

Roll No.

Total No. of Pages : 02

Total No. of Questions : 09

B.Tech. (SE) (Sem.-4)  
**DESIGN AND ANALYSIS OF ALGORITHMS**

Subject Code : SE404-19

M.Code : 80028

Date of Examination : 07-07-22

Time : 3 Hrs.

Max. Marks : 60

**INSTRUCTIONS TO CANDIDATES :**

1. SECTION-A is COMPULSORY consisting of TEN questions carrying TWO marks each.
2. SECTION-B contains FIVE questions carrying FIVE marks each and students have to attempt any FOUR questions.
3. SECTION-C contains THREE questions carrying TEN marks each and students have to attempt any TWO questions.

**SECTION-A**

1. Write briefly :

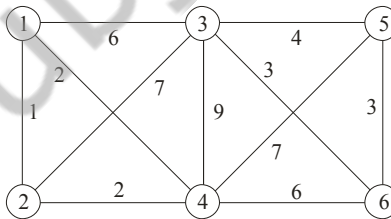
- a) Using the definitions of  $O$  and  $\Omega$ , show that:  $6n^2 + 20n$  has  $O(n^3)$  but not  $\Omega(n^3)$
- b) Solve the, following recurrence:  $f(n) = 9f(n/3) + n^2$  for  $n > 2, f(1) = 1$
- c) What do you mean by a negative cycle in a graph? Which algorithm detects one?
- d) Define transitive closure of a graph. What is its importance?
- e) What do you mean by all pair shortest path problem?
- f) What is a minimum spanning tree?
- g) What are approximation algorithms?
- h) What is Satisfiability Theorem?
- i) Is Heuristics a good way for solving comple problems? Justify
- j) What is algorithm Analysis?

## SECTION-B

- Let  $A[1..n]$  be an array of  $n$  integers and  $x$  an integer. Derive a divide and conquer algorithm to find the frequency of  $x$  in  $A$ , i.e., the number of times  $x$  appears in  $A$ . What is the time complexity of your algorithm?
- Solve the following instance of the knapsack problem. There are five items of sizes 3, 5, 7, 8, and 9 and values 4, 6, 7, 9, respectively, and 10, and the knapsack capacity is 22; using any suitable optimization technique.
- Show that the time complexity of breadth first search when applied on a graph of with  $n$  vertices and  $m$  edges is  $O(n + m)$ . What is its theta complexity?
- Explain Backtracking technique using a suitable example.
- Explain the concept of P, NP and NP hard problems? How do we find solution to NP complete problem?

## SECTION-C

- Write an Insertion Sort algorithm that uses Binary Search to find the position where the next insertion should take place. Determine the worst-case, average-case, and best-case time complexities for it.
- Apply PRIM's Algorithm to solve the MST on below giving stepwise details of data structure used :



- Explain the solution strategy for dynamic programming? How can this be used to solve the TSP problem? Give the pseudo code along with complexity.

**NOTE : Disclosure of Identity by writing Mobile No. or Making of passing request on any page of Answer Sheet will lead to UMC against the Student.**