CSE 331: Introduction to Algorithm Analysis and Design Fall 2011

Homework 6

Due Friday, October 28, 2011 by 1:15pm in class

IMPORTANT: Please submit each problem separately, i.e. each problem should begin on a new page and only the pages for one problem should be stapled together. Failure to do so might result in some problem(s) not being graded.

For general homework policies and our suggestions, please see the policy document.

---

1. (40 points) Say we are given an instance of the shortest $s-t$ path problem on a directed graph G. Assume that all the edge weights are distinct and positive. Let P be a minimum cost path for this instance. Now consider a new instance where each edge cost c_e is replaced by a new edge cost of c_e^3 (but the graph G remains the same).

   Mr. GraphKnowItAll claims that P is still a minimum cost $s-t$ path in the new instance. Decide if Mr. GraphKnowItAll is full of it or not– if so, give a counter example. If not, give a short explanation why Mr. GraphKnowItAll is correct.

2. (45 points) Ms. Greedy Chef has to schedule n clients for lunch for the next day. These n clients are regular customers, so for each customer i, Ms. Chef knows the exact time b_i it would take her to cook the burger exactly the way i likes it. Unfortunately, Ms. Chef's restaurant is new and the kitchen is small, so she can only cook one burger at a time. Ms. Chef is also gunning for a record, so she has decided not to waste any time moving from one burger to the next. However, given that her customers are very picky, Ms Chef has to spend one time unit between making two successive burgers to clean up her utensils. Help Ms. Chef design an $O(n \log n)$ time algorithm that schedules all the n clients, which provably minimizes the *sum* of the *service time* for every client, where the service time for client i is b_i plus the time i had to wait before Ms. Chef started i's burger. You should assume that all the clients came into the restaurant at the same time. As usual, you must prove the optimality of your algorithm.

   Here is a simple example: say $n = 3$ and

   $$b_1 = 5, \quad b_2 = 10, \quad b_3 = 4.$$

   Now consider the schedule

   $$1, \text{ break }, 2, \text{ break }, 3$$

   i.e. 1 gets her burger first and then leaves, 2 gets his burger after 1 and 3 gets his burger after 2. Note that the service time for 1 is 5. The service time for client 2 is his wait time (which is $5 + 1 = 6$) plus b_2, which is 16. The service time for client 3 is his wait time (which is $5 + 1 + 10 + 1 = 17$) plus b_3, which is 21. Thus, the sum of the service times for this schedule is $5 + 16 + 21 = 42$. This schedule, however, is *not* optimal.

3. (15 points) Exercise 6 in Chapter 4.

   Note: As mentioned in the previous homework, in many real life problems, not all parameters are equally important. Also sometimes it might make sense to "combine" two parameters into one. Keep these in mind when tackling this problem.

   Hint: In the solution that I have in mind, the analysis of the algorithm's correctness follows the exchange argument.