstrate ethical considerations in digital forensic practices.					
TEXTBOOKS: 1. William Oettinger, “Learn Computer Forensics: A beginner’s guide to searching, analyzing, and securing digital evidence”, Packt Publishing, 1 st Edition, 2020 2. Samir Datt, “Learning Network Forensics”, Packt Publishing, 1 st Edition, 2016 3. Lee Reiber, “Mobile Forensic Investigations: A Guide to Evidence Collection, Analysis, and Presentation”, McGraw Hill, 2 nd Edition, 2018					
REFERENCES:					

1. Rohit Tamma, Oleg Skulkin, Heather Mahalik, Satish Bommisetty, "Practical Mobile Forensics", Packt Publishing, 3rd Edition, 2018
2. Gerard Johansen, "Digital Forensics and Incident Response: Incident response tools and techniques for effective cyber threat response", Packt Publishing, 3rd Edition, 2022

**MINOR DEGREE CURRICULUM OFFERED BY
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
(FOR OTHER B.E. / B.Tech PROGRAMMES)**

S. No.	NAME OF THE MINOR DEGREE WITH SPECIALIZATION
1	Cyber Security
2	Cloud Computing
3	Full Stack Development

CYBER SECURITY

S. No.	Course Code	Course Title	Contact Periods	L	T	P	C
1	24CS005	Computer Networks	3	3	0	0	3
2	24CS006	Cryptography and Cyber Security	3	3	0	0	3
3	24CS007	Ethical Hacking	3	3	0	0	3
4	24CS004	Digital and Mobile Forensics	3	3	0	0	3
5	24CS812	Capstone Project	12	0	0	12	6

CLOUD COMPUTING

S. No.	Course Code	Course Title	Contact Periods	L	T	P	C
1	24CS008	Operating Systems	3	3	0	0	3
2	24CS009	Distributed and Cloud Computing	3	3	0	0	3
3	24CS010	Cloud Foundations	3	3	0	0	3
4	24CS011	Cloud Architecting	3	3	0	0	3
5	24CS812	Capstone Project	12	0	0	12	6

FULL STACK DEVELOPMENT

S. No.	Course Code	Course Title	Contact Periods	L	T	P	C
1	24CS012	Web Development Frameworks	4	2	0	2	3
2	24CS013	MERN Stack Development	4	2	0	2	3

3	24CS014	Mobile Architecture and Application Development	4	2	0	2	3
4	24CS015	Web Application Security	3	3	0	0	3
5	24CS812	Capstone Project	12	0	0	12	6

24CS005	COMPUTER NETWORKS	L	T	P	C
		3	0	0	3
OBJECTIVES:					
The Course will enable learners to:					
● Study the fundamental concepts of computer networks and physical layer. ● Apply the knowledge of various protocols and techniques used in the data link layer. ● Implement the services of network layer and network layer protocols. ● Illustrate different protocols used in the transport layer. ● Build applications using the application layer protocols.					
UNIT I	INTRODUCTION AND PHYSICAL LAYER				9
Data Communications - Network Types - Protocol Layering - Network Models (OSI, TCP/IP) Networking Devices: Hubs, Bridges, Switches - Performance Metrics - Transmission media - Guided media -Unguided media- Switching-Circuit Switching - Packet Switching.					
UNIT II	DATA LINK LAYER				9
Introduction - Link-Layer Addressing- Error Detection and Correction - Wired LANs: Ethernet - Wireless LANs - Introduction - IEEE 802.11, Bluetooth					
UNIT III	NETWORK LAYER				9
Network Layer Services - IPV4 Addresses - Forwarding of IP Packets - Network Layer Protocols: IP, ICMP - Unicast Routing Algorithms - Protocols - Multicasting Basics - IPV6 Addressing - IPV6 Protocol.					
UNIT IV	TRANSPORT LAYER				9
Introduction - Transport Layer Protocols - Services - Port Numbers - User Datagram Protocol -Transmission Control Protocol - SCTP.					
UNIT V	APPLICATION LAYER				9
Application layer-WWW and HTTP - FTP - Email -Telnet -SSH - DNS - SNMP					
TOTAL: 45 PERIODS					
OUTCOMES:					
Upon completion of the course, the students will be able to:					
CO1: Understand the fundamental concepts of computer networks.					
CO2: Apply the various routing protocols to solve real-world problems.					
CO3: Build simple applications to solve societal problems.					
CO4: Apply the simulation tools to implement various protocols used in the various layers.					

CO5: Analyze the various application layer protocols. CO6: Apply the mathematical knowledge to do performance analysis of various routing protocols.
TEXT BOOK: 1. Data Communications and Networking, Behrouz A. Forouzan, McGraw Hill Education, 5th Ed., 2017.
REFERENCES: 1. Computer Networking- A Top Down Approach, James F. Kurose, University of Massachusetts and Amherst Keith Ross, 8th Edition, 2021. 2. Computer Networks, Andrew S. Tanenbaum, Sixth Edition, Pearson, 2021. 3. Data Communications and Computer Networks, P.C. Gupta, Prentice-Hall of India, 2006. 4. Computer Networks: A Systems Approach, L. L. Peterson and B. S. Davie, Morgan Kaufmann, 3rd ed., 2003.

24CS006	CRYPTOGRAPHY AND CYBER SECURITY	L	T	P	C
		3	0	0	3
OBJECTIVES: ● To understand the fundamentals of network security and security architecture. ● To learn the different symmetric key cryptographic algorithms. ● To study the various asymmetric key cryptographic algorithms and techniques. ● To know the importance of message authentication and integrity. ● To learn the various cyber-crimes and cyber security.					
UNIT I	INTRODUCTION TO SECURITY				9
Computer Security Concepts - The OSI Security Architecture - Security Attacks - Security Services and Mechanisms - A Model for Network Security - Classical encryption techniques: Substitution techniques, Transposition techniques, Steganography - Foundations of modern cryptography: Perfect security					
UNIT II	SYMMETRIC CIPHERS				9
SYMMETRIC KEY CIPHERS: SDES - Block Ciphers - DES, Strength of DES - Differential and linear cryptanalysis - Block cipher design principles - Block cipher mode of operation - Evaluation criteria for AES - Pseudorandom Number Generators - RC4 - Key distribution.					
UNIT III	ASYMMETRIC CRYPTOGRAPHY				9
MATHEMATICS OF ASYMMETRIC KEY CRYPTOGRAPHY: Primes - Primality Testing - Factorization - Euler's totient function, Fermat's and Euler's Theorem - Chinese Remainder Theorem Exponentiation and logarithm ASYMMETRIC KEY CIPHERS: RSA cryptosystem - Key distribution Key management - Diffie Hellman key exchange					

UNIT IV	INTEGRITY AND AUTHENTICATION ALGORITHMS	9
Hash function - Security of hash function: HMAC, CMAC - SHA - Digital signature and authentication protocols - DSS - Schnorr Digital Signature Scheme - ElGamal cryptosystem - Entity Authentication: Biometrics, Passwords, Challenge Response protocols - Authentication applications - Kerberos MUTUAL TRUST: Key management and distribution - Symmetric key distribution using symmetric and asymmetric encryption - Distribution of public keys		
UNIT V	CYBER CRIMES AND CYBER SECURITY	9
Cyber Crime and Information Security - classifications of Cyber Crimes - Tools and Methods - Password Cracking, Keyloggers, Spywares, SQL Injection - Network Access Control - Cloud Security - Web Security - Wireless Security		
TOTAL: 45 PERIODS		
OUTCOMES: Upon completion of the course, the students will be able to: CO1: Understand the fundamentals of networks security, security architecture, threats and vulnerabilities CO2: Apply the different cryptographic operations of symmetric cryptographic algorithms CO3: Apply the different cryptographic operations of public key cryptography CO4: Apply the various Authentication schemes to simulate different applications CO5: Understand various cyber-crimes and cyber security. CO6: Apply the Penetration Tools.		
TEXT BOOK: 1. William Stallings, Cryptography and Network Security: Principles and Practice, Pearson education 8th Edition, 2023. 2. Nina Godbole, Sunit Belapure, "Cyber Security: Understanding Cyber crimes, Computer Forensics and Legal Perspectives", First Edition, Wiley India, 2011.		
REFERENCES: 1. C K Shyamala, N Harini and Dr. T R Padmanabhan: Cryptography and Network Security, Wiley India Pvt.Ltd 1 st Edition,2011 2. Behrouz A. Ferouzan, Debdeep Mukhopadhyay, "Cryptography and Network Security", 3rd Edition, Tata Mc Graw Hill, 2015. 3. Wade Trappe, Lawrence C. Washington: Introduction to Cryptography with Coding Theory, 3rd Edition, 2020. 4. Charles Pfleeger, Shari Pfleeger, Jonathan Margulies, "Security in Computing", Fifth Edition, Prentice Hall, New Delhi, 2015.		

24CS007	ETHICAL HACKING	L	T	P	C
		3	0	0	3
OBJECTIVES: The Course will enable learners to: • To understand Information Security, Cyber threats, attacks, web security. • To know about different modes of hacking tools and phases of penetration tests and Methodologies. • To Gain the knowledge of the use and availability of tools to support an ethical hack • To Gain the knowledge of interpreting the results of a controlled attack					
UNIT I	Fundamentals of Ethical Hacking				9
Introduction to Cyber Security-Overview of Cyber threats - Data and Network Security Attacks - Threats: MAC spoofing - Access control Network protocol and services-Hacking terms - Ethical Hacking overview -Modes of Ethical Hacking - Ethics and Legality.					
UNIT II	Hacking Methodology Reconnaissance				9
Foot printing: Reconnaissance - Footprinting theory - Penetration test - Phases of Penetration test - Methods of Footprinting - Network Information gathering process - Terminologies of Foot printing -Footprinting through search engine directives - Whois tool -NetCraft - Extract Information from DNS - Foot printing from Email servers - Shodan - Dig - MetaGooFil - Social Engineering.					
UNIT III	Scanning and Enumeration				9
Scanning: Concept of Nmap -Port scanning with Nmap - Subnet - Scanning IPs with Nmap Pings and Ping sweeps - Port - Three way handshake - NmapSyn scanning - Nmap TCP Scan - Nmap UDP Scan - Bypass of IPS and IDS - Nmap Script Engine Enumeration: Service Fingerprinting - Vulnerability Scanners - Basic Banner Grabbing - Common Network services - SMTP - DNS - RPCBIND Enumeration - SMB - NetBIOS					
UNIT IV	System and Network Vulnerability				9
Metasploit - Penetration testing with framework Metasploit - Scan services to identify vulnerabilities - Scan FTP services - Scan HTTP services - Exploitation - Post exploitation techniques - Meterpreter - Rootkit - Backdoor - Password hashes - Privilege Escalation - Scanning vulnerable services with Nessus					
UNIT V	Software Vulnerability (OWASP 10)				9
Fundamentals of OWASP Zed Attack Proxy (ZAP) - Web app vulnerability scan - Code Injection Attacks - Broken Authentication - Sensitive Data Exposure - XML External Entities - Broken Access Control - Security misconfiguration - Website pen testing - Cross Site Scripting (XSS) - Insecure Deserialization - Using Components with known vulnerabilities - Insufficient logging and monitoring.					
TOTAL: 45 PERIODS					
OUTCOMES: Upon completion of the course, the students will be able to: CO1: Identify cybersecurity threats and network vulnerabilities to enhance data and system security. CO2: Perform Penetration Testing using tools Metasploit and Nmap to evaluate system defenses. CO3: Apply ethical hacking techniques reconnaissance, scanning, and enumeration to evaluate security posture. CO4: Detect and exploit vulnerabilities in networks and systems while adhering to ethical standards.					

CO5: Analyze Software Vulnerabilities and mitigate risks. CO6: Develop and implement countermeasures against attacks such as SQL injection, DoS, and malware.	
TEXTBOOKS:	
1. McClure, S., Scambray, J. and Kurtz, G., 2012. Hacking Exposed 7Network Security Secrets and Solutions. New York: McGraw-Hill. 2. Engebretson, P., 2013. The Basics Of Hacking And Penetration Testing. Amsterdam: Syngress, an imprint of Elsevier.	
REFERENCES:	
1. Zaid Sabih, Learn Ethical Hacking from Scratch, 2018, PACKT publishing, ISBN: 978-1-78862-205-9 2. Harsh Bothra, Hacking be a hacker with ethics, Khanna Publishing, 2016, ISBN: 978-03-86173-05-8	
LIST OF SOFTWARE:	
1. Kali Linux -Metasploit Framework (MSF) & Nmap 2. WireShark, Jcrypt Tool 3. Burp suite	

24CS004	DIGITAL AND MOBILE FORENSICS	L	T	P	C
		3	0	0	3
OBJECTIVES: The Course will enable learners to: • Learn how to acquire digital forensic evidence. • Learn how to investigate different digital artifacts and write reports • Understand network forensics processes and procedures • Understand mobile forensics processes and procedures. • Able to analyze SIM cards and analyze mobile file systems.					
UNIT I	ACQUIRING DIGITAL FORENSICS EVIDENCE				9
Types of Computer-Based Investigations - The Forensic Analysis Process-Acquisition of Evidence - Computer Systems.					
UNIT II	DIGITAL FORENSICS INVESTIGATION& REPORTING				9
Computer Investigation Process-Windows Artifact Analysis-RAM Memory Forensic Analysis-Email Forensics-Investigation Techniques-Internet Artifacts.					
UNIT III	NETWORKING FORENSICS				9
Characteristics in the network world-Identifying threats to the enterprise-Data breach surveys-Defining network forensics-Differentiating between computer forensics and network forensics-Digital footprints-Collecting network traffic using tcp dump-Collecting network traffic using Wireshark-Collecting network logs-Acquiring memory using FTK Imager-Tapping into network traffic-Packet sniffing and analysis using Wireshark-Packet sniffing and analysis using Network Miner					
UNIT IV	MOBILE FORENSICS FUNDAMENTALS				9
Mobile Devices vs. Computer Devices in the World of Forensics-Living in the Cloud: The Place to Hide and Store Mobile Data-Preparing, Protecting, and Seizing Digital Device Evidence					
UNIT V	ANALYSING MOBILE INTERNALS				9

Analyzing SIM Cards - Advanced Android Analysis - Advanced iOS Analysis-Case Study: Use Andriller or equivalent to extract data from Android			
TOTAL: 45 PERIODS			
OUTCOMES: Upon completion of the course, the students will be able to: CO1: Understand the process of acquiring digital forensics evidence. CO2: Apply digital forensics investigation techniques and report findings. CO3: Analyze network forensics data to identify threats and breaches. CO4: Explore fundamentals of mobile forensics and evidence preparation. CO5: Investigate mobile device internals for forensic analysis. CO6: Demonstrate ethical considerations in digital forensic practices.			
TEXTBOOKS: 1. William Oettinger, "Learn Computer Forensics: A beginner's guide to searching, analyzing, and securing digital evidence", Packt Publishing, 1 st Edition, 2020 2. Samir Datt, "Learning Network Forensics", Packt Publishing, 1 st Edition, 2016 3. Lee Reiber, "Mobile Forensic Investigations: A Guide to Evidence Collection, Analysis, and Presentation", McGraw Hill, 2 nd Edition, 2018			
REFERENCES: 1. Rohit Tamma, Oleg Skulkin, Heather Mahalik, Satish Bommisetty, "Practical Mobile Forensics", Packt Publishing, 3 rd Edition, 2018 2. Gerard Johansen, "Digital Forensics and Incident Response: Incident response tools and techniques for effective cyber threat response", Packt Publishing, 3 rd Edition, 2022			

CLOUD COMPUTING

24CS008	OPERATING SYSTEMS	L	T	P	C
		3	0	0	3
OBJECTIVES: The Course will enable learners to: • Explain the basic concepts of operating systems and process. • Discuss threads and analyse various CPU scheduling algorithms. • Describe the concept of process synchronization and deadlocks. • Analyse various memory management schemes. • Describe I/O management and file systems.					
UNIT I	INTRODUCTION TO OPERATING SYSTEMS AND PROCESSES				9
Introduction: Computer system organization - architecture - Resource management - Protection and Security - Virtualization - Operating System Structures: Services - User and Operating-System Interface - System Calls - System Services - Design and Implementation - Building and Booting an Operating System - Processes: Process Concept - Process Scheduling - Operations on Processes - Inter process Communication - IPC in Shared-Memory Systems - IPC in Message-Passing Systems					
UNIT II	THREADS AND CPU SCHEDULING				9
Threads & Concurrency: Overview - Multicore Programming - Multithreading					

Models - Thread Libraries - Implicit Threading - Threading Issues - CPU Scheduling: Basic Concepts - Scheduling Criteria - Scheduling Algorithms - Thread Scheduling - Multi-Processor Scheduling - Real-Time CPU Scheduling

UNIT III

PROCESS SYNCHRONISATION AND DEADLOCKS

9

Process Synchronization: The critical-section problem - Peterson's Solution, Synchronization hardware, Mutex locks, Semaphores, monitors - Classic problems of synchronization: Bounded Buffer Problem - Reader's & Writer Problem, Dining Philosopher Problem. Deadlock: System model - Deadlock characterization, Methods for handling deadlocks - Deadlock prevention - Deadlock avoidance - Deadlock detection - Recovery from deadlock.

UNIT IV

MEMORY MANAGEMENT

9

Memory Management: Contiguous Memory Allocation - Paging - Structure of the Page Table - Swapping - Virtual Memory: Demand Paging - Copy-on write - Page Replacement - Allocation of frames - Thrashing - Memory Compression

UNIT V

STORAGE MANAGEMENT

9

Mass Storage Structure: Overview of Mass Storage Structure- HDD scheduling - Swap Space Management, I/O systems: I/O Hardware, Application I/O interface, Kernel I/O Subsystem, File System Interface: File Concept - Access Methods - Directory Structure - Protection, File-System Implementation: File-System Structure-File-System Operations - Directory Implementation - Allocation Methods - Free-Space Management, - Case Study-Linux

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Demonstrate the basic concepts of operating systems and process.

CO2: Implement process management techniques using inter-process communication.

CO3: Implement the concepts of process synchronization and deadlocks.

CO4: Apply various memory management schemes for the suitable scenario.

CO5: Describe various I/O and file management techniques.

CO6: Develop practical skills in developing system-level programming.

TEXTBOOKS:

1. Abraham Silberschatz, Peter Baer Galvin and Greg Gagne, "Operating System Concepts" II, 10th Edition, John Wiley and Sons Inc., 2018.
2. Andrew S Tanenbaum, "Modern Operating Systems", Pearson, 5th Edition, 2022 New Delhi.

REFERENCES:

1. William Stallings, "Operating Systems: Internals and Design Principles", 7th Edition, Prentice Hall, 2018.
2. Achyut S. Godbole, Atul Kahate, "Operating Systems", McGraw Hill Education, 2016.

24CS009	DISTRIBUTED AND CLOUD COMPUTING	L	T	P	C
		3	0	0	3
OBJECTIVES: • To articulate the concepts and models underlying distributed computing • To maintain consistency and perform efficient coordination in distributed systems through the use of logical clocks, global states, and snapshot recording algorithms. • To learn different distributed mutual exclusion algorithms. • To develop the ability to understand the cloud infrastructure and virtualization that help in the development of cloud. • To explain the high-level automation and orchestration systems that manage the virtualized infrastructure.					
UNIT I	INTRODUCTION	9			
Definition - Relation to computer system components - Message-passing systems versus shared memory systems - Primitives for distributed communication - Synchronous versus asynchronous executions. A model of distributed computations: A distributed program - A model of distributed executions - Models of communication networks - Global state of a distributed system.					
UNIT II	LOGICAL TIME, GLOBAL STATE, AND SNAPSHOT ALGORITHMS	9			
Logical time - Scalar Time - Vector Time. Global state and snapshot recording algorithms: System model - Snapshot algorithms for FIFO channels and non-FIFO channels.					
UNIT III	DISTRIBUTED MUTUAL EXCLUSION ALGORITHMS	9			
Introduction - Lamport's algorithm - Ricart-Agrawala algorithm - Quorum-based mutual exclusion algorithms - Maekawa's algorithm - Suzuki-Kasami's broadcast algorithm..					
UNIT IV	CLOUD INFRASTRUCTURE AND VIRTUALIZATION	9			
Data Center Infrastructure and Equipment - Virtual Machines - Containers - Virtual Networks - Virtual Storage.					
UNIT V	AUTOMATION AND ORCHESTRATION	9			
Automation - Orchestration: Automated Replication and Parallelism - The MapReduce Paradigm: The MapReduce Programming Paradigm - Splitting Input - Parallelism and Data size - Data access and Data Transmission - Apache Hadoop - Parts of Hadoop - HDFS Components - Block Replication and Fault Tolerance - HDFS and MapReduce - Microservices.					
TOTAL: 45 PERIODS					
OUTCOMES: Upon completion of the course, the students will be able to: CO1: Articulate the main concepts and models underlying distributed computing. CO2: Learn how to maintain consistency and perform efficient coordination in distributed systems through the use of logical clocks, global states, and snapshot recording algorithms.					

CO3: Learn different distributed mutual exclusion algorithms CO4: Develop the ability to understand the cloud infrastructure and virtualization that help in the development of cloud. CO5: Explain the high-level automation and orchestration systems that manage the virtualized infrastructure.
TEXT BOOKS: 1. Ajay D. Kshemkalyani, Mukesh Singhal, “Distributed Computing: Principles, Algorithms, and Systems”, Cambridge University Press, 2011. (Unit 1, 2, 3) 2. Douglass E. Comer, “The Cloud Computing Book: The future of computing explained”, CRC Press, 2023. (Unit 4, 5)
REFERENCES: 1. Arshdeep Bahga, Vijay Madisetti, “Cloud Computing: A Hands-on Approach”, Universities Press Private Limited, 2014. 2. Rajkumar Buyya, Christian Vecchiola, S. ThamaraiSelvi, “Mastering Cloud Computing”, Tata Mcgraw Hill, 2024. 3. Kai Hwang, Geoffrey C. Fox, Jack G. Dongarra, “Distributed and Cloud Computing, From Parallel Processing to the Internet of Things”, Morgan Kaufmann Publishers, 2017. 4. Hagit Attiya, Jennifer Welch, “Distributed Computing: Fundamentals, Simulations and Advanced Topics”, John Wiley & Sons, Inc., 2004.

24CS010	CLOUD FOUNDATIONS	L	T	P	C
		3	0	0	3
OBJECTIVES: • To describe the different ways a user can interact with Cloud. • To discover the different compute options in Cloud and implement a variety of structured and unstructured storage models. • To confer the different application managed service options in the cloud and outline how security is administered in Cloud. • To demonstrate how to build secure networks in the cloud and identify cloud automation and management tools. • To determine a variety of managed big data services in the cloud.					
UNIT I	INTRODUCTION TO CLOUD				9
Introduction to Cloud Computing - Cloud Versus Traditional Architecture - IaaS, PaaS, and SaaS - Cloud Architecture - The Console - Understanding projects - Billing - Install and configure Cloud SDK - Use Cloud Shell - APIs - Cloud Console Mobile App.					

UNIT II	COMPUTE AND STORAGE	9
Compute options in the cloud - Exploring IaaS with Compute Engine - Configuring elastic apps with autoscaling - Exploring PaaS - Event driven programs - Containerizing and orchestrating apps - Storage options in the cloud - Structured and unstructured storage in the cloud - Unstructured storage using Cloud Storage - SQL managed services - NoSQL managed services.		
UNIT III	APIs AND SECURITY IN THE CLOUD	9
The purpose of APIs - API Services - Managed message services - Introduction to security in the cloud - The shared security model - Encryption options - Authentication and authorization with Cloud IAM - Identify Best Practices for Authorization using Cloud IAM.		
UNIT IV	NETWORKING, AUTOMATION AND MANAGEMENT TOOLS	9
Introduction to networking in the cloud - Defining a Virtual Private Cloud - Public and private IP address basics - Cloud network architecture - Routes and firewall rules in the cloud - Multiple VPC networks - Building hybrid clouds using VPNs - Different options for load balancing - Introduction to Infrastructure as Code - Terraform - Monitoring and management tools.		
UNIT V	BIG DATA AND MACHINE LEARNING SERVICES	9
Introduction to big data managed services in the cloud - Leverage big data operations - Build Extract, Transform, and Load pipelines - Enterprise Data Warehouse Services - Introduction to machine learning in the cloud - Building bespoke machine learning models with AI Platform - Pre-trained machine learning APIs.		
List of Exercise/Experiments: 3. Create a cluster, run a simple Apache Spark job in the cluster, then modify the number of workers in the cluster. 4. Set up your Python development environment, get the relevant SDK for Python, and run an example pipeline using the Cloud Console.		
TOTAL: 45 PERIODS		
OUTCOMES: At the end of this course, the students will be able to: CO1: Explain the basic concepts of cloud computing and the different cloud service models. CO2: Deploy and manage various cloud compute and storage options. CO3: Implement security measures in the cloud, including Cloud IAM for authentication and authorization. CO4: Design and manage cloud networks and use automation tools like Terraform. CO5: Use cloud monitoring tools to optimize cloud resource performance. CO6: Leverage cloud-based big data and machine learning services to build data pipelines and models.		

REFERENCES:

1. <https://cloud.google.com/docs>
2. <https://www.cloudskillsboost.google/paths/36>
3. <https://nptel.ac.in/courses/106105223>
4. Anthony J. Sequeira, "AWS Certified Cloud Practitioner (CLF-C01) Cert Guide", First Edition, Pearson Education, 2020.
5. [AWS Documentation \(amazon.com\)](#)
6. [AWS Skill Builder](#)
7. AWS Academy Cloud Foundations Course - https://www.awsacademy.com/vforcesite/LMS_Login

24CS011	CLOUD ARCHITECTING	L	T	P	C
		3	0	0	3
OBJECTIVES: The Course will enable learners: • To make architectural decisions based on AWS architectural principles and best practices. • To describe the features and benefits of Amazon EC2 instances, and compare and contrast managed and unmanaged database services. • To create a secure and scalable AWS network environment with VPC, and configure IAM for improved security and efficiency. • To use AWS services to make infrastructure scalable, reliable, and highly available. • To use AWS managed services to enable greater flexibility and resiliency in an infrastructure.					
UNIT I	INTRODUCING CLOUD ARCHITECTING AND STORAGE LAYER				9
Cloud architecting - The AWS Well-Architected Framework - AWS global infrastructure - Amazon S3 - Amazon S3 Versioning - Storing data in Amazon S3 - Moving data to and from Amazon S3 - Amazon S3 Transfer Acceleration - Choosing Regions for your architecture.					
UNIT II	COMPUTE LAYER AND DATABASE LAYER				9
Adding compute with Amazon EC2 - Choosing an Amazon Machine Image (AMI) to launch an Amazon EC2 instance - Selecting an Amazon EC2 instance type - Using user data to configure an EC2 instance - Adding storage to an Amazon EC2 instance - Amazon EC2 pricing options - Amazon EC2 considerations - Database layer considerations - Amazon Relational Database Service (Amazon RDS) -					

Amazon DynamoDB - Database security controls - Migrating data into AWS databases.		
UNIT III	CREATING AND CONNECTING NETWORKS	9
Creating an AWS networking environment - Connecting your AWS networking environment to the internet - Securing your AWS networking environment - Connecting your remote network with AWS Site-to-Site VPN - Connecting your remote network with AWS Direct Connect - Connecting virtual private clouds (VPCs) in AWS with VPC peering - Scaling your VPC network with AWS Transit Gateway - AWS Transit Gateway - Connecting your VPC to supported AWS services. Securing User and Application Access: Account users and AWS Identity and Access Management (IAM) - Organizing users - Federating users - Multiple accounts.		
UNIT IV	RESILIENT CLOUD ARCHITECTURE	9
Scaling your compute resources - Scaling your databases - Designing an environment that's highly available - Monitoring - Reasons to automate - Automating your infrastructure - Automating deployments - AWS Elastic Beanstalk - Overview of caching - Edge caching - Caching web sessions - Caching databases.		
UNIT V	BUILDING DECOUPLED ARCHITECTURES, MICROSERVICES AND SERVERLESS ARCHITECTURE	9
Decoupling your architecture - Decoupling with Amazon Simple Queue Service (Amazon SQS) - Decoupling with Amazon Simple Notification Service (Amazon SNS) - Sending messages between cloud applications and on-premises with Amazon MQ. Introducing microservices - Building microservice applications with AWS container services - Introducing serverless architectures - Building serverless architectures with AWS Lambda - Extending serverless architectures with Amazon API Gateway - Orchestrating microservices with AWS Step Functions - Disaster planning strategies - Disaster recover patterns.		
TOTAL: 45 PERIODS		
OUTCOMES: Upon Completion of this course, the students will be able to: CO1: Explain cloud architecture principles and AWS storage solutions. CO2: Deploy and manage AWS compute and database resources securely. CO3: Design and configure secure AWS networks using VPC and IAM. CO4: Implement scalable and resilient AWS architectures with high availability. CO5: Build decoupled and serverless applications using AWS services like Lambda. CO6: Develop disaster recovery strategies for AWS environments.		
REFERENCES: 1. AWS Certified Solutions Architect Official Study Guide by Joe Baron, Hisham Baz, Tim Bixler 2. Architecting the Cloud by Michael Kavis. 3. AWS Documentation (amazon.com) 4. AWS Skill Builder 5. AWS Academy Cloud Architecting Course - https://www.awsacademy.com/vforcesite/LMS_Login		

24CS012	WEB DEVELOPMENT FRAMEWORKS (Lab Integrated)	L	T	P	C
		2	0	2	3
COURSE OBJECTIVES: The Course will enable the learners: • To understand web semantics and related tools and framework • To get hands on latest JS based web frameworks • To develop a scalable and responsive web application • To develop an industry ready application web enterprise feature					
UNIT I	ADVANCED TYPESCRIPT				6+6
Introduction to HTML5 and CSS3, Media Queries, JS, DOM, BootStrap, Variables, Loops, Operators, Scope, Hoisting, Arrays, Spread, REST, DeStructuring Introduction, Advantage of Using TS over JS, and where to Use and not to use TS - Understanding the Compiler (Transpiler), and its options, Scope of TS - Variable Scopes, Static Data Types - String, Number, Interface, Date - Union, Tuple, Undefined Data Types, Unknown vs any vs Never - Object Oriented , Arrow Funcions - Types, KeyOf, TypeOf, InstanceOf ,Narrowing, Conditional Types - Generics, Enum ,Required / Partial / Optional - Arrays, Modules ,Async Processing w Call backs ,Type Inference, Type Compatibility, Utility Type - Unit Testing, TSLint					
<u>List of Exercise/Experiments</u> 1. Create a TS Object for Bank Account (with attributes like à customer name, account type, balance, data of creation, bank name, branch name, pan card number). Using JS Object keyword, try to perform following activities ➤ List down all the entries of the bank object ➤ Check the existence of a key ➤ If key found, get the value for the key 2. Spread Operator ➤ Merge Customer and Account Arrays ➤ Update the Customer Object with the new values ➤ Develop a function that takes an Spread Argument and calculates total balance.					
UNIT II	INTRODUCTION TO REACTJS				6+6
Introduction to React - ES6 Features, What is React?, React Features - Setting up React Development Environment:- Node.js and npm installation, Create React App, Project structure - JSX (JavaScript XML):- What is JSX?,JSX Syntax and Rules,JSX Expressions - Components in React:- Functional Components, Class Components, Props and PropTypes - State and Lifecycle :- State and setState, Lifecycle Methods, Mounting, Updating, and Unmounting, Handling Events in React - Event Handling in					

React :- Synthetic Events, Event Binding, - Conditional Rendering:-If-else Statements, Ternary Operator, Logical && Operator
 Lists and Keys:- Rendering Lists, Keys and Reconciliation, Extracting Components.

List of Exercise/Experiments

1. A leading bank from APAC wants to modernize their banking services and decided to build a online multi channel mobile ecommerce platform. As part of the drive, starting building following feature set in a staggered model emplying ReactJS as front end library and associated libs from React eco system. Feature to be implemented are
 - a. User Login Page
 - b. Account Summary
 - c. Funds Transfer (within bank and outside bank)
 - d. Recurring and Fixed deposits
 - e. Letter of Credit
 - f. Salary or 3rd Party Payment
2. Unit 2 Scope - Project Setup, Web App Layout Completion using BootStrap or Tailwind, Login Page Implementation, Landing Page Implementation, Authentication and Authorization Implementation.

UNIT III	REACTJS COMPONENTS	6+6
-----------------	---------------------------	------------

Forms and Controlled Components :- Form Handling in React, Controlled Components, Uncontrolled Components - Basic Hooks :- useState, useRef, useEffect, - Routing in React:- Introduction to React Router, Route, Link, and Switch Components, Route Parameters, useNavigate, useParams, - REST API - Axios GET/PUT/Delete/Remove, Interceptor, Headers, Authorization Token, Promise and Observables (via rxjs)

List of Exercise/Experiments

1. Extend the Project developed in previous section with newly learnt concepts
 - a. Unit 3 Scope - Forms and Validation (React Form Validation), Integration of Back End Apis via Axios, API Security Implementation, Routes and Navigation with Private Routes, Usage of useEffect, useContext hooks

UNIT IV	REACT PRO TOOLKIT: ERROR MANAGEMENT, ABSTRACTIONS & DATA HANDLING	6+6
----------------	--	------------

Error Handling :- Error Boundaries, componentDidCatch, Error Handling Strategies Higher-Order Components (HOCs) :- What are HOCs?, Creating and Using HOCs, HOCs vs Render Props, Code Splitting and Lazy Loading, Server-Side Rendering - Data Fetching with React Query :- React Query, Introduction to React Query, Query Keys and Query Functions, Query Invalidation and Refetching.

List of Exercise/Experiments

1. Extend the Project developed in previous section with newly learnt concepts a. Unit 4 Scope - Completion of Remaining Modules, Error Handling, HOC and AUX implementation, Lazy loaded components for improved performance Extend the Project developed in previous section with newly learnt concepts		
UNIT V	REACT UNDER THE HOOD: TESTING, CONTEXT API, AND REDUX	6+6
Testing React Components :- Introduction to Testing, Jest Framework, React Testing Library Context API :- Creating Context, Providing and Consuming Context, useContext Hook Redux Overview :- What is Redux?, Redux Principles, Single Source of Truth - Redux Actions and Reducers :- Redux Actions, Redux Reducers, Combining Reducers.		
<u>List of Exercise/Experiments</u>		
1. Extend the Project developed in previous section with newly learnt concepts a. Unit 5 Scope - Unit Testing using JEST, Redux implementation for state management.		
<u>Business Use Case Implementations</u>		
1. Student Management System 2. Retail Bank System 3. eCommerce System 4. Student LMS Management System		
TOTAL: 30(T) + 30 (P) = 60 PERIODS		
<u>COURSE OUTCOMES:</u>		
Upon completion of the course, the students will be able to:		
CO1. Understand and apply modern web technologies including HTML5, CSS3, JavaScript, and advanced TypeScript concepts to build dynamic web components. CO2. Develop responsive and modular front-end applications using ReactJS, including component creation, state management, and routing. CO3. Implement advanced React features like hooks (useState, useEffect, useRef), React Router, and REST API integration using Axios for dynamic content handling. CO4. Utilize higher-order components (HOCs), lazy loading, and server-side rendering to optimize and abstract React applications. CO5. Perform unit testing using Jest and RTL, and manage global application state efficiently using Context API and Redux. CO6. Design and deliver scalable and real-world enterprise web applications with complete user interface flow, security, and error handling.		
<u>TEXTBOOKS:</u>		
1. <u>David Flanagan</u> , Javascript The Definitive Guide, Paperback, 7 th Edition, 2020. 2. David Choi ,Full-Stack React, TypeScript, and Node: Build cloud-ready web applications using React 17 with Hooks and GraphQL Paperback - Import, 18 December 2020.		

3. Mehul Mohan, Advanced Web Development with React Paperback - 1 January 2020.

E-RESOURCES:

1. Parental Website - <https://reactjs.org/>
2. The Road to Learn React: Your journey to master plain yet pragmatic React.js by Robin Wieruch
3. Learning React: Functional Web Development with React and Redux by Alex Banks and Eve Porcello
4. Learning React by KirupaChinnathambi
5. "React Up & Running" by StoyanStefanov
6. <https://www.edureka.co/reactjs-redux-certification-training>
7. CodePen
8. CodeSandbox (Preferred)
9. Stackblitz

LIST OF EQUIPMENTS:

- NodeJS (v22.11.2)
- Github as code repository
- Visual studio code as IDE
- RTL as unit testing framework
- Responsive design w bootstrap
- ReactJS installation (v17)
- Chrome / FireFox Browsers (latest)
- Responsive using Media Queries & Bootstrap Material & Antdesign
- Design based Apps

24CS013	MERN STACK DEVELOPMENT (Lab Integrated)	L	T	P	C
		2	0	2	3
OBJECTIVES: The Course will enable learners: • Design applications using Node .JS • Create architecture involving Express and graphQL • Develop applications using mongoDB • Apply the concepts of React Components and State • Build web applications using React Router, Forms and Bootstrap					
UNIT I	INTRODUCTION TO MERN and NODE JS				6+6

Introduction - MERN Components - Node JS: Introduction to Node JS, Setting up Node.js, Node.js Modules - HTTP Servers and Clients - Request Handling - Database connectivity - Data Storage and Retrieval - Dynamic Client/Server Interaction with Socket.IO

List of Exercise/Experiments:

1. Create your own modules and return Current date and time.
2. Create the HTTP server using createServer() method that listens to server ports and gives a response back to the client.

UNIT II	EXPRESS	6+6
----------------	----------------	------------

Express - Routing - Request Matching - Route parameters - Route Lookup - Handler Function Request Object - Response Object - Middleware - REST API - GraphQL - About API - List API
 - List API Integration - Custom Scalar Types - Create API Integration - Query Variables - Input Validations - Displaying Errors

List of Exercise/Experiments:

1. Create an application using Express.js to print Hello world on the Homepage.
2. Build a Simple Node.js/Express server that handles GET and POST request and returns data in JSON format.

UNIT III	MongoDB	6+6
-----------------	----------------	------------

MongoDB Basics - CRUD Operations - NODE.js driver - Schema Initialization - Reading from MongoDB - Writing to MongoDB - UI Server - Multiple Environments - Proxy-based Architecture - ESLint - ESLint for Front End - React PropTypes - Back End Modules - Front End Modules and Webpack - Transform and bundle - Libraries Bundle - Module Replacement Debugging - Defineplugin - Product Optimization.

List of Exercise/Experiments:

3. Build an application to perform Basic CRUD operation in MongoDB using Node/Express.
4. Building the MongoDB database for the My To-do List app.

UNIT IV	REACT COMPONENTS AND STATE	6+6
----------------	-----------------------------------	------------

React Components- Issue Tracker - React Classes - Composing Components - Passing Data
 - Dynamic Composition - React State - Hooks - Event handling - Stateless Components Designing Components.

List of Exercise/Experiments:

1. Write a program to create a simple calculator Application using React JS
2. Build a simple React application that displays the list of items and allows the user to add new items to the list.

UNIT V	REACT ROUTER, FORMS AND BOOTSTRAP	6+6
---------------	--	------------

React Router - Simple Routing - Route Parameters - Query Parameters - Links - Programmatic Navigation - Nested Routes - React Forms - Controlled Components - Specialized Input Components - Update API - Delete API - React Bootstrap - Buttons
-Navigation - Panels - Tables - Forms - Grid - Inline Forms - Horizontal Forms - Validation Alerts - Toasts - Modals.

List of Exercise/Experiments:

1. Create a Simple Login form using React JS.
2. Build an application for an E-Commerce platform.
3. Build a full-stack MERN app that allows the user to register, login, and create a list of items that are stored in a MongoDB database. The app should also display the list of items using React components.

TOTAL: 30(T) + 30(P) = 60 PERIODS

OUTCOMES:

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

CO1: Develop applications using Node.js for backend functionality and server-side logic.

CO2: Handle data queries with GraphQL on an Express server for efficient API management.

CO3: Build applications using MongoDB to perform CRUD operations and manage databases.

CO4: Apply dynamic composition and event handling techniques.

CO5: Implement React forms and Bootstrap for creating responsive, user-friendly interfaces.

CO6: Design and develop full-stack applications with the MERN stack.

TEXT BOOKS:

1. Vasan Subramanian, Pro MERN Stack - Full stack web app development, 2nd Edition, Apress, 2019
2. David Herron , Node.js Web Development - Fourth Edition, Packt Publishing, 2018.

REFERENCES:

1. Adam Freeman, Essential TypeScript, Apress, 2019.
2. Shama Hoque, Full-Stack React Projects, 2nd edition, Apress, 2022
3. Karl Seguin, "The Little Mongo DB Book",
<https://github.com/karlseguin/the-little-mongodb-book>.
4. <https://aws.amazon.com/education/awseducate/>
5. <http://packaging.ubuntu.com/html/packaging-new-software.html>
6. https://www.tutorialspoint.com/nodejs/nodejs_express_framework.htm

LIST OF EQUIPMENTS:

Node, Express, MongoDB, React

24CS014	MOBILE ARCHITECTURE AND APPLICATION DEVELOPMENT (Lab Integrated)	L	T	P	C
		2	0	2	3
OBJECTIVES: The Course will enable learners to: - Understand Android SDK - Understand and develop Android applications - Learn the concepts of iOS application development - Develop Hybrid platform application - Develop cross-platform applications with event handling					
UNIT I	MOBILE ARCHITECTURE AND BASICS IN APP DEVELOPMENT	6+6			
Mobile Device Operating Systems – Special Constraints & Requirements – Commercial Mobile Operating Systems – Software Development Kit: iOS, Android, BlackBerry, Mobile App Architecture: Android - iOS - Hybrid - Cross Platform - Factors of Mobile App Architecture - Windows Phone – MCommerce – Structure – Pros & Cons – Mobile Payment System – Security Issues.					
List of Exercise/Experiments 1. develop an application that uses GUI components, Font, Layout Managers and event listeners. 2. develop a native application that uses GPS location information.					
UNIT II	ANDROID APPLICATION	6+6			
Introduction to Android: The Android Platform - Android SDK, Eclipse Installation, Android Installation, Building First Android application - Life cycle of android application - Android User Interface Design Essentials - Android APIs - Deploying android applications.					
List of Exercise/Experiments 1. Create an android application that converts the user input text to voice. 2. Develop a Mobile application for simple and day to day needs.					
UNIT III	IOS APPLICATION	6+6			
Introduction - Performance metrics - Core optimizations - concurrent programming - Application life cycle - User Interface - Network -Security - Testing - Tools - Instrumentation and Analytics					
List of Exercise/Experiments 1. Implement an iOS app allowing users to type or paste text and have it spoken aloud using Apple's speech synthesis. 2. Develop an ios application for simple and day to day needs.					
UNIT IV	HYBRID APP ENVIRONMENT	6+6			

Hybrid Web App, Benefits of Hybrid App, Criteria for creating Native App, Tools for creating Hybrid App, Cons of Hybrid App, Popular Hybrid App Development Frameworks, Ionic, Apache Cordova

List of Exercise/Experiments

1. 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.
2. Design and develop an android application using Apache Cordova to find and display the current location of the user.

UNIT V

CROSS-PLATFORM APP DEVELOPMENT

6+6

What is Cross-platform App, Benefits of Cross-platform App, Criteria for creating Cross- platform App, Tools for creating Cross-platform App, Cons of Cross-platform App, Popular Cross-platform App Development Frameworks, Flutter, Xamarin, React-Native, Basics of React Native, Native Components, JSX, State, Props

List of Exercise/Experiments

1. Design and develop a cross platform application for day to day task (to-do) management.
2. Build a cross platform application for a simple expense manager which allows entering expenses and income on each day and displays category wise weekly income and expenses.

TOTAL: 30(L)+30(P) = 60 PERIODS

COURSE OUTCOMES:

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

CO1: Identify various concepts of mobile architecture.

CO2: Create, test and debug Android application by setting up Android development

CO3: Develop iOS application with performance metrics

CO4: Develop hybrid applications with basic event handling.

CO5: Implement cross platform applications with basic GUI and event handling.

CO6: Deploy applications to the Android Market place for distribution.

TEXT BOOKS:

1. Gaurav Vaish, "High Performance iOS Apps", O'Reilly Media, 1st ed. 2016.
2. Lauren Darcey and Shane Conder, "Android Wireless Application Development", Pearson Education, 2nd ed.2012.
3. Prasanth Kumar Pattnaik, Rajib Mall, "Fundamentals of Mobile Computing", PHI Learning Pvt. Ltd, New Delhi-2012.

REFERENCES:

1. Pawan Lingras, Matt Triff, Rucha Lingras, "Building Cross-Platform Mobile and Web Apps for Engineers and Scientists: An Active Learning Approach", Cengage Learning, 2016. 2. Mahesh Panhale, Beginning Hybrid Mobile Application Development, Apress, 2016. 3. Android Application development James C. Sheusi Cengage learning 2017. 4. Erik Hellman, "Android Programming - Pushing the Limits", 1st Edition, Wiley India Pvt Ltd, 2014.
LIST OF SOFTWARES:
3. Software required: Android Studio, macOS. 4. Software Tools required: Android 11 SDK, Xcode.

24CS015	WEB APPLICATION SECURITY	L	T	P	C
		3	0	0	3
OBJECTIVES: The Course will enable learners to: • Understand the fundamentals of web application security • Focus on wide aspects of secure development and deployment of web applications • Learn how to build secure APIs • Learn the basics of vulnerability assessment and penetration testing • Get an insight about Hacking techniques and Tools					
UNIT I	FUNDAMENTALS OF WEB APPLICATION SECURITY				9
The history of Software Security-Recognizing Web Application Security Threats, Web Application Security, Authentication and Authorization, Secure Socket layer, Transport layer Security, Session Management-Input Validation.					
UNIT II	SECURE WEB APPLICATIONS AND APIs				9
Introduction to Web & API Security- Security Testing - Security Incident Response Planning - Secure Development Frameworks - Authentication and Session Management - API Security Fundamentals - Advanced API Security Mechanisms - Securing Service-to-Service APIs - Securing Microservice APIs - Securing Incoming Requests - Case Study: Vulnerability Identification using OWASP ZAP.					
UNIT III	VULNERABILITY ASSESSMENT AND PENETRATION TESTING				9
Vulnerability Assessment Lifecycle, Vulnerability Assessment Tools: Cloud-based vulnerability scanners, Host-based vulnerability scanners, Network-based vulnerability scanners, Database- based vulnerability scanners, Types of Penetration Tests: External Testing, Web Application Testing, Internal Penetration Testing, SSID or Wireless Testing, Mobile Application Testing.					
UNIT IV	HACKING TECHNIQUES AND TOOLS				9

Social Engineering, Injection, Cross-Site Scripting (XSS), Broken Authentication and Session Management, Cross-Site Request Forgery, Security Misconfiguration, Insecure Cryptographic Storage, Failure to Restrict URL Access, Tools: Comodo, OpenVAS, Nexpose, Nikto, Burp Suite, etc. Case Study: Attack the website using Social Engineering method.

UNIT V	SECURITY FOR AI-ENABLED APPLICATIONS	9
---------------	---	----------

Introduction to AI Security - AI Attack Surface - Data Security in AI - Common AI Threats - Securing AI Applications - API and Model Security - Testing and Monitoring AI Systems - Responsible and Ethical AI - Use Case: Secure Test Data Generation System (Scenario: AI generates test data for an application)

TOTAL: 45 PERIODS

COURSE OUTCOMES:

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

- CO1:** Explain the basic concepts of web application security and the need for it.
- CO2:** Develop a thorough understanding of the process for securely designing, implementing, testing, and deploying web applications.
- CO3:** Acquire the skill to design and develop Secure Web Applications that use Secure APIs.
- CO4:** Develop the ability to grasp the importance of conducting vulnerability assessments and penetration testing.
- CO5:** Acquire the skill to think like a hacker and to use hackers tool sets.
- CO6:** Develop the ability to implement and manage web application security measures effectively, ensuring ongoing protection against evolving threats.

TEXT BOOKS:

1. Andrew Hoffman, Web Application Security: Exploitation and Countermeasures for Modern Web Applications, First Edition, 2020, O'Reilly Media, Inc.
2. Bryan Sullivan, Vincent Liu, Web Application Security: A Beginners Guide, 2012, The McGraw-Hill Companies.
3. Neil Madden, API Security in Action, 2020, Manning Publications Co., NY, USA.
4. Rishabh Bhardwaj, " Vulnerability Assessment and Penetration Testing (VAPT): Detailed guide with highlighted threats, risk exposure, and remediations", 1st Edition, BPB Publications, 2025

REFERENCES:

1. Michael Cross, Developer's Guide to Web Application Security, 2007, Syngress Publishing, Inc.
2. Ravi Das and Greg Johnson, Testing and Securing Web Applications, 2021, Taylor & Francis Group, LLC.
3. Prabath Siriwardena, Advanced API Security, 2020, Apress Media LLC, USA.
4. Malcom McDonald, Web Security for Developers, 2020, No Starch Press, Inc.
5. Allen Harper, Shon Harris, Jonathan Ness, Chris Eagle, Gideon Lenkey, and Terron Williams Grey Hat Hacking: The Ethical Hacker's Handbook, Third Edition, 2011, The McGraw-Hill Companies.

