COL 351 : Analysis and Design of Algorithms

Tutorial Sheet - 2

Question 1 The manager of a student union in IIT Delhi is in-charge of a group of n students, each of whom is scheduled to work one shift during the week. Each shift is a single contiguous interval of time. There can be multiple shifts going on at once. The Manager is trying to choose a *minimum* subset of these n students to form a supervising committee that she can meet weekly. She considers such a committee to be *complete* if, shift of each student not in the committee overlaps (at least partially) with the shift of some student who is in the committee. In this way, each student's performance can be observed by at least one person who's serving in the committee.

(a) Formulate this problem mathematically.

(b) Give an efficient algorithm that takes the schedule of n shifts and produces a complete supervising committee containing as few students as possible.

Question 2 There are n workshops to be held in technical fest at IIT Delhi. Each workshop is scheduled to operate for a pre-decided single contiguous interval of time. Your task is to allot lecture rooms to different workshops such that no two overlapping workshops get the same lecture room.

(a) Devise a greedy algorithm for this problem that uses as few lecture rooms as possible.

(b) How will your solution change if the following additional constraint is imposed - The gap between two consecutive workshops in each lecture room must be at least 30 minutes.

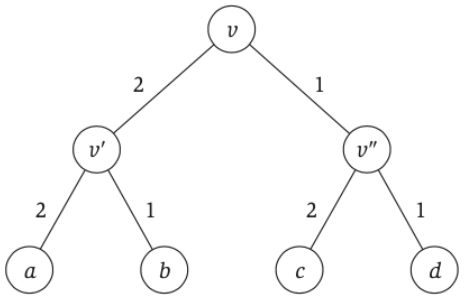
Question 3 There is a complete binary tree with n leaves representing a circuit. Here n is a power of two. Each edge e of the tree has an associated delay $d_e \geq 0$. The root generates a clock signal which is propagated along the edges to the leaves. The time taken for a message to reach from the root to a given leaf is the sum of the delays of all the edges on the path from the root to the leaf.

Now, we want all the leaves to be completely synchronized, i.e. they receive the signal at the same time. To make this happen, we will have to increase the delay of certain edges, so that all root-to-leaf paths have the same delay. Our goal is to achieve this synchronization in a way that keeps the total delay enhancement in the circuit to be minimum.

(a) Design a greedy step that transforms the given instance to a smaller instance of the same problem.

(b) Next establish a relation between optimal solution of given instance and the optimal solution of smaller instance.

(c) Design an $O(n)$ time greedy algorithm based upon (a) and (b), and prove the correctness.



Question 4 You are starting a start-up that needs to obtain licenses for n different pieces of cryptographic software. Due to regulations, you can only obtain these licenses at the rate of at most one per month.

Each license is currently selling for a price of 100 INR.

However, they are all becoming more expensive according to exponential growth curves: in particular, the cost of license j increases by a factor of $r_j > 1$ each month, where r_j is a given parameter. This means that if license j is purchased after t months from now, it will cost $(100 \cdot r_j^t)$. Further, all the price growth rates are distinct, that is $r_i \neq r_j$ for licenses $i \neq j$ even though they start at the same price of 100 INR.

The question is the following: Given that your company can only buy at most one license a month, in which order should it buy the licenses so that the total amount of money it spends is as small as possible? Give an algorithm that takes the n rates of price growth $r_1, r_2, \ldots, r_n$ and computes an order in which to buy the licenses so that the total amount of money spent is minimized. The running time of the algorithm should be polynomial in n.