

MCA-14-33**DESIGN AND ANALYSIS OF ALGORITHMS**

Maximum marks: 100 (External: 80, Internal: 20)

Time: 3 hours

Note: Examiner will be required to set NINE questions in all. Question Number 1 will consist of objective type/short-answer type questions covering the entire syllabus. In addition to question no. 1, the examiner is required to set eight more questions selecting two from each unit. Student will be required to attempt FIVE questions in all. Question Number 1 will be compulsory. In addition to compulsory question, student will have to attempt four more questions selecting one question from each Unit. All questions will carry equal marks.

UNIT – I

Introduction: Algorithms, Role of algorithms in computing, Complexity of algorithms, Analyzing algorithms, designing algorithms, asymptotic notations.

Divide and Conquer: Complexity of iterative programs and recursive programs, solving recurrence equations: back substitution method, recursion tree method, masters theorem.

Analysis of heap sort and quick sort; Counting sort, Radix sort, Bucket sort, Lower bounds for sorting.

UNIT – II

Hash Tables, Hash functions, Collision handling in hashing, analyzing various operations on Binary search tree. Introduction to Red-black trees.

Dynamic Programming (DP): Elements of DP, Matrix chain multiplication, Longest common subsequence, optimal binary search trees.

UNIT – III

Greedy Techniques (GT): Elements of GT, Activity selection problem, Huffman codes, Knapsack Problem.

Graph Algorithms: Single source shortest path: Analysis of Dijkstra's Algorithm, Limitations of Dijkstra's Algorithm, Negative weight cycle, Bellman-Ford algorithm. All Pairs Shortest Path: Relation of Shortest path and matrix multiplication, Analysis of Floyd Warshall algorithm. Maximum Flow: Flow network, Ford-Fulkerson method.

UNIT – IV

Strings: Storage of strings, naive string-matching algorithm, Rabin-Karp string matching algorithm.

Computational complexity: Notion of Polynomial time algorithms, Complexity classes: P, NP, NP-Hard and NP-Complete, Polynomial time verification, Reducibility, NP-Completeness, Examples of NP-Complete and NP-Hard problems: Traveling Salesman Problem, Knapsack, Bin Packing, Satisfiability, Vertex Cover, Clique, Independent Set. Introduction to approximation algorithms.

Text Books:

1. Cormen, Leiserson, Rivest, "Introduction to Algorithms", PHI India.
2. Neapolitan R., "Foundations of Algorithms", Jones and Bartlett Learning.

Reference Books:

1. Cooper A., "Computability Theory", Chapman and Hall/ CRC Press.
2. Robert Sedgewick, "Algorithms in C", Pearson Education India.
3. Steven Skiena, "The Algorithm Design Manual", Springer India.
4. Reiter, Johnson, "Limits of Computation", Chapman and Hall/ CRC Press.