

Design and Analysis of Algorithm

It is an OPEN NOTE examination. You can ONLY keep hand-written notes with you and nothing else. Attempt ALL Questions (carrying 120 marks). Maximum you can score 100. Best wishes to all of you.

Question 1. Let $A = ((a_{i,j}))$ be an $n \times n$ matrix of integers (not necessarily distinct) such that elements of each row and column are sorted in increasing order. Describe an $O(n)$ algorithm to search an element (and report in case it is absent) in A . [15]

Question 2. You are managing a cloud server that must schedule n computational tasks $\{T_1, T_2, \dots, T_n\}$. Each task T_i has two attributes:

- (i) A **deadline** d_i (a positive integer time slot by which the task must be completed),
- (ii) A **profit** p_i earned if the task is completed before or on its deadline.

Only one task can be executed at each integer time unit, and each task takes exactly one time unit to execute. Write an algorithm using a Heap to schedule a subset of the tasks so as to maximize the total profit while respecting deadlines. Explain why your algorithm returns the maximum profit among all possible scheduling. [20]

Question 3. Let T be a rooted tree with n nodes. The **Lowest Common Ancestor (LCA)** of two nodes u and v is defined as the lowest node in T that has both u and v as descendants (where a node is considered a descendant of itself). Design an algorithm to answer LCA queries efficiently by reducing the LCA problem to a Range Minimum Query (RMQ) problem. [20]

Question 4. Describe the Miller-Rabin algorithm which has error probability at most 10^{-10} . Justify that your algorithm has desired error probability (you may assume that one iteration has an error probability at most $1/2$). Let n be the input and $n - 1 = 2^r m$ where m is odd. Write down the exact number of modular multiplications (including those which have been used to compute modular exponentiations) in terms of m and r . [15]