CS 577: Introduction to Algorithms 9/17/2015

Homework 2

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

This assignment covers divide and conquer. The guidelines are the same as for homework 1 – and will be the same for all subsequent homeworks. Good luck!

Review problems

1. You are given an infinite array A in which the subarray $A[0..n-1]$ contain integers in sorted order and the rest of the cells are filled with ∞. However, you are not given the value of n. Describe an algorithm that takes an integer x as input and finds a position in the array containing x, if such a position exists, in $O(\log n)$ time.

   If you are disturbed by the fact that the array A has infinite length, then assume instead that it is of length n (which you still do not know) and that the implementation of the array data type in your programming language return the error message ∞ whenever an element of A at index $i \geq n$ is accessed.

2. This problem deals with a simple version of hidden surface removal, a well-known problem in computer graphics.

   You are given n non-vertical lines in the plane, labeled $L_1, \ldots, L_n$, with the ith line specified by the equation $y = a_i x + b_i$. You can assume that no three of the lines meet at a single point. We say that line L_i is *uppermost* at a given x-coordinate x_0 if its y-coordinate at x_0 is greater than the y-coordinates of all the other lines at x_0: $a_i x_0 + b_i > a_j x_0 + b_j$ for all $j \neq i$. We say that line L_i is *visible* if there is some x-coordinate at which it is uppermost – intuitively, some portion of it can be seen if you look down from "$y = \infty$".

   Give an algorithm that takes n lines as input, and in $O(n \log n)$ time, returns all of the ones that are visible.

Graded written problem

3. [10 points] You are given an array $A[0..n-1]$ of positive integers. Describe an algorithm running in $O(n \log n)$ time that finds the largest value of $m(i,j) \cdot (j - i + 1)$, where i and j are integers with $0 \leq i \leq j \leq n-1$, and $m(i,j) \doteq \min_{i \leq k \leq j} A[k]$. Intuitively, you are trying to maximize the area of a rectangle that fits inside the histograpm given by A, namely the rectangle with base $[i,j]$ on the x-axis and height $m(i,j)$.

Additional written problem

4. You are given two arrays A and B sorted in non-decreasing order, and an integer k satisfying $1 \leq k \leq |A| + |B|$. Develop an algorithm that finds the kth smallest element in the concatenation of A and B and looks at only $O(\log k)$ positions of the arrays A and B.

Optional programming problem

5. [2.5 points] Solve SPOJ problem Insertion Sort (problem code CODESPTB).

Challenge problem

The following is one of the nicest introductory algorithms problems I know. Give it a try!

6. You are given n coins, at least one of which is bad. All the good coins weigh the same, and all the bad coins weigh the same. The bad coins are lighter than the good coins.

   Find the exact number of bad coins by making $O((\log n)^2)$ weighings on a balance. Each weighing tells you whether the total weight of the coins on the left side of the balance is smaller than, equal to, or larger than the total weight of the coins on the right side.