CS4102 - ALGORITHMS - S'16 — ABHI SHELAT
DUE WED MAR 30, 2016 AT 5P VIA COURSE WEBSITE

H5.AS4BD

You may collaborate with other students on the homework but you must submit your own individually written solution, identify your collaborators, and acknowledge any external sources that you consult. Do no submit a solution that you cannot explain to me.

PROBLEM 1 *Lifeguards*

You run the pool on Saturdays. There must always be a lifeguard on duty. There are n lifeguards who can work, but they are busy college students with complex social obligations; lifeguard i can work starting at time a_i and ending at time b_i on Saturdays. Your goal is to cover the entire day from 8a–8p *using the fewest lifeguards as possible*.

The first idea that comes to mind is to start with an empty schedule, and then add the guard who covers *the most amount of time that is not already covered*. Show that this algorithm fails by providing a counter-example.

Devise and analyze a greedy algorithm that solves the problem in linear time.

PROBLEM 2 *ExpressEspresso*

Every morning, customers $1, \ldots, n$ show up to get their espresso drink. Suppose the omniscient barista knows all of the customers, and knows their orders, $o_1, \ldots, o_n$. Some customers are nicer people; let T_i be the barista's expected tip from customer i. Some drinks like a simple double-shot, are easy and fast, while some other orders, like a soy-milk latte take longer. Let t_i be the time it takes to make drink o_i. Assume the barista can make the drinks in any order that she wishes, and that she makes drinks back-to-back (without any breaks) until all orders are done. Based on the order that she chooses to complete all drinks, let D_i be the time that the barista finishes order i. Devise an algorithm that helps the barista pick a schedule that minimizes the quantity

$$\sum_{i}^{n} T_i D_i$$

In other words, she wants to serve the high-tippers the fastest but she also wants to take into consideration the time it takes to make each drink. (Hint: think about a property that is true about an optimal solution.)

PROBLEM 3 *Homework*

College life is hard. You are given n assignments $a_1, \ldots, a_n$ in your courses. Each assignment $a_i = (d_i, t_i)$ has a deadline d_i when it is due and an estimated amount of time it will take to complete, t_i. You would like to get the most out of your college education, and so you plan to finish all of your assignments. Let us assume that when you work on one assignment, you give it your full attention and do not work on any other assignment.

In some cases, your outrageous professors demand too much of you. It may not be possible to finish all of your assignments on time given the deadlines $d_1, \ldots, d_n$; indeed, some assignments may have to be turned in late. Your goal as a sincere college student is to minimize the lateness of any assignment. If you start assignment a_i at time s_i, you will finish at time $f_i = s_i + t_i$. The lateness value—denoted ℓ_i—for a_i is the value

$$\ell_i = \begin{cases} f_i - d_i & \text{if } f_i > d_i \\ 0 & \text{otherwise} \end{cases}$$

Devise a polynomial-time algorithm that computes a schedule O that specifies the order in which you complete assignments which minimizes the maximum ℓ_i for all assignments, i.e.

$$\min_O \max_i \ell_i$$

In other words, you do not want to turn in *any* assignment *too* late, so you minimize the lateness of the latest assignment you turn in. Prove that your algorithm produces an optimal schedule. Analyze the running time of your algorithm.

PROBLEM 4 *Populist*

Let $G = (V, E)$ be a connected graph with distinct positive edge costs, and let T be a spanning tree of G (not necessarily the minimum spanning tree). The *elite edge* of T is the edge with the greatest cost. A spanning tree is said to be *populist* if there is no other spanning tree with a less costly elite edge. In other words, such a tree minimizes the cost of the most costly edge (instead of minimizing the overall cost).

1. Is every minimum spanning tree of G a populist spanning tree of G?
2. Is every populist spanning tree of G a minimum spanning tree?

Justify both of your answers.