R.M.D. ENGINEERING COLLEGE

(An Autonomous Institution)

B.E. - COMPUTER SCIENCE AND ENGINEERING

REGULATIONS – 2024

CHOICE BASED CREDIT SYSTEM

I - VIII SEMESTER CURRICULUM

(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++	ESC	6	3	0	3	4.5
5.	24CS102	Software Development Practices	ESC	6	3	0	3	4.5
6.	24CH101	Engineering Chemistry	BSC	5	3	0	2	4
7.	24EC101	Digital Principles and System Design	ESC	5	3	0	2	4
LABORATORY COURSES								
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	BSC	5	3	0	2	4
4.	24CS201	Data Structures	ESC	6	3	0	3	4.5
5.	24CS202	Java Programming	ESC	6	3	0	3	4.5
6.	24PH201	Physics for Information Science	BSC	5	3	0	2	4
7.	24AI201	Introduction to Artificial Intelligence	ESC	4	2	0	2	3
LABORATORY COURSES								
8.	24GE211	Idea Lab II	EEC	2	0	0	2	1
MANDATORY COURSES								
9.	24CH102	Environmental Science & Sustainability (Non Credit)	MC	2	2	0	0	0
10.	24MC201	Yoga for Stress Management (Non Credit)	AC	1	0	0	1	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	3	0	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.		Indian Constitution(Non Credit)	MC	1	1	0	0	0
			TOTAL	**33**	**18**	**1**	**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.	24IT402	Web Development Frameworks	PCC	6	3	0	3	4.5
5.		Professional Elective I	PEC	4	2	0	2	3
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.		Value Education (Non Credit)	AC	1	1	0	0	0
			TOTAL	**31**	**15**	**0**	**16**	**22**

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 III	PEC	3	3	0	0	3
THEORY COURSES WITH LABORATORY COMPONENT								
3.	24CS502	Distributed and Cloud Computing	PCC	4	2	0	2	3
4.		Machine Learning	PCC	5	3	0	2	4
5.		Professional Elective II	PEC	4	2	0	2	3
LABORATORY COURSES								
6.		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.		Essence of Indian Traditional Knowledge (Non Credit)	MC	1	1	0	0	0
			TOTAL	**31**	**14**	**0**	**17**	**21.5**

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.		Design Thinking	HSMC	3	1	0	2	2
LABORATORY COURSES								
7.		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								
10.		Personality Development (Non Credit)	AC	2	2	0	0	0
			TOTAL	**32**	**17**	**0**	**15**	**22.5**

SEMESTER-VII

Sl. No.	COURSE CODE	COURSE TITLE	CATEGORY	CONTACT PERIODS	L	T	P	C
THEORY COURSES								
1.		Professional Elective VI	PEC	3	3	0	0	3
2.		Open Elective II	OEC	3	3	0	0	3
THEORY COURSES WITH LABORATORY COMPONENT								
3.	24CS701	Data Analytics	PCC	5	3	0	2	4
4.	24CS702	Wireless and Mobile Communication	PCC	4	2	0	2	3
5.		Professional Elective V	PEC	4	2	0	2	3
LABORATORY COURSES								
6.		Internship/Seminar (4 weeks)	EEC	4	0	0	4	2
			TOTAL	**23**	**13**	**0**	**10**	**18**

SEMESTER-VIII

Sl. No.	COURSE CODE	COURSE TITLE	CATEGORY	CONTACT PERIODS	L	T	P	C
EMPLOYABILITY ENHANCEMENT COURSES								
1.	24CS811	Project Work	EEC	16	0	0	16	8
			TOTAL	**16**	**0**	**0**	**16**	**8**

TOTAL CREDITS: 163

Summary

S. No.	Subject Area	CREDITS AS PER SEMESTER								Total Credits	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										
TOTAL		23	23	25	22	21.5	22.5	18	8	163	

HSMC – Humanities and Social Sciences including Management courses; **BSC** – Basic Science Courses; **ESC** – Engineering Science Courses including workshop, drawing, basics of electrical/mechanical/computer etc.; **PCC** – Professional Core Courses; **PEC** – Professional Elective Courses relevant to chosen specialization/branch; **OEC** – Open Subjects–Electives from other technical and/or emerging subjects **EEC** – Project Work, Seminar and Internship in Industry or elsewhere

PROFESSIONAL ELECTIVES/ HONOUR VERTICALS

CYBER SECURITY

Sl. 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.	24IT903	Rest Application Development Using Spring Boot and JPA (Lab Integrated)	PEC	4	2	0	2	3
4.	24CS903	Blockchain Technology	PEC	3	3	0	0	3
5.	24CS904	Cloud and Data Security	PEC	3	3	0	0	3

6.	24CS905	Enterprise Cyber Security	PEC	3	3	0	0	3
7.	24CS906	Digital and Mobile Forensics	PEC	3	3	0	0	3
8.	24CS907	Vulnerability Analysis and Penetration Testing*		3	3	0	0	3
9.	24CS908	Engineering Secure Software Systems*		3	3	0	0	3
10.	24CS909	Network Design and Programming*		3	3	0	0	3
11.	24CS910	Fault Tolerant Computing*		3	3	0	0	3
12.	24CS928	Capstone Project*		12	0	0	0	6

*** Honor Subjects**

CLOUD COMPUTING

Sl. No.	COURSE CODE	COURSE TITLE	CATEGORY	CONTACT PERIODS	L	T	P	C
1.	24CS911	Cloud Foundations (Lab Integrated)	PEC	4	2	0	2	3
2.	24IT903	Rest Application Development Using Spring Boot and JPA (Lab Integrated)	PEC	4	2	0	2	3
3.	24CS912	Virtualization	PEC	3	3	0	0	3
4.	24CS913	Data Engineering in Cloud	PEC	3	3	0	0	3
5.	24CS914	Devops (Lab Integrated)	PEC	4	2	0	2	3
6.	24CS915	Machine Learning for NLP in Cloud	PEC	3	3	0	0	3
7.	24CS916	Microservice Architecture	PEC	3	3	0	0	3
8.	24CS917	Software Defined Networks*		3	3	0	0	3
9.	24CS918	Storage Technologies*		3	3	0	0	3
10.	24CS919	Cloud Security Foundations*		3	3	0	0	3
11.	24CS920	Cloud Services Management*		3	3	0	0	3
12.	24CS928	Capstone Project*		12	0	0	0	6

*** Honor Subjects**

FULL STACK TECHNOLOGY

Sl. No.	COURSE CODE	COURSE TITLE	CATEGORY	CONTACT PERIODS	L	T	P	C
1.	24CS921	UI/UX Design (Lab Integrated)	PEC	4	2	0	2	3
2.	24IT903	Rest Application Development Using Spring Boot and JPA (Lab Integrated)	PEC	4	2	0	2	3
3.	24CS922	Software Testing and Automation	PEC	3	3	0	0	3
4.	24CS903	Blockchain Technology	PEC	3	3	0	0	3
5.	24CS923	Usability Design of Software Application	PEC	3	3	0	0	3
6.	24CS914	Devops (Lab Integrated)	PEC	4	2	0	2	3
7.	24CS916	Microservice Architecture	PEC	3	3	0	0	3
8.	24CS924	MERN Stack Development (Lab Integrated)*		4	2	0	2	3
9.	24CS925	Web Application Security*		3	3	0	0	3
10.	24CS926	Generative AI Fundamentals*		3	3	0	0	3
11.	24CS927	Mobile Architecture and Application Development (Lab Integrated)*		4	2	0	2	3
12.	24CS928	Capstone Project*		12	0	0	0	6

*** Honor Subjects**

R2024 CURRICULUM
B.E (HONOURS) IN COMPUTER SCIENCE AND ENGINEERING WITH SPECIALIZATION IN

Sl. No.	NAME OF THE HONOURS DEGREE WITH SPECIALIZATION
1	Cyber Security
2	Cloud Computing
3	Full Stack Technology
4	Data Science
5	Artificial Intelligence

HONOURS - DATASCIENCE

Sl. No.	COURSE CODE	COURSE TITLE	CATEGORY	CONTACT PERIODS	L	T	P	C
1.	24AM901	Foundations of Data Science	PEC	4	2	0	2	3
2.	24AM502	Data Exploration and Visualization	PEC	4	2	0	2	3
3.		Healthcare Analytics	PEC	4	3	0	0	3
4.	24CS926	Generative AI Fundamentals	PEC	4	3	0	0	3
5.	24AM902	Text and Speech Analytics	PEC	4	3	0	0	3
6.	24AM905	Image and Video Analytics	PEC	4	3	0	0	3
7.	24AM908	Cognitive Science and Analytics	PEC	4	3	0	0	3
8.	24CS928	Capstone Project	EEC	12	0	0	12	6

HONOURS - ARTIFICIAL INTELLIGENCE

Sl. No.	COURSE CODE	COURSE TITLE	CATEGORY	CONTACT PERIODS	L	T	P	C
1.	24CS929	Soft Computing	PEC	4	2	0	2	3
2.	24AM933	Knowledge Engineering	PEC	4	3	0	0	3
3.		Reinforcement Learning	PEC	4	3	0	0	3
4.	24AM932	Recommender Systems	PEC	4	3	0	0	3
5.		Natural Language Processing	PEC	4	3	0	0	3
6.	24AM910	Applied AI and ML	PEC	4	3	0	0	3
7.	24CS926	Generative AI Fundamentals	PEC	4	3	0	0	3
8.	24CS928	Capstone Project*	EEC	12	0	0	12	6

R.M.D. ENGINEERING COLLEGE

(An Autonomous Institution)

B.E. - COMPUTER SCIENCE AND ENGINEERING

REGULATIONS – 2024

CHOICE BASED CREDIT SYSTEM

SYLLABUS

SEMESTER – III

24MA301	DISCRETE MATHEMATICS (Common to B.E. CSE, B.Tech. IT and AIML)	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable 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

COURSE 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 BOOKS:

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.

Course Code	UNIVERSAL HUMAN VALUES 2: UNDERSTANDING HARMONY (Common to all Branches)	L	T	P	C
24GE301		2	1	0	3

OBJECTIVES:

Students completing this course are expected 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

- Self-Exploration–what is it? - Its 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 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.

Include practice sessions to discuss natural acceptance in human being as the innate acceptance for 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 being 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, meaning of Prosperity in detail
- Programs to ensure Sanyam and Health.

Include 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 disease

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

Understanding values in human-human relationship; meaning of Justice (nine universal values in relationships) and program for its fulfilment 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 relationship
- Understanding the harmony in the society (society being an extension of 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.

Include practice sessions to reflect on relationships in family, hostel and institutes extended family, real life examples, teacher-student relationship, goal of education etc. Gratitude as a universal value in relationships. Discuss with 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 fulfilment among the four orders of nature- recyclability and self-regulation in nature
- Understanding Existence as Co-existence of mutually interacting units in all-pervasive space
- Holistic perception of harmony at all levels of existence.
- Include practice sessions to discuss human being as 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 the 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 above production systems.
- Case studies of typical holistic technologies, management models and production systems.
- Strategy for transition from the present state to Universal Human Order: a. At the level of individual: as socially and ecologically responsible engineers, technologists and managers
  b. At the level of society: as mutually enriching institutions and organizations
- Sum up.

Include practice exercises and case studies will be taken up in practice (tutorial) sessions eg. To discuss the conduct as an engineer or scientist etc.

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

CO1	be aware of themselves, and their surroundings (family, society, nature).
CO2	be more responsible in life, and in handling problems with sustainable solutions, while keeping human relationships and human nature in mind.
CO3	have better critical ability
CO4	become sensitive to their commitment towards what they have understood (human values, human relationships, and human society).
CO5	be able to apply what they have learnt to their own self in different day-to-day settings in real life, at least a beginning would be made in this direction.

TEXT BOOKS:

1. Human Values and Professional Ethics by R R Gaur, R Sangal, G P Bagaria, Excel Books, NewDelhi, 2010.

REFERENCES:

1. A Nagaraj, “Jeevan Vidya: Ek Parichaya”, Jeevan Vidya Prakashan, Amarkantak,1999.
2. E. F Schumacher, "Small is Beautiful", Vintage classics, London, 1993.
3. A.N. Tripathi, “Human Values”, New Age Intl. Publishers, New Delhi, Third Edition 2020.
4. Maulana Abdul Kalam Azad, "India Wins Freedom", Oriental blackswan private limited, Hyderabad, 2020.
5. Mahatma Gandhi, “Hind Swaraj or Indian Home Rule”, Maheswari Publications, Delhi, 2020.
6. Romain Rolland, "The life of Vivekananda and the universal gospel", Publication house of Ramakrishna Math, Kolkata, Thirty second edition 2018.
7. Romain Rolland, "Mahatma Gandhi: The man who become one with the universal being “, Srishti Publishers & Distributors, New Delhi, Sixth Edition 2013.

24CS301	COMPUTER ORGANIZATION AND ARCHITECTURE (Common to CSE, IT,CSBS 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 interfacings
- 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.

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.

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.

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: Explain the basic principles and operations of digital computers.
CO2: Analyse the performance of computers by identifying factors that contribute to performance.
CO3: Compare various I/O methods and understand memory management principles.
CO4: Explain data flow in arithmetic algorithms.
CO5: Demonstrating the concept of parallelism in hardware and software.
CO6: Design hardware to solve computationally intensive problems.

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:

1. Cay S. Horstmann, "Core Java Volume I--Fundamentals," 12th Edition, 2019.
2. Joshua Bloch, "Effective Java," 3rd Edition, 2018.
3. Raoul-Gabriel Urma, "Java 8 in Action: Lambdas, Streams, and Functional-Style Programming," 1st Edition, 2014.
4. Herbert Schildt, "Java: The Complete Reference," 11th Edition, 2018.
5. Alan Mycroft and Martin Odersky, "Programming in Scala," 4th Edition, 2020.

REFERENCES:

1. Bruce Eckel, "Thinking in Java," 4th Edition, 2006.
2. Herbert Schildt, "Java: A Beginner's Guide," 8th Edition, 2019.
3. Richard Warburton, "Java 8 Lambdas: Pragmatic Functional Programming," 1st Edition, 2014.

LIST OF EQUIPMENTS:

JDK/Eclipse

24CS303	DATABASE MANAGEMENT SYSTEMS (Lab Integrated) (Common to all Branches)	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.
- To understand SQL and effective relational database design concepts.
- To learn relational algebra, calculus and normalization.
- To know the fundamental concepts of transaction processing, concurrency control techniques, recovery procedure and data storage techniques.
- To 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 – eMall
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

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 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:

After completing the course, students will have the ability 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:

1. Elmasri R. and S. Navathe, “Fundamentals of Database Systems”, Pearson Education,

7th Edition, 2016.

2. Abraham Silberschatz, Henry F.Korth, “Database System Concepts”, Tata McGraw Hill , 7th Edition, 2021.

REFERENCES:

1. 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.
2. 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, 6th Edition, 2015.
7. Database Management System Part – 1
   https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_012758066672820224 56_shared/overview
8. Database Management System Part – 2
   https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_012767300562919424 1_shared/overview
9. Online Resources:
10. https://infyspringboard.onwingspan.com/web/en/page/home

24CS304	OPERATING SYSTEMS (Common to CSE, IT and AIML)	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:

1. Write a program to implement the following actions using pthreads
   a. 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
   b. 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.
2. 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, Dinning Philosopher Problem. Deadlock: System model - Deadlock characterization, Methods for handling deadlocks - Deadlock prevention - Deadlock avoidance - Deadlock detection - Recovery from deadlock.

List of Exercise/Experiments:

1. 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.
   a. When a process A1 is accessing the database another process of the same category is permitted.
   b. When a process B1 is accessing the database neither process A1 nor another 74 process B2 is permitted.
   c. 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.

2. 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:

1. Analysis and Simulation of Memory Allocation and Management Techniques
   i. First Fit ii. Best Fit iii. Worst Fit
2. Implementation of Page Replacement Techniques
   i. FIFO ii. LRU iii. 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

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.

SEMESTER – IV

24MA401	PROBABILITY AND STATISTICS (Common to B.E. CSE, B.Tech. IT and AIML)	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	15

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	15

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	15

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	15

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	15

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 BOOK:

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:

3. 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 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:

1. Write a program to implement sum of subset problem.
2. 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	L	T	P	C
		3	0	3	4.5

COURSE OBJECTIVES:

The Course will enable the learners:

- To understand web semantics and related tools and framework
- Able 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 Compatability, Utility Type - Unit Testing,TSLint

List of Exercise/Experiments

1. Create a TS Object for Bank Account (w 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
   - User Login Page
   - Account Summary
   - Funds Transfer (within bank and outside bank)
   - Recurring and Fixed deposits
   - Letter of Credit
   - Salary or 3rd Party Payment
2. Unit 2 Scope – Project Setup, Web App Layout Completion using BootStrap or Tailwind, Login PageImplementation, 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
   - Unit 3 Scope – Forms and Validation (React Form Validation), Integration of Back End Apisvia Axios,API Security Implementaiton, Routes and Navigation with Priviate Routes, Usage
     of useEffect, UseContext hooks

UNIT IV	REACT PRO TOOLKIT: ERROR MANAGEMENT, ABSTRACTIONS AND 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.

List of Exercise/Experiments

1. Extend the Project developed in previous section with newly learnt concepts
    - Unit 4 Scope – Completion of Remaining Modules, Error Handling, HOC and AUX implementation, Lazy loaded components for improved performance
2. 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 LibraryContext 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.

List of Exercise/Experiments

1. Extend the Project developed in previous section with newly learnt concepts
    - Unit 5 Scope – Unit Testing using JEST, Redux implementation for state management.

Business Use Case Implementations

1. Student Management System
2. Retail Bank System
3. eCommerce System
4. Student LMS Management System

TOTAL: 45+45=90 PERIODS

COURSE OUTCOMES:

After completing the course, students will have the ability to

CO1. Understand and apply modern web technologies including HTML5, CSS3, JavaScript, and advancedTypeScript 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 RESTAPI integration using Axios for dynamic content handling.

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

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

CO6. Design and deliver scalable and real-world enterprise web applications with complete user interfaceflow, 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 17with 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 byRobin Wieruch
3. Learning React: Functional Web Development with React and Redux by Alex Banksand 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

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

PROFESSIONAL ELECTIVES/HONOUR VERTICALS

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+30=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 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**

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+30=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

22CS903	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. Case study - Web3j.

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 – Case study - Corda.

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.

22CS904	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.

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**

ntroduction- 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	9

WLAN and its inherent insecurities Bypassing WLAN Authentication uncovering hidden SSIDs MAC Filters Bypassing open and shard authentication - Attacking the client caffe latte attack Deauthenticating the client cracking WEP with the hirte attack AP-less WPA cracking - Advanced WLAN Attacks Wireless eavesdropping using MITM session hijacking over wireless - WLAN Penetration Test Methodology

UNIT V	PENETRATION TESTING	9

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, 2nd 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.

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 + 30 = 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)
6. AWS Skill Builder
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 Tsafrir, “Hardware and Software Support for Virtualization”, Morgan & Claypool Publishers, 2017.
2. Matthew Portnoy, “Virtualization Essentials”, Third Edition, Sybex - John Wiley & Sons, 2023.
3. Dave Shackleford, “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**	**P**	**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.

TEXT BOOKS:

1. Martin Kleppman, “Data Engineering: Building Reliable Scalable Data Systems”, O'Reilly Media, 2017.
2. Wes McKinney, “Python for Data Analysis”, 2nd Edition, O'Reilly Media, 2017.
3. Martin Kleppman, “Designing Data-Intensive Applications”, O'Reilly Media, 2017.

REFERENCES:

1. AWS Documentation (amazon.com)
2. AWS Skill Builder
3. AWS Academy Data Engineering Course - https://www.awsacademy.com/vforcesite/LMS_Login

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 Kauffman, 2016.
2. Thomas D Nadeau, Ken Gray, "SDN: Software Defined Networks", O'Reilly Media, 2013.
3. Fei Hu, "Network Innovation through OpenFlow and SDN: Principles and Design", 1st Edition, CRC Press, 2014.
4. 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

C06: 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:

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

REFERENCES:

3. Rajkumar Buyya, James Broberg, Andrzej, "Cloud Computing: Principles and Paradigms", Wiley India Publications 2011.
4. 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

C06: 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.

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+30=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.

24CS924	MERN FULL STACK DEVELOPMENT (Lab Integrated)	L	T	P	C
		2	0	2	3

OBJECTIVES:

The Course will enable learners to:

- 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:

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 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+30=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 (Unit 2 to 5)
2. David Herron , Node.js Web Development - Fourth Edition, Packt Publishing, 2018. (Unit 1)

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

HONORS COURSES

ARITIFICIAL INTELLIGENCE

	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 McCulloh-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+30 = 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:

Introduction To Soft Computing - https://onlinecourses.nptel.ac.in/noc23_cs40/preview

DATA SCIENCE

24AM901	FOUNDATIONS OF DATA SCIENCE	L	T	P	C
		2	0	2	3

OBJECTIVES:

The Course will enable learners to:

- To learn the fundamentals of Data Science.
- To describe and understand data for analysis.
- To describe and analyze relationships between variables.
- To apply various data pre-processing strategies.
- To 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 ofData.

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	DESCRIBING DATA	6+6

Types of Data - Types of Variables -Describing Data with Tables and Graphs –Describing Data withAverages - 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	DESCRIBING RELATIONSHIPS	6+6

Correlation –Scatter plots –correlation coefficient for quantitative data –computational formula forcorrelation coefficient –

Regression –regression line –least squares regression line –

Standard error of estimate – interpretation of r2 –multiple regression equations –regression towards themean.

List of Exercise/Experiments:

1. Use a scatter plot to see how two variables are related. Calculate the correlation coefficient tomeasure 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 linefits the data.
3. Use a Multiple Linear Regression model to predict a target variable from multiple inputs, andplot the predicted vs. actual values to evaluate model performance.
4. Create a dashboard or summary using charts and key numbers.

UNIT IV	DATA PREPROCESSING	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, andcreating 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	**DATA WRANGLING**	**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 and3D surface plots.
2. Build a DataFrame, perform aggregation and transformation, filter and sort data, removeduplicates, group, and join tables.
3. Simulate SQL-style operations like filtering, ordering, deduplication, and grouping in aDataFrame.

TOTAL:30+30 = 60 PERIODS

OUTCOMES:

At the end of this 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. 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)

REFERENCES:

1. David Cielen, Arno D. B. Meysman, and Mohamed Ali, “Introducing Data Science”, Manning Publications, 2016. (Unit 1)
2. Hui Lin, Ming Li, Practitioner’s Guide to Data Science, 1st Edition, Chapman and Hall/CRC, 2023.
3. Jiawei Han, Micheline Kamber, Jian Pei, "Data Mining: Concepts and Techniques", 3rd Edition, Morgan Kaufmann, 2012.
4. 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.