DEPARTMENT OF COMPUTER APPLICATION

SCHEME OF INSTRUCTION AND SYLLABUS

Computer Application

Academic Year: 2024-25

Department of Computer Application

United University

Rawatpur-Jhalwa (Prayagraj)
Uttar Pradesh

DEPARTMENT OF COMPUTER APPLICATION

University Vision

"To establish a Value based Global University having dynamic learning environment encouraging creativity and innovation, research inspired experimental learning and focusing on topics that are pertinent to the development of the region, the Country and the World."

University Vision

- To provide a dynamic, inspiring, and varied learning environment with global exposure.
- To position the institution as a premier hub for research and experiential learning.
- To develop into an adaptable university meeting the demands of society and business.
- To incorporate Value thinking, integrity, wisdom and passion in professional for their career and life

DEPARTMENT OF COMPUTER APPLICATION

Department Vision

The Vision of the Department of Computer Applications is to be a trailblazing institution that plays a transformative role in the nation's progress by producing exceptional human resources in information technology and related fields, meeting the dynamic demands of the country's IT industry for sustainable development. We envision driving cutting-edge research, advancing the frontiers of computer science and engineering, and making ground-breaking contributions through high-impact research publications and enduring patents. Embracing our social responsibility, we are dedicated to serving the local and national communities, fostering awareness of IT-related products, and emphasizing the critical significance of knowledge management. By nurturing a culture of innovation, inclusivity, and ethical leadership, we strive to shape a brighter future and create a positive and lasting impact on society and the ever-evolving technological landscape.

Department Mission

The Department of Computer Applications is committed to attain excellence in education, research, and service. We aim to produce highly skilled and motivated graduates through a comprehensive curriculum that fosters problem-solving abilities, teamwork, and a deep understanding of theory and practical applications. Our passion for research drives us to explore fundamental principles and innovative technologies, both within computer science and interdisciplinary fields.

Additionally, we actively serve our communities at local and national levels, while upholding ethical responsibilities to our profession and society. By nurturing a culture of innovation and entrepreneurship, we empower our students to become visionary leaders, driving positive change and making a lasting impact on the ever-evolving world of technology and beyond.

DEPARTMENT OF COMPUTER APPLICATION

Program Outcomes

On successful completion of the BCA programme the student will be able to:

PO1: Computational Knowledge: Understand and apply foundational mathematical concepts, computing principles, and domain-specific knowledge to conceptualize computing models for solving defined problems.

PO2: Problem Analysis: Ability to identify, analyze, and formulate complex computing problems using fundamentals of computer science and application domains relevant to the BCA field.

PO3: Design / Development of Solutions: Transform complex business scenarios and contemporary issues into well-defined problems, and propose integrated solutions using emerging technologies and tools appropriate for the BCA domain.

PO4: Conduct Investigations of Computing Problems: Devise and conduct experiments, interpret data, and draw conclusions to address complex computing problems in the context of BCA applications.

PO5: Modern Tool Usage: Select and utilize modern computing tools, techniques, and skills necessary for developing innovative software solutions in the BCA field.

PO6: Professional Ethics: Apply and commit to professional ethics and cyber regulations relevant to the BCA profession, understanding the importance of integrity, privacy, and security.

PO7: Life-long Learning: Recognize the significance of continuous learning and development, staying updated with advancements in the BCA domain throughout their professional careers.

PO8: Project Management: Understand and apply project management principles with computing knowledge to manage projects related to BCA applications in multidisciplinary environments.

PO9: Communication Efficacy: Communicate effectively with the computing community and society, demonstrating the ability to comprehend and present technical concepts in clear and accessible ways.

PO10: Societal & Environmental Concern: Recognize the impact of computer technology on society, including economic, environmental, and social aspects, and adhere to ethical practices and responsibilities in BCA professional practice.

PO11: Individual & Team Work: Work proficiently as a member or leader in diverse teams, promoting collaborative problem-solving in the context of BCA projects and applications.

PO12: Innovation and Entrepreneurship: Identify opportunities, foster an entrepreneurial vision, and use innovative ideas to create value and contribute to the betterment of individuals and society, leveraging BCA skills and knowledge.

DEPARTMENT OF COMPUTER APPLICATION

Program Specific Outcomes

PSO1:

Explore technical comprehension in varied areas of Computer Applications and experience a conducive environment in cultivating skills for thriving career and higher studies.

PSO2:

Comprehend, explore and build up computer programs in the allied areas like Algorithms, System Software, Multimedia, Web Design and Data Analytics for efficient design of computer-based systems of varying complexity.

PSO3:

Demonstrate a clear understanding of both conceptual and application-oriented skills in computer applications within a business context, effectively applying these skills to design, develop, and implement practical solutions that meet organizational needs.

DEPARTMENT OF COMPUTER APPLICATION

SCHEME OF INSTRUCTION

COURSE CATEGORY ABBREVIATIONS

1. Professional Core (PC)
2. Professional Elective (PE)
3. Open Elective (OE)
4. Basic Sciences (BS)
5. Humanities and Social Sciences (HS)
6. Project Work, Seminar, Internship (PWSI)
7. Mandatory Audit Courses (AU)

DEPARTMENT OF COMPUTER APPLICATION

COURSE STRUCTURE

Bachelor of Computer Application (BCA-IBM)

3 Year (6 Semester) Programme

[Academic Session 2024-25 onwards]

Semester I

				Contact Hours			21
S. No.	Course Code	Course Category	Course Name	L	T	P	C
1	CAUCBC106T	PC	Problem Solving Techniques	3	-	-	3
2	CAUCBC114T	PC	Python Programming	3	-	-	3
3	CAUCBC104T	PC	Digital Logic Design	4	-	-	4
4	CMSFPCA11T	HS	Business Communications	3	-	-	3
5	SCSEPCA11T	BS	Environmental Science	2	-	-	2
6	CAUCBC106P	PC	Problem Solving Techniques Lab	-	-	2	1
7	CAUCBC114P	PC	Python Programming Lab	-	-	4	2
8	CAUCBC104P	PC	Digital Logic Design Lab	-	-	2	1
9	PTSPPCA12T	HS	Professional Proficiency	1	-	2	2
				16	**-**	**10**	**21**

DEPARTMENT OF COMPUTER APPLICATION

Semester II

				Contact Hours			26
S. No.	Course Code	Course Category	Course Name	L	T	P	C
1	CAUCBC212T	PC	Data Structure	3	-	-	3
2	CAUCBC211T	PC	Computer Organization & Architecture	4	-	-	4
3	CAUCBC203T	PC	Database Management System	4	-	-	4
4	SCSMPCA22T	BS	Basic Mathematics	4	-	-	4
5	CAUCBC212P	PC	Data Structure Lab	-	-	4	2
6	CAUCBC211P	PC	Computer Organization Lab	-	-	2	1
7	CAUCBC203P	PC	Database Management System Lab	-	-	2	1
8	PTSPPCA22T	HS	Professional Proficiency	1	-	2	2
Internship (3-4 weeks) shall be conducted during summer break after II and IV semester and subsequently assessed in III and V semester respectively.				**16**	**-**	**10**	**21**

DEPARTMENT OF COMPUTER APPLICATION

Semester III

				Contact Hours			30
S. No.	Course Code	Course Category	Course Name	L	T	P	C
1	**CAUCBC301T**	PC	Object Oriented Programming	4	-	-	4
2	**CAUCBC306T**	PC	Web Design	2	-	-	2
3	**CAUCBC411T**	PC	Software Engineering	3	-	-	3
4	**CAUCBC312T**	PC	Computer Networks	4	-	-	4
5	**CAUCBC301P**	PC	Object Oriented Programming Lab	-	-	2	1
6	C**AUCBC306P**	PC	Web Design Lab	-	-	4	2
7	**CAUCBC411P**	PC	Software Engineering Lab	-	-	2	1
8	**CAUCBC312P**	PC	Computer Networks Lab	-	-	2	1
9	**CAUCBC314P**	PWSI	Internship Assessment (B.C.A.) - I	-	-	4	4
10	**PTSPPCA31T**	HS	Professional Proficiency	1	-	2	2
11	**TTCS**	AU	Technical Training	2	-	-	0
				16	**-**	**16**	**24**

DEPARTMENT OF COMPUTER APPLICATION

Semester IV

				Contact Hours			24
S. No.	Course Code	Course Category	Course Name	L	T	P	C
1	**CAUCBC402T**	PC	Operating System	4	-	-	4
2	**CAUCBC403T**	PC	Design and Analysis of Algorithms	4	-	-	4
3		PC	Advanced Web Design	2	-	-	2
4		PE	Professional Elective-I	3	-	-	3
5	**CAUCBC402P**	PC	Operating System Lab	-	-	2	1
6	**CAUCBC403P**	PC	Design and Analysis of Algorithms Lab	-	-	2	1
7		-	Advanced Web Design Lab	-	-	4	2
8	PPS	PE	Professional Elective-I Lab	-	-	2	1
9	**CAUCBC412P**	PWSI	Mini Project (B.C.A.)	-	-	4	4
10		HS	Professional Proficiency	1	-	2	2
11		AU	Technical Training	2	-	-	0
				16	**-**	**16**	**24**

DEPARTMENT OF COMPUTER APPLICATION

Semester V

				Contact Hours			24
S. No.	Course Code	Course Category	Course Name	L	T	P	C
1	CAUCBC502T	PC	Machine Learning	3	-	-	3
2		PE	Professional Elective-II	3	-	-	3
3		PE	Professional Elective-III	3	-	-	3
4		OE	Open Elective - I	3	-	-	3
5	CSPEAI002P	PC	Machine Learning Lab	-	-	2	1
6	CAUCBC501P	PE	Professional Elective-II Lab	-	-	2	1
7	CSPEDS002P	PE	Professional Elective-III Lab	-	-	2	1
8		PWSI	Internship Assessment (B.C.A.) - II	-	-	4	4
		HS	Professional Proficiency				
9	PTSPPCA50T	AU	Technical Training	1	-	2	2
							21

DEPARTMENT OF COMPUTER APPLICATION

Semester VI

				Contact Hours			24
S. No.	Course Code	Course Category	Course Name	L	T	P	C
1		PE	Professional Elective-IV	3	-	-	3
2		OE	Open Elective –II	3	-	-	3
3		PE	Professional Elective-IV Lab	-	-	2	1
4	CAUCBC601P	PWSI	Major Project (B.C.A.)	-	-	14	14
		AU	Technical Training	2	-	2	0
				8	**-**	**18**	**21**

Note: Students will identify the project problem from their chosen specialization or interdisciplinary in nature as Project based on Product Design Technology.

COURSE CODE & NAME: CAUCBC106T / Problem Solving Techniques

COURSE OUTCOMES

1. Understanding the evolution of programming languages and differentiate between machine level, assembly, and higher-level languages
2. Apply program design techniques using hierarchy charts and express program logic through flowcharts and pseudocode.
3. Analyze algorithms for various problem-solving scenarios, including input-output statements, decision-making, and looping statements.
4. Evaluate the effectiveness of debugging techniques by identifying and categorizing different types of errors (syntax, semantic, and runtime).
5. Create algorithms and implement solutions for complex problems involving arithmetic operations, arrays, and series patterns

UNIT I:

Computer Problem Solving: Evolution of programming, languages – Introduction to machine level language, Assembly language and Higher level languages. Programing Life Cycle , Understanding the Problem Statement, Planning Program design using Hierarchy charts, Expressing Program logic using flowcharts /Pseudocode, Coding using a programing language such as 'C'/'Python', Documenting, Compiling, Debugging and Executing.

UNIT II:

Algorithm Development: Definition, Algorithm: a solution to a problem, Input-Output Statements, Decision Making Statements, Looping Statements, Examples. Flowcharting: Definition, Input-Output Statements, Decision Making Statements, Looping Statements, Module representation, Drawing conventions and standards, Example. Debugging: Bug, errors : syntax ,semantics and runtime, Compilation, Interoperation, Program debugging.

UNIT III:

Logic Building and Problem Solving Based on Number: Addition/ Subtraction/ Multiplication/ Division of two numbers, Power of a Number, Prime Number, Reversing a Number, HCF of two numbers, LCM of two numbers. Perfect No, Factor of a number, Strong number, Perfect number, Auto-morphic number, Harshad number, Abundant number, Friendly pair.

UNIT IV:

Logic Building and Problem Solving Based on Arrays: Largest element in an array, Smallest Element in an Array, Smallest and largest element in an array, Second Smallest Element in an Array, sum of elements in an array, Reverse an Array.

UNIT V:

Logic Building and Problem Solving Based on Series and patterns: Fibonacci Series, Triangular number series, Square number series, Cube number series, Alphabet Triangle, Number Triangle, Fibonacci Triangle

TEXTBOOKS/ REFERENCE BOOKS

1. "Introduction to Programming with C" by David I. Schneider
2. "Python Programming: An Introduction to Computer Science" by John Zelle
3. "Algorithms Unlocked" by Thomas H. Cormen

COURSE CODE & NAME: CAUCBC114T / Python Programming

COURSE OUTCOMES

1. Understanding basic programming skills using Python programming language.
2. Understanding the notion of data types and complex data types such as lists, tuples etc.
3. Understanding the concept of decision making and iterative control structure in python.
4. Understanding the concepts of functions and file handling in Python.

UNIT I:

Introduction to Python Language: Introduction to Python: Python variables, Python basic Operators, Understanding python blocks. Python Data Types, Declaring and using Numeric data types: int, float etc.

UNIT II:

Control Structures: Python Program Flow Control Conditional blocks: if, else and else if, Simple for loops in python, For loop using ranges, string, list and dictionaries. Use of while loops in python, Loop manipulation using pass, continue, break and else. Programming using Python conditional and loop blocks.

UNIT III:

Strings, Lists, Tuples and Dictionaries: Python Complex data types: Using string data type and string operations, Defining list and list slicing, Use of Tuple data type. String, List and Dictionary, Manipulations Building blocks of python programs, string

manipulation methods, List manipulation. Dictionary manipulation, Programming using string, list and dictionary in-built functions. Python Functions, Organizing python codes using functions.

UNIT IV:

Functions & Modules: Defining a function, Calling a function, Types of functions, Function Arguments, Anonymous functions, Global and local variables. Importing module, Math module, Packages and their composition

UNIT V:

File Handling: Python File Operations: Reading files, Writing files in python, Understanding read functions, read(), readline(), readlines(). Understanding write functions, write() and writelines() Manipulating file pointer using seek Programming, using file operations.

TEXTBOOKS/ REFERENCE BOOKS

1. R Nageswar Rao, Core Python Programming, 2018.
2. Eric Mathews, Python Crash Course, 2019.
3. Practical Programming: An introduction to Computer Science Using Python, second edition, Paul Gries, Jennifer Campbell, Jason Montojo, The Pragmatic Bookshelf.
4. Exploring Python, Timothy A. Budd, Mc Graw Hill Education

COURSE CODE & NAME: CAUCBC104T / Digital Logic Design

COURSE OUTCOMES

1. Understand and apply number systems and logic gates in digital circuits.
2. Analyze and evaluate Boolean functions using minimization techniques.
3. Design and construct combinational logic circuits with decoders and multiplexers.
4. Synthesize and compare sequential logic circuits using various flip-flops.
5. Understand and differentiate between digital integrated circuit logic families.

UNIT I:

Number system and Logic Gates: Introduction of number systems, Radix, Radix Interco versions. Radix Complement, Diminished radix complement

UNIT II:

Boolean algebra: Basic theorem of Boolean algebra. Boolean function and minimization, Karnaugh map Universal Gates, Realization of Primary gates using Universal gates only. Minterm and Maxterm Realization of Boolean Functions, Gate-level minimization: The map method up to four variables, don't care conditions, SOP and POS simplification, Quine Mc- Cluskey Method.

UNIT III:

Combinational logic circuits: Binary adder and Subtractor circuits, Magnitude comparator, Decoders, Encoders, Multiplexer, and demultiplexer, Realization of switching expressions by decoders, encoders, multiplexer and Demultiplexer.

UNIT IV:

Sequential Logic Circuits: Sequential circuits, latches, and Flip Flops, difference between latch and flip flop, SR flip flop, JK flip flop, Master Slave flip flop, comparison.

UNIT V:

Digital Integrated Circuits: Characteristics of digital ICs, Introduction to logic families-RTL, DTL, TTL, ECL. MOS and CMOS circuits and comparison

TEXTBOOKS/ REFERENCE BOOKS

1. Digital Design: M. Morris Mario (PHI)
2. Digital circuits & logic design: S. C. Lee (PHI)
3. Digital electronics: W. H. Gothmann (PHI)
4. Switching theory: A. K. Gautam (Katsons)
5. R.P. Jain, "Modern Digital Electronics," Tata McGraw Hill, 4th edition, 2009.
6. A. Anand Kumar, "Fundamental of Digital Circuits," PHI 4th edition, 2018.
7. W. H. Gothmann, "Digital Electronics- An Introduction to Theory and practice," PHI, 2Nd edition, 2006.
8. D.V. Hall, "Digital Circuits and Systems," Tata McGraw Hill, 1989.

9. A. K. Singh, “Foundation of Digital Electronics & Logic Design,” New Age Int. Publishers.
10. Subrata Ghosal, “Digital Electronics,” Cengage publication, 2nd edition, 2018

COURSE CODE & NAME: CMSFPCA11T / Business Communications

COURSE OUTCOMES

1. Understanding the evolution of programming languages and differentiate between machine level, assembly, and higher-level languages
2. Apply program design techniques using hierarchy charts and express program logic through flowcharts and pseudocode.
3. Analyze algorithms for various problem-solving scenarios, including input-output statements, decision-making, and looping statements.
4. Evaluate the effectiveness of debugging techniques by identifying and categorizing different types of errors (syntax, semantic, and runtime).
5. Create algorithms and implement solutions for complex problems involving arithmetic operations, arrays, and series patterns

UNIT I:

Business Communication: Introduction, Role of Communication In Business, Definitions of Communication, Purpose Of Communication, Communication Situation, Communication Process, Forms Of Communication (Formal & Grapevine), Barriers Of Communication, Seven Cs Of Communication.

UNIT II:

Oral Communication: Oral Communication, Advantages of Oral Communication, Limitations of Oral Communication, Two Sides of Oral Communication, Principles Of Effective Communication, Effective Listening, Non-verbal Communication

UNIT III:

Written Communication: Written Communication, Purpose of Writing, Principles of Effective Writing, Writing Techniques, Electronic Writing Process

UNIT IV:

Business Letters and Report Writing: Business Letters and Its Need, Types of Letter, Structure of Business Letter, Form of Letters, Report Writing, Types Of Business Reports, Characteristics and Purpose of a Good Report, Guiding Principles of Writing A Report, Preparing A Report, Structure of A Report

UNIT V:

Presentation Skill: Presentation, Elements Of Presentation, Designing A Presentation, Using Visual Aids, Appearance And Posture, Tips For An Effective Presentation Communication and Technology: The Role Of Technological Advancement, Communication Network, Intranet, Internet, E-mails, Teleconferencing, Videoconferencing.

TEXTBOOKS/ REFERENCE BOOKS

1. Vikram Bisen and Priya, “Business Communication”, New Age International Publishers.
2. B. M. Shaikh, “Business Communication”, Vision Publication.

3. Urmila Rai and S. M. Rai, “Business Communications”, Himalaya Publication House.
4. Asha Kaul, “Effective Business Communications”, PHI Learning private Ltd.
5. Dr. Anjali P. Kalkar,” Business Communications”, Success Publications.

COURSE CODE & NAME: SCSEPCA11T/ Environmental Science

COURSE OUTCOMES

1. Comprehend the importance of ecosystem and sustainable
2. Demonstrate interdisciplinary nature of environmental issues
3. Identify different types of environmental pollution and control measures.
4. Adopt cleaner productive technologies
5. Identify the role of non-conventional energy resources in environmental protection.
6. Analyse the impact of human activities on the environment

UNIT I:

Introduction to Environmental Studies: Multidisciplinary nature of environmental studies; Scope and importance; Concept of sustainability and sustainable development. Ecosystems: Structure and function of ecosystem; Energy flow in an ecosystem: food chains, food webs and ecological pyramids. Nutrient cycle (carbon cycle, nitrogen cycle, Sulphur cycle, water cycle, oxygen cycle).

UNIT II:

Renewable and non-renewable energy resources, Land resources and land use change; Land degradation, soil erosion and desertification. Deforestation: Causes and impact due to mining dam building on environment. Flood and drought.

UNIT III:

Environmental Pollution: air pollution, water pollution, thermal pollution, noise pollution, soil pollution; Solid Waste Management; Environmental Impact Assessment.

UNIT IV:

Biodiversity and Conservation: Levels of biological diversity: genetic, species and ecosystem diversity; hot spots; threats to biodiversity; Conservation of biodiversity: in-situ and ex -situ conservation of biodiversity.

UNIT V:

Impact of energy usage on environment: Global warming, Climate change, Depletion of ozone layer, Acid rain. Environmental ethics, Role of NGOs, Environmental Laws: Environment Protection Act. Air (Prevention and Control of Pollution) Act. Water (Prevention and control of Pollution) Act. Wildlife Protection. Act. Forest Conservation Act.

TEXTBOOKS/ REFERENCE BOOKS

1. Environmental and Pollution Awareness by Sharma BR; Satya Prakashan, New Delhi.
2. Environmental Protection Law and Policy in India by Thakur Kailash; Deep and Deep Publications, New Delhi.
3. Environmental Pollution by Dr. RK Khitoliya; S Chand Publishing, New Delhi
4. Environmental Science by Deswal and Deswal; Dhanpat Rai and Co. (P) Ltd. Delhi.

COURSE CODE & NAME: PTSPPCA12T/ Professional Proficiency

COURSE OUTCOMES

1. Identify and analyze basic grammar elements, including tenses and subject-verb agreement, through reading comprehensions.
2. Apply grammatical concepts in constructing coherent sentences and engaging in daily conversations effectively.
3. Demonstrate effective self-introduction and body language skills to enhance communication confidence in professional settings.
4. Create professional documents such as emails, letters, and resumes, applying appropriate formatting and language conventions.
5. Evaluate and solve quantitative aptitude problems, including simplification and coding-decoding, to enhance logical reasoning skills.

UNIT I:

Hard Skills: Basic Grammar-Tenses, Subject-Verb- Agreement, Article, formation of sentences. Introduction to technical terms.

UNIT II:

Soft Skills: Self Introduction, body language, Voice modulation

UNIT III:

Professional training: E-mail writing, Letter writing, Minutes writing, Resume building

UNIT IV:

Quantitative Aptitude & Logical Reasoning: Simplification & Approximation, Alphanumeric Series & Miscellaneous, Coding-Decoding

TEXTBOOKS/ REFERENCE BOOKS

1. Advance Maths- RakeshYadav
2. Verbal and Non-Verbal Reasoning- R.S Agarwal
3. English Grammar, Composition and Usage by N.K. Agrawal & F.T. Wood, Macmillan India Ltd., New Delhi.
4. English Grammar & Composition by Wren & Martin, S.Chand & Co. Ltd., New Delhi.

COURSE CODE & NAME: CAUCBC212T/ Data Structure

COURSE OUTCOMES

1. Understand and apply fundamental data structures and algorithms for efficient data organization.
2. Analyze and evaluate the efficiency of algorithms and data structures through complexity analysis.
3. Design and implement algorithms for data manipulation using various data structures.
4. Synthesize and optimize searching and sorting techniques for effective data retrieval.
5. Demonstrate proficiency in implementing graph algorithms and understanding their applications in problem-solving.

UNIT I:

Introduction to Data Structure: Introduction: Basic Terminology, Elementary Data Organization, Algorithm, Efficiency of an Algorithm, Abstract Data Types (ADT)

Arrays: Definition, Single and Multidimensional Arrays, Representation of Arrays: Row Major Order, and Column Major Order, Application of arrays, Sparse Matrices and their representations.

Linked lists: Array Implementation and Dynamic Implementation of Singly Linked Lists, Doubly Linked List, Circularly Linked List, Operations on a Linked List. Insertion, Deletion, Traversal, Polynomial Representation and Addition, Generalized Linked List.

UNIT II:

Stacks: Primitive Stack operations: Push & Pop, Array and Linked Implementation of Stack in C, Application of stack: Prefix and Postfix Expressions, Evaluation of postfix expression, Recursion, Tower of Hanoi Problem, Simulating Recursion, Principles of recursion, Tail recursion, Removal of recursion Queues,

Operations on Queue: Create, Add, Delete, Full and Empty, Circular queues, Array and linked implementation of queues in C, Dequeue and Priority Queue.

UNIT III:

Trees: Basic terminology, Binary Trees, Binary Tree Representation: Array Representation and Dynamic Representation, Complete Binary Tree, Algebraic Expressions, Extended Binary Trees, Array and Linked Representation of Binary trees, Tree Traversal algorithms: Inorder, Preorder and Postorder.

UNIT IV:

Graphs: Terminology, Sequential and linked Representations of Graphs: Adjacency Matrices, Adjacency List, Adjacency Multi list, Graph Traversal: Depth First Search and Breadth First Search, Connected Component, Spanning Trees, Minimum Cost Spanning Trees: Prims and Kruskal algorithm. Transitive Closure and Shortest Path algorithm: Warshal Algorithm and Dijikstra Algorithm

UNIT V:

Searching: Sequential search, Binary Search, Comparison and Analysis Internal Sorting: Insertion Sort, Selection, Bubble Sort, Quick Sort, Two Way Merge Sort, Heap Sort, Radix Sort, Practical consideration for Internal Sorting. Search Trees: Binary Search Trees (BST), Insertion and Deletion in BST, Complexity of Search Algorithm, AVL trees, Introduction to m-way Search Trees, B Trees. Hashing: Hash Function, Collision Resolution Strategies Storage Management: Garbage Collection and Compaction.

TEXTBOOKS/ REFERENCE BOOKS

1. Aaron M. Tenenbaum, YedidyahLangsam and Moshe J. Augenstein, "Data Structures Using C and C++", PHI Learning Private Limited, Delhi India
2. Horowitz and Sahani, "Fundamentals of Data Structures", Galgotia Publications Pvt Ltd Delhi India.
3. Lipschutz, "Data Structures" Schaum's Outline Series, Tata McGraw-hill Education (India) Pvt. Ltd.
4. Thareja, "Data Structure Using C" Oxford Higher Education.

COURSE CODE & NAME: CAUCBC211T/ Computer Organization & Architecture

COURSE OUTCOMES

1. Analyze the internal organization of a computer system, including CPU functional units and data representation methods.
2. Apply arithmetic operations and instruction execution techniques in basic computer organization, including ALU operations and control cycles.
3. Evaluate different CPU architectures and their characteristics, including CISC and RISC, alongside data transfer and manipulation techniques.
4. Design and differentiate between hardwired and micro-programmed control units, and assess memory organization and hierarchy in computer systems.
5. Investigate input-output organization, including various modes of data transfer and the characteristics of multiprocessor systems.

UNIT I:

Introduction to Computers: Basic of Computer, internal organization of CPU, Functional Units of Digital System and Interconnection, Software, Basic Operational Concepts, Data Representation, Fixed-Point Representation, Floating-Point Representation, Bus Architecture, Types of Buses, Bus Arbitration, Registers, Processor Organization, General Registers Organization, Stack Organization, Addressing Modes.

UNIT II:

ALU and Micro Operations: Multiplication: Signed Operand Multiplication, Booths Algorithm and Array Multiplier. Division and Logic Operations. Floating Point Arithmetic Operation, Basic Computer Organization: Instruction Codes, Computer Registers, Computer Instructions, Timing and Control, Instruction Cycle, Memory-Reference Instructions, Input-Output and Interrupt.

UNIT III:

Central Processing Unit: Introduction, General Register Organization, Stack Organization, Instruction Formats, Addressing Modes with Numerical Examples, Data Transfer and Manipulation, Program Control, Program Interrupt, Types of Interrupts, CISC Characteristics, RISC Characteristics. Introduction to Parallel Processing, Pipelining

UNIT IV:

Control Design: Hardwired & Micro Programmed (Control Unit), Control Memory, Micro program Example.

Memory Organization: Basic concept of memory and hierarchy, semiconductor RAM memories, 2D & 2 1/2D memory organization. Memory Hierarchy, Main Memory, Auxiliary Memory, Associative Memory, Cache Memory, Virtual Memory.

UNIT V:

Input-Output Organization: Peripheral Devices, Input-Output Interface, Asynchronous Data Transfer, Modes of Transfer: Programmed I/O, Interrupt-Initiated I/O, Direct Memory Access, Priority Interrupt.

Multiprocessors: Introduction, Characteristics or Multiprocessors, Interconnection Structures, Inter Processor Arbitration, Serial Communication: Synchronous and Asynchronous Communication.

TEXTBOOKS/ REFERENCE BOOKS

1. D. A. Patterson and J. L. Hennessy, Computer Organisation and Design: The Hardware/Software Interface, 5/e, Morgan Kaufmann, 2014.
2. J. L Hennessy and D. A. Patterson, A. Computer Architecture: A Quantitative approach, 6/e, Morgan Kaufmann, 2017.
3. V. P. Heuring and H. F. Jordan, Computer System Design and Architecture, Prentice Hall, 2003.
4. D. A. Patterson and J. L. Hennessy,Computer Organisation and Design: The Hardware/ Software Interface, 5/e, Harcourt Asia Pte Ltd (Morgan Kaufman), 2014

COURSE CODE & NAME: CAUCBC203T/ Database Management System

COURSE OUTCOMES

1. Understand database concepts, structures, query language, E R model and relational model.
2. To design and build a simple database system and demonstrate competence with the fundamental tasks involved with modelling, designing, and implementing a DBMS.
3. Understand Functional Dependency and Functional Decomposition.
4. Apply various Normalization techniques.
5. Perform PL/SQL programming using concept of Cursor Management, Error Handling, Package and Triggers.
6. Understand the principles of storage structure and recovery management.

UNIT I:

Introduction: Data, information and knowledge, Characteristics of database approach, Data independence, Architecture of database system, Data dictionary, Types of database language, database system life cycle, Overview of hierarchical, network and relational model. Relations and Codd's rules, Concepts of keys, Relation Algebra - Select, Project, Joins, Set operations, Update operations — tuple relational calculus, Relational Calculus vs. relational algebra. Data definition, date manipulation, view definition, nested queries, updation, Embedded SQL, Handling of nulls and cursors.

UNIT II:

Data Models: Conceptual, Logical and Physical design, ER models, ER diagrams, Strong and weak entity sets. Generalization. Specialization and Aggregation, Conversion of ER model into relational schemas.

UNIT III:

Normalization: Normalization concepts, Functional dependencies and dependency preservation, Normal forms 1NF, 2NF, 3NF, BCNF, 4NF, 5NF, DKNF, Indexing, file organization, De-normalization, Clustering of tables and indexes.

UNIT IV:

Transaction Handling: Transaction recovery, System recovery, two phase commit. Concurrency problems, locking, deadlocks, security, discretionary and mandatory access control, data encryption

UNIT V:

Distributed databases: Overview of query processing, concurrency control and recovery in distributed databases, overview of client/server architecture and its relationship with distributed databases, performance benchmark and performance tuning of databases

TEXTBOOKS/ REFERENCE BOOKS

1. Date C J, "An Introduction To Database System", Addision Wesley
2. Korth, Silbertz, Sudarshan, "Database Concepts", Tata Mcgraw-hill Education (India) Pvt. Ltd.
3. Elmasri, Navathe, "Fundamentals Of Database Systems", Pearson Education New Delhi India.

4. Bipin C. Desai, “An introduction to Database Systems”, Galgotia Publication Pvt. Ltd. New Delhi.
5. Majumdar & Bhattacharya, “Database Management System”, Tata Mcgraw-hill Education (India) Pvt. Ltd.
6. G. K. Gupta, “Database Management System”, Tata Mcgraw-hill Education (India) Pvt. Ltd.
7. Ramakrishnan, Gehrke, “Database Management System”, McGraw Hill (India) Pvt Ltd. New Delhi.
8. II Chakravarti, “Advanced Database Management System” Wiley Dreamtech Publications.

COURSE CODE & NAME: SCSMPCA22T/ Basic Mathematics

COURSE OUTCOMES

1. Understand the foundations of mathematics.
2. Be able to perform basic computations in higher mathematics.
3. Be able to write and understand basic proofs.
4. Use mathematical ideas to model real-world problems.

UNIT I:

Matrices: Definition, Types of Matrices, Addition, Subtraction, Scalar Multiplication and Multiplication of Matrices, Adjoint, Inverse, Cramers Rule, Rank of Matrix, Eigen Vectors of a Matrix, Caley-Hamilton Theorem (without proof).

UNIT II:

Limit, Continuity & Complex Number: Limit at a Point, Properties of Limit, Computation of Limits of Various Types of Functions, Continuity and Discontinuities at a Point, Continuity Over an Interval. Complex Numbers: Modules, Argument of complex number, Polar form, Complex conjugate, Algebraic operations, Roots of a complex number.

UNIT III:

Differentiation: Derivative, Derivatives of Sum, Differences, Product & Quotients, Chain Rule, Derivatives of Composite Functions, Logarithmic Differentiation, Rolle's Theorem, Mean Value Theorem, Maxima & Minima, Successive Differentiation & Liebnitz Theorem.

UNIT IV:

Integration: Indefinite integrals, Basic formulae. Integration by parts, Integration by substitution, Definite integrals. Properties of definite integrals, Evaluation of double integration & triple integration, Application of definite integral to find Area and Volume.

UNIT V:

Vector Algebra: Definition of a vector in 2 and 3 Dimensions; Double and Triple Scalar and Vector Product. Dot and cross- product of vectors, Gradient of vectors. Divergence and curl of vectors.

TEXTBOOKS/ REFERENCE BOOKS

1. K. C. Sinha A Text Book of Calculus.
2. K. Sharma A text book of matrix.
3. Vector Analysis: Schaum'S Outlines Series
4. Advanced Engineering Mathematics- Erwin Kreyszig.
5. Calculus: Volume I-Aposto.
6. Higher Engineering Mathematics, B S Grewal.

COURSE CODE & NAME: PTSPPCA22T/ Professional Proficiency

COURSE OUTCOMES

1. Identify and analyze basic grammar elements, including tenses and subject-verb agreement, through reading comprehensions.
2. Apply grammatical concepts in constructing coherent sentences and engaging in daily conversations effectively.
3. Demonstrate effective self-introduction and body language skills to enhance communication confidence in professional settings.
4. Create professional documents such as emails, letters, and resumes, applying appropriate formatting and language conventions.
5. Evaluate and solve quantitative aptitude problems, including simplification and coding-decoding, to enhance logical reasoning skills.

UNIT I:

Hard Skills: Transformation of sentences (from affirmative to negative) degree of Adjective ,Preposition, vocabulary .

UNIT II:

Soft Skills: Types of communication 7C's ascent, Tone, Etiquettes, Time management. Management of work & Time , Introduction of different IT'S giants

UNIT III:

Quantitative Aptitude & Logical Reasoning: Analogy and classification, Average, LCM & HCF

TEXTBOOKS/ REFERENCE BOOKS

1. Advance Maths- RakeshYadav
2. Verbal and Non-Verbal Reasoning- R.S Agarwal
3. English Grammar, Composition and Usage by N.K. Agrawal & F.T. Wood, Macmillan India Ltd., New Delhi.
4. English Grammar & Composition by Wren & Martin, S.Chand & Co. Ltd., New Delhi.

COURSE CODE & NAME: CAUCBC301T/ Object Oriented Programming

COURSE OUTCOMES

1. Analyze and apply object-oriented programming principles such as abstraction, encapsulation, inheritance, and polymorphism using Java.
2. Design and implement Java classes and interfaces, demonstrating mastery of packages and multithreading concepts.
3. Evaluate and manage exceptions in Java applications, utilizing effective error handling and multithreading techniques.
4. Develop interactive graphical user interfaces (GUIs) using AWT and Swing, incorporating event handling and layout management.
5. Implement JDBC to connect and interact with databases, executing queries and managing results efficiently.

UNIT I:

OOPS concept & Java Language Basics: Object oriented approach. Basic terms and ideas Abstraction, Encapsulation. Inheritance & Polymorphism. Structured vs. Object Oriented Programming. Benefits of Object oriented programming.

Introduction To Java: Basic features, Java Virtual Machine Concepts, java environment. Primitive data types, tokens, variables constants & Java keywords. Java Operators. Java program structure. A simple Java program. Expressions, Statements, Control Statements, Selection Statements, Iterative Statements, Jump Statements. Arrays, Vector. String handling & wrapper classes.

UNIT II:

OOP Package & Multithreading: Classes and Methods- Implementing & designing classes, constructors, polymorphism & inheritance. Interfaces. Interface: defining Interface, Extending Interface, implementing Interface & Accessing Interface Variable Package: System packages, using system package, import. Adding a class to a package, Hiding classes

UNIT III:

Exception Handling: Concepts of Exceptions, types of exceptions, try; catch & finally keywords, throwing exceptions & nested try and catch. Multithreaded Programming: Life cycle of a Thread, creating Threads, extending Threads class, Stopping& blocking a thread, using thread methods, thread exceptions, thread priority, and synchronization.

UNIT IV:

Graphics & GUI: Working with Windows Graphics and Text. Using AWT Controls, Layout Managers, Event Handling & Menus. Swing based GUI

UNIT V:

JDBC: JDBC- Overview, JDBC implementation, Connection class & Statements. Catching Database Results, handling database Queries.

TEXTBOOKS/ REFERENCE BOOKS

1. E. Balagunisamy. “Programming in Java”, TMH Publications.
2. Java The Complete Reference, Herbert Schildt 7th Edition. Tata McGraw- Hill Edition.
3. S. Horstmann, Gary Cornell – “Core Java 2 Volume II – Advanced Features” Addison Wesley.

COURSE CODE & NAME: CAUCBC306T / Web Design

COURSE OUTCOMES

1. Manipulate elements on a webpage and responding to user interactions
2. Identify the basis of designing a Web site and adding additional functionality using Bootstrap
3. Describe and identify the use of Bootstrap and successfully place it into Web pages.
4. Describe how intended website design features will specifically benefit a target user group content strategy.

UNIT I:

Introduction to Web Design: Introduction of Internet, WWW, Website, Working of Websites, Webpages, Web Server, Front End, Back End, Client and Server Scripting Languages, Responsive Web Designing, Types of Websites (Static and Dynamic Websites), Downloading free Editors like Notepad++, Sublime Text Editor, File creation and editing, saving. Architecture , Http and Https, URL: Relative Path and Absolute Path

UNIT II:

HTML Basics and HTML5: Introduction, Basic Structure of HTML, Head Section and Elements of Head Section, Formatting Tags: Bold, Italic, Underline, Strikethrough, Div, Pre Tag Anchor links and Named Anchors Image Tag, Paragraphs, Comments, Tables: Attributes–(Border, Cellpadding, Cellspacing , height , width), TR, TH, TD, Rowspan, Colspan Lists : Ordered List , Unordered List , Definition List, Forms, Form Elements, Input types, Input Attributes, Text Input Text Area, Dropdown, Radio buttons , Check boxes, Submit and Reset Buttons Frames: Frameset, nested Frames

HTML 5 Introduction, HTML5 New Elements: Section, Nav, Article, Aside, Audio Tag, Video Tag, HTML5 Form Validations: Require Attribute, Pattern Attribute, Autofocus Attribute, email, number type, date type , Range type, HTML embed multimedia, HTML Layout, HTML Iframe

UNIT III:

CSS: Introduction to CSS, Types of CSS, CSS Selectors: Universal Selector, ID selector, Tag Selector, Class Selector, Sub Selector, Attribute Selector, Group Selector, CSS Properties: Back Ground properties, Block Properties, Box properties, List properties, Border Properties, Positioning Properties, CSS Lists CSS Tables, CSS Menu Design CSS Image Gallery, website design using Bootstrap Framework: Colors, Containers, Panels, Border, Fonts, Text, tables, List, Images, Grid

UNIT IV:

JavaScript and Angular JS: Introduction to Client Side Scripting Language, Variables in Java Script, Operators in JS, Conditions Statements, JS Popup Boxes, JS Events, Basic Form Validations in JavaScript. Introduction to Angular JS: Expressions, Modules and Directives.

UNIT V:

CSS Bootstrap Framework and Photo Editor: Colors, Containers, Panels, Border, Fonts, Text, tables, List, Images, Grid Photo Editing Tools: Selection Tools, Paint Tools, Transform Tools, Text Tool, Layers, Brightness/ Contrast, Improve Colors and tone, Filters

TEXTBOOKS/ REFERENCE BOOKS

1. 'The Complete Reference HTML & XHTML' by Thomas Powell, 5th Edition, Tata McGraw-Hill Company Limited.
2. Learning web designing: a beginners guide to HTML, CSS, JavaScript,and web graphics' by Niederst Robbins, 4th Edition, Oreilly Publication
3. Mastering HTML, CSS & JAVAScript Web Publishing' by Laura Lemay, Rafe Coburn, Jennifer Kyrnin, 7th edition, SAMS publication.
4. Web Enabled Commercial Application Development using HTML, JavaScript, DHTML and PHP' by Ivan Bayross, 4th Edition, BPB Publications.

COURSE CODE & NAME: CAUCBC411T/ Software Engineering

COURSE OUTCOMES

1. Apply the principles of the engineering processes in software development.
2. Demonstrate software project management activities such as planning, scheduling and estimation.
3. Model the requirements for the software projects.
4. Design and Test the requirements of the software projects.
5. Implement the software development processes activities from requirements to validation and verification.
6. Apply and evaluate the standards in process and in product.

UNIT I:

Introduction: Introduction and overview of Software Engineering, Software Crisis, Scope and necessity of software engineering, Software Engineering Processes, Software Development Life Cycle (SDLC) model: Water Fall Model, Prototype Model, Spiral Model, Evolutionary Development Models, Iterative Enhancement Models.

UNIT II:

Software Requirement Analysis & Specification: Requirement Engineering, Problem Analysis: Data Flow Diagram, Data Dictionaries, ER Diagram, Approaches to Problem Analysis, SRS Document. Software Quality Assurance (SQA): Verification and Validation, SQA Plans, Software Quality Frameworks, ISO 9000 Models, SEI-CMM Model.

UNIT III:

System Design: Conceptual and Technical Design, Objectives of Design, Modularization, Design Structure Charts, Pseudo Codes, Flow Charts, Coupling and Cohesion Measures, Design Strategies: Function versus Object Oriented Design, Top-Down and Bottom-Up Design.

Software Measurement and Metrics: Various Size Oriented Measures: Halestead's Software Science, Function Point (FP) Based Measures, Cyclomatic Complexity Measures: Control Flow Graphs.

UNIT IV:

Software Testing: Software verification & validation, Strategic Approach to Software Testing, Testing Fundamentals Test Plan, Test Design, Test Execution, Reviews, Inspection Auditing, Alpha and Beta Testing of Products, Recent Trends in Software Design/Specialized Software Testing, Related Tools and Standards.

UNIT V:

Software Maintenance and Software Project Management: Software Maintenance, Types of Maintenance, Overview of RE-engineering Reverse Engineering, Software Configuration Management, Cost Estimation-Constructive Cost Models (COCOMO), Project Scheduling, Resource Allocation Models, Software Risk Analysis and Management.

TEXTBOOKS/ REFERENCE BOOKS

1. Roger Pressman, Software Engineering: A Practitioner‟s Approach, 7th Edition, McGraw Hill
2. Ian Sommerville, Software Engineering, 9th Edition, Addision-Wesley, 2016
3. Pankaj Jalote, A Concise Introduction to Software Engineering, Springer, 2008
4. William E. Lewis , Software Testing and Continuous Quality Improvement, Third Edition, Auerbach Publications, 2008

COURSE CODE & NAME: CAUCBC312T / Computer Networks

COURSE OUTCOMES

1. Analyze the fundamental concepts of data communication and networking models, including OSI and TCP/IP protocols.
2. Evaluate various transmission media and multiplexing techniques, assessing their impact on data integrity and efficiency.
3. Apply error detection and correction methods to ensure reliable data transmission in different network scenarios.
4. Design and implement IP addressing schemes, including subnetting and supernetting, to optimize network performance.
5. Create network applications using transport layer protocols, demonstrating proficiency in DNS, email, file transfer, and web technologies.

UNIT I:

Introduction: Data Communication, Component, Data Flow, Data Representation, Network Criteria, Physical Structure, Network models, Categories of Network-LAN, MAN, WAN, Protocol and Standard, OSI Model, TCP-IP Protocol, Network Devices: Hub, Switch, Router, Bridge, Gateway

UNIT II:

Physical layer: Data, Signal, Digital Transmission, Impairment, Digital to Digital Conversion- Line Coding Scheme, Block Coding, Transmission Modes, Analog Transmission. Multiplexing- Frequency Division Multiplexing, Wavelength Division Multiplexing, Time Division Multiplexing- Synchronous & Statistical Transmission Media-Guided: Twisted Pair, Coaxial, Optical Fiber & Unguided Media, **Switching:** Circuit Switched network, Packet Switched Network

UNIT III:

Data link Layer: Error detection and Correction- types of error, hamming Distance, CRC, Checksum. Framing, Flow and error control, Noise less Channel- Simplest, Stop and Wait, Noisy Channel- Stop and wait ARQ, Go-Back-N ARQ

Multiple Access: Random Access Protocol- ALOHA, CSMA, CSMA/CD, CSMA/CA, Controlled Access Protocol, Ethernet and Cabling.

UNIT IV:

Network layer: IP Addresses, Classful Addressing, Supernetting, Subnetting, Classless Addressing, IPv6, Routing Protocol: Distance Vector Routing, Link State Routing.

UNIT V:

Transport & Application Layer: User Datagram Protocol, Transmission Control Protocol(TCP), Domain Name System, Electronic Mail, file Transfer, Remote Login, WWW, HTTP

TEXTBOOKS/ REFERENCE BOOKS

1. A. S. Tanenbaum (2003), Computer Networks, 4th edition, Pearson Education/ PHI, New Delhi, India.
2. Behrouz A. Forouzan (2006), Data communication and Networking, 4th Edition, Mc Graw-Hill, India.
3. Kurose, Ross (2010), Computer Networking: A top down approach, Pearson Education, India.

COURSE CODE & NAME: PTSPPCA22T/ Professional Proficiency

COURSE OUTCOMES

1. Identify and analyze basic grammar elements, including tenses and subject-verb agreement, through reading comprehensions.
2. Apply grammatical concepts in constructing coherent sentences and engaging in daily conversations effectively.
3. Demonstrate effective self-introduction and body language skills to enhance communication confidence in professional settings.
4. Create professional documents such as emails, letters, and resumes, applying appropriate formatting and language conventions.
5. Evaluate and solve quantitative aptitude problems, including simplification and coding-decoding, to enhance logical reasoning skills.

UNIT I:

Hard Skills: Idioms & Phrases, modal verbs Phrasal verb, para jumble, spot the error, sentence - making using idioms & phrasal verb,

UNIT II:

Soft Skills: Presentation, type and deliverance, stress management, team work, motivating skills, dressing etiquettes

UNIT III:

Quantitative Aptitude & Logical Reasoning: Blood Relation, Direction and Distance, Percentage, Inequality, Practice sheet having 25 question cover all topic till III semester to be distributed every week

TEXTBOOKS/ REFERENCE BOOKS

1. Advance Maths- RakeshYadav
2. Verbal and Non-Verbal Reasoning- R.S Agarwal
3. English Grammar, Composition and Usage by N.K. Agrawal & F.T. Wood, Macmillan India Ltd., New Delhi.
4. English Grammar & Composition by Wren & Martin, S.Chand & Co. Ltd., New Delhi.

COURSE CODE & NAME: CAUCBC402T / Operating System

COURSE OUTCOMES

1. Understanding of communication interface between user and computer hardware
2. Structure, Functions, Services, components, working of Operating System
3. Process Representation and Handling, scheduling, synchronization
4. Understanding of Memory management, File and Security issues.

UNIT I:

Introduction: Definition and types of operating system, Evolution, Batch Processing System, Multiprogramming, Time Sharing, Parallel System, Real Time System, Distributed System, Network System, Operating System Structure, Components of Operating System, services, functions, System Calls, System programs, Kernel and its types, Virtual Machines

UNIT II:

Process Management: Concept of Process, process queues, process Scheduling, Cooperating Process, Threads, Inter-Process Communication, CPU scheduling Criteria, Scheduling algorithms, Multiple Processor Scheduling, Real Time Scheduling, Algorithm Evaluation.

UNIT III:

Process Synchronization and Deadlock: The Critical Section Problem, Synchronization Hardware, Semaphores, Monitors, Classical Problems of Synchronization, Critical Region, Deadlock System Model, Characterization, Deadlock Prevention, Detection and Avoidance, Recovery from Deadlock, Combined approach to handle Deadlock, Banker's Algorithm

UNIT IV:

Memory Management: Logical and Physical Address Space, Swapping, Contiguous Allocation, Dynamic Memory Allocation, Fragmentation, Memory Freeing, Virtual Memory Concepts and Its Implementation, Demand Paging and its Performance, Page Replacement Algorithms, Allocation of Frames, Thrashing, Page size and other consideration, Demand Segmentation.

UNIT V:

File Management and Security: File System, Secondary Storage structure, Concept of File, Access Methods, Directory Implementation, Efficiency and Performance, Recovery Security: Safeguards, Penetration, Access and Information Flow control, Protection Problems, Formal Model of Protection

TEXTBOOKS/ REFERENCE BOOKS

1. A. Silberschatz, P. B. Galvin, and G. Gagne, Operating System Principles, 9/e, John Wiley, 2013.
2. A. S. Tanenbaum, Modern Operating Systems, 4/e, Pearson Education, 2017.
3. G. J. Nutt, Operating Systems - A Modern Perspective, 3/e, Pearson Education, 2009.
4. W. Stallings, Operating Systems: Internals and design Principles, 7/e, Pearson Education, 2012.

COURSE CODE & NAME: CAUCBC403T / Design and Analysis of Algorithms

COURSE OUTCOMES

1. Analyze and evaluate the efficiency of various sorting algorithms and their complexities, applying appropriate performance measurements.
2. Construct and manipulate advanced data structures such as AVL Trees, Red-Black Trees, and Binomial Heaps to solve complex computational problems.
3. Implement divide-and-conquer strategies and greedy algorithms to optimize solutions for problems like matrix multiplication and minimum spanning trees.
4. Apply dynamic programming techniques to solve complex problems including resource allocation and shortest path calculations effectively.
5. Critique and differentiate between string matching algorithms and understand the implications of NP-completeness in algorithm design and analysis.

UNIT I:

Introduction: Algorithms, Analysing algorithms, Complexity of algorithms, Growth of functions, Performance measurements, Analysis of sorting and order Statistics algorithms - Shell sort, Quick sort, Merge sort, Heap sort, Comparison of sorting algorithms, Sorting in linear time.

UNIT II:

Advanced Data Structures: AVL Tree, Red-Black Tree, Trie, B – trees, Comparison of Tree structures,

Binomial Heaps, Fibonacci Heaps, Augmenting Data Structures- Optimal Binary Search Trees, Amortized Algorithm and Analysis

UNIT III:

Divide and conquer with examples such as Sorting, Matrix Multiplication, Convex hull and Searching.

Greedy methods with examples such as Optimal Reliability Allocation, Knapsack, Minimum Spanning trees – Prim's and Kruskal's algorithms, Single source shortest paths - Dijkstra's and Bellman Ford algorithms.

UNIT IV:

Dynamic programming with examples such as Knapsack, All pair shortest paths – Warshal's and Floyd's algorithms, Resource allocation problem.

Backtracking, **Branch and Bound** with examples such as Travelling Salesman Problem, Graph Coloring, n-Queen Problem, Hamiltonian Cycles and Sum of subsets.

UNIT V:

Advanced Topics: String Matching: Naïve, KMP, RK etc., Theory of NP-completeness, Approximation algorithms and Randomized algorithms

TEXTBOOKS/ REFERENCE BOOKS

1. Thomas H. Coreman, Charles E. Leiserson and Ronald L. Rivest, “Introduction to Algorithms”, Printice Hall of India.
2. Thomas H. Coreman, “Algorithms Unlocked”, MIT Press, 2013
3. E. Horowitz & S Sahni, "Fundamentals of Computer Algorithms"
4. Aho, Hopcraft, Ullman, “The Design and Analysis of Computer Algorithms” Pearson Education, 2008.
5. Jon Kleinberg and Éva Tardos, Algorithm Design, Pearson, 2005.
6. Michael T Goodrich and Roberto Tamassia, Algorithm Design: Foundations, Analysis, and Internet Examples, Second Edition, Wiley, 2006.
7. Harry R. Lewis and Larry Denenberg, Data Structures and Their Algorithms, Harper Collins, 1997
8. Robert Sedgewick and Kevin Wayne, Algorithms, fourth edition, Addison Wesley, 2011.
9. Harsh Bhasin,”Algorithm Design and Analysis”, First Edition, Oxford University Press.
10. Gilles Brassard and Paul Bratley, Algorithmics: Theory and Practice, Prentice Hall,1995.

COURSE CODE & NAME: ----------/ Advanced Web Design

COURSE OUTCOMES

1. Understand the fundamental concepts of Python programming, including data structures and object-oriented principles.
2. Apply data science techniques using Python libraries such as NumPy and Pandas for effective data manipulation and visualization.
3. Analyze web application development using the Django framework, including project creation, URL mapping, and template rendering.
4. Evaluate database management skills by performing CRUD operations with MySQL and MongoDB, ensuring proper transaction handling and error management.
5. Create dynamic web applications that integrate front-end and back-end technologies, utilizing best practices in version control and deployment.

UNIT I:

Python Basics: Introduction to python, control statements, list, ranges & tuples in python, python dictionaries and sets, python built in function, python object oriented concepts, exceptions, python regular expressions, python multithreaded programming

UNIT II:

Data science using python: Introduction to numpy, Creating arrays, Indexing Arrays, Array Transposition, Universal Array Function, Array Processing, Array Input and Output, Matplotlib: Data Visualization, Python for Data Visualization, Pandas**.**

UNIT III:

Django web framework in python: Django overview, Creating a project, Apps life cycle, Admin interface, Creating views, URL Mapping, Template system, Models, Form details, Testing, Page redirection, Sending Emails, Deploying Django framework, Form processing, File uploading, Cookie handling, Sessions, caching and comments, RSS, AJAX, Sending Emails, GitHub , Bigbucket

UNIT IV:

Database: Database handling with mysql, python mysql database access, create database connection,dml and ddl operations with databases, performing transactions, handling database errors, disconnecting database, database handling with mongodb, Sql vs nosql, mongodb, pymongo, establishing a connection, accessing database,dml and ddl operations

UNIT V:

The Django Administration Site: Activating the Admin Interface, Using the Admin Interface, Users, Groups and Permissions, Customizing the Admin Interface Customizing the Admin Interface's Look and Feel, Customizing the Admin Index Page, When and Why to Use the Admin Interface Cookies, Getting and Setting Cookies, The Mixed Blessing of Cookies, Setting Test Cookies, Users and Authentication Enabling Authentication Support, Using Users, Logging In and Out ,Limiting Access to Logged-in Users, Managing Users, Permissions and Groups, Using Authentication Data in Templates, Permissions Groups

TEXTBOOKS/ REFERENCE BOOKS

1. The Complete Reference HTML & XHTML by Thomas Powell, 5th Edition, Tata McGraw-Hill Company Limited.
2. Learning web designing: a beginner& guide to HTML, CSS, JavaScript and web graphics' by Niederst Robbins, 4th Edition, Oreilly Publication
3. Mastering HTML, CSS & JAVAScript Web Publishing' by Laura Lemay, Rafe Coburn, Jennifer Kyrnin, 7th edition, SAMS publication.
4. Web Enabled Commercial Application Development using HTML, JavaScript, DHTML and PHP' by Ivan Bayross, 4th Edition, BPB Publications.

COURSE CODE & NAME: CAUCBC502T / Machine Learning

COURSE OUTCOMES

1. Appreciate the importance of visualization in the data analytics solution
2. Apply structured thinking to unstructured problems
3. Understand a very broad collection of machine learning algorithms and problems
4. Learn algorithmic topics of machine learning and mathematically deep enough to introduce the required theory
5. Develop an appreciation for what is involved in learning from data.

UNIT I:

Introduction to Machine Learning: Fundamentals of ML, supervised, unsupervised, reinforcement learning;
Supervised Learning: Classification: KNN, Centroid Method, Perceptron, Support Vector Machines, Multi-level Perceptron, Decision tree
Regression: Simple Linear Regression, Multiple Linear Regression, Logistic regression

UNIT II:

Unsupervised Learning : Clustering: Centroid-based Clustering, Density-based Clustering, Distribution-based Clustering, Hierarchical Clustering, Dimensionality Reduction: PCA, LDA

UNIT III:

Bayesian and Computational Learning: Bayes Theorem, Concept Learning, Maximum Likelihood, Bayes Optimal Classifier, Naïve Bayes Classifier, Bayesian Belief Network, Probability Learning.

UNIT IV:

Neural Network: Introduction to neural networks, Fundamental concepts- neuron models and basic learning rules; Single layer neural Networks, input layer, output layer, hidden layers, Multilayer Neural Networks, Backpropagation.

UNIT V:

Deep Learning Techniques: Gradient Descent; Convolutional Neural Network (CNN)- Convolution, activation, pooling, Self-Organizing Feature Maps (SOMs).

TEXTBOOKS/ REFERENCE BOOKS

1. Kevin Murphy , Machine Learning: a Probabilistic Perspective, 2012.
2. Chris Bishop, Pattern Recognition and Machine Learning, 2006.
3. Ian Goodfellow, Yoshua Bengio, Aaron Courville, Deep Learning, 2016
4. Richard Szeliski, Computer Vision: Algorithms and Applications, 2010.
5. Seth Weidman, Deep Learning from Scratch: Building with Python from First Principles, 2019
6. Rishal Hurbans, Grokking Artificial Intelligence Algorithms, 2020

COURSE CODE & NAME: CAUCBC501T / Web Technology

COURSE OUTCOMES

1. Understand the fundamentals of web technologies, including HTML structure, client-server architecture, and the evolution of the web from Web 1.0 to Web 2.0.
2. Demonstrate proficiency in CSS for styling web pages, including layout techniques, responsive design, and the use of frameworks like Bootstrap.
3. Develop interactive web applications using JavaScript, focusing on client-side scripting, event handling, and jQuery for enhanced user experience.
4. Utilize AJAX and XML for asynchronous web communication and data interchange, including knowledge of XML schema languages and XSLT.
5. Implement server-side scripting with PHP, covering database interactions, form handling, and the integration of frameworks like CodeIgniter and Laravel.

UNIT I:

Introduction to Web and Hyper Text Markup Language: Internet, Intranet, WWW, Static and Dynamic Web Page; Web Clients; Web Servers; Client Server Architecture: Single Tier, Two-Tier, Multi-Tier; HTTP: HTTP Request and Response; URL, Client Side Scripting, Server Side Scripting, Web 1.0, Web 2.0.

Introduction to HTML: Elements of HTML Document; HTML Elements and HTML Attributes, Headings, Paragraph, Division, Formating: b, i, small, sup, sub; Spacing: Pre, Br; Formatting Text, Phrases: span, strong, Image element; Anchors; Lists: Ordered and Unordered and Definition; Tables; Frames; Forms: Form Elements, ID attributes, Class Attributes of HTML Elements; Meta Tag, Audio, Video, Canvas, Main, Section, Article, Header, Footer, Aside, Nav, Figure Tags; HTML Events: Window Events, Form Element Events, Keyboard Events, Mouse Events.

UNIT II:

Cascading Style Sheets: Introduction; Cascading Style Sheets (CSS); CSS Syntax; Inserting CSS: Inline, Internal, External, ID and Class Selectors; Colors; Backgrounds; Borders; Text; Font; List; Table; CSS Box Model; Normal Flow Box Layout: Basic Box Layout, Display Property, Padding, Margin; Positioning: Relative, Float, Absolute; CSS3 Borders, Box Shadows, Text Effects and shadow; Basics of Responsive Web Designs; Media Queries, Introduction to Bootstrap

UNIT III:

Client Side Scripting with JavaScript: Structure of JavaScript Program; Variables and Data Types; Statements: Expression, Keyword, Block; Operators; Flow Controls, Looping, Functions; Popup Boxes: Alert, Confirm, Prompt; Objects and properties; Constructors; Arrays; Built-in Objects: Window, String, Number, Boolean, Date, Math, RegExp, Form, DOM; User Defined Objects; Event Handling and Form Validation, Error Handling, Handling Cookies, jQuery Syntax; jQuery Selectors, Events and Effects; Introduction to JSON.

UNIT IV:

AJAX and XML: Basics of AJAX; Introduction to XML and its Application; Syntax Rules for creating XML document; XML Elements; XML Attributes; XML Tree; XML Namespace; XML schema languages: Document Type Definition(DTD), XML Schema Definition (XSD); XSD Simple Types, XSD Attributes; XSD Complex Types; XML Style Sheets (XSLT), XQuery.

UNIT V:

Server Side Scripting using PHP: PHP Syntax, Variables, Data Types, Strings, Constants, Operators, Control structure, Functions, Array, Creating Class and Objects, PHP Forms, Accessing Form Elements, Form Validation, Events, Cookies and Sessions, Working with PHP and MySQL, Connecting to Database, Creating, Selecting, Deleting, Updating Records in a table, Inserting Multiple Data, Introduction to CodeIgniter, Laravel, Wordpress etc.

TEXTBOOKS/ REFERENCE BOOKS

1. Web Enabled Commercial Application Development using HTML, JavaScript, DHTML and PHP' by Ivan Bayross, 4th Edition, BPB Publications.
2. Learning web designing: a beginner's guide to HTML, CSS, JavaScript,and web graphics' by Niederst Robbins, 4th Edition, Oreilly Publication
3. Web Enabled Commercial Application Development using HTML, JavaScript, DHTML and PHP' by Ivan Bayross, 4th Edition, BPB Publications.
4. The Complete Reference HTML & XHTML' by Thomas Powell, 5th Edition, Tata McGraw-Hill Company Limited.
5. HTML 4.0' by E. Stephen Mack, Janan Platt, Anaya Multimedia publication.
6. Mastering HTML, CSS & JAVAScript Web Publishing' by Laura Lemay, Rafe Coburn, Jennifer Kyrnin, 7th edition, SAMS publication.

COURSE CODE & NAME: CAUEBC301T / Data Mining & Warehousing

COURSE OUTCOMES

1. Understand the fundamental concepts and characteristics of Data Warehousing and its architecture.
2. Analyze data cleaning methods and pre-processing techniques to enhance data quality in a Data Warehouse.
3. Evaluate and differentiate between OLTP and OLAP systems, including various data cube computation methods.
4. Apply association rule mining techniques and classification methods to derive insights from data.
5. Synthesize clustering methods to group data objects based on similarity and dissimilarity measures.

UNIT I:

Introduction to Data Warehouse, Building a Data Warehouse, Data Pre-processing & Data cleaning Data Cleaning methods, Data reduction, Descriptive Data Summarization, Data Discretization, Concept Hierarchy Generation

UNIT II:

Data Warehouse-Definition and Characteristics, Essential component of a Data Warehouse,3-layered architecture of Data Warehouse, Implementation Issues related to DW,H/w and S/w requirements for a Data Warehouse, Enterprise Data Warehouse, Data Mart, C/S Computing model and Data Warehouse, Data warehouse Schema.

UNIT III:

Comparison OLTP and OLAP, ROLAP,MOLAP and DOALP, Data Cube Computation methods, Advanced SQL support for OLAP, Multidimensional Modelling, Attribute-oriented Induction.

UNIT IV:

Basic Concepts of Association rule mining, Frequent ItemSet Mining, Classification by Decision tree induction, Bayesian Classification, Rule based classification, Associative Classification, Lazy Learners, Rough Set approach.

UNIT V:

Data Objects and Attribute Types, Basic Statiscal Descriptions of Data, Measuring Data Similarity and Dissimilarity, Partitioning based Clustering methods, K-means, Hierarchical Clustering methods, Density based Clustering methods, DBSCAN, BIRCH algorithms

TEXTBOOKS/ REFERENCE BOOKS

1. Data Warehousing in the Age of Big Data by Krish Krishnan
2. Fundamentals of Data Warehousing by Paulraj Ponniah
3. Data Mining: Concepts and Techniques by Jiawei Han, Micheline Kamber, and Jian Pei
4. Data Warehousing: A Comprehensive Approach by Arun K. Pujari
5. The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling by Ralph Kimball and Margy Ross

6. Data Mining: Practical Machine Learning Tools and Techniques by Ian H. Witten, Eibe Frank, and Mark A. Hall
7. Introduction to Data Mining by Pang-Ning Tan, Michael Steinbach, and Vipin Kumar

COURSE CODE & NAME: CAUCBC114P / Python Programming Lab

COURSE OUTCOMES

1. Understand and comprehend the basics of python programming.
2. Demonstrate the principles of structured programming and be able to describe, design, implement, and test structured programs using currently accepted methodology.
3. Explain the use of the built-in data structures list, sets, tuples and dictionary.
4. Make use of functions and its applications.

List of Experiments:

1. Write a program to demonstrate different number data types in Python
2. Write a program to compute distance between two points taking input from the user using Pythagorean Theorem.
3. Write a Program for checking whether the given number is a even number or not.
4. Write a Python script that prints prime numbers less than 20.
5. Write a program to create, concatenate and print a string and accessing sub-string from a given string.
6. Write a program to create, append, and remove lists in python.
7. Write a program to demonstrate working with tuples in python.
8. Write a python program to define a module to find Fibonacci Numbers and import the module to another program.
9. Write a python program to define a module and import a specific function in that module to another program
10. Write a script named copyfile.py. This script should prompt the user for the names of two text files. The contents of the first file should be input and written to the second file.
11. Write a Python class to implement pow(x, n).

COURSE CODE & NAME: CAUCBC104P / Digital Logic Design Lab

COURSE OUTCOMES

1. Analyze and verify the truth tables of basic and universal gates to demonstrate their functionality.
2. Apply simplification techniques to logical expressions and implement them using basic and universal gates.
3. Evaluate the significance of NAND and NOR gates as universal gates through practical experiments.
4. Construct and implement Half Adders and Full Adders using both basic and universal gates to understand their operations.
5. Design and realize Half Subtractors, Full Subtractors, and a Binary to Grey code generator using appropriate gate configurations.

List of Experiments:

1. To study and verify the truth table of basic gates.
2. To study and verify the truth table of universal gate.
3. To simplify the given expression and to realize it using basic gates and universal gate.
4. To realize why NAND gate is known as the universal gate.
5. To realize why NOR gate is known as the universal gate.
6. Realization of Half Adder and Full Adder by using Basic gates.
7. Realization of Half Adder and Full Adder by using universal gate.
8. Realization of Half Subtractor and Full Subtractor by using Basic gates.
9. Realization of Half Subtractor and Full Subtractor by using universal gate.
10. Realization of Binary to grey generator.

COURSE CODE & NAME: CAUCBC212P / Data Structure Lab

COURSE OUTCOMES

1. Analyze and compare various sorting algorithms through both recursive and non-recursive implementations to evaluate their efficiency.
2. Design and implement searching algorithms to enhance data retrieval processes in programming.
3. Construct stack data structures using both arrays and linked lists, demonstrating proficiency in memory management.
4. Develop queue data structures with arrays and linked lists, emphasizing their application in real-world scenarios.
5. Implement and analyze tree structures and graph algorithms, applying traversal, insertion, deletion, and pathfinding techniques to solve complex problems.

List of Experiments:

1. Implement Sorting Algorithms-Non-Recursively and Recursively.
2. Implement Searching Algorithms.
3. Implementation of Stack using Array and Linked List.
4. Implementation of Queue using Array and Linked List.
5. Implementation of Tree Structures, Binary Tree, Tree Traversal, Binary Search Tree, Insertion an Deletion in BST.
6. Graph Implementation, BFS, DFS, Minimum cost spanning tree, shortest path algorithm.

COURSE CODE & NAME: CAUCBC211P/ Computer Organization Lab

COURSE OUTCOMES

1. Understanding basic design of computer with modern computer architecture
2. Understanding CPU organization, Memory Organization, I/O Organization
3. Understanding various data transfer schemes with interrupt Handling
4. Understanding architecture, and concept of Parallel Computing

List of Experiments:

1. Implementing HALF ADDER, FULL ADDER using basic logic gates
2. Implementing Binary -to -Gray, Gray -to -Binary code conversions.
3. Implementing 3-8 line DECODER.
4. Implementing 4x1 and 8x1 MULTIPLEXERS.
5. Verify the excitation tables of various FLIP-FLOPS.
6. Design of an 8-bit Input/ Output system with four 8-bit Internal Registers.
7. Design of an 8-bit ARITHMETIC LOGIC UNIT.

COURSE CODE & NAME: CAUCBC203P/ Database Management System Lab

COURSE OUTCOMES

1. Understand database concepts and structures and query language
2. Understand the E R model and relational model
3. To design and build a simple database system and demonstrate competence with the fundamental tasks involved with modelling, designing, and implementing a DBMS.
4. Understand Functional Dependency and Functional Decomposition.
5. Apply various Normalization techniques.
6. Perform PL/SQL programming using concept of Cursor Management, Error Handling, Package and Triggers.
7. Understand the principles of storage structure and recovery management.

List of Experiments:

1. Student should decide on a case study and formulate the problem statement.
2. Conceptual Designing using ER Diagrams (Identifying entities, attributes, keys and relationships between entities, cardinalities, generalization, specialization etc.)
3. Converting ER Model to Relational Model (Represent entities and relationships in Tabular form, Represent attributes as columns, identifying keys)
4. Note: Student is required to submit a document showing the database tables created from ER Model.
5. Creation of Tables using SQL- Overview of using SQL tool, Data types in SQL, Creating. Tables (along with Primary and Foreign keys), Altering Tables and Dropping Tables
6. Practicing DML commands- Insert, Select, Update, Delete
7. Practicing Queries using ANY, ALL, IN, EXISTS, NOT EXISTS, UNION, INTERSECT, CONSTRAINTS etc.
8. Practicing Sub queries (Nested, Correlated) and Joins (Inner, Outer and Equi).
9. Practice Queries using COUNT, SUM, AVG, MAX, MIN, GROUP BY, HAVING, VIEWS Creation and Dropping.

COURSE CODE & NAME: CAUCBC301T/ Object Oriented Programming

COURSE OUTCOMES

1. Introduction to OOPs concepts & Object oriented Programming using java
2. Multithreading & Java applets concepts
3. To understand the various concepts of AWT, Swings and event handling of Java.
4. To understand the various concepts of AWT, Swings and event handling of Java.

List of Experiments:

1. Student should To write programs to illustrate the uses of decision control structures: if, nested if, switch case etc.
2. To write programs to illustrate the uses of loop control structures: do, while, for etc.
3. To write programs to illustrate the uses of array, Vector & String.
4. To write programs to illustrate the uses of creating and working with class and object.
5. To write programs to illustrate the uses of OOPs concepts: data abstraction, data hiding, encapsulation, inheritance & polymorphism (method overloading and overriding).
6. To write programs to illustrate the uses of Interfaces and packages.
7. To write programs using Multithreading & exceptions handling mechanism.
8. To write GUI programs using AWT controls.
9. To write GUI programs to implement various layouts
10. To write GUI programs to handle mouse & key events.
11. To write programs to retrieve data from data base using JDBC drivers.

COURSE CODE & NAME: CAUCBC312P / Computer Networks Lab

COURSE OUTCOMES

1. Identify and describe different types of network cables and their applications in networking.
2. Demonstrate the ability to create cross-wired and straight-through cables using appropriate tools.
3. Analyze and explain the functionality of various network devices and their roles in a network.
4. Apply IPv4 and IPv6 addressing schemes to configure and connect computers in a Local Area Network.
5. Evaluate and configure network topologies using simulation software, including Distance Vector and Link State Routing protocols.

List of Experiments:

1. List of Experiments:
2. Study of different types of Network cables and practically implement the cross-wired cable and straight through cable using clamping tool.
3. Study of Network Devices in Detail.
4. Study of network IPv4 and IPv6.
5. Connect the computers in Local Area Network.
6. Study of basic network command and Network configuration commands.
7. Configure a Network topology using packet tracer software.
8. Configure a Network topology using packet tracer software.
9. Configure a Network using Distance Vector Routing protocol.
10. Configure Network using Link State Vector Routing protocol.

COURSE CODE & NAME: CAUCBC402P / Operating System Lab

COURSE OUTCOMES

1. Install and configure a Linux operating system with essential settings and user management.
2. Utilize Unix/Linux utility commands for effective system navigation and file management.
3. Apply text editing skills using the vi editor for efficient file manipulation.
4. Implement C programs for scheduling algorithms and concurrency issues in Unix/Linux.
5. Analyze and manipulate the Unix/Linux file system and environment variables for optimization.

List of Experiments:

1. Installation of Linux operating system.
   a. Partitioning drives
   b. Configuring boot loader (GRUB/LILO)
   c. Network configuration
   d. Setting time zones
   e. Creating password and user accounts
   f. Shutting down
2. Study of Unix/Linux general purpose utility command list obtained from (man, who, cat, cd, cp, ps, ls, mv, rm, mkdir, rmdir, echo, more, date, time, kill, history, chmod, chown, finger, pwd, cal, logout, shutdown) commands.
3. Study of vi editor.
4. Study of Bash shell, Bourne shell and C shell in Unix/Linux operating system.
5. Study of Unix/Linux file system (tree structure).
6. Study of .bashrc, /etc/bashrc and Environment variables.
7. To write a C-program to implement various scheduling algorithms.
8. To write a C-program to implement the producer – consumer problem using semaphores
9. To write a c program to implement IPC using shared memory.
10. To write a C program to implement banker‟s algorithm for deadlock avoidance.
11. To write a c program to implement Threading and Synchronization Applications.
12. To write a C program for implementation memory allocation methods for fixed partition using first fit.

COURSE CODE & NAME: CAUCBC403P / Design and Analysis of Algorithms Lab

COURSE OUTCOMES

1. Analyze and compare the efficiency of various sorting algorithms through practical implementation.
2. Apply greedy algorithm techniques to solve real-world problems effectively.
3. Demonstrate the use of divide and conquer strategies in algorithm design through hands-on coding.
4. Implement dynamic programming solutions to optimize complex computational problems.
5. Utilize advanced data structures like AVL, RB, TRIE, and B-Tree to perform fundamental operations efficiently.

List of Experiments:

1. Implementing sorting algorithms.
2. Implementing at least 1 Example/Application of Greedy Algorithm.
3. Implementing at least 1 Example/Application of Divide and Conquer Algorithm.
4. Implementing at least 1 Example/Application of DP Algorithm
5. Implementing at least 1 Example/Application of Backtracking Algorithm.
6. Implementing at least 1 Example/Application of Branch & Bound Algorithm.
7. Implementing at least 1 String Matching Algorithm
8. Implementing AVL tree with operations: Insertion, deletion, search, traversal.
9. Implementing RB tree with operations: Insertion, deletion, search, traversal.
10. Implementing TRIE with operations: Insertion, deletion.
11. Implementing B-Tree tree with operations: Insertion, deletion.