CS 577: Introduction to Algorithms 10/15/2015

Practice Problems for the Midterm Exam

Instructor: Dieter van Melkebeek TAs: Kevin Kowalski, Andrew Morgan, Bryce Sandlund

Questions:

1. Consider the following specification and (incomplete) pseudocode.

Algorithm 1: `MergeSortedArraysAndCountInversions`

Input: Arrays $L[1,\ell]$ and $R[1,r]$ of integers, each sorted in non-decreasing order.
Output: (c, A), where c equals the number of inversions in the concatenation of L and R, and A is the concatenation sorted in non-decreasing order.

```
1  i ← 1;
2  j ← 1;
3  k ← 1; c ← 0;
4  while i ≤ ℓ and j ≤ r do
5      if L[i] ≤ R[j] then
6          A[k] ← L[i];
7          i ← i + 1;
8          ???
9      else
10         A[k] ← R[j];
11         j ← j + 1;
12         ???
13     k ← k + 1;
14 ???
15 return (c, A);
```

   (a) Fill in lines 8, 12, and 14 in the pseudocode so as to satisfy the specification as well as the following loop invariant: c equals the number of inversions in $L[1, i-1]R$.

   The latter invariant is *not* equivalent to the one from class, so the resulting pseudocode will need to be different, too. Each of the missing lines may actually consist of multiple lines or may be empty.

   (b) Prove correctness.

2. Let P be a set of n points in the plane. A point (x, y) in P is called undominated if for every other point (x', y') in P, either $x' < x$ or $y' < y$ (or both).

   Develop an $O(n \log n)$ algorithm for finding all of the undominated points.

3. You are given a communication network in the form of a rooted tree. The root wants to broadcast a message to all of the nodes in the tree. The message gets forwarded as follows: At every step, every node that has received the message can forward it to one of its neighbors.

Give an $O(n \log n)$ algorithm that computes the minimum number of steps needed for every node to receive the message.

For example, for a path of length $2k$ rooted in the middle, the answer is $k + 1$.

4. To get in shape you have decided to start running to school. You want a route that goes entirely uphill and then entirely downhill. Your run starts at home and ends at school. You have a map detailing the roads with m road segments and n intersections. Each road segment has a positive length and each intersection has an elevation.

   Give an efficient algorithm to find a shortest route that meets your specifications, or report that none exists. You may assume that every road segment is either uphill or downhill, and that home and school lie at intersections.

5. For a dinner after a sports event, you want to assign seats such that no two members of the same team sit at the same table. There are n teams of size $s_1, s_2, \ldots, s_n$, respectively, and m tables, of size $t_1, t_2, \ldots, t_m$, respectively.

   Develop an algorithm that determines whether such a seating arrangement is possible. Your algorithm should run in time $O(N \log N)$, where $N = (\sum_{i=1}^{n} s_i) + (\sum_{j=1}^{m} t_j)$.

6. Suppose two trees, T_1 and T_2, have the following edge weights:

$$
\begin{aligned}
T_1 &: 1, 2, 2, 2, 3, 3, 3, 4, 4, 5, 5, 5, 6, 6, 10, 12, 13. \\
T_2 &: 1, 2, 2, 2, 3, 3, 3, 4, 4, 5, 5, 6, 6, 7, 8, 12, 13.
\end{aligned}
$$

   Can T_1 and T_2 be minimum spanning trees of the same graph G?