*Data Structures and Algorithms* April 28, 2007
Pennsylvania State University CSE 465
Professors Sofya Raskhodnikova & Adam Smith Handout 25

# Homework 11 – Due Friday, May 3, 2007

**Reminder**

- Collaboration is permitted, but you must write the solutions by yourself without assistance, and be ready to explain them orally to a member of the course staff if asked. You must also identify your collaborators. Getting solutions from outside sources such as the Web or students not enrolled in the class is strictly forbidden.
- To facilitate grading, please write down your solution to each problem on a *separate sheet* of paper. Make sure to include all identifying information and your collaborators on each sheet. Your solutions to different problems will be graded separately, possibly by different people, and returned to you independently of each other.

**Reading and Exercises** Sections 15.1 – 15.3 of CLRS, and exercises in these sections.

**Problems to be handed in**

1. **(Short Answers)** For each of the following give a short answer.
   (a) Recall that when we considered Weighted Interval Scheduling problem in class, we sorted all intervals by finishing times and defined for each interval $j = 1, \ldots, n$
   $$p(j) = \begin{cases} \text{interval } i & \text{where } i \text{ is the largest index such that } i < j \text{ and } i \text{ does not overlap } j, \\ 0 & \text{if no such interval exists.} \end{cases}$$
   Give an $o(n^2)$ time algorithm for computing $p(j)$ for all intervals $j = 1, \ldots, n$, given an instance of Weighted Interval Scheduling with $n$ intervals, sorted by finishing times. (Note that the time requirement is stated using little-$o$ notation.)
   (b) Recall that in the lecture of Friday, April 27 we showed that the recursive algorithm for Weighted Interval Scheduling takes $\Omega(2^n)$ time on an instance with $n$ non-overlapping intervals. Draw the tree of recursive calls for this instance, as we did in class. Mark the subset of these calls that the memoized algorithm makes on this instance.
   (c) Exercise 15.1-1 from CLRS.
   (d) Exercise 15.2-2 from CLRS.
2. **(Binomial coefficient)** Recall that the binomial coefficient
   $$\binom{n}{k} = \begin{cases} \binom{n-1}{k-1} + \binom{n-1}{k} & \text{if } 0 < k < n, \\ 1 & \text{otherwise.} \end{cases}$$
   Give a dynamic programming algorithm for calculating the binomial coefficient on input $n$ and $k$. Analyze time and space complexity of your algorithm. You may assume that addition can be performed in constant time.

3. **(Billboards)** You are managing the construction of billboards along a stretch of highway. The possible sites for the billboards are given by real numbers $x_1, ..., x_n$, each of which specifies the position along the highway measured in miles from its western end. Assume that the highway is a straight line. If you place a billboard at location $x_i$, your company will make a profit of $r_i > 0$ dollars.

   Regulations require that every pair of billboards be at least 5 miles apart. You'd like to place billboards at a subset of the sites so as to maximize total profit, subject to this restriction. The input is given as a list of $n$ pairs $(x_1, r_1), ..., (x_n, r_n)$ where the $x_i$'s are sorted in increasing order. Give a dynamic programming algorithm for this problem. Analyze the space and time complexity of your algorithm.