# CS3510-A: Design and Analysis of Algorithms, Spring 2021

Homework-5

February 16, 2021

**DUE DATE: Tuesday, February 23, 1l:59pm**

**Note-1:** Your homework solutions should be electronically formatted as a single PDF document that you will upload on Gradescope. If you have to include some handwritten parts, please make sure that they are very clearly written and that you include them as high resolution images.
**Note-2:** Please think twice before you copy a solution from another student or resource (book, web site, etc). It is not worth the risk and embarrassment.
**Note-3:** You need to **explain/justify** your answers. Do not expect full credit if you just state the correct answer.
**Note-4: You will get 2 extra points if you submit electronically typed solutions instead of hand-written.**

## Problem-1 (30 points)

A binary tree is called "complete" if (1) every internal node has two children, and (2) every leaf node has the same depth (distance from the root).

Describe a divide-and-conquer algorithm that computes the largest complete subtree T* of a given binary tree T. The algorithm should return both the root and the depth of T*. Note that the leaves of T* may not be leaves in the T (i.e., T* may be "internal" in T). Also analyze the run time of your algorithm.

### Solution

## Problem-2 (35 points)

Suppose that we have two sorted lists $a$ and $b$. The size of $a$ is $n$ entries and the size of $b$ is $m$ entries. Design an algorithm to find the $k$'th smallest element in the union of $a$ and $b$ in $O(log(n+m))$ time. Please also analyze the run time of your algorithm.

### Solution

## Problem-3 (35 points)

You are given $n$ non-vertical lines in the plane, labeled $L_1, ..., L_n$ , with the $i$'th line specified by the equation $y = a_i * x + b_i$ . We will make the assumption that no three of these lines meet at a single point and these lines extend infinitely.

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 $L_i$ is uppermost. Intuitively, some portion of $L_i$ can be "seen" if you look down from $y = +\infty$.

Design an algorithm that takes $n$ lines as input and in $O(n \log n)$ time returns the set of lines that are visible. Fig. 1 gives an example.

Please also analyze the run time of your algorithm.

**Note:** Make sure that your solution uses the divide-and-conquer approach.

**Hint:** You can first sort all lines by their slopes, and then split the problem.

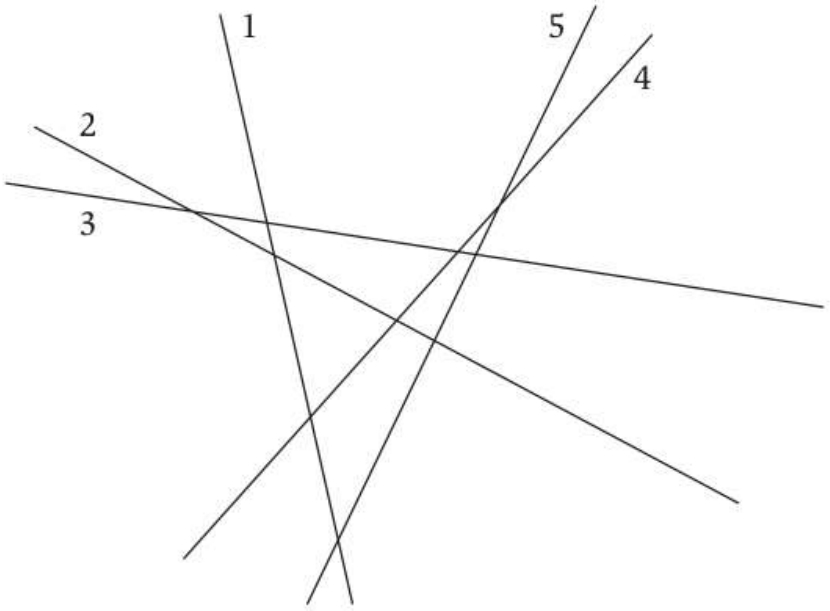


Figure 1: All the lines except for 2 are visible.

### Solution