**CSE 441T/541T Advanced Algorithms** **Fall 2008**

# Homework 1: Greedy Algorithms

*Assigned: September 3, 2008* *Due Date: September 17, 2008*

**Whenever you are asked to give an algorithm for a problem, I expect you to do all of the following:**

1. **give a clear, concise description of your algorithm;**
2. **prove that the algorithm outputs an optimal solution;**
3. **give the asymptotic time complexity of the algorithm;**
4. **prove that the time complexity is as claimed.**

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

## Problems

1. (20 pts) Each of the following problems suggests a greedy algorithm for a specified task. Prove that the greedy algorithm is optimal, or give a counterexample to show that it is not.

   (a) **Problem**: You are given a set of $n$ jobs. Job $i$ starts at a fixed time $s_i$, ends aat a fixed time $e_i$, and results in a *profit* $q_i$. Only one job may run at a time. The goal is to choose a subset of the available jobs to maximize total profit.

   **Algorithm**: First, sort the jobs so that $q_1 \geq q_2 \geq \cdots \geq q_n$. Then, try to add each job to the schedule in turn. If job $i$ does not conflict with any jobs schedule so far, add it to the schedule; otherwise, discard it

   (b) **Problem**: You are given a set of $n$ jobs, each of which runs in unit time. Job $i$ has an integer-valued deadline time $d_i \geq 0$ and a real-valued penalty $p_i \geq 0$. Jobs may be scheduled to start at any integer time $(0, 1, 2,$ etc), and only one job may run at a time. If job $i$ completes at or before time $d_i$, then it incurs no penalty; otherwise, it incurs penalty $p_i$. The goal is to schedule all jobs so as to minimize the total penalty incurred.

   **Algorithm**: Define slot $k$ in the schedule to run from time $k-1$ to time $k$. First, sort the jobs so that $p_1 \geq p_2 \geq \cdots \geq p_n$. Then, add each job to the schedule in turn. When adding job $i$, if any time slot between 1 and $d_i$ is available, then schedule $i$ in the latest such slot. Otherwise, schedule job $i$ in the latest available slot $\leq n$.

2. (15 pts) You are an alien spy, worried that the worlds radio telescopes will discover your home planet. After decades of undercover work as a PhD student in Physics, you've become the scheduler for the biggest world radio telescope. You're home planet will be particularly susceptible to being noticed during a time interval $[S, F)$. During this time, there are a set of research projects that have been proposed: $A = \{a_1 \ldots a_n\}$, which can justify using the radio telescope from an interval $[s_i, f_i)$. (These intervals cannot be lengthened or shortened and may overlap).

You're goal is to avoid suspicion, but minimize the chance that your home planet will be discovered. That is, you must create an efficient algorithm to select the **fewest** possible jobs from $A$ such that at least one job is running at all times (otherwise, the schedule will look suspicious).

3. (20 pts) "Survivor Anchorage" proposes to take the popular TV show up north. One of the challenges is based on the entire team skiing quickly down a mountainside. Knowing that you aren't the best skier, you need to help your team by defining an efficient algorithm to match skis to skiers. In particular, skiers go fastest with skis whos length is about their height.

   Your team consists of $n$ members, with heights $h_1, h_2, \ldots, h_n$, and your team gets a delivery of $n$ pairs of skis, with lengths $l_1, l_2, \ldots, l_n$. Your goal is to write an algorithm to assign one pair of skis to each skier to minimize the sum of the absolute differences between the height of the skier $h_i$ and its length of the corresponding ski $l_j$.

## Advanced Problem

**This problem is *required* only for CSE 541 students.** 441 students may receive extra credit for a correct solution.

4. (15 pts) You are given n events where each takes one unit of time. Event i will provide a profit of $g_i$ dollars ($g_i > 0$) if started at or before time $t_i$, where $t_i$ is an arbitrary real number. (Note: If an event is not started by $t_i$ then there is no benefit in scheduling it at all. All events can start as early as time 0.) Given the most efficient algorithm you can to find a schedule that maximizes the profit.