

BANGALORE UNIVERSITY

SCHEME AND SYLLABUS

For the course

BACHELOR OF COMPUTER APPLICATIONS (BCA)

NEP2021 Scheme

Academic Year 2021-22 and onwards

Department of Computer Science and Applications
BANGALORE UNIVERSITY, BANGALORE

MEMBERS OF THE BoS IN COMPUTER SCIENCE

1	Dr. Muralidhara B L Professor Department of Computer Science Bangalore University	CHAIRPERSON
2	Dr. Guru D.S Professor PG Department of Computer Science Mysore Univeristy	Member
3	Dr. Susesha Professor, PG Department of Computer Science Mysore Univeristy	Member
4	Dr. Prabhakar C.J Professor Kuvempu University, Shimogga	Member
5	Dr. Chandrakanth Naikodi Associate Professor Department of Computer Science Davanagere University	Member
6	Dr. Prathibha V Kalburgi Ramaiah College of Arts Science, and Commerce Bangalore	Member
7	Mrs. Amalorpavam Sambram Academi of Management Studies Bangalore	Member
8	Dr. H.K. Gundurao Associate Professor Vijaya College, Bangalore	Member
9	Dr. Bhagyawana S Mudigowda Associate Professor Maharani Cluster University, Bangalore	Member
10	Smt. Nagarathnamma S.M Associate Professor Maharani Cluster Univeristy, Bangalore	Member

BCA SYLLABUS (NEP)

Semester	Course Code	Title of the Paper	Credits	Languages, Skill Enhancement (SEC), and Ability Enhancement Courses (AEC)	Credits	Total Credits
I	CA-C1T	Discrete Structure	3	OE1: Open Elective	3	26
	CA-C2T	Problem solving Techniques	3	Language L1	3	
	CA-C3T	Data Structure	3	Language L2	3	
	CA-C4L	Problem solving Lab	2	SEC I : Office Management Tools	2	
	CA-C5L	Data Structure Lab	2	Physical Education	1	
				Health & Wellness	1	
II	CA-C6T	Computer Architecture	3	OE2: Open Elective	3	26
	CA-C7T	Object Oriented Programming using Java	3	Language L1	3	
	CA-C8T	Database Management System	3	Language L2	3	
	CA-C9L	Java Lab	2	Environmental studies	2	
	CA-C10L	Database Management System Lab	2	Physical Education	1	
				NCC/NSS/CL/R&R	1	
III	CA-C11T	Operating Systems	3	OE3: Open Elective	3	26
	CA-C12T	Computer Networks	3	Language L1	3	
	CA-C13T	Python Programming	3	Language L2	3	
	CA-C14L	Computer Networks Lab	2	SEC II : Computer Assembly and Repair	2	
	CA-C15L	Python Programming Lab	2	Physical Education	1	
				NCC/NSS/CL/R&R	1	
IV	CA-C16T	Software Engineering	3	OE4: Open Elective	3	26
	CA-C17T	Design and Analysis of Algorithm	3	Language L1	3	
	CA-C18T	Internet Technologies	3	Language L2	3	
	CA-C19L	Design and Analysis of Algorithm Lab	2	The Constitution of India	2	
	CA-C20L	Internet Technologies Lab	2	Physical Education	1	
				NCC/NSS/CL/R&R	1	
V	CA-C21T	Artificial Intelligence	3	CA-V1 Vocation Course I : Quantitative	3	23
	CA-C22T	Data Analytics	3	CA-E1 Elective I : a. Data Mining b. Computer Graphics	3	
	CA-C23T	Web Programming	3	SEC III : Cyber Crime, Cyber Law, and Intellectual Property Right	2	
	CA-C24L	Data Analytics Lab	2	Physical Education	1	

Semester	Course Code	Title of the Paper	Credits	Languages, Skill Enhancement (SEC), and Ability Enhancement Courses (AEECC)	Credits	Total Credits
	CA-C25L	Web Programming Lab	2	NCC/NSS/CL/R&R	1	
Semester	Course Code	Title of the Paper	Credits	Languages, Skill Enhancement (SEC), and Ability Enhancement Courses (AEECC)	Credits	Total Credits
VI	CA-C26T	Theory of Computation	3	CA-V2 Vocation Course II : Electronic Content Design	3	23
	CA-C27T	Machine Learning	3	CA-E2 Elective II : a. Operations Research b. Software Testing	3	
	CA-C28T	Mobile Application Development	3	Professional Communication	2	
	CA-C29L	Machine Learning Lab	2	Physical Education	1	
	CA-C30L	Mobile Application Development Lab	2	NCC/NSS/CL/R&R	1	
VII	CA-C31T	Cloud Computing	3	CA-V3 Vocation Course III : Technical Writing	3	21
	CA-C32T	Internet of Things	3	CA-E3 Elective III : a. Modeling and Simulation b. Compiler Design	3	
	CA-C33T	Internship	2	Research Methodology	3	
	CA-C34L	Cloud Computing Lab	2			
	CA-C35L	Internet of Things Lab	2			
VIII	CA-C36T	Block Chain Technologies	3	CA-V4 Vocation Course IV : Project Management	3	20
	CA-C37T	Cryptography and System Security	3	CA-E4 Elective IV : a. Human Computer Interface b. Parallel Algorithms	3	
	CA-C38T	Block Chain Technologies Lab	2	Research Project	6	

CA-C1T: DISCRETE STRUCTURES

Total Teaching Hours: 48

No. of Hours / Week: 03

UNIT – I

[12 Hours]

Set Theory and Logic: Fundamentals of Set theory, Set Operations and the Laws of Set Theory, Counting and Venn Diagrams, Cartesian Products and Relations, Functions–One-to-One, Onto Functions, Function Composition and Inverse Functions. Mathematical Induction, The well ordering principle, Recursive Definitions, Structural Induction, Recursive algorithms. Fundamentals of Logic, Propositional Logic, Logical Connectives and Truth Tables, Logic Equivalence, Predicates and Quantifiers.

UNIT - II

[12 Hours]

Counting and Relations: Basics of counting, Pigeonhole Principle, Permutation and Combinations, Binomial coefficients. Recurrence relations, Modeling with recurrence relations with examples of Fibonacci numbers and the tower of Hanoi problem. Divide and Conquer relations with examples (no theorems). Definition and types of relations, Representing relations using matrices and digraphs

UNIT - III

[12 Hours]

Matrices: Definition, order of a matrix, types of matrices, operations on matrices, determinant of a matrix, inverse of a matrix, rank of a matrix, linear transformations, applications of matrices to solve system of linear equations.

UNIT - IV

[12 Hours]

Graph Theory: Graphs: Introduction, Representing Graphs, Graph Isomorphism, Operations on graphs. Trees: Introduction, Applications of Trees, Tree Traversal, Spanning Trees, Minimum Spanning Trees, Prim's and Kruskal's Algorithms. Connectivity, Euler and Hamilton Paths, Planar Graphs. Directed graphs: Fundamentals of Digraphs, Computer Recognition - Zero-One Matrices and Directed Graphs, Out-degree, in-degree, connectivity, orientation, Eulerian and Hamilton directed graphs, tournaments.

Text Books:

1. Ralph P. Grimaldi: Discrete and Combinatorial Mathematics, 5th Edition, Pearson Education, 2004.
2. C. L. Liu: Elements of Discrete Mathematics, Tata McGraw-Hill, 2000.
3. F. Harary: Graph Theory, Addition Wesley, 1969.
4. Richard Bronson, Schaum's Outline of Matrix Operations, McGraw-Hill publications, 2nd Edition, 2011

Reference Books:

1. Kenneth H Rosen. Discrete Mathematics and its Applications, McGraw-Hill publications, 7th edition, 2007.
2. J. P. Tremblay and R.P. Manohar. Discrete Mathematical Structures with applications to Computer Science, Mc Graw Hill Ed. Inc. 1975.
3. Charles G Cullen. Matrices and Linear Transformations, Dover Publications Inc., Second Edition, 1990

Web Resources:

1. <https://www.my-mooc.com/en/categorie/mathematics>
2. <http://www.nptelvideos.in/2012/11/discrete-mathematical-structures.html>
3. <https://ocw.mit.edu/courses/mathematics/>

CA-C2T: PROBLEM SOLVING TECHNIQUES

Total Teaching Hours: 48

No. of Hours / Week: 03

UNIT - I

[12 Hours]

Introduction: The Role of Algorithms in Computing, Algorithms as a technology, Analyzing algorithms, Designing algorithms, Growth of Functions, Asymptotic notation, Standard notations and common functions. Fundamental Algorithms: Exchanging the values of two variables, Counting, Summation of a set of numbers, Factorial Computation, Generating of the Fibonacci sequence, Reversing the digits of an integer, Character to number conversion.

UNIT - II

[12 Hours]

C Programming: Getting Started, Variables and Arithmetic expressions. Input and Output: Standard input and output, formatted output- printf, variable length argument list, formatted input-scanf. Control Flow: Statements and Blocks, If-else, else-if, switch, loops: while loop, for loop, do while, break and continue, goto and labels. Pointers and Arrays: pointers and address, pointers and function arguments, multidimensional array, initialization of pointer arrays, command line arguments.

UNIT - III

[12 Hours]

Factoring Methods: Finding the square root of a number, the smallest Divisor of an integer, the greatest common divisor of two integers, computing the prime factors of an integer, generation of pseudo random numbers, raising a number to a large power. Array Techniques: Array order Reversal, Array counting or Histogramming, Finding the maximum number in a set, removal of duplicates from an ordered array, partitioning an array, Finding the kth smallest element, multiplication of two matrices.

UNIT - IV

[12 Hours]

Merging: the two-way merge. Sorting: Sorting by selection, sorting by exchange, sorting by insertion, sorting by diminishing increment, sorting by partitioning. Searching: binary search, hash search. Text processing and Pattern searching: text line length adjustment, keyword searching in text, text line editing, linear pattern search

Text Books:

1. R.G.Dromey, "How to Solve it by Computer", Pearson Education India, 2008.
2. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, "Introduction to Algorithms", 3rd Edition, The MIT Press Cambridge, Massachusetts London, England, 2008
3. Brian M. Kernighan, and Dennis M. Ritchie, "The C Programming Language", 2nd edition, Princeton Hall Software Series, 2012.

Reference Books:

1. Steven S. Skiena, "The Algorithm Design Module", 2nd Edition, Springer-Verlag London Limited, 2008.
2. Donald E. Knuth, "The Art of Computer Programming", Volume 1: Fundamental Algorithms, 3rd Edition, Addison Wesley Longman, 1997.
3. Donald E. Knuth, "The Art of Computer Programming", Volume 2: Seminumerical Algorithms, 3rd Edition, Addison Wesley Longman, 1998.
4. Greg Perry and Dean Miller, "C programming Absolute Beginner's Guide", 3rd edition, Pearson Education, Inc, 2014.

Web Resources:

1. <http://algorithmsforinterviews.com> "Algorithms for Interviews"

CA-C3T: DATA STRUCTURES

Total Teaching Hours: 48

No. of Hours / Week: 03

UNIT-I [12 Hours]

Introduction and Overview: Definition, Elementary data organization, Data Structures, data Structures operations, Abstract data types, algorithms complexity, time-space trade off. Preliminaries: Mathematical notations and functions, Algorithmic notations, control structures, Complexity of algorithms, asymptotic notations for complexity of algorithms. Arrays: Definition, Linear arrays, arrays as ADT, Representation of Linear Arrays in Memory, Traversing Linear arrays, Inserting and deleting, Multi-dimensional arrays, Matrices and Sparse matrices.

UNIT-II [12 Hours]

Linked list: Definition, Representation of Singly Linked List in memory, Traversing a Singly linked list, Searching in a Singly linked list, Memory allocation, Garbage collection, Insertion into a singly linked list, Deletion from a singly linked list; Doubly linked list, Header linked list, Circular linked list. Stacks: Definition, Array representation of stacks, Linked representation of stacks, Stack as ADT, Arithmetic Expressions: Polish Notation, Conversion of infix expression to postfix expression, Evaluation of Post fix expression, Application of Stacks, Recursion, Towers of Hanoi, Implementation of recursive procedures by stack. Queues: Definition, Array representation of queue, Linked list representation of queues. Types of queue: Simple queue, Circular queue, Double-ended queue, Priority queue, Operations on Queues, Applications of queues.

UNIT-III [12 Hours]

Binary Trees: Definitions, Tree Search, Traversal of Binary Tree, Tree Sort, Building a Binary Search Tree, Height Balance: AVL Trees, Contiguous Representation of Binary Trees: Heaps, Lexicographic Search Trees: Tries, External Searching: B-Trees, Applications of Trees. Graphs: Mathematical Back ground, Computer Representation, Graph Traversal, Topological Sorting

UNIT-IV [12 Hours]

Searching: Introduction and Notation, Sequential Search, Binary Search, Comparison of Methods. Sorting: Introduction and Notation, Insertion Sort, Selection Sort, Shell Sort, Divide And Conquer, Merge sort for Linked List, Quick sort for Contiguous List. Hashing: Sparse Tables, Choosing a Hash function, Collision Resolution with Open Addressing, Collision Resolution by Chaining.

Text Books:

1. Seymour Lipschutz, "Data Structures with C", Schaum's outLines, Tata Mc Graw Hill, 2011.
2. Robert Kruse, C.L.Tondo, Bruce Leung, Shashi Mogalla, "Data Structures and Program Design using C", Pearson Education, 2009.

Reference Books:

1. Mark Allen Weiss, "Data Structures and Algorithm Analysis in C", Second Edition, Pearson Education, 2013.
2. Forouzan, "A Structured Programming Approach using C", 2nd Edition, Cengage Learning India, 2008.

CA-C4P: Problem Solving Lab using C

Write, and execute C program for the following:

1. to read radius of a circle and to find area and circumference
2. to read three numbers and find the biggest of three
3. to check whether the number is prime or not
4. to read a number, find the sum of the digits, reverse the number and check it for palindrome
5. to read numbers from keyboard continuously till the user presses 999 and to find the sum of only positive numbers
6. to read percentage of marks and to display appropriate message
(Demonstration of else-if ladder)
7. to find the roots of quadratic equation
8. to read marks scored by n students and find the average of marks
(Demonstration of single dimensional array)
9. to remove Duplicate Element in a single dimensional Array
10. to perform addition and subtraction of Matrices
11. to find factorial of a number
12. to generate fibonacci series
13. to remove Duplicate Element in a single dimensional Array
14. to find the length of a string without using built in function
15. to demonstrate string functions
16. to read, display and add two m x n matrices using functions
17. to read a string and to find the number of alphabets, digits, vowels, consonants, spaces and special characters.
18. to Swap Two Numbers using Pointers
19. to demonstrate student structure to read & display records of n students
20. to demonstrate the difference between structure & union.

CA-C5P: DATA STRUCTURES LAB

NOTE: For all the programs write the output, flowchart and number of basic operations performed.

1. Given {4,7,3,2,1,7,9,0} find the location of 7 using Linear and Binary search and also display its first occurrence.
2. Given {5,3,1,6,0,2,4} order the numbers in ascending order using Bubble Sort Algorithm
3. Perform the Insertion and Selection Sort on the input {75,8,1,16,48,3,7,0} and display the output in descending order.
4. Write a program to insert the elements {61,16,8,27} into singly linked list and delete 8,61,27 from the list. Display your list after each insertion and deletion.
5. Write a program to insert the elements {61,16,8,27} into linear queue and delete three elements from the list. Display your list after each insertion and deletion.
6. Write a program to insert the elements {61,16,8,27} into circular queue and delete 4 elements from the list. Display your list after each insertion and deletion.
7. Write a program to insert the elements {61,16,8,27} into ordered singly linked list and delete 8,61,27 from the list. Display your list after each insertion and deletion.
8. Write a program to add $6x^3+10x^2+0x+5$ and $4x^2+2x+1$ using linked list.
9. Write a program to push 5,9,34,17,32 into stack and pop 3 times from the stack, also display the popped numbers.
10. Write a recursive program to find GCD of 4,6,8.
11. Write a program to insert the elements {5,7,0,6,3,9} into circular queue and delete 6,9&5 from it(using linked list implementation)..
12. Write a program to convert an infix expression $x^y/(5*z)+2$ to its postfix expression
13. Write a program to evaluate a postfix expression $5\ 3+8\ 2\ -\ *$.
14. Write a program to create a binary tree with the elements {18,15,40,50,30,17,41} after creation insert 45 and 19 into tree and delete 15,17 and 41 from tree. Display the tree on each insertion and deletion operation
15. Write a program to create binary search tree with the elements {2,5,1,3,9,0,6} and perform inorder, preorder and post order traversal.
16. Write a program to Sort the following elements using heap sort {9,16,32,8,4,1,5,8,0}
17. Given $S1=\{\text{"Flowers"}\}$; $S2=\{\text{"are beautiful"}\}$ I. Find the length of S1 II. Concatenate S1 and S2 III. Extract the substring "low" from S1 IV. Find "are" in S2 and replace it with "is"

CA-C6T: COMPUTER ARCHITECTURE

Total Teaching Hours: 48

No. of Hours / Week: 03

UNIT - I

[12 Hours]

Number Systems: Binary, Octal, Hexa decimal numbers, base conversion, addition, subtraction of binary numbers, one's and two's complements, positive and negative numbers, character codes ASCII, EBCDIC. Computer Arithmetic: Addition and Subtraction, Multiplication and Division algorithms, Floating-point Arithmetic Operations, Decimal arithmetic operations. Structure of Computers: Computer types, Functional units, Basic operational concepts, Von-Neumann Architecture, Bus Structures, Software, Performance, Multiprocessors and Multicomputer, Digital Logic Circuits: Logic gates, Boolean algebra, Map Simplification. Combinational Circuits: Half Adder, Full Adder, flip flops. Sequential circuits: Shift registers, Counters, Integrated Circuits, Mux, Demux, Encoder, Decoder. Data representation: Fixed and Floating point.

UNIT - II

[12 Hours]

Basic Computer Organization and Design: Instruction codes, Computer Registers, Computer Instructions and Instruction cycle. Timing and Control, Memory-Reference Instructions, Input-Output and interrupt. Central processing unit: Stack organization, Instruction Formats, Addressing Modes, Data Transfer and Manipulation, Complex Instruction Set Computer (CISC) Reduced Instruction Set Computer (RISC), CISC vs RISC

UNIT - III

[12 Hours]

Register Transfer and Micro-operations: Register Transfer Language, Register Transfer, Bus and Memory Transfers, Arithmetic Micro-Operations, Logic Micro-Operations, Shift Micro-Operations, Arithmetic logic shift unit. Micro-programmed Control: Control Memory, Address Sequencing, Micro-Program example, Design of Control Unit. Input Output: I/O interface, Programmed IO, Memory Mapped IO, Interrupt Driven IO, DMA. Instruction level parallelism: Instruction level parallelism (ILP)-over coming data hazards, limitations of ILP

UNIT - IV

[12 Hours]

Memory System: Memory Hierarchy, Semiconductor Memories, RAM(Random Access Memory), Read Only Memory (ROM), Types of ROM, Cache Memory, Performance considerations, Virtual memory, Paging, Secondary Storage, RAID. Multiprocessors And Thread level Parallelism: Characteristics of multiprocessors, Multi-Threaded Architecture, Distributed Memory MIMD Architectures, Interconnection structures,

TEXT BOOKS:

1. Mano M Morris, "Computer System Architecture", 3rd edition Pearson India(2019).
2. William Stallings, "Computer Organization and Architecture designing for performance", 10th edition, Pearson(2016)

REFERENCE BOOKS

1. Subrata Ghoshal, "Computer Architecture And Organization", Pearson India(2011).
2. Andrew S. Tanenbaum "Structured Computer Organization", 5th edition, Pearson Education Inc(2006).
3. Carl Hamacher, Zvonks Vranesic, SafeaZaky, "Computer Architecture And Organization", 5th edition McGraw Hill New Delhi, India(2002).
4. Kai Hwang, "Advanced Computer Architecture - Parallelism, Scalability, Programmability", Tata Mcgraw-Hill (2008).

CA67T: OBJECT ORIENTED PROGRAMMING USING JAVA

Total Teaching Hours: 48

No. of Hours / Week: 03

UNIT-I

[12 Hours]

Introduction to Java: Basics of Java programming, Data types, Variables, Operators, Control structures including selection, Looping, Java methods, Overloading, Math class, Arrays in java. Objects and Classes: Basics of objects and classes in java, Constructors, Finalizer, Visibility modifiers, Methods and objects, Inbuilt classes like String, Character, String Buffer, File, this reference

UNIT-II

[12 Hours]

Inheritance and Polymorphism: Inheritance in java, Super and sub class, Overriding, Object class, Polymorphism, Dynamic binding, Generic programming, Casting objects, Instance of operator, Abstract class, Interface in java, Package in java, UTIL package.

UNIT-III

[12 Hours]

Event and GUI programming: Event handling in java, Event types, Mouse and key events, GUI Basics, Panels, Frames, Layout Managers: Flow Layout, Border Layout, Grid Layout, GUI components like Buttons, Check Boxes, Radio Buttons, Labels, Text Fields, Text Areas, Combo Boxes, Lists, Scroll Bars, Sliders, Windows, Menus, Dialog Box, Applet and its life cycle, Introduction to swing, Exceptional handling mechanism. I/O programming: Text and Binary I/O, Binary I/O classes, Object I/O, Random Access Files.

UNIT-IV

[12 Hours]

Multithreading in java: Thread life cycle and methods, Runnable interface, Thread synchronization, Exception handling with try-catch-finally, Collections in java, Introduction to JavaBeans and Network Programming.

Textbooks:

1. E. Balagurusamy, Programming with JAVA, McGraw Hill, New Delhi, 2007

Reference Books:

1. Raj Kumar Buyya, Object Oriented Programming with JAVA, McGraw Hill, 2009
2. Herbert Schildt, Java A Beginner's Guide – Create, Compile, and Run Java Programs Today, Sixth Edition, Oracle Press, 2014
3. Ken Arnold, James Gosling, "The Java Programming Language, Fourth Edition, Addison Wisely, 2005
4. Herbert Schildt, 'The Complete Reference Java, 7th Edition, McGraw Hill, 2007

Web Resources

1. <https://docs.oracle.com/javase/tutorial/>
2. <https://javabeginnerstutorial.com/core-java-tutorial/>

CA-C8T: DATABASE MANAGEMENT SYSTEMS

Total Teaching Hours: 48

No. of Hours / Week: 03

UNIT - 1

[12 Hours]

Databases and Database Users: Introduction, An example, Characteristics of the Database Approach, Actors on the Scene, Workers behind the Scene, Advantages of Using DBMS Approach, A Brief History of Database Applications, When Not to Use a DBMS. Database System Concepts and Architecture: Data Models, Schemas, and Instances, Three-schema Architecture and Data Independence, Database Languages and Interfaces, The Database System Environment, Centralized and Client-Server Architectures, Classification of Database Management Systems.

UNIT - 2

[12 Hours]

Data Modeling Using Entity-Relationship Model: Using High-Level Conceptual Data Models for Database Design, An Example Database Application, Entity Types, Entity Sets, Attributes and Keys, Relationship Types, Relationship Sets, Roles and Structural Constraints, Weak Entity Types, Refining the ER Design Company Database Diagrams, Naming Conventions and Design. Issues, File organization and storage, secondary storage devices, type of single level ordered index, multi-level indexes, indexes on multiple keys, other types of indexes.

UNIT – 3

[12 Hours]

Relational Model and Relational Algebra: Relational Model Concepts, Relational Model Constraints and Relational Database Schemas, Update Operations, Transactions and Dealing with Constraint Violations, Unary Relational Operations: SELECT and PROJECT, Relational Algebra Operations from SET Theory, Binary Relational Operations: JOIN and DIVISION, Additional Relational Operations, Examples of Queries in Relational Algebra. Relational Database Design: Anomalies in a database, functional dependency, normal forms, lossless join and dependency, BCNF, normalization through synthesis, higher order normal forms. SQL- SQL Data Definition and Data Types, Specifying Constraints in SQL, Schema Change Statements in SQL, Basic Queries in SQL, More Complex SQL Queries, Insert, Delete and Update Statements in SQL, Specifying Constraints as Assertion and Trigger, Views(Virtual Tables) in SQL, Embedded SQL, Dynamic SQL,

UNIT – 4

[12 Hours]

Introduction to transaction processing, transaction and system concepts, desirable properties of transactions, transaction support in SQL. Concurrency control techniques: two-phase locking techniques, concurrency control based on timestamp ordering. Recovery techniques: recovery concepts, recovery in multi-database systems, database backup and recovery from catastrophic failures.

Text Books:

1. Elmasri and Navathe: Fundamentals of Database Systems, 7th Edition, Addison -Wesley, 2016.
2. Silberschatz, Korth and Sudharshan Data base System Concepts, 7th Edition, Tata McGraw Hill, 2019.

References:

1. C.J. Date, A. Kannan, S. Swamynatham: An Introduction to Database Systems, 8th Edition, Pearson education, 2009
2. Database Management Systems :Raghu Ramakrishnan and Johannes Gehrke: , 3rd Edition, McGraw-Hill, 2003

CA-C9P: JAVA PROGRAMMING LAB

1. Write a simple java application, to print the message, “Welcome to java”
2. Write a program to display the month of a year. Months of the year should be held in an array.
3. Write a program to demonstrate a division by zero exception
4. Write a program to create a user defined exception say Pay Out of Bounds. .
5. Write a java program to add two integers and two float numbers. When no arguments are supplied, give a default value to calculate the sum. Use function overloading.
6. Write a program to perform mathematical operations. Create a class called AddSub with methods to add and subtract. Create another class called MulDiv that extends from AddSub class to use the member data of the super class. MulDiv should have methods to multiply and divide A main function should access the methods and perform the mathematical operations.
7. Write a program with class variable that is available for all instances of a class. Use static variable declaration. Observe the changes that occur in the object’s member variable values.
8. Write a java program to create a student class with following attributes: Enrollment_id: Name, Mark of sub1, Mark of sub2, mark of sub3, Total Marks. Total of the three marks must be calculated only when the student passes in all three subjects. The pass mark for each subject is 50. If a candidate fails in any one of the subjects his total mark must be declared as zero. Using this condition write a constructor for this class. Write separate functions for accepting and displaying student details. In the main method create an array of three student objects and display the details.
9. In a college first year class are having the following attributes Name of the class (BCA, BCom, BSc), Name of the staff No of the students in the class, Array of students in the class
10. Define a class called first year with above attributes and define a suitable constructor. Also write a method called best Student () which process a first-year object and return the student with the highest total mark. In the main method define a first-year object and find the best student of this class
11. Write a Java program to define a class called employee with the name and date of appointment. Create ten employee objects as an array and sort them as per their date of appointment. ie, print them as per their seniority.
12. Create a package ‘ student.Fulltime.BCA ‘ in your current working directory
 - a. Create a default class student in the above package with the following attributes: Name, age, sex. b. Have methods for storing as well as displaying
13. Write a small program to catch Negative Array Size Exception. This exception is caused when the array is initialized to negative values.
14. Write a program to handle Null Pointer Exception and use the “finally” method to display a message to the user.
15. Write a program which create and displays a message on the window
16. Write a program to draw several shapes in the created window
17. Write a program to create an applet and draw grid lines
18. Write a program which creates a frame with two buttons father and mother. When we click the father button the name of the father, his age and designation must appear. When we click mother similar details of mother also appear.
19. Create a frame which displays your personal details with respect to a button click
20. Create a simple applet which reveals the personal information of yours.
21. Write a program to move different shapes according to the arrow key pressed.
22. Write a java Program to create a window when we press M or m the window displays Good Morning, A or a the window displays Good After Noon E or e the window displays Good Evening, N or n the window displays Good Night
23. Demonstrate the various mouse handling events using suitable example.
24. Write a program to create menu bar and pull-down menus.

CA- C10P: DATABASE MANAGEMENT SYSTEMS LAB**PART – A**

1. Draw E-R diagram and convert entities and relationships to relation table for a given scenario.
Two assignments shall be carried out i.e. consider two different scenarios (eg. bank, college)

Consider the Company database with following Schema

EMPLOYEE (FNAME, MINIT, LNAME, SSN, BDATE, ADDRESS, SEX, SALARY, SUPERSSN, DNO)

DEPARTMENT (DNAME, DNUMBER, MGRSSN, MSRSTARTDATE)

DEPT_LOCATIONS (DNUMBER, DLOCATION)

PROJECT (PNAME, PNUMBER, PLOCATION, DNUM)

WORKS_ON (ESSN, PNO<HOURS)

DEPENDENT (ESSN, DEPENDENT_NAME, SEX, BDATE, RELATIONSHIP)

2. Perform the following:
 - a. Viewing all databases, Creating a Database, Viewing all Tables in a Database, Creating Tables (With and Without Constraints), Inserting/Updating/Deleting Records in a Table, Saving (Commit) and Undoing (rollback)
3. Perform the following:
 - a. Altering a Table, Dropping/Truncating/Renaming Tables, Backing up / Restoring a Database.
4. For a given set of relation schemes, create tables and perform the following Simple Queries, Simple Queries with Aggregate functions, Queries with Aggregate functions (group by and having clause).
5. Execute the following queries
 - a. How the resulting salaries if every employee working on the 'Research' Departments is given a 10% raise.
 - b. Find the sum of the salaries of all employees of the 'Accounts' department, as well as the maximum salary, the minimum salary, and the average salary in this department
6. Execute the following queries
 - a. Retrieve the name of each employee Controlled by Department number 5 (use EXISTS operator).
 - b. Retrieve the name of each dept and number of employees working in each Department which has at least 2 employees
7. Execute the following queries
 - a. For each project, retrieve the project number, the project name, and the number of employee who work on that project.(use GROUP BY)
 - b. Retrieve the name of employees who born in the year 1990's
8. For each Department that has more than five employees, retrieve the department number and number of employees who are making salary more than 40000.
9. For each project on which more than two employees work, retrieve the project number, project name and the number of employees who work on that project.

10. For a given set of relation tables perform the following: Creating Views (with and without check option), Dropping views, Selecting from a view

PART B

Create the following tables with properly specifying Primary keys, Foreign keys and solve the following queries.

BRANCH (Branchid, Branchname, HOD)
 STUDENT (USN, Name, Address, Branchid, sem)
 BOOK (Bookid, Bookname, Authorid, Publisher, Branchid)
 AUTHOR (Authorid, Authorname, Country, age)
 BORROW (USN, Bookid, Borrowed_Date)

1. Perform the following:
 - a. Viewing all databases, Creating a Database, Viewing all Tables in a Database, Creating Tables (With and Without Constraints), Inserting/Updating/Deleting Records in a Table, Saving (Commit) and Undoing (rollback)
Execute the following Queries:
2.
 - a. List the details of Students who are all studying in 2nd sem BCA.
 - b. List the students who are not borrowed any books.
3.
 - a. Display the USN, Student name, Branch_name, Book_name, Author_name, Books_Borrowed_Date of 2nd sem BCA Students who borrowed books.
 - b. Display the number of books written by each Author.
4.
 - a. Display the student details who borrowed more than two books.
 - b. Display the student details who borrowed books of more than one Author.
5.
 - a. Display the Book names in descending order of their names.
 - b. List the details of students who borrowed the books which are all published by the same publisher.

Consider the following schema:

STUDENT (USN, name, date_of_birth, branch, mark1, mark2, mark3, total, GPA)

6. Perform the following:
 - a. Creating Tables (With and Without Constraints), Inserting/Updating/Deleting Records in a Table, Saving (Commit) and Undoing (rollback)
7. Execute the following queries:
 - a. Find the GPA score of all the students.
 - b. Find the students who born on a particular year of birth from the date_of_birth column.
8.
 - a. List the students who are studying in a particular branch of study.
 - b. Find the maximum GPA score of the student branch-wise.

CA-C11T: OPERATING SYSTEMS

Total Teaching Hours: 48

No. of Hours / Week: 03

UNIT - I [12 Hours]

Introduction: Computer System Organization, Architecture, Structure, Operations, Process Management, Memory Management, Storage Management, Kernel Data Structures, Computing Environments. Operating System Structures: Services, System Calls, Types, Operating System Structure, System Boot. Processes: Process Concept, Scheduling, Operations, Interprocess Communication. Multithreaded Programming: Multicore Programming, Multithreading Models.

UNIT -II [12 Hours]

Process Synchronization: The Critical-Section Problem, Peterson's Solution, Synchronisation Hardware, Mutex Locks, Semaphores, Classic Problems of Synchronization, Monitors, Synchronization Examples. Process Scheduling: Criteria, Scheduling Algorithms, Multi-Processor Scheduling, Real-time CPU Scheduling. Deadlocks: System model, Characterization, Methods for handling deadlocks, Deadlock Prevention, Avoidance, Detection and Recovery from deadlock.

UNIT - III [12 Hours]

Memory Management Strategies: Background, Swapping, Contiguous Memory Allocation, Segmentation, Paging, Structure of the Page Table. Virtual Memory Management: Demand Paging; Copy-on-Write, Page Replacement; Allocation of Frames; Thrashing, Memory-Mapped Files, Allocating Kernel Memory. File System: File Concept, Access Methods, Directory and Disk Structure, Protection. File-System Implementation: Structure

UNIT - IV [12 Hours]

File-System and Directory Implementation, Allocation Methods, Free Space Management, Efficiency and Performance, Recovery. Mass-Storage Structure: Overview, Disk Scheduling, Disk Management. Distributed Systems: Advantages, Types of Network- based OS, Robustness, Design Issues, Distributed File Systems. Case Studies: The Linux System, Windows 10 (Process, Memory, storage management).

Text Books:

1. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne: Operating Systems Concepts, 9th Edition, 2016 India, Wiley.

Reference Books:

1. William Stallings, "Operating Systems-Internals and Design Principles", Pearson, IX Edition, 2018
2. D M Dhamdhare: Operating Systems – A Concept Based Approach, III Edition, Tata McGraw – Hill, 2015.
3. Harvey M Deitel, Paul J Deitel, Dr Choffnes, "Operating Systems", Pearson Education Limited (Publisher), 3rd Edition, 2013.
4. J. Archer Harris, John Cordani, " Operating Systems", Schaum's Outline, Indian Edition, Mc Graw Hill Education (India), First Edition
5. Gary Nutt, Nabendu Chaki, Sarmistha Neog, "Operating Systems" Pearson Education Limited, 3rd Edition, 2016.

CA-C12T: COMPUTER NETWORKS

Total Teaching Hours : 48

No. of Hours / Week: 03

UNIT - I [12 Hours]

Introduction: Data Communications, Networks, Network Types, Internet History, Network Models: Protocol Layering, The OSI Model, TCP/IP Protocol Suite, Introduction to Physical Layer: Transmission Impairments, Data Rate Limits, Performance, Introduction to Data-Link- Layer: Link-Layer Addressing, Error Detection and Correction: Block Coding, Cyclic Codes, Checksum

UNIT - II [12 Hours]

Data Link Control: Data-Link Layer Protocols, HDLC, Point-To-Point (PPP), Media Access Control (MAC): ALOHA, CSMA, CSMA/CD, CSMA/CA, Reservation, Polling, Token Passing, FDMA, TDMA, CDMA

UNIT - III [12 Hours]

Introduction to Network Layer: Network-Layer Services, Packet Switching, Network-Layer Performance, IPV4 Addresses, Network Layer Protocols: Internet Protocol (IP), ICMPv4, Mobile IP, Unicast Routing: Routing Algorithms, Unicast Routing Protocols, Next Generation IP: IPv6 Addressing

UNIT - IV [12 Hours]

Introduction to Transport Layer: Introduction, Transport-Layer Protocols, Transport-Layer Protocols: User Datagram Protocol, Transmission Control Protocol: TCP Services, TCP Features, Segment, A TCP Connection, TCP Congestion Control, Flow Control, Error Control, Application Layer: WWW, E-MAIL, Domain Name System (DNS), Quality of Service: Flow Control To Improves QoS, Integrated Services

Text Books:

1. Behrouz A. Forouzan, "Data Communications and Networking", 5th Edition, McGraw Hill Education, 2013.

Reference Books:

1. Andrew S. Tanenbaum, David J. Wetherall, "Computer Networks", 5th Edition, Prentice Hall, 2011.
2. Larry L. Peterson and Bruce S. Davie, "Computer Networks A System Approach", 5th Edition, MKP, 2012.
3. James F. Kurose , Keith W. Ross, " Computer Networking, A Top-Down Approach", 5th Edition, Pearson, 2012.

Web Resources:

1. <https://www.geeksforgeeks.org/computer-network-tutorials/>
2. <https://codescracker.com/networking/>
3. https://youtube.com/playlist?list=PLxCzCOWd7aiGFBD2-2joCpWOLUrDLvVV_

CA-C13T: PYTHON PROGRAMMING

Total Teaching Hours : 48

No. of Hours / week: 03

UNIT - I [12 Hours]

Parts Python Programming Language: Python Interpreter/Shell, Identifiers, Keywords, Statements and Expressions, Variables, Operators, Precedence and Associativity, Data types, Indentation, Comments, Reading Input, Print Output, Type Conversions, The type() function and Is operator, Dynamic and Strongly Typed Language. **Control Flow Statements:** The if Decision Control Flow Statement, The if...else DecisionControl FlowStatement, The if...elif... else Decision Control Statement, Nested if Statement, The while Loop, TheforLoop, The continue and break Statements. **Functions:** Built-In Functions, Commonly Used Modules, Function Definition and Calling the Function, The return Statement and void Function, Scope and Life time of Variables, Default Parameters, Command Line Arguments. **Strings:** Creating and Storing Strings, Basic String operations, Accessing Characters in StringbyIndexNumber, String Slicing and Joining, String methods

UNIT - II [12 Hours]

Lists: Creating Lists, BasicListOperations, Indexing and Slicing in Lists, Built-In Functions Used on Lists, List Methods, ThedelStatement, **Dictionaries:** Creating Dictionary, Accessing and modifying key:value pairs in Dictionaries, Built-In Functions Used on Dictionaries, Dictionary methods, ThedelStatement. **Tuples and Sets:** Creating Tuples, Basic Tuple Operations, Indexing and Slicing in Tuples, Built-In Functions Used onTuples, Relations between Tuples and Lists, Relations between Tuples and Dictionaries, Tuple Methods, Using zip() Function, Sets, Set Methods, Frozenset.

UNIT - III [12 Hours]

Files: Types of files, Creating and Reading Text Data, File Methods to Read and Write Data, Reading and Writing Binary Files, The Pickle module, Reading and writing CSV files, **Object-Oriented Programming:** Classes and Objects, Creating Classes in Python, Creating Objects in Python, The Constructor Method, Classes with Multiple Objects, Class Attributes versus Data attributes, Encapsulation, Inheritance, The Polymorphism.

UNIT - IV [12 Hours]

Data Visualization: Generating Data-Installing Matplotlib, Plotting a Simple Line Graph, Random Walks, Rolling Dice with Plotly. Downloading Data- The CSV File Format, Mapping Global Data Sets: JSON Format, **Working with APIs:** Using a Web API, Visualizing Repositories Using Plotly.

Text Books:

1. Gowrishankar S, Veena A, "Introduction to Python Programming", 1st Edition, CRC Press/Taylor & Francis, 2018. ISBN-13: 978-0815394372. [Unit I&II- 2,3,4,5,6,7,8,9 Unit III-11,12].
2. Eric Matthes, "Python Crash Course- A Hands-On, Project-Based Introduction to Programming", 2nd Edition, No Starch Press, 2019. [Unit III-15, Unit IV-16]
3. Wesley J. Chun, "Core Python Applications Programming", 3rd Edition, Pearson Education, 2016. [Unit IV- 2,5].

Reference Books:

1. Kamthane, A. N., & Kamthane, A.A. , "Programming and Problem Solving with Python", McGraw Hill Education, 2017.
2. Mark Lutz, "Learning Python", 5th edition, Orelly Publication, 2013, ISBN 978- 1449355739.
3. Ljubomir Perkovic, "Introduction to Computing Using Python- An Application Development Focus", Wiley, 2012.

CA-C14L: COMPUTER NETWORKS LAB

1. Execute the following commands:
arp, ipconfig, hostname, netdiag, netstat, nslookup, pathping, ping route, tracert
2. Study of different types of network cables.
3. Practically implement the cross-wired cable and straight wired cable using crimping tool.
4. Study of network IP address configuration: (Classification of address, static and dynamic address)
5. Study of network IP address configuration: (IPv4 and IPv6, Subnet, Supernet)
6. Study of network devices: (Switch, Router, Bridge)
7. Configure and Connect the computer in LAN.
8. Block the website using "Windows Defender Firewall" in windows 10.
9. Share the folder in a system, and access the files of that folder from other system using IP address .
10. Share the printer in Network, and take print from other PC.
11. Configuration of wifi hotspot, and connect other devices (mobile / laptop).
12. Configuration of switches.
13. Configuration of I/O box fixing.
14. Making your own patch cord.
15. Configuration of VLAN using Packet Tracer/ GNS3
16. Configuration of VPN using Packet Tracer/ GNS3

References:

1. Paul Browning, "101 CompTIA Networks+ LABS", 2018, Reality Press Ltd.

Web References:

1. [youtube.com/watch?v=rurs7cdT5cc](https://www.youtube.com/watch?v=rurs7cdT5cc)
2. https://www.youtube.com/watch?v=_IOZ8_cPgu8
3. <https://www.alphr.com/block-websites-windows/>

CA-C15L: PYTHON PROGRAMMING LAB

1. Write a program to demonstrate basic data type in python
2. Create a list and perform the following methods
 - 1) insert() 2) remove() 3) append()
 - 4) len() 5) pop() 6) clear()
3. Create a tuple and perform the following methods
 - 1) Add items 2) len() 3) check for item in tuple 4) Access items
4. Create a dictionary and apply the following methods
 - 1) Print the dictionary items 2) access items 3) use get()
 - 4) change values 5) use len()
5. Write a program to create a menu with the following options
 1. TO PERFORM ADDITION 2. TO PERFORM SUBTRACTION
 3. TO PERFORM MULTIPLICATION 4. TO PERFORM DIVISION
 Accepts users input and perform the operation accordingly. Use functions with arguments.
6. Write a python program to print a number is positive/negative using if-else.
7. Write a program for filter() to filter only even numbers from a given list.
8. Write a python program to print date, time for today and now
9. Write a python program to add some days to your present date and print the date added.
10. Write a program to count the numbers of characters in the string and store them in a dictionary data structure
11. Write a program to count frequency of characters in a given file.
12. Using a numpy module create an array and check the following: 1. Type of array 2. Axes of array 3. Shape of array 4. Type of elements in array
13. Write a python program to concatenate the dataframes with two different objects
14. Write a python code to read a csv file using pandas module and print the first and last five lines of a file.
15. Write a python program which accepts the radius of a circle from user and computes the area (use math module)
16. Use the following data (load it as CSV file) for this exercise. Read this file using Pandas or NumPy or using in-built matplotlib function.

Months	Pen	Book	Marker	Chair	Table	Pen stand	Total units	Total profit
1	2500	1500	5200	9200	1200	1500	21100	211000
2	2630	1200	5100	6100	2100	1200	18330	183300
3	2140	1340	4550	9550	3550	1340	22470	224700
4	3400	1130	5870	8870	1870	1130	22270	222700
5	3600	1740	4560	7760	1560	1740	20960	209600
6	2760	1555	4890	7490	1890	1555	20140	201400
7	2980	1120	4780	8980	1780	1120	29550	295500
8	3700	1400	5860	9960	2860	1400	36140	361400
9	3540	1780	6100	8100	2100	1780	23400	234000
10	1990	1890	8300	10300	2300	1890	26670	266700
11	2340	2100	7300	13300	2400	2100	41280	412800
12	2900	1760	7400	14400	1800	1760	30020	300200

- a. Get total profit of all months and show line plot with the following Style properties
Generated line plot must include following Style properties: –

- Line Style dotted and Line-color should be blue
 - Show legend at the lower right location.
 - X label name = Months
 - Y label name = Sold units
 - Line width should be 4
- b. Display the number of units sold per month for each product using multiline plots. (i.e., Separate Plotline for each product.
- c. Read chair and table product sales data and show it using the bar chart.
- The bar chart should display the number of units sold per month for each product. Add a separate bar for each product in the same chart.
- d. Read all product sales data and show it using the stack plot

SEC II: COMPUTER ASSEMBLY

1. Demonstration of Hardware peripherals: CPU, RAM, SMPS, Motherboard, NIC card, Processor, Processor cooling fan, PCI card, HDD.
2. Demonstration of various ports: CPU , VGA port, PS/2 (keyboard, mouse) ,USB, LAN, Speaker, Audio.
3. Identify the Computer Name and Hardware Specification (RAM capacity, Processor type, HDD, 32 bit/ 64 bit)
4. Identify and Troubleshoot the problems of RAM (beep sound with blue screen), SMPS and motherboard (CPU is not switched ON)
5. Configure BIOS settings- disable and enable USB and LAN.
6. Identify, how to recover the hidden files from corrupted pendrive using command.
7. Recover the contents from crashed Hard Disk using Disk Drill software.
8. Install Operating System – Windows family (Windows 7/ Windows 10) and also make partitions.
9. Install Operating System - Unix family (Linux/UBUNTU)
10. Install Application software – python 3.8, MS- Office 2010/2013, MySQL, TOAD, Openoffice, etc.,
11. Install any one of the antivirus software (Avast, Kaspersky, etc..) and observe the variations before and after installation.
12. Add new Hardware device (keyboard, mouse, Speaker, Microphone)
13. Connect the LCD Projector with Laptop / CPU.
14. Adding additional RAM to the system.(expanding RAM size).
15. Graphic Card insertion.
16. Assemble and Disassemble Desktop System.

References:

1. Dan Gookin ,Troubleshooting & Maintaining Your PC ALL-IN-ONE, 3rd Edition,2017, John Wiley & Sons.
2. Mike Meyers, Scott Jernigan, Dan Lachance, ”CompTIA Fundamentals + Exam Guide (All-in-One), 2nd Edition, 2019, Mc Graw Hill Education.

Web References:

1. https://www.youtube.com/watch?v=ItxwyMR0SnY&list=PLeH4ngtDM7eE-I_mdWuXWyZrI_FMHnyJ0&index=5
2. <https://www.cleverfiles.com/howto/crashed-hard-drive-recovery.html>

CA-C16T : SOFTWARE ENGINEERING

Total Teaching Hours: 48

No. of Hours / Week: 03

UNIT – I

[12 Hours]

Introduction to Software Engineering: Evolution and impact of Software engineering, software life cycle models: Waterfall, prototyping, Evolutionary, and Spiral models. Feasibility study, Functional and Non-functional requirements, Requirements gathering, Requirements analysis and specification.

Agile development: Agile, Agility and cost of change; Agile Process, Extreme programming; Other agile process models.

UNIT - II

[12 Hours]

Formal Modeling and verification: The cleanroom strategy; Functional specification; Cleanroom design; Cleanroom testing; Formal methods: Concepts; Applying mathematical notation for formal specification; Formal specification languages. Software Project Management: The management spectrum; The management of people, product, process and project; The W5HH Principle; Critical practices. **Software testing strategies:** A Strategic Approach to Software Testing, Test Strategies for Conventional Software, Test Strategies for Object-Oriented Software, Test Strategies for WebApps, System Testing, Software Testing Fundamentals, White-Box Testing, Black-Box Testing

UNIT - III

[12 Hours]

Software Project Scheduling: Basic concepts and principles of project scheduling; Defining task set and task network; Scheduling; Earned value analysis. Risk Management: Reactive versus proactive strategies; Software risks; risk identification; Risk projection; Risk refinement; Risk mitigation, monitoring and management; The RMMM plan. Maintenance and Reengineering: Software maintenance; Software supportability; Reengineering; Business process reengineering; Software reengineering; Reverse engineering; Restructuring; Forward engineering; The economics of reengineering.

UNIT - IV

[12 Hours]

Software Process Improvement (SPI): Approaches to SPI; Maturity models; The SPI process; The CMMI; The People CMM; Other SPI frameworks: SPICE, Bootstrap, PSP and TSP, ISO; SPI return on investment. Software Configuration Management (SCM): Basic concepts; SCM repository; The SCM process; Configuration management for web applications; SCM standards.

Text Books:

1. Fundamentals of Software Engineering by Rajib Mall, – PHI-3rd Edition, 2009.
2. Roger S. Pressman, “*Software Engineering: A Practitioner’s Approach*”, Alternate Edition, 7th Edition, McGraw Hill, 2010.

Reference Books:

1. Software Engineering, by Ian Sommerville, Pearson Education Inc., New Delhi, (2009).
2. Software Engineering: A Practitioner’s Approach”, by Roger S. Pressman, McGraw-Hill. (2005).
3. Pankaj Jalote, “An Integrated Approach to Software Engineering”, Narosa Publishing House Pvt Ltd, Darya Ganj, New Delhi 110002

CA-C17T : THE DESIGN AND ANALYSIS OF ALGORITHMS

Total Teaching Hours: 48

No. of Hours / Week: 03

UNIT - I

[12 Hours]

Introduction: Algorithms, Fundamentals of Algorithmic Problem Solving, Important Problem Types, Fundamental Data Structures. Fundamentals of the Analysis of Algorithm Efficiency: The Analysis Framework, Asymptotic Notations and Basic Efficiency Classes, Mathematical Analysis of Non-recursive and Recursive Algorithms, Empirical Analysis of Algorithms

UNIT – II

[12 Hours]

Brute Force Method: Selection Sort and Bubble Sort, Sequential Search, Brute-Force String Matching, Exhaustive Search, Depth-First Search and Breadth-First Search. Decrease and Conquer: Insertion Sort, Topological Sorting, Algorithms for Generating Combinatorial Objects, Decrease-by-a-Constant-Factor Algorithms. Divide and Conquer: Merge Sort, Quick Sort, Binary Tree Traversals and Related Properties, Strassen's Matrix Multiplication.

UNIT - III

[12 Hours]

Space and Time Tradeoffs: Sorting by Counting, Input Enhancement in String Matching, Hashing. Dynamic programming: Binomial Coefficient, Principle of Optimality, Optimal Binary Search Trees, Knapsack Problem and Memory Functions, Warshall's and Floyd's Algorithms. Greedy Technique: Prim's Algorithm, Kruskal's Algorithm, Dijkstra's Algorithm, Huffman Trees.

UNIT – IV

[12 Hours]

Limitations of Algorithm Power: Lower-Bound Arguments, Decision Trees, P, NP and NP Complete Problems. Coping with the Limitations of Algorithm Power: Back Tracking: n Queens problem, Hamiltonian Circuit Problem, Subset-Sum Problem. Branch-and-Bound: Assignment Problem, Knapsack Problem, Traveling Salesman Problem.

Textbooks:

1. Anany Levitin, "Introduction to the Design and Analysis of Algorithms", 3rd Edition, Pearson, 2012.
2. Horowitz, Sahni, Rajasekaran, "Fundamentals of Computer Algorithms", 2/e, Universities Press, 2007.

Reference Books:

1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, "Introduction to Algorithms", 3rd Edition, The MIT Press, 2009.
2. A.V. Aho, J.E. Hopcroft, J.D. Ullmann, "The design and analysis of Computer Algorithms", Addison Wesley Boston, 1983.
3. Jon Kleinberg, Eva Tardos, "Algorithm Design", Pearson Education, 2006.

CA-C18T : INTERNET TECHNOLOGIES

Total Teaching Hours : 48

No. of Hours / Week: 03

UNIT - I

[12 Hours]

INTERconnected NETWORK: Internet: The Giant Wide Area Network, Communicating over the Internet, Accessing the Internet, Internet Organisations, Cyber Ethics, **Internet Applications:** Internet services, Electronic Mail(E-Mail), File Transfer, Real-Time User Communication, Remote Login, Usenet, **World Wide Web:** The Web, The Working Web, Web Terminology, Web Architecture, World Wide Web Challenges.

UNIT - II

[12 Hours]

Hypertext Transfer Protocol (HTTP):HTTP, HTTP Version, HTTP connections, HTTP Communication, Hypertext Transfer Protocol Secure, Hypertext Transfer Protocol State Retention: Cookies, Hypertext Transfer Protocol Cache, **Evolution of Web:** The Generations of Web, Web 1.0, Web 2.0, Web 3.0, Big Data: A Special Discussion, **Web IR: Information Retrieval on the Web:** Web Information Retrieval, Web Information Retrieval Tools, Web Information Retrieval Architecture (Search Engine Architecture), Web Information Retrieval Performance Metrics, Web Information Retrieval Models, Google PageRank.

UNIT - III

[12 Hours]

Web Development Basics: Elements of Web Development , Client-Side and Server-Side Scripting, Model-View-Controller Architecture for Web Application Development, **Client-Side Technologies:** HTML: Hypertext Markup Language, CSS: Cascading Style Sheets, JavaScript, Bootstrap Framework, AngularJS Framework, **Server-Side Technologies:** Server-Side Scripting, Personal Home Pages, Node.js: Server-Side JavaScript.

UNIT - IV

[12 Hours]

Web Application Frameworks:Django ,Ruby on Rails.**Web Databases:** Web Database, Structured Query Language: Relational Databases, NoSQL Databases: Non-relational and Distributed Data, Understanding Popular Databases. **Research Trends on the Web:** Contextual Information Retrieval, Web Mining.

TEXT BOOKS:

1. Akshi Kumar, "Web Technology: Theory and Practice", CRC Press, 2019.

REFERENCE BOOKS :

1. Web Technology: A Developer's Perspective, N.P. Gopalan and J. Akilandeswari, PHI, Learning, Delhi, 2013.
2. Internetworking Technologies, An Engineering Perspective, Rahul Banerjee, PHI Learning, Delhi, 2011.

CA-C19L: Algorithms Lab

1. Write a program to implement linear search algorithm Repeat the experiment for different values of n , the number of elements in the list to be searched and plot a graph of the time taken versus n .
2. Write a program to implement binary search algorithm. Repeat the experiment for different values of n , the number of elements in the list to be searched and plot a graph of the time taken versus n .
3. Write a program to solve towers of honai problem and execute it for different number of disks
4. Write a Program to Sort a given set of numbers using selection sort algorithm. Repeat the experiment for different values of n , the number of elements in the list to be sorted and plot a graph of the time taken versus n . The elements can be read from a file or can be generated using the random number generator.
5. Write a program to find the value of a^n (where a and n are integers) using both brute-force based algorithm and divide and conquer based algorithm
6. Write a Program to Sort a given set of elements using quick sort algorithm. Repeat the experiment for different values of n , the number of elements in the list to be sorted and plot a graph of the time taken versus n .
7. Write a Program to find the binomial co-efficient $C(n, k)$, [where n and k are integers and $n > k$] using brute force based algorithm and also dynamic programming based algorithm
8. Write a Program to implement Floyd's algorithm and find the lengths of the shortest paths from every pairs of vertices in a given weighted graph
9. Write a program to evaluate a polynomial using brute-force based algorithm and using Horner's rule and compare their performances
10. Write a Program to solve the string matching problem using Boyer-Moore approach.
11. Write a Program to solve the string matching problem using KMP algorithm
12. Write a program to implement BFS traversal algorithm
13. Write a program to find the minimum spanning tree of a given graph using Prim's algorithm
14. Write a Program to obtain the topological ordering of vertices in a given digraph. Compute the transitive closure of a given directed graph using Warshall's algorithm.
15. Write a Program to Find a subset of a given set $S = \{s_1, s_2, \dots, s_n\}$ of n positive integers whose sum is equal to a given positive integer d . For example, if $S = \{1, 2, 5, 6, 8\}$ and $d = 9$ there are two solutions $\{1, 2, 6\}$ and $\{1, 8\}$. A suitable message is to be displayed if the given problem instance doesn't have a solution.

CA-C20L :INTERNET TECHNOLOGIES LAB

1. Demonstrate E-Mail working (Sending ,Receiving, forward)
2. How to create, organize meeting in Zoom/ GoogleMeet
3. Create a form by using various attributes of the input tags (text box, multiline textbox, option button, check box)
4. Create a simple HTML page by using some of the basic tags (hyperlink, marquee, image)
5. Create a web page with multiple types of style sheet used in a single page
6. Write a CGI sample program to send output back to the user
7. Create Time-Table using table tag
8. Creation of Frames in browser window using HTML.
9. Write a java script program to create dialogue boxes using alert, confirm and prompt methods
10. Write a java script program on Form Validations.
11. Write a java script program to perform four arithmetic operations: Addition, Subtraction, Multiplication and Division on two numbers.
12. Create a web site of our College.