C4-R3: ALGORITHM ANALYSIS AND DESIGN

NOTE:

1.	**Answer question 1 and any FOUR questions from 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) Solve $T(n)=2T(\lfloor \sqrt{n} \rfloor)+\lg n$.

b) Define P, NP and NP-complete problem. Give suitable example.

c) What do you mean by backtracking and why is it required? Why is it so called?

d) Show that the worst case running time of HEAPIFY on a heap of size n is Ω (lgn).

e) If $f(n) = a_m n^m + a_{m-1} n^{m-1} + \ldots\ldots + a_0$ is a polynomial of degree m, then prove that $f(n) = \Theta(n^m)$.

f) Distinguish between divide-and-conquer and dynamic programming with suitable examples.

g) Write a short note on *amortized analysis of algorithm*.

(7x4)

2.

a) Write an algorithm for deleting an element from a binary search tree? Determine its time complexity.

b) Write an algorithm for RIGHT-ROTATE in the construction of AVL trees. Explain with one example. Determine its time complexity.

(9+9)

3.

a) Give an $O(n^2)$ time algorithm to find the longest monotonically increasing subsequence of a sequence of n numbers.

b) Compute the prefix function Π for the pattern a b a b a b a b c a.

c) Write Boyer-Moore Matcher algorithm for string matching.

(8+3+7)

4.

a) What are the differences between *heuristic* and approximation algorithms?

b) Design an approximation algorithm for colouring a planar graph.

c) Design a heuristic algorithm for chromatic partitioning of a simple, connected, undirected graph.

(4+7+7)

5.

a) Give an algorithm that determines whether or not a given undirected graph G=(V, E) contains a cycle.

b) What is the running time of Breadth first search if its input graph is represented by an adjacency matrix?

c) Devise a O(n+m) time algorithm for computing a component graph of a directed graph G=(V,E), where |V| = n and |E| = m. Make sure that your algorithm produces at most one edge between any pair of vertices in the component graph.

(7+3+8)

6.

a) Write an algorithm to compute the k-th smallest element of a list of n numbers, where $k \leq n$. Determine the number of comparisons required to compute it, and deduce the time complexity of your algorithm.

b) Write the Kruskal's algorithm for computing a minimum spanning tree of a simple, connected, undirected graph G. Trace this algorithm to compute a minimum spanning tree for such a graph G that contains at least 9 vertices and 13 weighted edges.

(9+9)

7.

a) What is convex hull? Discuss Graham's algorithm for computing the convex hull for a given set of points on a plane.

b) Explain the terms *flow* and *capacity* in a network. What are meant by properly and improperly oriented edges? Discuss how flow is related to these kinds of edges.

(9+9)