

## Homework Assignment 1

September 7, 2004

Due Date: September 21

In all problems (throughout this course) when you are asked to give an algorithm you are expected to: (1) give a clear description of the algorithm, (2) prove the algorithm outputs an optimal solution, (3) give the time complexity of the algorithm, and (4) prove that the algorithm has the stated time complexity.

You must submit your homework with a signed cover sheet attached to the front.

## Core Problems

- (20 pts) In each of the following two problems a greedy algorithm is suggested. You are to prove whether or not the given greedy algorithm computes an optimal solution.

- You are given  $n$  jobs where job  $i$  runs from 1 unit, has an integer deadline time  $d_i \geq 0$ , and real-valued penalty  $p_i \geq 0$ . The  $n$  jobs can be scheduled at times  $0, 1, \dots, n-1$  with each job scheduled exactly once and only one job can run at a time. If job  $i$  is completed by time  $d_i$  then there is no cost for it, but if job  $i$  completes after time  $d_i$  then a penalty of  $p_i$  is incurred. Your goal is to find a schedule which minimizes the total penalty.

Here is the proposed greedy algorithm. Sort the jobs so that  $p_1 \geq p_2 \geq \dots \geq p_n$ . Let the  $n$  possible time slots be initially empty where slot  $i$  is the slot that runs from time  $i-1$  to time  $i$ . Consider the jobs in the order  $1, 2, \dots, n$ . When considering job  $j$ , if any of time slots  $1, \dots, d_j$  are available then job  $j$  is scheduled in the latest such slot. Otherwise, schedule job  $j$  in the latest available time slot (from those after  $d_j$ ).

- Consider the following scheduling problem. We have  $n$  jobs, where only one job can be processed at a time. Job  $i$  must start at time  $s_i$ , end at time  $e_i$ , and if run will result in a profit of  $p_i$ . The goal is to find a schedule that results in the maximum total profit.

Here is the proposed greedy algorithm. Sort the input so that  $p_1 \geq p_2 \geq \dots \geq p_n$ . Consider the jobs in order of job 1 to job  $n$ . If job  $i$  doesn't conflict with the previously schedule jobs then schedule it from time  $s_i$  to  $e_i$ . Otherwise, job  $i$  is not scheduled.

- (15 pts) You have  $n$  people who can work at a store, where person  $i$  will work from time  $b_i$  until time  $e_i$ . Also, you are given an opening time  $b$  and closing time  $e$  for the store. The goal is to find the minimum number of people to work for the day so that at least one person is at the store between times  $b$  and  $e$ . You can assume that for all  $i$ ,  $b_i \geq b$  and  $e_i \leq e$ .

Give a greedy algorithm that optimally solves this problem. (Your solution should include the four parts listed at the top of this assignment.)

- (20 pts) A ski rental agency has  $m$  pair of skis, where the height of the  $i$ th pair of skis is  $s_i$ . There are  $n$  skiers who wish to rent skis, where the height of the  $i$ th skier is  $h_i$ . Your goal is to assign skis to skiers so that the sum of the absolute differences of the heights of each skier and her skis is minimized. (The rest of the problem is on the back.)

- (a) Give the most efficient algorithm you can to obtain an optimal solution to this problem when  $m = n$ .
- (b) Now consider this problem when  $m \geq n$ . Prove whether or not the following greedy algorithm is optimal.

Let  $H$  be the set of heights for the skiers

Let  $S$  be the set of ski lengths

Repeat until each skier has skis

    Pick a height  $h$  in  $H$  and ski length  $s$  in  $S$  such that

$|h-s|$  is the minimum possible

    Match the person with height  $h$  to skis of length  $s$

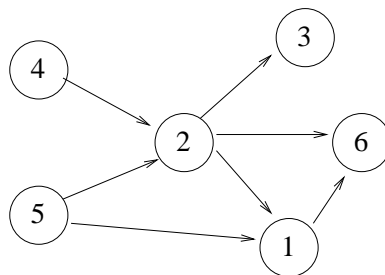
    Remove  $h$  from  $H$

    Remove  $s$  from  $S$

### Advanced Problems, required for CSE 541T (extra credit for CSE 441T)

4. (20 pts) Consider the following scheduling problem. There are  $n$  jobs to be processed by a single machine which can execute at most one job at a time.

- Each job  $j$  requires a processing time of  $p_j$  and specifies a “anger” function  $f_j(t)$  which captures how angry the customer will be if job  $j$  is not completed until time  $t$ . The “anger” function can be any non-decreasing function. That is, for  $t_1 < t_2$ , it must be  $f_j(t_1) \leq f_j(t_2)$ . In other words, the customer will only get angrier as it takes longer. But these anger functions could look very different. One customer may not be angry at all until some time at which he became very angry. Another customer may get a little bit angrier over time.
- As part of the input you are also given a set of precedence constraints of the form job  $i$  must be scheduled prior to job  $j$ . The precedence constraints form a directed acyclic graph. For example:



- Your goal is as follows. Let  $S$  be a legal schedule (i.e. the precedence constraints are satisfied and only one job is processed at a time). For  $1 \leq j \leq n$ , let  $C_j$  be the time when job  $j$  is completed in  $S$ . The goal is to find the schedule  $S$  that **minimizes**  $\max_{1 \leq j \leq n} (f_j(C_j))$ . That is, you want to minimize the highest anger that will be incurred.

Give a greedy algorithm that optimally solves this problem. Your solution should include the four parts listed at the top of this assignment. If you aren't familiar with topological sort I recommend you read about it. It is covered in the text book.