MinorI (CSL 356/ COL 351)

Read the instructions carefully:

- You must write a proof of correctness in each of the 3 questions.
- The algorithm should be explained in English (you can use pseudocode, but it should be easy to understand). Do NOT use examples/figures to explain your algorithm. We will NOT read any such explanation.
- The proofs should be brief and statements in the proof should follow a logical sequence. Do NOT use examples or special cases to explain the proof. We will NOT read any such part of the proof.
- Each question carries 10 marks.

1. You are given an array consisting of n integers (which could be positive or negative). A sub-array of an array is a continuous segment of the array. For example, if the array A is $[1, -3, 5, 4, -6, -9, 1, 7]$, then $[1, -3, 5, 4], [-3, 5], [5, 4, -6, -9]$ are examples of sub-arrays of A. Give an algorithm to find the sub-array for which the sum of the elements in it is maximized. The running time should be $O(n \log n)$.

2. You wish to invite a set of people to a party. There are a set of n people to choose from, and you are given which pair of people know each other (you can assume that this information is given as a graph, where the set of vertices is the set of n people, and we have an edge between two people if and only if they know each other). You would like to invite as many people as possible, but subject to two constraints: among the set of invited people, each person should have at least 5 other persons whom he/she knows, and 5 other people whom he/she does **not** know. Give an algorithm to solve this problem (i.e., find a subset of people of maximum size such that each person in the subset satisfies the two constrains). The running time should be a polynomial in n.

3. You are organizing a sports event, where each contestant has to swim 20 rounds in a pool, and then run 3 kilometers. However, the pool can be used by only one person at a time. In other words, the first contestant swims 20 rounds, gets out and then starts running. As soon as this first person is out of the pool, a second contestant begins swimming the 20 rounds; as soon as he/she is out and starts running, a third contestant begins swimming... and so on.)

   You are given a list of n contestants, and for each contestant you are given the time it will take him/her to complete swimming 20 rounds of the pool, and the time it will take him/her to run 3 kilometers. Your job is to decide on a schedule for the event, i.e., an order in which to sequence the starts of the contestants. The completion time of a schedule is the earliest time at which all contestants will be finished with swimming and running. Give an efficient algorithm that produces a schedule whose completion time is as small as possible.

   Example: Suppose there are two contestants C_1, C_2 with swimming and running times being $(10, 5)$ for C_1 and $(2, 8)$ for C_2. If we schedule them as C_1, C_2, then C_1 will finish swimming and running by time $10 + 5 = 15$. C_2 can start swimming at time 10, and so, will finish by time $10 + 2 + 8 = 20$. Note that both contestants will finish by time 20, and so the completion time is 20. If we order them as C_2, C_1, then C_2 will finish by time 10 and C_1 by time 17. Therefore, the completion time is 17. Thus, the best ordering is C_2, C_1.

Scanned by CamScanner