Bharati Vidyapeeth

(Deemed to be University)

Pune, India

College of Engineering, Pune

B. Tech (Information Technology)

(CBCS-2023 COURSE)

Program Curriculum

As Per NEP 2020 Guidelines

VISION OF THE UNIVERSITY

Social Transformation through Dynamic Education

MISSION OF THE UNIVERSITY

- To make available quality education in different areas of knowledge to the students as per their choice and inclination
- To offer education to the students in a conducive ambience created by enriched infrastructure and academic facilities in its campuses.
- To bring education within the reach of rural, tribal and girl students by providing them substantive fee concessions and subsidized hostel and mess facilities
- To make available quality education to the students of rural, tribal and other deprived sections of the population

VISION OF THE INSTITUTE

To be World Class Institute for Social Transformation Through Dynamic Education.

MISSION OF THE INSTITUTE

- To provide quality technical education with advanced equipment, qualified faculty members, infrastructure to meet needs of profession and society.
- To provide an environment conducive to innovation, creativity, research and entrepreneurial leadership.
- To practice and promote professional ethics, transparency and accountability for social community, economic and environmental conditions.

VISION OF THE DEPARTMENT

To be a leading Programme, transforming students into skilled IT professionals.

MISSION OF THE DEPARTMENT

• Amplify the student's technical skills by conducting continuing education programs, organizing and participating in various technical events.

• Provide comprehensive support in synchronization with industry to achieve professional and technological excellence.

• Provide an environment for effective social and ethical skills.

PROGRAM EDUCATIONAL OBJECTIVES

PEO1: Cultivate IT graduates for industry, pertaining to Information Technology solutions.

PEO2: Practice technical competency and teamwork abilities.

PEO3: Exhibit social responsibilities by following ethical practices in graduate's professional pursuits.

PROGRAM OUTCOMES

1. Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

2. Problem analysis: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

3. Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified

needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations

4. Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

5. Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

6. The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

7. Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

8. Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

9. Individual and teamwork: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

10. Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

11. Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one’s own work,

as a member and leader in a team, to manage projects and in multidisciplinary environments.

12. Life-long learning: Recognize the need for and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change.

PROGRAM SPECIFIC OUTCOMES

At the end of the program, Graduates will be able to

PSO1: Use knowledge of core and allied courses for developing a computer-based system to deliver a quality product for real-world problems of society.

PSO2: Apply modern IT tools and techniques for perusing student's professional career by practicing effective communication with team members.

PSO3: Develop time-bound, cost-effective, and sustainable solutions by following professional ethics.

CORELATION BETWEEN GRADUATE ATTRIBUTES AND PROGRAMME OUTCOMES

Graduate Attributes/ Programme Outcomes	a	b	c	d	e	f	g	h	i	j	k	l
Engineering Knowledge	✓											
Problem Analysis		✓										
Design/Development of Solutions			✓									
Conduct Investigations of Complex Problems				✓								
Modern Tool Usage					✓							
The Engineer and Society						✓						
Environment and Sustainability							✓					
Ethics								✓				
Individual and Teamwork									✓			
Communication										✓		
Project Management and Finance											✓	
Life-Long Learning												✓

A. DEFINITION OF CREDITS:

1 Hour Lecture (L) per week	1 Credit
1 Hour Tutorial (T) per week	1 Credit
2 Hours Practical (Lab) per week	1 Credit

B. Course Code and Definition

Course Code	Definitions
L	Lecture
T	Tutorial
P	Practical
TW	Term Work
O	Oral
SEE	Semester End Examination
MJ	Major (Core) Courses
MI	Minor Courses
GE	General Elective Courses
OE	Open Elective Courses
SE	Skill Enhancement Courses
AE	Ability Enhancement Courses
VE	Vocational Enhancement Courses
VS	Vocational Skill Courses
VA	Value Added Courses
CC	Co-Curricular Courses

ID	Inter-disciplinary Courses
MD	Multidisciplinary Courses
RP	Research *I* Project Courses
PC	Practical Courses
BS	Basic Science
ES	Engineering Science
AC	Audit Course
EC	Extracurricular Activities
BM	Basic Mathematics
BP	Basic Physics
BC	Basic Chemistry
UH	Universal Human Values

BHARATI VIDYAPEETH (DEEMED TO BE UNIVERSITY)
COLLEGE OF ENGINEERING, PUNE
B. Tech. (Information Technology): Semester –I (CBCS-2023 COURSE)

Sr. No	Category	Subject Code	Subject	Teaching Scheme			Examination Scheme-Marks						Credits			
				L	P	T	ESE	Internal Assessment	TW	PR	OR	Total	Th	Pr/Or	Tut	Total
1.	BM	BM1113101	Engineering Mathematics- I	3	-	1	60	40	-	-	-	100	3	-	1	4
2.	BC	BC1113102	Engineering Chemistry	3	2	-	60	40	50	-	-	150	3	1	-	4
3.	ES	ES1109103	Digital Electronics	4	2	-	60	40	50	-	-	150	4	1	-	5
4.	MJ	MJ1110104	Structured Programming	4	-	-	60	40	-	-	-	100	4	-	-	4
5.	MJ	MJ1110105	Web Technologies	4	2	-	60	40	25	-		125	4	1	-	5
6.	AE	AE1113106	Communication Skills	-	2	-	-	-	50	-	-	50	-	1	-	1
7.	SE	SE1110107	Information Technology Laboratory-I	-	2	-	-	-	25	-	25	50	-	1	-	1
8.	SE	SE1111108	Computer Workshop Technology	-	2	-	-	-	25	-	-	25	-	1	-	1
			Total	**18**	**12**	**1**	**300**	**200**	**225**	**0**	**25**	**750**	**18**	**6**	**1**	**25**

BHARATI VIDYAPEETH (DEEMED TO BE UNIVERSITY)
COLLEGE OF ENGINEERING, PUNE
B. Tech. (Information Technology): Semester –II (CBCS-2023 COURSE)

Sr. No	Category	Subject Code	Subject	Teaching Scheme			Examination Scheme-Marks						Credits			
				L	P	T	ESE	Internal Assessment	TW	PR	OR	Total	Th	Pr/Or	Tut	Total
1.	BM	BM1113201	Engineering Mathematics- II	3	-	1	60	40	-	-	-	100	3	-	1	4
2.	BP	BP1113202	Engineering Physics	3	2	-	60	40	50	-	-	150	3	1	-	4
3.	MJ	MJ1110203	Content Management System	4	2	-	60	40	50	-	-	150	4	1	-	5
4.	MJ	MJ1110204	Computer Communication & Networks	4	2	-	60	40	25	-	-	125	4	1	-	5
5.	MJ	MJ1110205	Object Oriented Programming	4	-	-	60	40	-	-	-	100	4	-	-	4
6.	UH	UH1113206	Universal Human Values	-	2	-	-	-	50	-	-	50	-	1	-	1
7.	SE	SE1110207	Information Technology Laboratory-II	-	2	-	-	-	25	-	25	50	-	1	-	1
8.	SE	SE1111208	Computer Aided Drawing & Design	-	2	-	-	-	25	-	-	25	-	1	-	1
			Total	**18**	**12**	1	**300**	**200**	**225**	**0**	**25**	**750**	**18**	**6**	**1**	**25**

BHARATI VIDYAPEETH (DEEMED TO BE UNIVERSITY)
COLLEGE OF ENGINEERING, PUNE
B. Tech. (Information Technology): Semester –III (CBCS-2023 COURSE)

Sr. No	Category	Subject Code	Subject	Teaching Scheme			Examination Scheme-Marks						Credits			
				L	P	T	ESE	Internal Assessment	TW	PR	OR	Total	Th	Pr/ Or	Tut	Total
1.	MJ	MJ1110301	Discrete Structure & Graph Theory	3		-	60	40	-	-	-	100	3	-	-	3
2.	MJ	MJ1110302	Database Management Systems	3	2	-	60	40	25	25	-	150	3	1	-	4
3.	MJ	MJ1110303	Operating System	3	2	-	60	40	25	-	25	150	3	1	-	4
4.	MJ	MJ1110304	Microprocessors & Microcontrollers	3	-	-	60	40	-	-	-	100	3	-	-	3
5.	MJ	MJ1110305	Data Structures	3	2	-	60	40	25	25	-	150	3	1	-	4
6	SE	SE1110306	Information Technology Laboratory-III	-	2	1	-	-	25	25	-	50	-	1	1	2
			Total	**15**	**08**	**1**	**300**	**200**	**100**	**75**	**25**	**700**	**15**	**4**	**1**	**20**
7	AE	AE1110307	MOOC-I*	-	-	-	-	-	-	-	-	-	-	-	-	2
8	VA	VA1110308	Value Added Course-I* A) Internet Security B) Data Analysis Tools	2	-	-	-	100	-	-	-	100	-	-	-	2

*** Indicate this is mandatory course.**

BHARATI VIDYAPEETH (DEEMED TO BE UNIVERSITY)
COLLEGE OF ENGINEERING, PUNE
B.Tech. (Information Technology): Semester –IV (CBCS-2023 COURSE)

Sr. No	Category	Subject Code	Subject	Teaching Scheme			Examination Scheme-Marks						Credits			
				L	P	T	ESE	Internal Assessment	TW	PR	OR	Total	Th	Pr/Or	Tut	Total
1.	MJ	MJ1110401	Formal Languages & Automata Theory	3		-	60	40	-	-	-	100	3	-	-	3
2	MJ	MJ1110402	Advanced Database Systems	3	2	-	60	40	25	25	-	150	3	1	-	4
3.	MJ	MJ1110403	Software Engineering	3	2	-	60	40	25	-	25	150	3	1	-	4
4.	MJ	MJ1110404	Computer Organization & Architecture	3	-	-	60	40		-	-	100	3	-	-	3
5.	MJ	MJ1110405	Applied Algorithm	3	2	-	60	40	25	25	-	150	3	1	-	4
6.	SE	SE1110406	Information Technology Laboratory-IV	-	2	1	-	-	25	25	-	50	-	1	1	2
			Total	**15**	**08**	**1**	**300**	**200**	**100**	**75**	**25**	**700**	**15**	**4**	**1**	**20**
7	AC	AC1113407	Indian Knowledge System*	2	-			100	-	-	-	100	-	-	-	2
8	EC	EC1110408	Social Activity*	-	-	-	-	-	-	-	-	-	-	-	-	2

*** Indicate this is mandatory course.**

BHARATI VIDYAPEETH (DEEMED TO BE UNIVERSITY)
COLLEGE OF ENGINEERING, PUNE

B. Tech. (Information Technology): Semester –III/IV/V/VI (CBCS-2023 COURSE)

MINOR DEGREE: CLOUD TECHNOLOGIES

Sr. No	Course Code	Subject	Teaching Scheme			Examination Scheme-Marks						Credits			
			L	P	T	ESE	Internal Assessment	TW	PR	OR	Total	Th	Pr/Or	Tut	Total
1.	MI1110301	SEM-III: Distributed Computing	3	2	-	60	40	25	-	25	150	3	1	-	4
2	MI1110401	SEM-IV: Cloud Systems and Infrastructures	3	2	-	60	40	25	-	25	150	3	1	-	4
3.	MI1110501	SEM-V: Cloud Virtualization	3	2	-	60	40	25	-	25	150	3	1	-	4
4.	MI1110601	SEM-VI: Cloud Containers and Orchestrations	3	2	-	60	40	25	-	25	150	3	1	-	4
5.	RP1110602	Sem-VI Project		8	-			50		50	100		4	-	4
		Total	**12**	**16**	**-**	**240**	**160**	**150**	**-**	**150**	**700**	**12**	**8**	**-**	**20**

B. Tech
Information Technology
Semester III

Discrete Structure & Graph Theory

TEACHING SCHEME		EXAMINATION SCHEME		CREDIT SCHEME	
Theory;	3 Hours/Week	End Semester Examination:	60 Marks	Theory	3
		Internal Assessment:	40 Marks		
Total:	3 Hours/Week	Total:	100 Marks	Total:	3

Course Objectives:

1. To apply and relate knowledge of mathematics in computer science.
2. To learn proof theory with propositional calculus and induction.
3. To map, represent and solve network problem with trees and graphs.

Prerequisite:

Basic mathematics and programming fundamentals.

Course Outcomes: On completion of the course, students will have the ability to

1. Formulate real world problems into statement forms using sets and relations which can be solved or proved mathematically using set theory and logic.
2. Design mathematical model from theoretical statements.
3. Apply counting techniques to real world problems.
4. Apply knowledge of graphs to solve network problems.
5. Design searching algorithm efficiently by applying tree and tree traversal logic.
6. Apply algebraic structure and coding theory in computer science.

Unit I **(6 Hours)**

Propositional Logic and Proof Theory: Sets, Set operations, Finite and Infinite sets, Venn diagram, Principle of inclusion and exclusion, Multisets. Propositions, Conditional Propositions, Logical Connectivity, Propositional calculus, Universal and ExistentialQuantifiers, Normal forms.

Unit II: **(6 Hours)**

Induction and Relations: Types of mathematical inductions, Problems on mathematical induction. Basics of combinatorics, Pigeonhole Principle, Properties of Binary Relations, Closure of relations, Warshall's algorithm, Equivalence.

Unit-III:

Probability: Sample Space, Events of an Experiment, Properties of Probability, permutations, combinations, generalized permutations and combinations (with/without repetitions), Probability theory, Discrete Random variables, Bernoulli's Distribution, Binomial coefficients, and identities.

Unit IV **(6 Hours)**

Graph theory: Basic terminology, multi graphs and weighted graphs, paths and circuits,shortest path in weighted graph, Dijkstra's algorithm, Hamiltonian and Euler paths and circuits, factors of a graph, planer graph and Travelling salesman problem.

Unit V **(6 Hours)**

Trees: Trees, rooted trees, path length in rooted trees, prefix codes, binary search trees,tree traversal, spanning trees and cut set, minimal spanning trees, Kruskal's and Prim's algorithms for minimal Spanning tree. The Max flow- Min Cut Theorem (Transport network). Case Study- Game Tree, Mini-Max Tree.

Unit VI **(6 Hours)**

Algebraic Structures: The structure of algebra, Algebraic Systems, Semi Groups, Monoids, Groups, Homomorphism and Normal Subgroups, coding theory, Polynomial Rings and polynomial Codes.

Textbooks:

1. Kenneth H. Rosen, Discrete Mathematics and its Applications, 7th Edition,McGraw Hill.
2. C. L. Liu, D. P. Mohapatra, Elements of Discrete Mathematics: A Computer Oriented Approach, 4th Edition, McGraw Hill.

Reference Books

1. Seymour Lipzschutz, M. Lipson, Discrete Mathematics, 3rd Edition, McGraw Hill.
2. P. Tremblay, R. Manohar, Discrete Mathematical Structures with Applications to Computer Science, McGraw Hill.

Project Based Learning Assignments

Note:- *Students in a group of 3 to 4 shall complete any one project from the following list

1. Study the writings of Lewis Carroll on symbolic logic. Describe in detail some of the models he used to represent logical arguments and the rules of inference he used in these arguments.
2. Describe a variety of different applications of the Fibonacci numbers to the biological and the physical sciences.
3. Explain how graph theory can help uncover networks of criminals or terrorists by studying relevant social and communication networks.
4. Explain what community structure is in a graph representing a network, such as a social network, a computer network, an information network, or a biological network. Define what a community in such a graph is, and explain what communities represent in graphs representing the types of networks listed.

5. Describe how Euler paths can be used to help determine DNA sequences.
6. Describe some of the strategies and algorithms used to solve the traveling salesperson problem.
7. Explain how graph multi-colorings can be used in a variety of different models.
8. Define a heap and explain how trees can be turned into heaps. Why are heaps useful in sorting?
9. Describe the techniques used by chess-playing programs such as Deep Blue or stockfish.
10. Discuss the algorithms used in IP multicasting to avoid loops between routers.
11. Compare and contrast some of the most important sorting algorithms in terms of their complexity and when they are used.
12. Describe an algorithm for finding the minimum spanning tree of a graph such that the maximum degree of any vertex in the spanning tree does not exceed a fixed constant k
13. Describe the origins of mathematical induction. Who were the first people to use it and to which problems did they apply it?

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit – VI

Database Management Systems

TEACHING SCHEME		EXAMINATION SCHEME		CREDIT SCHEME	
Lecture:	03 Hrs/Week	End Semester Examination	60 Marks	Lecture	03
Practical:	02 Hrs/Week	Internal Assessment	40 Marks		
		Term Work	25 Marks	Practical	01
		Practical	25 Marks		
Total	05 Hrs/Week	Total	150 Marks	Total	04

Course Objective:
Introduction to Database Management Systems, with an emphasis on efficiently and effective organization, maintenance and retrieval of information in Database Management Systems.

Prerequisite:
Students should have knowledge of
1) Basic understanding of data and data structure
2) Basic understanding of programming language

Course Outcomes: On completion of the course, students will have the ability to:

1. Model an application's data requirements using conceptual modelling tools
2. Implement concepts of relational database using SQL and PL/SQL
3. Demonstrate concepts of relational database design
4. Interpret the query processing and optimization activities in database
5. Interpret the transaction activities in database
6. Recognize the emerging database applications and security concerns

Unit I **(6 Hours)**

Introduction: Introduction to Database system architecture, Data Abstraction, Data Independence
Data models: Extended Entity-relationship model, network model, relational and object oriented data models, data manipulation operations.
Relational algebra: Fundamental and extended relational algebra operation

Unit II **(6 Hours)**

Integrity constraints : What are constraints, types of constrains, Relational database design: Domain and data dependency, Armstrong's axioms, Functional Dependencies, Normal forms (1NF, 2NF, 3NF, BCNF, 4NF), Dependency preservation, Lossless design

Unit III **(6 Hours)**

Introduction to SQL: Data definition language, Data Manipulation Language, Joined relations, Views

Introduction to PL/SQL: Functions, Procedures, Triggers, Cursors.

Unit IV **(6 Hours)**

Indexing strategies: Indices, B trees, B+ trees, Hashing , Query processing and optimization: Evaluation of relational algebra expressions, Query equivalence, Join strategies, Query optimization algorithms

Unit V **(6 Hours)**

Transaction processing: Concurrency control, ACID property, Serializability of scheduling, Locking and timestamp based schedulers, Multi-version and optimistic Concurrency Control schemes, Database recovery, SQL commands for Transactions

Unit VI **(6 Hours)**

Database Security: Authentication, Authorization and access control, DAC, MAC and RBAC models, Intrusion detection, SQL injection.

Data Intensive Computing: Introduction to big data, unstructured data processing using Hadoop , NoSQL and Cloud Databases: Architecture, Components, Characteristics

Textbooks

1. Silberschatz, Korth, "Data base System Concepts", 7th ed., McGraw hill.
2. Raghu Ramakrishnan and Johannes Gehrke, "Database Management Systems" (3/e), McGraw Hill.
3. Ivan Bayross, "SQL, PL/SQL the Programming Language of Oracle", BPB Publication.

Reference Books

1. Ramez Elmasri and Shamkant B. Navathe, Fundamentals of Database Systems (5/e), Pearson Education
2. C. J. Date, Kannan, "An Introduction to Database Systems", 8e, Addison-Wesley

List of Laboratory Exercises

1. Conceptual Designing using ER Diagrams (Identifying entities, attributes, keys and relationships between entities, cardinalities, generalization, specialization etc.)
2. Converting ER Model to Relational Model (Represent entities and relationships in Tabular form, Represent attributes as columns, identifying keys) Apply Normalization to remove the redundancies and anomalies in the above relational tables, Normalize up to Third Normal Form
3. Study and implementation of SQL : DDL Creation of above Tables using SQL- Data types in SQL, Creating Tables (along with Primary and Foreign keys), Altering Tables and Dropping Tables
4. Study and implementation of SQL : DML, Querying with set operations and wildcards
5. Study and implementation of aggregate functions, joins, nested subqueries in SQL for querying above tables

6. Study and implementation of views in SQL
7. Study and implementation of PL/SQL – Control statements
8. Study and implementation of PL/SQL Functions and stored procedure
9. Study and implementation of Triggers
10. Study and implementation of Cursors

Project Based Learning - Provisional List of Projects

Implement any database management systems with CRUD functionality and design normalised at least up to 3 NF. Apply appropriate integrity constraints and access privileges to database and relations. Show indexing. (Any implemented system must have at least 7 inter-related normalised relations.

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit – VI

Operating System

TEACHING SCHEME		EXAMINATION SCHEME		CREDIT SCHEME	
Theory	3 Hrs/Week	End Semester Examination	60 Marks	Theory	03
Practical	2 Hrs/Week	Internal Assessment	40 Marks	Practical	01
		Term Work	25 Marks		
		Oral	25 Marks		
Total	5 Hrs/Week	Total	150 Marks	Total Credits	04

Course Objectives:

1. To understand the mechanism of operating system such as process management, file management, Memory management and storage structures used in the operating system.
2. To be familiar with the basics of Linux operating system.

Course Prerequisites:

Students should have knowledge of

- Basics of computers
- General knowledge of operating system (used in PCs, Phones)

Course Outcome:

Students will be able to:

1) Install operating system and configure it.

2) Know different operating system services

3) Understand process management system calls.

4) Implement the various scheduling algorithms

5) Understand paging algorithms and its implementation

6) Understand file management and access methods

UNIT-I Operating system Overview (6 Hours)

Operating System – Concept, Evolution of Operating System, Components of operating system, Functions of OS, Views of OS: User view, System View

Different Types of Operating Systems-Batch operating system, Multi Programmed, Time Shared OS, Multiprocessor Systems, Distributed Systems, Real time systems. Mobile OS (Android,iOS).System Calls- Concept and its types,, OS Components: – Process Management, Main Memory

Management, File Management, I/O System management, Secondary storage management

UNIT-II Process Management (6 Hours)

Process Concept, Process states, Process control block, Process Scheduling- Scheduling Queues, Schedulers, Context switch, Inter-process communication (IPC): shared memory system & message passing system.

Threads – Benefits, users and kernel threads, Multithreading Models – Many to One, One to One, Many to Many.

Execute process commands- like ps, wait, sleep, exit, kill

UNIT-III CPU Scheduling and Algorithms (6 Hours)

Scheduling objectives, Scheduling Strategies: Pre-emptive, Non-Pre-emptive Scheduling, Scheduling criteria, CPU and I/O burst cycles, Uni-processor Scheduling: Types of scheduling: Preemptive, Non preemptive, Scheduling algorithms: FCFS, SJF, RR, Priority, Thread Scheduling, Real Time Scheduling,
Deadlock: Principles of deadlock, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection

UNIT-IV Memory Management (6 Hours)

Memory Management requirements, Memory partitioning: Fixed and Variable Partitioning, Fragmentation
Memory Allocation: Allocation Strategies (First Fit, Best Fit, and Worst Fit), , Swapping,
Paging and Segmentation: Page Replacement Policies (FIFO, LRU, Optimal, Other Strategies)
Virtual Memory: Concepts, management of VM, , Thrashing.

UNIT-V File Management (6 Hours)

File : Concepts, Attributes, Operations, types and File System Structure.
Access Methods : Sequential, Direct, Swapping
File Allocation Methods: Contiguous, Linked, Indexed.
Directory structure : Single level, two levels, tree-structured directory
Mass Storage system : Overview of Mass Storage Structure, Disk Structure, Disk Scheduling and Management, swap space management

UNIT-VI Multi-Processor Based Virtualization Concepts (6 Hours)

Virtual machines; supporting multiple operating systems simultaneously on a single hardware platform; running one operating system on top of another, True or pure virtualization. Paravirtualization; optimizing performance of virtualization system; hypervisor call interface.

List of Laboratory Exercises

1. Introduction to Operating system.
2. Basic Linux commands: Working with Directories
3. Basic Linux commands: Working with files
4. Write Shell Script for finding the global complete path for any file
5. Write Shell Script to broadcast a message to a specified user or a group of users logged on any terminal
6. Write Shell Script to copy the file system from two directories to a new directory in such a way that only the latest file is copied in case there are common files in both the directories
7. Implementation of FCFS (First Come First Serve) CPU Scheduling
8. Implementation of SJF (Shortest Job First) CPU Scheduling
9. Implementation of FIFO Replacement Algorithm.
10. Implementation of Optimal Page Replacement Algorithm

Project Based Learning

1) Design of First-fit, worst-fit and best-fit for given allocation memory requirements
2) Simulation of the behavior of the multiprogramming operating system and use CPU scheduler, and CPU Execution
3) Design the FCFS, SSTF, and SCAN disk-scheduling algorithms to simulate a simple disk drive, which has a specified number of logical blocks numbered from 0 onwards.
4) A Java simulator program to analyze the dependency of Page Faults on the Page Frames for incoming page requests.
5) CPU Scheduling Algorithm to calculate Throughput, Utilization, Turn Around time, Waiting Time. Gantt chart displayed for all n processes.
6) To simulate Round Robin algorithm
7) To implement FIFO page replacement policy.
8) To implement Optimal page replacement policy.
9) To implement LRU page replacement policy.
10) To implement Banking algorithm.

Textbooks:

1) Operating System Principles, Abraham Silberchatz, Peter B.Galvin,Greg Gagne,8th Edition, Wiley Student Edition
2) Operating System-Internals and Design Principles, W.Stallings, 6th Edition, Pearson.
3) Principles of Operating System,Naresh Chauhan,Oxford University Press

Reference Books:

1) Modern Operating System, Andrew s Tanenbaum, 3rd Edition, PHI
2) Operating System A concept-based Approach, 2nd Edition, D.M.Dhamdhere, TMH.
3) Principle Of Operating Systems, B.LStuart, Cengage Learning, India Edition

Syllabus for Unit Tests:

Unit Test -1: Unit – I, Unit – II, Unit – III

Unit Test -2: Unit – IV, Unit –V, Unit - VI

Microprocessors & Microcontrollers

TEACHING SCHEME		EXAMINATION SCHEME		CREDIT SCHEME	
Theory	3 Hours/Week	End Semester Examination	60 Marks	Theory	3
		Internal Assessment	40 Marks		
Total	3 Hours/Week	Total	100 Marks	Total Credits	3

Course Objectives:
This course facilitates the learners with the basic knowledge of microprocessors and microcontrollers. Also, the course supports the learners with detailed study of ARM processor andAVR Microcontroller.

Prerequisite:
Digital Electronics, Knowledge of Structured Programming

Course Outcomes: On completion of the course, students will have the ability to
1. Learn basics of 16/32-bit Microprocessors.
2. Cognize the ARM Cortex Processor with its architecture and programming.
3. Discover Intel Pentium and i7 processor with its architecture and pipelining.
4. Comprehend basics of 8/16-bit Microcontrollers.
5. Uncover the details of AVR Microcontroller with its architecture and programming.
6. Understand the basics of Arduino and Raspberry Pi Controllers.

Unit I **(6 Hours)**

Introduction to Microprocessors: Basics of 16-bit and 32-bit processor (Intel 8086 and 80386 processors), Multicore Architecture, Hyperthreading Technology,Instruction Set Architectures (ISA), Multiprocessor Organizations, Inter-Processor Communication (IPC). Brief introduction to assembly language programming

Unit II **(6 Hours)**

Intel Pentium Processor: Features and Internal Architecture, Superscalar Operation, Integer &Floating- Point Pipeline Stages, Branch Prediction Logic, CacheOrganization and MESI Protocol, Comparative study of 8086, 80386, Pentium I, Pentium II and Pentium III, Hyper Threading technology and its use in Pentium 4, Intel i7 processor

Unit III **(6 Hours)**

ARM Cortex: ARM Micro-architecture (ARMv7/v8/v9/v11), ARM architectures:Pipelining, ARM OS, Introduction to ARM Programming.

Unit IV **(6 Hours)**

Introduction to Microcontrollers: Microprocessors vs Microcontrollers, Basics of 8- bit and 16-bit Microcontrollers (Intel 8051 and 8096 microcontrollers), Applicationsof microcontrollers.

Unit V **(6 Hours)**

AVR Microcontroller: Types of AVR Microcontrollers, ATmega16/32 8-bit AVR microcontroller: Features, Pin Description, Internal Architecture, Data and Program Memory, Brief introduction to AVR Programming using C/Java/Assembly language

Unit VI **(6 Hours)**

Introduction to Arduino and Raspberry Pi : Introduction, Difference, Arduino Unoand Raspberry Pi Pico (RP2040), microcontrollers, Programming concepts of ArduinoUno with C/C++/Python and IDE, Programming concepts of Raspberry Pi Pico with C/MicroPython.

Textbooks:

1 Arm Microprocessor Systems Cortex-M Architecture Programming and Interfacing, MuhammadTahir, T&F India.
2 The Definitive Guide to ARM Cortex-M3 and Cortex-M4 Processors, Joseph Yiu.
3 ARM A32 Assembly Language, Bruce Smith.
4 8051 Microcontrollers, Satish Shah, Oxford University Press.
5 Microprocessors and Interfacing, N.Senthil Kumar, M.Saravanan, Oxford University Press
6 Programming and Interfacing Atmel AVR Microcontrollers, Grace, Cengage Learning.
7 Practical AVR Microcontrollers, Alan Trevennor, Technology In Action.
8 Getting Started with Arduino, Massimo Banzi and Michael Shiloh.
9 Getting Started with Raspberry Pi, Matt Richardson and Shawn Wallace.

Reference Books:

1 The Definitive Guide to ARM Cortex-M3 Processors, Stellaris, Texas Instruments.
2 ARM System-on-Chip Architecture, Steve Furber.
3 ARM processor, Santul Bisht, Lambert Publications
4 Modern Assembly Language Programming with the ARM Processor, Larry D Pyeatt.
5 Programming and Customizing AVR Microcontroller, Dhananjay Gadre.
6 Arduino Cookbook 2nd Edition, Michael Margolis.
7 Raspberry Pi The Ultimate Guide, Geoff Adams.
8 Internet of Things with Raspberry Pi and Arduino, Anita Gehlot.

Project Based Learning:

Students in a group of 2 to 3 shall complete and submit any one working project under Arduino/Raspberry Pi/8051 or any other microcontroller in the curriculum. A sample list is as given below:

1. Design and Implementation of Automatic Turn off for Water Pump with Four Different Time Slots using ARM (Processor/Controller)/AVR Controller/8051 Microcontroller
2. Design and Implementation of Gas Leak Detector with Automatic Air Exhaust Using ARM Cortex
3. Design and Implementation of ARM Based Liquid Level Detection & Flow Control
4. Design and Implementation of Motion Based Door Opener (in malls, big shops) using ARM (Processor/Controller)/AVR Controller/8051 Microcontroller
5. Design and Implementation of Voice Controlled Air Purifier based on Arduino and Raspberry Pi
6. Design and Implementation of Face Recognition Door Lock System based on Arduino and Raspberry Pi
7. Design and Implementation of Vehicle Number Plate Recognition based on Arduino and Raspberry Pi

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit – VI

Data Structures

TEACHING SCHEME		EXAMINATION SCHEME		CREDIT SCHEME	
Lecture:	3 Hours/Week	End Semester Examination:	60 Marks	Theory	3
Practical:	2 Hours/Week	Internal Assessment:	40 Marks	Practical	1
		Term Work	25 Marks		
		Practical:	25 Marks		
Total	5 Hours/week	Total	150 Marks	Total	4

Course Objective:
The objective of the course is to familiarize students with various data structures and fundamentals of algorithms.

Prerequisite:
Fundamental knowledge programming and problem-solving steps

Course Outcomes: On completion of the course, students will have the ability to:
1. Understand the fundamentals of data structure and algorithms
2. Execute linear sequential data structures
3. Implement linear linked organization data structures
4. Execute nonlinear data structure-trees
5. Implement nonlinear data structure-graph
6. Know hashing and file organization concepts

Unit I **(6 Hours)**

Introduction to Algorithm and Data Structures: Introduction to data structures, types of data structure, abstract data types (ADT), introduction to algorithms, characteristics of algorithms, algorithm design tools: pseudo code and flowchart, relationship among data, data structure and algorithms, analysis of algorithms, asymptotic notation.

Unit II **(6 Hours)**

Sequential Organization Data Structures: Stacks: primitive operations, stack as an ADT, realization of stacks using array, multi-stack, applications of stack, simulating recursion using stack

Queue: primitive operations, queues as ADT, realization of queue using array, circular queue, double ended queue, priority queue, applications of queue.

Unit III **(6 Hours)**

Linked Organization Data Structures: Introduction, sequential vs. linked organizations, static vs. dynamic memory allocation, realization of linked lists, dynamic memory management, linked list as ADT, types of linked list, polynomial manipulations, linked stack, linked queue, generalized linked list, applications oflink list.

Unit IV **(6 Hours)**

Non-Linear Data Structure-Tree: Tree terminology, types of trees, binary tree as an ADT, realization of tree, tree traversals, binary search tree, operations on BST, threaded binary tree, AVL tree,heap tree, applications of trees.

Unit V **(6 Hours)**

Non-Linear Data Structure-Graph: Graph terminologies, graph as an ADT, realization of graphs using adjacency matrix and adjacency list, graph traversals: breadth first search traversal, depth first search traversal, spanning tree, prim's and kruskal's algorithms, topological sorting, applications of graph

Unit VI **(6 Hours)**

Hashing and File Organization:

Hashing: introduction, key terms, hash function, Collision Resolution strategies,hash table overflow, skip list, comparison of hashing and skip lists.

File: concept of file, file organization, sequential file organization, direct access file organization, indexed sequential file organization.

Textbooks:

1. Y. Langsam, M. Augenstin, A. Tannenbaum, "Data Structures using C and C++", Prentice Hall of India, , ISBN-81-203-1177-9.
2. E. Horowitz, S. Sahni, D. Mehta, "Fundamentals of Data Structures in C++", Galgotia Book Source, New Delhi, ISBN 16782928
3. S. Lipschutz, "Data Structures", McGraw Hill Pub.
4. Patil V., "Data Structures using C++", Oxford university press, ISBN 0-19-806623-6
5. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein,"Introduction to Algorithms"

Reference Books

1. G. A.V, PAI , "Data Structures and Algorithms ", McGraw Hill, ISBN -13: 978-0-07-066726-6
2. M. Welss, "Data Structures and Algorithm Analysis in C++", Pearson Education, ISBN-81-7808-670-0

List of Laboratory Exercise

1. Write a program to implement functions (insert, delete, display) on stack, queue and circular queue data structure.
2. Write a program to convert and solve expression from
   (a)Infix to Prefix
   (b) Infix to Postfix
   Evaluate Postfix expression
3. Write a program to implement Singly Linked List manipulation for storing student information (PRN, Name, Marks).
   a. Display data of top rank student.
   How many students secure first class and above rank?
4. Write a program to implement Doubly Linked List manipulation for storing Employee information (Name, Salary, Age).

a) Display data of employees having salary more than 50,000.
b) Display list of employees having age less than 30 and salary greater than 30,000.

5. Write a program to implement Binary Search Tree storing city names and Traversal in BST (Inorder, Preorder, Postorder).
6. Write a program to implement Threaded Binary Tree and its Traversals.
7. Write a program to implement graph traversals: BFS and DFS.
8. Write a program to implement Prims and Krushkals algorithms MST.

Project Based Learning Assignments

Note:- *Students in a group of 3 to 4 shall complete any one project from the following list

1. Design and develop a project for Basic Search Engine
2. Design and develop a project for Sudoku game
3. Design and develop a project for flight route planner
4. Design and develop a project for Music Player
5. Create a mini project to construct game: Tic-Tac-Toe
6. Design and develop a project for Phone Directory using doubly link list
7. Create a mini project to construct game: Snakes and Ladder

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Information Technology Laboratory-III

TEACHING SCHEME		EXAMINATION SCHEME		CREDIT SCHEME	
			Marks		Credits
Tutorial	01 Hours/Week	Term Work	25 Marks	Tutorial	01
Practical:	02 Hours/Week	Practical	25 Marks	Practical	01
Total	03 Hours/Week	Total	50 Marks	Total	02

Course Objective:

1. Compute time and space complexity for a given program.
2. Demonstrate concepts OOPS using java.
3. Solve specified requirement.
4. Infer various approaches to decide the efficiency of the given approach.
5. Formulate a given problem by providing the proof of behaviour of the given model.
6. Design an application using a platform-independent approach.

Prerequisite:

Basic understanding of Object-Oriented Programming language and logic to solve. given problem.

Course Outcomes: On completion of the course, students will have the ability to:

1. Design a solution to a given problem applying logic and features of the java language
2. Develop their logical skill through various assignments and practicals.
3. Divide complex problem into subpart and then handle every part to achieve the Goal.
4. Model a solution to any real-world problem
5. Analyze the significance of platform independence.
6. Design application using object-oriented norms.

Unit I **(6 Hours)**

Introduction to Java: Java Fundamentals, Features of Java OOPs concepts Java virtual machine Reflection byte codes Byte code interpretation Data types, variable, arrays, expressions, operators, and control structures - if, switch, and loops like for, do while, while. Introduction to Objects and classes.

Unit II **(6 Hours)**

Classes and objects: Java Classes, Abstract classes Static classes Inner classes Packages, Wrapper classes. Interfaces, This, Super, Access control, Inheritance, Encapsulation, Polymorphism, Data Binding, data abstraction.

(6 Hours)

Unit III

String and Arrays: One dimensional Array, Multidimensional array, Array of an object, Introduction to vector. String, StringBuilder, String Buffer, String methods, manipulations.

Unit IV **(6 Hours)**

Exception Handling: Checked exceptions, unchecked exceptions, and Errors, try catch block, throws, User-defined exception – Throw, Common exception classes.

Unit V **(6 Hours)**

Threading and multithreading: Lifecycle of Thread, Basic functions of thread, multithreading, synchronization.

Unit VI **(6 Hours)**

Collections and Generics: Introduction to collection framework, List, Set, Maps, utility class, Reflection API, Generics.

Textbooks

1. OCA Java SE 8 Programmer I Study Guide (Exam 1Z0-808) (Oracle Press) 3rd Edition. by Edward Finegan, Robert Liguori.
2. OCA Java SE 8 Programmer, Exam Guide (Exams 1Z0-808) 1st Edition, Kathy Sierra, Bert Bates.
3. Programmer's Guide to Java SE 8 Oracle Certified Associate (OCA), Khalid A. Mughal and Rolf W Rasmussen.

Reference Books

1. Headfirst Java, 2nd Edition by Kathy Sierra, Bert Bates.
2. Java: The Complete Reference, Eleventh Edition 11th Edition, Herbert Schildt.
3. OCAJP Associate Java 8 Programmer Certification Fundamentals: 1Z0-808, Hanmant Deshmukh.

List of Laboratory Exercises

1. Write a program to demonstrate the working of control structures in java.
2. Write a program for single inheritance and multilevel inheritance
3. Write a program to perform SUM of all values of integer array
4. Write a program to compare String in JAVA
5. Write a program to Split word into number of substring
6. Write a program to perform Runtime Polymorphism
7. Write a program to demonstrate Palindrome.
8. Write a program for Try Catch block in java
9. What is thread ? WAP for creating thread.
10. Write a program to perform Matrix Multiplication.

Note: * Students can use Leet Code website for assignments.

Value Added Course -I: A) Internet Security

TEACHING SCHEME		EXAMINATION SCHEME		CREDIT SCHEME	
Theory:	02 Hrs/Week	Internal Assessment	100 Marks	Theory	02
Total	02 Hrs/Week	Total	100 Marks	Total	02

Course Objective:

1. Gain knowledge of the various aspects of network architecture and protocols, Network performance.
2. Explore common vulnerabilities and threats in web applications.
3. Learn techniques to secure web applications and prevent attacks.
4. Gain practical experience through assignments, labs, and projects.

Prerequisite:

Basic networking, web development, cybersecurity awareness, OS basics, critical thinking

Course Outcomes: On completion of the course, students will have the ability to:

1. Analyse and evaluate the cyber security needs of an organization.
2. Describe system and Network vulnerability.
3. Evaluate network defence tools.
4. Understand the cyber laws.
5. Classify cyber forensics investigation and cyber security solutions for information assurance.
6. Investigate a cybercrime, prepare report, and apply laws for the case.

Unit I: Introduction to Cyber Security **(4 Hours)**

CYBER SECURITY- Introduction to Cyber Security, Implementing Hardware Based Security, Software Based Firewalls, Security Standards, Assessing Threat Levels, Forming an Incident Response Team, Reporting Cybercrime, Operating System Attacks, Application Attacks, Future scope of cyber security.

Unit II : System and Network vulnerability **(4 Hours)**

Systems Vulnerability Scanning Overview of vulnerability scanning, Open Port / Service Identification, Banner / Version Check, Traffic Probe, Vulnerability Probe, Vulnerability Examples.

Networks Vulnerability Scanning - Netcat, Socat, understanding Port and Services tools - Datapipe, Fpipe, WinRelay, Network Reconnaissance – Nmap, THC-Amap and System tools. Network Sniffers and Injection tools – Tcpdump and Windump, Wireshark, Ettercap, Hping Kismet

Unit III: Introduction to Network Defence Tool **(4 Hours)**

Network Defence tools Firewalls and Packet Filters: Firewall Basics, Packet Filter Vs Firewall, Packet Characteristic to Filter, Stateless Vs Stateful Firewalls, Network Address Translation (NAT) and Port Forwarding, Snort: Introduction Detection System

Unit IV: Introduction to Cyber Crime and law **(4 Hours)**

Introduction to Cyber Crime and law Cyber Crimes, Types of Cybercrime, Hacking, Attack vectors, Cyberspace and Criminal Behavior, Clarification of Terms, Traditional Problems Associated with Computer Crime, Introduction to Incident Response, Digital Forensics, Realms of the Cyber world, Recognizing and Defining Computer Crime, Contemporary Crimes, Contaminants and Destruction of Data, Indian IT ACT 2000.

Unit V: Cyber Forensic Investigation **(4 Hours)**

Cyber Forensic Investigation - Introduction to Cyber Forensic Investigation, Investigation Tools, eDiscovery, Digital Evidence Collection, Evidence Preservation, E-Mail Investigation, E-Mail Tracking, IP Tracking, E-Mail Recovery, Encryption and Decryption methods, Search and Seizure of Computers, Recovering deleted evidence, Password Cracking

Unit VI: Introduction to Cyber Crime Investigation **(4 Hours)**

Introduction to Cyber Crime Investigation Keyloggers and Spyware, Virus and Warms, Trojan and backdoors, Steganography, DOS and DDOS attack, SQL injection, Buffer

Textbooks

1. CyBoK, The Cyber Security Book of Knowledge, Oct 2019.
2. Nina Godbole and Sunit Belpure, Cyber Security Understanding Cyber Crimes, Computer Forensics and Legal Perspectives, Wiley
3. B. B. Gupta, D. P. Agrawal, Haoxiang Wang, Computer and Cyber Security: Principles, Algorithm, Applications, and Perspectives, CRC Press, ISBN 9780815371335, 2018.

Reference Books

1. Cyber Security Understanding Cyber Crimes, Computer Forensics and Legal Perspectives by Nina Godbole and Sunit Belpure, Publication Wiley.
2. Cyber Security and Cyber Laws Paperback – 2018 by Alfred Basta, Nadine Basta , Mary Brown , Ravinder Kumar, publication Cengage.
3. Cyber SecurityEssentials, James Graham, Richard Howard and Ryan Otson, CRC Press.

4. Introduction to Cyber Security , Chwan-Hwa(john) Wu,J.David Irwin.CRC PressT&FGroup

List of Project Based Learning

1. Setting up web application firewalls.
2. Performing port scanning and vulnerability assessment (using tools like Nmap).
3. Configuring secure network protocols (e.g., SSH, HTTPS).
4. Implementing host-based intrusion detection systems (HIDS).
5. Setting up security information and event management (SIEM) systems.
6. Demonstrate perform cryptographic attacks (e.g., brute force, chosen plaintext).
7. Conducting wireless network penetration testing..
8. Performing social engineering exercises (e.g., phishing simulations, pretexting)..
9. Discussing the implications of cybersecurity on privacy and civil liberties.
10. Understanding cybersecurity laws and regulations (e.g., GDPR, HIPAA).

VAC-I: B) Data Analysis Tools

TEACHING SCHEME		EXAMINATION SCHEME		CREDIT SCHEME	
			Marks		Credits
Theory	02 Hours/Week	Internal Assessment	100	Theory	02
Total	02 Hours/Week	Total	100	Total	02

Course Objective: Learn new skills and discover the power of Microsoft products in data analysis.

Prerequisite: Basic Knowledge of Spreadsheets.

Course Outcomes: On completion of the course, students will have the ability to:

1. Work with Excel basic and advanced formulas for data manipulation.
2. Perform data analysis using spreadsheets.
3. profile, clean, and load data into Power BI before you model your data.
4. Build out your semantic model to get the data insights.
5. Create an interactive report using common types of data visualization with Power BI Desktop.
6. Discover the benefits of the Microsoft modern analytics technology suite of tools.

Unit I **(4 Hours)**

Introduction: About Excel & Microsoft, Columns & Rows, Functionality Using Ranges, Using Formulas, Formula Functions, Advance Formulas.

Unit II **(4 Hours)**

Data Analysis with Excel: Spreadsheet Charts, What if analysis, Sorting, Filter, Text to Column, Data Validation, PivotTables.

Unit III **(4 Hours)**

Basics of Data analysis with PowerBI: Get data with Power BI Desktop, Clean, transform, and load data in Power BI, retrieve data from a variety of data sources, including Microsoft Excel, relational databases, and NoSQL data stores.

Unit IV **(4 Hours)**

Model data with Power BI: Design a semantic model in Power BI, Write DAX formulas for Power BI Desktop models.

Unit V **(4 Hours)**

Build Power BI visuals and reports: Design effective reports in Power BI, Create dashboards in Power BI

Unit VI **(4 Hours)**

Perform analytics in Power BI: Use visuals in Power BI, Introduction to modern analytics using Excel and Power BI, Transition from Excel to Power BI

Textbooks

1. Data Analysis with Microsoft Excel by K. Berk , Partrick Carey
2. Data Analysis and Business Modelling Using Microsoft Excel by Manohar, Hansa Lysander, Print Book ISBN : 9788120352889
3. Microsoft Power BI Quick Start Guide by Devin Knight, Packt Publishing, ISBN: 9781789138221.
4. Beginning Microsoft Power BI: A Practical Guide to Self-Service Data Analytics by Dan Clark, Publisher: Apress, ISBN-13: 978-1484256190

Reference Books

1. Business Statistics Using Excel: A Complete Course in Data Analytics, December 2023 by R. Panneerselvam
2. Data Analysis with Microsoft Power BI: A Kagen the Damned Novel, 28 January 2020 by Brian Larson.

Project Based Learning:

The faculty member will assign mini-projects among each group of 2-3 students, which will be evaluated as part of PBL.

Minor Course-I: Distributed Computing

TEACHING SCHEME		EXAMINATION SCHEME		CREDIT SCHEME	
	Hours/Week		Marks		Credits
Lecture:	3	End Semester Examination	60	Lecture	3
Practical:	2	Internal Assessment	40	Pract/Oral	1
		Oral	25		
		Termwork	25		
Total:	5	Total	150	Total	4

Course Objective: To provide an understanding of key concepts underlying the function of distributed computing systems

Prerequisite: Students should have knowledge of Fundamentals of Data Structures, Operating Systems, Networking concepts

Course Outcomes: On completion of the course, students will have the ability to:

1. Discover the fundamentals of distributed computing environment
2. Observe the inter process communication
3. Infer the concepts of process and threads
4. Interpret the concepts of clock synchronization
5. Infer the concepts of Distributed file system
6. Interpret the concepts distributed shared memory

Unit I Fundamentals: **(6 Hours)**

Definition and evolution of Distributed Computing System, Models and Types of Distributed Computing Systems, Issues and Goals in designing Distributed System, Distributed Computing Environment, Peer to peer systems and its middleware, Routing overlays, Mobile and Ubiquitous computing, Data Center architecture and management

Unit II Communication: **(6 Hours)**

Interprocess communication (IPC): Introduction and need Message Passing system: Desirable features of good message passing system, Issues in IPC, Group and multicast communication, Remote Procedure Calls (RPC): The RPC Model, Implementation of RPC mechanisms (Stubs and marshalling) Java RMI: Architecture, Implementation (Stubs and Skeletons)

Introduction of Web services and SOAP.

Unit III Processes and Threads: **(6 Hours)**

Process Migration: Introduction, Features, Mechanisms, Advantages, use in heterogeneous systems.

Threads: Concept, Motivation, Models, Issues, Synchronization, Scheduling, Implementing.

Unit IV Synchronization and Distributed Transactions: **(6 Hours)**

Clock synchronization: Drifting, Issues, Algorithms, Event Ordering Deadlock: Conditions, Modeling, Handling, Avoidance, Prevention, Detection, Election Algorithms

Distributed Transaction: Introduction, Locks, Optimistic Concurrency Control, Timestamp Ordering

Unit V Distributed File system **(6 Hours)**

Distributed Files Systems: Advantages, Features, Models, Caching, Replication, Fault Tolerance

Unit VI Distributed Shared memory **(6 Hours)**

Distributed Shared Memory: Architecture, Design and Implementation Issues, Advantages, Granularity, Structure of Shared Space, Consistency Models, Replacement Strategy, Thrashing

Textbooks

1. Pradeep K. Sinha, "Distributed Operating Systems: Concepts and Design", Wiley-IEEE Press.
2. Andrew S. Tanenbaum, Maarten van Steen, "Distributed Systems: Principles and Paradigms", Prentice Hall India Learning Private Limited,
3. George Coulouris, Jean Dollimore, Tim Kindberg, Gordon Blair, "Distributed Systems: Concepts and Design", Pearson Education India;

Reference Books

1. Ajay D. Kshemkalyani, Mukesh Singhal, "Distributed Computing: Principles, Algorithms, and Systems", Cambridge University Press- South Asian edition
2. Abraham Silberschatz, Peter B. Galvin, Greg Gagne , "Operating System Concepts", Wiley, 8th Edition

List of Assignments

The course coordinator shall design suitable assignments involving case studies of Opensource distributed systems, Distributed Computing models, Distributed security systems etc.

List of Laboratory Exercises

The course coordinator shall design suitable laboratory exercises on Opensource distributed systems, inter-process communications, threads, election algorithms etc.

Project Based Learning –

The course coordinator shall design suitable projects to students in groups of four

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

B. Tech
Information Technology
Semester IV

Formal Languages & Automata Theory

TEACHING SCHEME		EXAMINATION SCHEME		CREDIT SCHEME	
Lecture:	3 Hours/Week	End Semester Examination:	60 Marks	Theory	3
		Internal Assessment:	40 Marks		
Total	3 Hours/Week	Total	100 Marks	Total	3

Course Objective:
Students will learn about a variety of issues in the mathematical development of computer. science theory, particularly finite representations for languages and machines. Students will gain more formal understanding of algorithms and procedures.

Prerequisite:
Students should have knowledge of set theory and state transition diagrams.

Course Outcomes: On completion of the course, students will have the ability to
1.Design automata machines for strings given.
2.Write a regular expression for the given string and find set of strings if regular expression is given.
3.Write grammar rules for the strings given.
4.Design push down automata for the string and grammar.
5.Design Turing machine and apply the same to solve algorithmic problems.
6.Apply knowledge computation in complexity theory.

Unit I : **(6 Hours)**
Finite Automata: Introduction to Finite Automata, Structural Representations, Deterministic finite Automata (DFA)-Formal Definition, Simplified notation: State transition graph, Transition table, Language of DFA, Nondeterministic finite Automata (NFA), NFA with epsilon transition, Language of NFA, Equivalence and Minimization of Automata, Conversion of NFA with epsilon to DFA Equivalence of Moore and Mealy Machine. Applications and Limitation of FA.

Unit II **(6 Hours)**
Regular expressions: Regular expression (RE), Definition, Operators of regular expression and their precedence, Algebraic laws for Regular expressions, Kleen's Theorem, Regular expression to DFA, DFA to Regular expression, Non-Regular Languages, Pumping Lemma for regular Languages, Closure properties of Regular Languages, Applications of regular expressions.

Unit III **(6 Hours)**
Grammar: Definition, Production rules, Derivation trees, Ambiguous Grammar, Removal of ambiguity, Regular Grammar, Inter-conversion between RE and Grammar, Reduced form of grammar. Linear grammar: left & right linear grammar, Inter- conversion. Chomsky hierarchy of languages, Context Free

Grammar- Definition, Context free language (CFL. Normal Forms- Chomsky Normal Form (CNF), Griebach Normal Form (GNF).

Unit IV **(6 Hours)**

Push Down Automata (PDA): Limitations of FA, PDA: Definition, Uses, Equivalence between FA and PDA, Designing of PDA, Deterministic Push Down Automata and Non-Deterministic Push Down Automata- Definition, Language accepted by PDA, Designing a PDA for CFG, Properties of CFL, Pumping Lemma for CFL. Limitations of PDA, Applications of PDA.

Unit V **(6 Hours)**

Turing Machine (TM): Definition, Model, Comparison of TM, FSM, PDA, Design of TM, Examples of TM- Combinational TM, Iterative TM, Recursive TM, Universal TM, TM as a language acceptor, Some Problems that cannot be solved by Turning Machines, Language accepted by TM, Church's Turing hypothesis, Multitask TM, TM limitations.

Unit VI **(6 Hours)**

Applications: Applications of Turing machine, Compiler phases, editors and its relevance to TM. Case studies.

Textbooks

1. "Introduction to Automata Theory, Languages and Computation", Hopcroft J, Motwani R, Ullman ,Addison-Wesley, ISBN 81-7808-347-7, Third Edition .
2. "Introduction to Theory of Computation", Michael Sipser, Course Technology, ISBN-10: 053494728X, Forth Edition. ISE.

Reference Books

1. "Introduction to Languages and Theory of Computation", John Martin. Fifth Edition, McGrawHill.
2. "Computational Complexity", Christos H. Papadimitriou, Pearson Education.

Project Based Learning Assignments

Note:- Students in a group of 3 to 4 shall complete any one project from the following list

1. Describe the process of designing the computer. How is it related with the simple automata?
2. Write project based on famous computer scientist Alan Turing. Select suitable material for reference and summarize.
3. Describe the set of problems which can be represented using machines. What are the criteria we can enlist for such representations?
4. Relate computational theory to World War II. What is the role of cryptography in World War II?
5. Invention of computer as a machine is related to formal automata. How today's complex and high-end computer systems can be mapped to these simple automata. Describe in detail.

6. Select a real-world problem and represent it mathematically. Design an automaton to solve this problem. Write detailed explanation of the entire process.
7. Study any text editor. Enlist its features. Map these features with the concepts you learned in the subject.
8. Enlist set of problems which can be solved, and which cannot be solved by memoryless automata. How does memory affects the power of automata? Explain in detail and justify your answer with example.
9. Why Ethereum blockchain must be deterministic? Study and explain application of computation theory to blockchain technology.
10. Can human brain be simulated by Tring machine? Write detailed essay and justify your conclusions with theorem you learned.
11. Study research paper published by Alan Turing and write a summary in your words.
12. What are the similarities and differences between human brain and machine? Support your answers with suitable mathematical model.
13. Study any chess game software. Write the process of developing such software. Describe how this is related to Turing machine.

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit – VI

Advanced Database Systems

TEACHING SCHEME		EXAMINATION SCHEME		CREDIT SCHEME	
	Hours/Week		Marks		Credits
Lecture	03	End Semester Examination	60	Lecture	03
		Internal Assessment	40	Practical	01
Practical	02	Term Work	25		
		Practical	25		
Total	05	Total	150	Total	04

Course Objective:

1. Exploring the working of large scale and emerging database management systems
2. Study and analysis of query processing and query optimization in distributed and parallel databases

Prerequisite:

Student should be well aware of database management systems, analysis of data structure and algorithms with sufficient programming experience

Course Outcomes:

On completion of the course, students will have the ability to:

1. Interpret the working of distributed database management system
2. Infer the processing and optimization of distributed queries
3. Recognise the architecture and query processing in parallel database management system
4. Analyse the concepts of advanced transaction management
5. Identify the concepts of different information retrieval systems
6. Discover the structure and significance of Big Data and NoSQL Databases

Unit I - Distributed databases: Architecture and Design (6 Hours)

Distributed Data processing: What is a DDBS?; Advantages and disadvantages of DDBS, Problem areas

Distributed DBMS Architecture: Transparencies in a distributed DBMS, Distributed DBMS architecture, Global directory issues,

Distributed Database Design: Alternative design methodologies and strategies, Distributed design issues, Types and role of Fragmentation, Types and role of replication, Data allocation

Unit II - Distributed query processing and optimization (6 Hours)

Distributed Query processing: Problem of query processing, Distributed query, Query decomposition, Distributed Query Processing Methodology, translation global queries to fragment queries

Distributed Optimization: Objectives of query optimization, Factors governing query optimization, Ordering of fragment queries, optimization of join operation, Load balancing, Distributed query optimization algorithms

Unit III - Parallel Database Management System **(6 Hours)**

Introduction: Types of parallelism in database systems, Parallel Query Processing, multiprocessor architectures, parallel relational operators, parallelism in main-memory DBMS, parallel handling of integrity constraints, Integrated I/O parallelism

Parallel Query Processing and Optimization: Inter-query parallelism, intra-query parallelism, intra-operation parallelism, inter-operation parallelism, objectives of parallel query optimization, parallel query optimization, load balancing, parallelism in join queries, testing the quality of query optimization

Unit IV – Advanced concepts in Transaction Management **(6 Hours)**

Transaction Management: ACID properties, pessimistic locking, optimistic locking, flat transactions, nested transactions, deadlock detection and management and their algorithms, Recovery Methods

Concurrency control and Reliability in Distributed Dtabases: Concurrency control in centralized database systems vs Concurrency control in DDBSs, Distributed concurrency control algorithms, Deadlock management, Reliability issues in DDBSs; Types of failures, Reliability techniques, Commit protocols, Recovery protocol.

Unit V –Advanced Querying and Information Retrieval **(6 Hours)**

Decision Support Systems, Data Analysis and OLAP, Data Mining, Data Warehousing, Information Retrieval Systems, Semantic Search

Database Tuning and Performance: Benchmarking, TPC benchmarks, object oriented benchmarks, TP Monitors, TPC and Wisconsin benchmarks, performance measurement, and performance tuning

Unit VI - Big Data, NoSQL and Vector Databases **(6 Hours)**

What is NoSQL? Why NoSQL? History of NoSQL Databases, Features of NoSQL, Types of NoSQL Databases, Query Mechanism tools for NoSQL, CAP Theorem

Big Data - Introduction, Types, Characteristics, Testing, Examples, Introduction to Hadoop, MongoDB- Introduction, Architecture, Features, Data Modelling in MongoDB

Vector Databases: Structure, Characteristics, Role, Challenges

Textbooks

1. Database System Concepts, Seventh Edition, AviSilberschatz, Henry F. Korth, S. Sudarshan, McGraw-Hill
2. Data Warehousing: Concepts, Techniques, Products and Applications, 3rd Edition, C.S.R. Prabhu, PHI Learning Pvt. Ltd.
3. Stefano *Ceri and* Giuseppe *Pelagatti,* "Distributed databases principles and systems", Tata McGraw Hill

Reference Books

1. NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence, Sadalage, P. & Fowler, Wiley Publications
2. M. Tamer Özsu and Patrick Valduriez, "Principles of Distributed Database Systems", Springer Science & Business Media, 3rd edition

List of Laboratory Exercises

The course instructor shall frame appropriate assignment on any NoSQL/ Vector Database tools with all phases from installation, Design, Modelling and CRUD functionalities

Project Based Learning

Implementation of Database in NoSQL/ Vector Database with monitoring, backup, security, report generation features.

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit – III
Unit Test -2	Unit – IV, Unit – V, Unit – VI

Software Engineering

TEACHING SCHEME		EXAMINATION SCHEME		CREDIT SCHEME	
	Hours/Week		Marks		Credits
Lecture:	03	End Semester Examination	60	Lecture	03
Practical	02	Internal Assessment	40		
		Term Work	25	Practical	01
		Oral	25		
Total	05	Total	150	Total	04

Course Objective:

1) Understand software development life cycle model
2) Understand change management, process and product metrics

Prerequisite:

Students should have knowledge of

Developing well-structured, modular programs

Course Outcomes: On completion of the course, students will have the ability to:

1. Use all the phases of the Software Development Life Cycle to develop software systems
2. Identify the requirements and implement appropriate software solutions for the real-world need
3. Apply systematic procedure for software design and deployment
4. Identify items of Software change management system
5. Compare and contrast the various testing and maintenance techniques
6. Understand software product and process quality standards.
7. Work efficiently in teams with effective communication and follow ethical principles.

Unit I **(6 Hours)**

Introduction: What is software? Types of software, Characteristics of Software, Attributes of good software, what is software engineering? key challenges of software engineering? Systems engineering & software Engineering.

Software Development Process Models: Traditional Life Cycle Models: Waterfall, V-model, Evolutionary, Spiral, CBSE, Unified Process, Rapid Application Development, Prototyping, Agile Software Engineering Process Models: Extreme Programming, Agile Software Development, Scrum.

Unit II **(6 Hours)**

Software Requirements Engineering and Analysis: Types of software requirements: Functional and non-functional requirements, Domain requirements, User requirements.

Analysis of requirements: Viewpoints, Interviewing, Scenarios, Use-cases, Process modelling with physical and logical DFDs, Entity Relationship Diagram, Data Dictionary.

Requirement validation, Requirement specification, Software requirement Specification (SRS) Structure, SRS format, Feasibility.

Unit III **(6 Hours)**

Software Design: Design concepts: Abstraction, Modularity, Information hiding, Functional Independence, Refinement, Refactoring, Data Design, Architectural Styles and Patterns, Architectural design & design process, Effective modular design, cohesion, coupling, Mapping Data flow into a Software Architecture.

Agile design practices: Role of design Principles including Single Responsibility Principle, Open Closed Principle, Liskov Substitution Principle, Interface Segregation Principles, Dependency Inversion Principle in Agile Design, Need and significance of Refactoring

Unit IV **(6 Hours)**

Change Management

Software configuration management (SCM), Elements of SCM Base lines, Software configuration items, SCM Repository, Scan process, Version Control, Change Control, Configuration Audit, Status Reporting, GitHub – Introduction, Continuous Integration, Benefits of Continuous Integration, Git Feature Branch Workflow: Working in Branches, Making pull request, Continuous Delivery Key Principles and practices

Unit V **(6 Hours)**

Software Implementation Techniques: Coding Practices, Refactoring, Maintenance and Reengineering-BPR model-Reengineering process model, Reverse and Forward Engineering

Software testing fundamentals: Software Testing Life Cycle, Internal and external views of Testing-white box test, basis path testing-control structure testing-black box testing- Regression Testing, Unit Testing, Integration Testing, Validation Testing, System Testing and Debugging, Defect: Defect Life Cycle

Unit VI **(6 Hours)**

Product metrics: Software quality, metrics for analysis model, metrics for design model, metrics for source code, metrics for testing, metrics for maintenance

Process Metrics: process framework, the capability maturity model integration (CMMI), process patterns, process assessment.

Quality Management: Quality concepts, software quality assurance, software reviews, formal technical reviews, statistical software quality assurance, software reliability, the ISO 9000 quality standards.

Textbooks

1. Rajib Mall, Fundamentals of Software Engineering, Prentice Hall India.
2. Pankaj Jalote, An integrated approach to Software Engineering, Springer/Narosa.
3. Roger S. Pressman, Software Engineering: A practitioner's approach, McGraw Hill.
4. Ian Sommerville, Software Engineering, Addison-Wesley

Reference Books

1. Kelkar S.A., ―Software Engineering, Prentice Hall of India Pvt Ltd, 2007.
2. Stephen R.Schach, Software Engineering, Tata McGraw-Hill Publishing Company Limited,2007

List of Laboratory Exercises

1. Preparing Software Requirements Specifications
2. E-R Modeling
3. Modeling Data Flow Diagrams
4. State chart and Activity Modeling
5. Estimation of Test Coverage Metrics and Structural Complexity
6. Designing Test Suites
7. Study design of Biometric Authentication software
8. Selemium Testing with any online application

Project Based Learning - Provisional List of Projects

1. Development of requirements specification.
2. Function oriented design
3. Demonstrate use of appropriate CASE tools and other tools such as configuration management tools
4. Demonstrate program analysis tools in the software life cycle.

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Computer Organization & Architecture

TEACHING SCHEME		EXAMINATION SCHEME		CREDIT SCHEME	
	Hours/Week		Marks		Credits
Lecture:	03	End Semester Examination	60	Lecture	03
		Internal Assessment	40		
Total	03	Total	100	Total	03

Course Objective:

1. To learn the low-level design and working of computer/processor
2. To learn parallel computing architectures and platforms

Prerequisite:

Digital Electronics, Microprocessor Architecture, Structured Programming

Course Outcomes: On completion of the course, students will have the ability to:

1. Understand the recent trends in Computer Architecture
2. Apply various design methodologies for the digital design of devices using VHDL.
3. Analyze, understand and apply the various design techniques of control unit of a processor
4. Understand the basic design of a processor and memory technologies
5. Analyze and understand different multiprocessor architectures
6. Analyze and understand different parallel processing architectures

Unit I – Recent Advances **(4 Hours)**

Technology trends in Computer Architecture, Performance Metrics, Improving performance, Moore's law, Cluster Computing, Cloud Computing,Quantum Computers, Hardware support for Operating Systems, Computer architecture for AI/ML : GPU,TPU. Multicore architecture

Unit II – Digital Logic Design, Simulation and Debugging with HDLs **(8 Hours)**

Introduction to hardware description languages. Study of any of the following HDLs : A) VHDL B) Verilog

Logic Gate and Logic circuit design using above HDLs. Introduction to HDL simulation and debugging. Introduction to FPGA.

Unit III – Control Unit Design **(6 Hours)**

Hardwired Control Unit, Micro-programmed Control Unit design, RecentTrends

Unit IV – Processor and Memory Design **(6 Hours)**

Basic design of a Processor, Control path, Data path

Cache memory: Working principle, Mapping functions, Replacement algorithms, Cache coherence, Examples, Atomic Memory, UFFO storage,UltraRAM, 3D NAND, Intel Optane memory, Recent Trends

Unit V – Multiprocessor Architectures **(4 Hours)**

Shared memory – Distributed Memory multiprocessor architectures, Message-Passing Multiprocessors, Dataflow machine architecture Supercomputer architecture, Recent Trends

Unit VI – Parallel Computing and Programming **(8 Hours)**

Pipelining, Data and Control Hazards, Stalls, RISC/Pentium-4 Pipeline, Complex Pipelines, Out-of-order Execution, Dynamic Scheduling, Tomasulo Algorithm, Register renaming, Register Scoreboarding, Basic compiler techniques for exposing instruction-level parallelism, Vector processors, Array processors, VLIW architecture, Multithreaded architecture, GPU Computing architecture, Nvidia Maxwell, CUDA, Writing a simple parallel algorithm, Parallel Programming languages, OpenMP, MPI, Pthreads, Amdahl's Law, Gustafson-Barsis's Law, Karp-Flatt Metric, isoefficiency, Recent Trends

Textbooks

1. Computer Organization and Architecture, William Stallings, Prentice Hall
2. Computer Organization and Embedded Systems, Hamacher&Zaky, McGraw Hill
3. Advanced Computer Architecture, Kai Hwang, Tata McGraw Hill
4. Fundamentals of Logic Design, Charles Roth & Larry Kinney, Cengage Learning
5. The Verilog: Hardware Description Language, Thomas &Moorby, Extra Materials
6. Advanced Computer Architecture and Parallel Processing, Rewini& Barr, Wiley Publications

Reference Books

1. Computer Organization and Design: The Software/Hardware Interface, David Patterson, Elsevier
2. Fundamentals and Standards in Hardware Description Languages, Jean Mermet, Springer Science
3. Parallel Computers: Architecture and Programming, V.Rajaraman&C.Murthy, Prentice Hall India
4. Introduction to Parallel Computing: From Algorithms to Programming, Roman Trobec, Springer

Project Based Learning

1. Case studies in recent trends in Computer Architecture
2. Case studies in Hardware Description Languages and Simulators
3. Recent Trends in Control Unit Design

4. Case studies in recent Memory Technologies
5. Case studies in recent trends in Multiprocessor Architectures
6. Case studies in recent trends in Parallel Computing

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit – III
Unit Test -2	Unit – IV, Unit – V, Unit – VI

Applied Algorithm

TEACHING SCHEME		EXAMINATION SCHEME		CREDIT SCHEME	
Lecture:	3 Hours/Week	End Semester Examination:	60 Marks	Theory	3
Practical:	2 Hours/Week	Internal Assessment:	40 Marks		
		Term Work:	25 Marks	Practical	1
		Practical:	25 Marks		
Total	5 Hours/Week	Total	150 Marks	Total	4

Course Objective:
Understand and compare important algorithmic design paradigms and analysis of algorithms. To choose and extend efficient algorithms required for designs.

Prerequisite:
Students should be well versed with algorithms and operations on basic data structures stacks, queues, linked lists, trees, graphs. Students should have knowledge of searching sorting algorithms.

Course Outcomes: On completion of the course, students will have the ability to:
1. Interpret the performance of algorithms using analysis techniques.
2. Examine the fundamental algorithmic strategies.
3. Compare the fundamental algorithmic strategies.
4. Implement graphs and trees algorithms.
5. Interpret the tractable or intractable problem.
6. Summarize the advance types of algorithms.

Unit I **(6 Hours)**

Introduction to Algorithm analysis: Characteristics of Algorithm. Analysis of Algorithm: Asymptotic analysis of Complexity Bounds – Best, Average and Worst-Case behavior, Performance Measurements of Algorithm, Time and Space Trade-Offs. Analysis of Recursive Algorithms through Recurrence Relations: Substitution Method, Recursion Tree Method and Masters' Theorem.

Unit II **(6 Hours)**

Algorithmic Strategies 1: Brute-Force technique, Heuristics, Greedy algorithms, Divide and Conquer, Illustrations of these techniques for Problem-Solving.

Unit III **(6 Hours)**

Algorithmic Strategies 2: Dynamic Programming, Branch and Bound algorithms, Backtracking, methodologies; Illustrations of these techniques for Problem-Solving.

Unit IV **(6 Hours)**

Graph and Tree Algorithms: Self-Balancing tress, B Trees, B+ Trees, Single source shortest path algorithms, all pair shortest path algorithms, Network Flow Algorithm

Unit V **(6 Hours)**

Tractable and Intractable Problems: Computability of Algorithms, Computability classes – P, NP, NP-complete and NP-hard. Cook's theorem, Standard NP-complete problems and Reduction techniques.

Unit VI **(6 Hours)**

Advanced Topics: Approximation algorithms, Randomized algorithms, Class of problems beyond NP – P SPACE, Introduction to Quantum Algorithms and parallel algorithms.

Textbooks

1. "Fundamental of Computer Algorithms", E. Horowitz and S. Sahni, Orient Black.
2. "Introduction to Algorithms", T. H. Cormen, C. E. Leiserson and R. L. Rivest,PHI Learning Pvt. Ltd. (Originally MIT Press).
3. "The Design and Analysis of Computer Algorithms", A. Aho, J. Hopcroft and J. Ullman, Pearson Education India.
4. Computer Algorithms: Introduction to Design and Analysis, S. Baase, Pearson Education India.
5. "The Art of Computer Programming", D. E. Knuth, Addison Wesley.

Reference Books

1. M. Welss, "Data Structures and Algorithm Analysis in C++", Pearson Education, ISBN- 81-7808-670-0.
2. G. A.V, PAI , "Data Structures and Algorithms ", McGraw Hill, ISBN -13: 978-0-07-066726-6.

List of Laboratory Exercises

1. From a given vertex in a weighted connected graph, find shortest paths to other vertices using Dijkstra's algorithm.
2. Write a Code to find the shortest path using Bellman-Ford algorithm.
3. Write and analyze code to sort an array of integers using merge sort.
4. Write and analyze to sort an array of integers using divide and conquer quick sort Method.
5. Write a program to implement Longest Common Subsequence problem using Dynamic Programming.
6. Write a program to Implement 0/1 Knapsack problem using Dynamic Programming.
7. Write a program to Implement N Queen's problem using Back Tracking.
8. Write a program to implement quick sort using randomize algorithm.
9. Write a program to implement network flow algorithm.

Project Based Learning Assignments

Note:- Students in a group of 3 to 4 shall complete any one project from the following list

1. Design and develop a project for shortest path calculation for travelling salesman problem
2. Design and develop a project for finding keywords from the paragraph
3. Design and develop a project for Customer Billing system

4. Design and develop a project for word dictionary using search tree concept
5. Design and develop a project for salary calculation of employees based on performance
6. Design and develop a project for password recovery system

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit – VI

Information Technology Laboratory-IV

TEACHING SCHEME		EXAMINATION SCHEME		CREDIT SCHEME	
	Hours/Week		Marks		Credits
Tutorial	01 Hours/Week	Term Work	25 Marks	Tutorial	01
Practical:	02 Hours/Week	Practical	25 Marks	Practical	01
Total	03 Hours/Week	Total	50 Marks	Total	02

Course Objective:

1. Understand web environment for building the application.
2. Implement web application.
3. Implement Servlet.
4. Implement Java Messaging Services.
5. Implement Java Mail API.

Prerequisite:

1) Core Java 2) Scripting languages.

Course Outcomes: On completion of the course, students will have the ability to:

1. Understand the lifecycle of web application.
2. Implement session management using servlet.
3. Apply standard and custom tags of JSP.
4. Design competitive web application which will work real web environment.
5. 5mplement Java Messaging Services.
6. Apply Java Mail API.

Unit I **(6 Hours)**

Introduction to Servlet: Web Application Basics, Architecture and challenges of Web, application. Introduction to servlet, Servlet life cycle, Developing and Deploying Servlets, Exploring Deployment Descriptor (web.xml), Session Management and Servlet Chaining: Handling Request and Response, Initializing a Servlet, Accessing Database.

Unit II **(6 Hours)**

JDBC: Design of JDBC, JDBC configuration, Executing, SQL statement, Query Execution, Scrollable and updatable, result sets, row sets, metadata, Transaction

Unit III **(6 Hours)**

Java Server Pages: Standard Tags: Basic JSP Architecture, Life Cycle of JSP (Translation, compilation), JSP Tags and Expressions, Role of JSP in MVC-2, JSP with Database, JSP Implicit Objects.

Unit IV **(6 Hours)**

Java Server Pages: Custom Tags: Tag Libraries, JSP Expression Language (EL), Using Custom Tag, JSP Capabilities Exception Handling Session Management Directives JSP with Java. Introduction to struts.

Unit V **(6 Hours)**

Java Messaging Services: JMS Architecture, Point-to-Point Messaging, Domain, Publisher/Subscriber, Messaging Domain, JMS API, JMS Queue.

Unit VI **(6 Hours)**

Overview of Hibernate, Hibernate Architecture, Hibernate Mapping Types, Hibernate O/R Mapping, Hibernate Annotation, Java Mail API: SMTP, POP, IMAP, MIME, NNTP, sending operations,

Textbooks

1. Web Technologies: HTML, JAVASCRIPT, PHP, JAVA, JSP, ASP.NET, XML and Ajax, Black Book: HTML, JavaScript, PHP, Java, Jsp, XML and Ajax, Black Book Paperback – 1 January 2009, Kogent Learning Solutions Inc.
2. Java EE 8 Cookbook: Build reliable applications with the most robust and mature technology for enterprise development, Packt Publication, Elder Moraes.
3. Headfirst Servlets and JSP: Passing the Sun Certified Web Component Developer Exam 2nd Edition, Bryan Basham, Kathy Sierra, Bert Bates.
4. Java Persistence with Hibernate by Christian Bauer, Gavin King

Reference Books

1. Beginning Java EE 7, Appress Publication, Antonio Goncalves.
2. Java EE 7 Essentials: Enterprise Developer Handbook 1st Edition, Headfirst Publication, Arun Gupta.
3. J2EE: The complete Reference Paperback, Jim Keogh.

List of Laboratory Exercises

1. Maintain record of students and perform CRUD functionality.
2. Write a program to redirect a request using a dynamic approach.
3. Write a program to pass the data using session.
4. Write a servlet to remove spam.
5. Maintain the record of faculty member using jsp action tags and directives.
6. Design a tag to perform the necessary editing in a given report.
7. Design reusable components of the form using taglib.
8. Implement sending and receiving mail utility using Java Mail API.
9. Implement Java Message Service queue.
10. Understand working of framework – struts- case study.

Indian Knowledge System

TEACHING SCHEME		EXAMINATION SCHEME		CREDIT SCHEME	
Lecture:	2 Hours/Week	Internal Assessment:	100 Marks	Theory :	2 Credits

Course Objectives:

1. To sensitize the students about Indian culture and civilization including its Knowledge System and Tradition.
2. To help student to understand the knowledge, art and creative practices, skills, and values in ancient Indian system
3. To help to study the enriched scientific Indian heritage.
4. To introduce the contribution from Ancient Indian system & tradition to modern science & Technology

Course Outcomes: After learning this course students will be able to understand

1. Concepts of Indian Knowledge System
2. India's contribution in Philosophy and Literature
3. India's involvement in Mathematics and Astronomy
4. India's role in Medicine and Yoga
5. India's influence in Sahitya
6. Concepts of Indian Shastra

UNIT-1: Introduction to Indian Knowledge System (4 Hours)

Definition, Concept and Scope of IKS, IKS based approaches on Knowledge Paradigm, IKS in ancient India and in modern India

Unit-II: Philosophy and Literature (4 Hours)

Contributions by Maharishi Vyas, Manu, Kanad, Pingala, Parasar, Banabhatta, Nagarjuna and Panini in Philosophy and Literature

Unit-III: Mathematics and Astronomy (4 Hours)

Contribution of Aryabhatta, Mahaviracharya, Bodhayan, Bhashkaracharya, Varahamihira and Brahmgupta in Mathematics and Astrononmy

Unit-IV: Medicine and Yoga **(4 Hours)**

Major contributions of Charak, Susruta, Maharishi Patanjali and Dhanwantri in Medicine and Yoga

Unit-V: Sahitya **(4 Hours)**

Introduction to Vedas, Upvedas, Upavedas (Ayurveda, Dhanurveda, Gandharvaveda) Puran and Upnishad) and shad darshan (Vedanta, Nyaya.Vaisheshik, Sankhya, Mimamsa, Yoga, Adhyatma and Meditation)

Unit-VI: Shastra **(4 Hours)**

Introduction to Nyaya, vyakarana, Krishi, Shilp, Vastu, Natya and Sangeet

Reference Books

1.Textbook on IKS by Prof. B Mahadevan, IIM Bengaluru

2. Kapur K and Singh A.K (Eds) 2005). Indian Knowledge Systems, Vol. 1. Indian Institute of Advanced Study, Shimla. Tatvabodh of sankaracharya, Central chinmay mission trust, Bombay, 1995.

3. The Cultural Heritage of India. Vol.I. Kolkata:Ramakrishna Mission Publication, 1972.

4. Nair, Shantha N. Echoes of Ancient Indian Wisdom. New Delhi: Hindology Books, 2008.

5. Dr. R. C. Majumdar, H. C. Raychaudhuri and Kalikinkar Datta: An Advanced History of India (Second Edition) published by Macmillan & Co., Limited, London, 1953.

6. Rao, N. 1970. The Four Values in Indian Philosophy and Culture. Mysore: University of Mysore.

7. Avari, B. 2016. India: The Ancient Past: A History of the Indian Subcontinent from c. *7000* BCE to CE 1200. London: Routledge.

8. Textbook on The Knowledge System of Bhārata by Bhag Chand Chauhan,

9. Histrory of Science in India Volume-1, Part-I, Part-II, Volume VIII, by Sibaji Raha, et al. National Academy of Sciences, India and The Ramkrishan Mission Institute of Culture, Kolkata (2014).

10. Pride of India- A Glimpse of India's Scientific Heritage edited by Pradeep Kohle et al. Samskrit Bharati (2006).

12. Vedic Physics by Keshav Dev Verma, Motilal Banarsidass Publishers (2012).

13. India's Glorious Scientific Tradition by Suresh Soni, Ocean Books Pvt. Ltd. (2010).

14.Kapoor, Kapil, Avadesh Kr. Singh (eds.) *Indian Knowledge Systems* (Two Vols), IIAS, Shimla, 2005

Minor Course-2: Cloud Systems & Infrastructures

TEACHING SCHEME		EXAMINATION SCHEME		CREDIT SCHEME	
Theory	3 Hours/Week	End Semester Examination	60 Marks	Theory	3
Practical	2 Hours/Week	Internal Assessment	40 Marks	Practical	1
		Termwork	25 Marks		
		Oral	25 Marks		
Total	5 Hours/Week	Total	150 Marks	Total	4

Course Objectives:
1.To provide with the students the essentials and fundamentals of cloud computing.
2.To learn the basics of virtualization and its importance.

Course Prerequisites:
Students should have knowledge of
- Operating System
- Computer Networks

Course Outcome:
Students will be able to:
1) Understand the main concepts, key technologies and fundamentals of cloud computing..

2) Analyse various cloud computing models and apply them to solve problems on cloud.

3) To discuss system virtualization and outline its role in enabling the cloud computing system model.

4) Understand cloud infrastructure and storage.

5) Analyse the core issues of cloud computing such as security, privacy, and interoperability.

6) Understand the trends in cloud enabling technologies.

UNIT-I : Introduction to Cloud Technologies (6 Hours)

Introduction to the Cloud Computing, History of cloud computing, Cloud service options, Cloud Deployment models, Business concerns in the cloud.

UNIT-II: Cloud computing Architecture (6 Hours)

Introduction, delivery models: The SPI framework, Meaning of cloud service model, Types of cloud service models, characteristics and benefits, Difference between all models, cloud deployment models,Cloud Reference Model, Types of

Clouds, Economics of Clouds, Open Challenges, Cloud Platforms in Industry: Amazon Web Services, Google AppEngine, And Microsoft Azure.

UNIT-III : Virtualization and cloud platform **(6 Hours)**

Exploring virtualization, Load balancing, Hypervisors, Machine imaging, Cloud marketplace overview, Comparison of Cloud providers.

UNIT-IV: Cloud Infrastructure and storage **(6 Hours)**

History of data centers and their components like IT equipment and facilities along with design considerations like efficiency power, requirements, redundancy etc.
Cloud storage systems, their concepts and object storage, databases, distributed file systems

UNIT-V: Cloud Security **(6 Hours)**

Securing the Cloud, The security boundary, Security service boundary, Security mapping, Securing Data, Brokered cloud storage access, Storage location and tenancy, Encryption, Auditing and compliance, Establishing Identity and Presence, Identity protocol standards

UNIT-VI: Advanced Topics in Cloud Computing **(6 Hours)**

Energy Efficiency in Clouds, Market Based Management of Clouds, Federated Clouds / InterCloud, Third Party Cloud Services.
Cloud Service Providers: EMC, EMC IT, Captiva Cloud Toolkit, Google, Cloud Platform, Cloud Storage, Google Cloud Connect, Google Cloud Print, Google App Engine, Amazon Web Services,Amazon Elastic Compute Cloud, Amazon Simple Storage Service, Amazon Simple Queue , Windows Azure, Microsoft Assessment and Planning Toolkit

List of Laboratory Assignments will be framed by course Co-ordinator on each Unit.

Project Based Learning

1) To transfer the files from one virtual machine to another virtual machine.

2) Installing and running GATE.

3) To Simulate a cloud scenario using CloudSim and run a scheduling algorithm that is not present in CloudSim.

4) To Implement IaaS using your resources.

5) Design and deploy a PaaS environment.

6) Installation and Configuration of virtualization using KVM.

7) Study Cloud Security management.

8) Case study on Amazon EC2/Microsoft Azure/Google Cloud Platform.

9) Study and implementation of Storage as a Service.

10) Study and implementation of identity management.

Textbooks:

1) Mastering Cloud Computing, Buyya R, Vecchiola C, Selvi S T, McGraw Hill Education (India), 2013.
2) Cloud Computing Bible, Barrie Sosinsky ,Wiley Publishing Inc. 2011
3) Cloud Computing from Beginning to End by Ray J Rafaels

Reference Books:

1) Cloud Computing - Principles and Paradigm, Buyya R, Broberg J, Goscinski A, , Wiley, 2011
2) Cloud Computing: Concepts, Technology & Architecture by Zaigham Mahmood, Ricardo Puttini, Thomas Erl,2013
3) The Basics of Cloud Computing: Understanding the Fundamentals of Cloud Computing in Theory and Practice by Derrick Rountree and Ileana Castrillo,2013

Syllabus for Unit Tests:

Unit Test -1 Unit – I, Unit – II, Unit – III

Unit Test -2 Unit – IV, Unit –V, Unit - VI

B. Tech (All Programmes) – 2023 Course

Rules and Regulations

(I) Theory

(A) Theory Examination

Theory examination consists of: (i) End semester examination (ESE), and (ii) Internal assessment (IA).

(i) ESE is of 60 marks for theory courses.

(ii) IA is of 40 marks. Out of 40 marks, 20 marks will be for Unit Tests and 20 marks will be for Project Based Learning for a given course. Two Unit Tests, each of 20 marks, will be conducted. Average of marks obtained in these two unit tests will be considered as UT marks. Roll numbers allotted to the students shall be the examination numbers for the conduction of unit tests.

(B) Standard of Passing

(i) There is a separate passing of 40% of 60 marks, i.e. 24 marks, for ESE for a given course.

(ii) There is a separate passing of 40% of 40 marks, i.e. 16, for IA for a given course.

(iii) A student who fails at ESE in a given course has to reappear only at ESE as a backlog student and clear the head of passing. Similarly, a student who fails at IA in a given course has to reappear only at IA as a backlog student and clear the head of passing

(II) Practical

(A) Practical Examination

Practical examination consists of: (i) Term work, and (ii) Practical/Oral examination for a given course based on term work.

(i) Term work (TW): TW marks are as mentioned in the curriculum structure.

(ii) Practical/Oral (PR/OR): PR/OR marks are as mentioned in the curriculum structure.

(B) Conduction of practical/oral examination

(i) A student will be permitted to appear for practical/oral examination only if he/she submits term work of a given course.

(ii) Practical/oral examination shall be conducted in the presence of internal and external examiners appointed by university.

(B) Standard of Passing

(i) A student shall pass both heads TW and PR/OR separately with minimum 40% of total marks of respective head.

(III) MOOC and Social Activity Course

(i) If a student completes one MOOC during a programme, he/ she will earn additional TWO credits, subjected to submission of the certificate of completion of the respective course. It is mandatory for a student to complete atleast two MOOC to obtain degree in a given discipline. Students shall register to MOOCs which are offered by any one the following agencies:

(a) SWAYAM : www.swayam.gov.in
(b) NPTEL : www.onlinecourse.nptel.ac.in
(c) Course Era : www.coursera.org
(d) edX online learning : www.edx.org
(e) MIT Open Course ware : www.ocw.mit.edu
(f) Udemy : www.udemy.com
(g) Spoken tutorial : www.spoken-tutorial.org

(ii) If a student completes social activity, he/she will earn additional TWO credits, subjected to submission of the certificate of completion of the respective course/ activity from the relevant authorities. It is mandatory for a student to complete atleast one social activities to obtain degree in a given discipline.

(iv) The additional credits for MOOC and Social Activity will be given only after verification of the authentic document by the Head of the Department and a separate mark-sheet will be submitted by the Head of the Department along with the course examiner

(IV) A. T. K. T

(i) A student who is granted term for B. Tech. Semester-I, III, V, VII will be allowed to keep term for his/her B. Tech. Semester-II, IV, VI, VIII examination, respectively even if he/she appears and fails or does not appear at B. Tech. Semester-I,III, V, VII examination respectively.

(ii) A student shall be allowed to keep term for the B. Tech. Semester-III course if he/she has a backlog of any number of Heads of passing at B. Tech. Semester-I & II taken together.

(iii) A student shall be allowed to keep term for the B. Tech. Semester-V of respective course if he/she has no backlog of B. Tech. Semester-I & II and he/she has a backlog of any number of Heads of passing at B. Tech. Semester-III & IV taken together.

(iv) A student shall be allowed to keep term for the B. Tech. Semester- VII of respective course if he/she has no backlog of B. Tech. Semester-I, II, III, IV and he/she has a backlog of any number of Heads of passing at B. Tech. Semester-V & VI taken together.

(V) Grade Point, Grade Letter and Equivalent Marks

The student must obtain a minimum Grade Point of 5.0 (40% marks) in ESE and also in combined ESE + IA. A student who fails in ESE of a course has to reappear only to ESE as a backlog student and clear that head of passing.

Award of the Class for the Degree considering CGPA: A student who has completed the minimum credits specified for the programme shall be declared to be passed in the programme. The CGPA will be computed every year of all the courses of that year. The grade will be awarded according to the CGPA of every year.

Range of CGPA	**Final Grade**	**Performance Descriptor**	**Equivalent range of Marks (%)**
9.50 ≤ CGPA ≤10.00	O	Outstanding	80 ≤ Marks ≤ 100
9.00 ≤ CGPA ≤ 9.49	A+	Excellent	70 < Marks <80
8.00 ≤ CGPA ≤ 8.99	A	Very Good	60 < Marks < 70
7.00 ≤ CGPA ≤ 7.99	B+	Good	55 < Marks < 60
6.00 ≤ CGPA ≤ 6.99	B	Average	50 < Marks < 55
5.00 ≤ CGPA ≤ 5.99	C	Satisfactory	40 ≤ Marks < 50
CGPA below 5.00	F	Fail	Marks Below 40

(VI) Minor Programme

(i) A students shall receive a MINOR degree when he/she acquire additional 20 credits in a given specialization defined by the UG programmes offered at the institute.

(ii) The theory and practical/oral components for a given course are mentioned in curriculum structure. The theory and examination for a given course are mentioned in Section I and II.

(iii) The grade point, grade letter and equivalent marks system for MINOR programme is mentioned in Section V.

(iv) The MINOR DEGREE programme is OPTIONAL. The interested students may opt MINOR programme.

(v) A student shall complete the MINOR program prior to his/her graduation.