P-880

Total Pages : 3 Roll No. .....

MCS-507

Design and Analysis of Algorithm

(MCA/MSCIT)

2nd / 4th Semester Examination, 2023 (June)

Time : 2 Hours] **Max. Marks : 70**

Note : This paper is of Seventy (70) marks divided into two (02) Sections A and B. Attempt the questions contained in these sections according to the detailed instructions given therein. Candidates should limit their answer to the questions on the given answer sheet. No additional (B) answer sheet will be issued.

SECTION–A

(Long Answer Type Questions)

Note : Section 'A' contains Five (05) long answer type questions of Nineteen (19) marks each. Learners are required to answer any Two (02) questions only.

(2×19=38)

1. Define Travelling Salesman Problem (TSP). Explain the basic steps that are to be followed to solve TSP using branch and bound.

2. Write and explain merge sort algorithm using divide and conquer strategy. Also analyze the complexity of merge sort algorithm.

3. State fractional knapsack problem. Give an algorithm for fractional knapsack problem using greedy strategy.

4. State and Explain N Queens Problem. Write the backtracking algorithm for solving N Queens problem.

5. Write the Kruskal's algorithm for finding minimum cost spanning tree and explain the Kruskal's algorithm with an example. Analyse the complexity of the algorithm.

SECTION–B

(Short Answer Type Questions)

Note : Section 'B' contains Eight (08) short answer type questions of Eight (08) marks each. Learners are required to answer any Four (04) questions only. (4×8=32)

1. Define algorithm. What are the characteristics of an algorithm?

2. What is binary search tree? Explain with example, how an element is searched in binary search tree?

3. Explain recursion in details with an example.

4. Write down and explain Greedy algorithm.

5. Write notes on Dynamic Programming Approach. List the sequence of steps to be followed in Dynamic Programming.

6. Define Strongly Connected Components of a graph with an example.

7. Write the algorithm for DFS and analyze its complexity.

8. Explain asymptotic notations in algorithm analysis.