C4-R4: ADVANCED ALGORITHMS

NOTE:

1. **Answer question 1 and any FOUR from questions 2 to 7.**
2. **Parts of the same question should be answered together and in the same sequence.**

Time: 3 Hours **Total Marks: 100**

1.

a) List out the properties of an efficient algorithm. Also mention various ways to measure running time of an algorithm.

b) How to choose an algorithm for solving a single problem, when you have multiple algorithms for solving a same problem are available? Which from the below function takes less time to solve the given problem? Justify your answer.
   i) $f(n) = 2f(n-1) + 1$
   ii) $f(n) = f(n-1) + n$

c) Is the sequence <23, 17, 14, 6, 13, 10, 1, 5, 7, 12> a max heap? Why?

d) What are the various ways to represent graph?

e) Prove that the height of a heap with n nodes is equal to $[\log_2 n]$.

f) State T/F with reason: A graph with n nodes and more than n-1 edges must contain cycle.

g) Define: Big-Oh, Omega & Theta notations with neat figures.

(7x4)

2.

a) What is recurrence? Solve the following recurrence using master method.
   $T(n) = 9\,T(n/3) + n$

b) Write down merge sort algorithm and derive recurrence equation for it.

c) Find the asymptotic bound for the following recurrence using recurrence tree method.

(6+6+6)

3.

a) What is a flow network? Also explain Residual Network.

b) Derive the recurrence relation for quick sort in best case and find the complexity.

c) Consider 5 times along their respective weight and profit:

Item	Weight	Profit
I1	5	30
I2	10	20
I3	20	100
I4	30	90
I5	40	160

The capacity of knapsack W=60. Find the solution to the fractional knapsack problem.

(6+6+6)

4.

a) Write the Kruskal's algorithm for finding the minimum spanning tree.

b) Write down: Depth First Search (DFS) algorithm.

c) Define Assembly Line Scheduling Problem. Solve it using Dynamic Programming and find its complexity.

(6+6+6)

5.

a) Find an optimal parenthesization of matrix-chain product whose sequence of dimensions is <4, 10, 3, 12, 20, 7>.

b) Explain the working of Radix Sort algorithm.

c) Find out longest common subsequence from the given two sequences of characters: P=<MLNOM> Q=<MNOM>.

(6+6+6)

6.

a) Define: P, NP Problems. Explain both in detail.

b) Explain Dijkstra's Algorithm with an example.

c) Explain in detail approximation Algorithm.

(6+6+6)

7.

a) Write down an algorithm for naïve string matching. Also derive its time complexity.

b) Select the maximum-size subset of mutually compatible activities for the following input. Activities are sorted in monotonically increasing order of finish time. Each activity has **Si** as start time and **Fi** as finish time. Apply Greedy strategy to find out the solution.

Activity	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**
Si	**1**	**3**	**0**	**5**	**3**	**5**	**6**	**8**	**8**	**2**	**12**
Fi	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**

c) Which data structure is used to implement BFS? Which data structure is used to implement DFS?

(8+8+2)