Algorithms Lab 6

(Divide-and-conquer, CLRS 4)
Laura Toma, csci2200, Bowdoin College

This week's topics

- Selection
- Divide-and-conquer:
  - Karatsuba integer multiplication
  - Strassen's algorithm for matrix multiplication

In-class (collaboration level 0[1])

1. Selection exercises
2. Divide-and-conquer: The maximum partial subarray problem.

Homework problems (Collaboration level 1[2])

1. You are given a sorted array of numbers where every value except one appears exactly twice; the remaining value appears only once. Design an efficient algorithm for finding which value appears only once.

   Example: Here are some example inputs to the problem:

   $$1, 1, 2, 2, 3, 4, 4, 5, 5, 6, 6, 7, 7, 8, 8$$

   $$10, 10, 17, 17, 18, 18, 19, 19, 21, 21, 23$$

   $$1, 3, 3, 5, 5, 7, 7, 8, 8, 9, 9, 10, 10$$

[1] Collaboration level 0: everything allowed!

[2] Collaboration level 1: verbal collaboration without solution sharing. You are allowed and encouraged to discuss ideas with other class members, but the communication should be verbal and additionally it can include diagrams on board. Noone is allowed to take notes during the discussion (being able to recreate the solution later from memory is proof that you actually understood it). Communication cannot include sharing pseudocode for the problem. Check complete guidelines at: https://turing.bowdoin.edu/dept/collab.php)

2. The skyline problem/the upper envelope problem: In this problem we design a divide-and-conquer algorithm for computing the skyline of a set of n buildings.

   A *building* B_i is represented as a triplet $(\mathbf{L_i}, H_i, \mathbf{R_i})$ where $\mathbf{L_i}$ and $\mathbf{R_i}$ denote the left and right x coordinates of the building, and H_i denotes the height of the building (note that the x coordinates are drawn boldfaced.)

   A *skyline* of a set of n buildings is a list of x coordinates and the heights connecting them arranged in order from left to right (note that the list is of length at most $4n$).

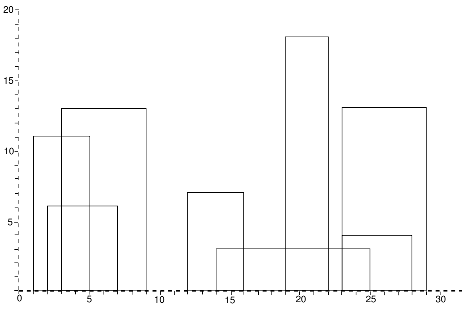
   Example: The skyline of the buildings

   $$\{(\mathbf{3}, 13, \mathbf{9}), (\mathbf{1}, 11, \mathbf{5}), (\mathbf{12}, 7, \mathbf{16}), (\mathbf{14}, 3, \mathbf{25}), (\mathbf{19}, 18, \mathbf{22}), (\mathbf{2}, 6, \mathbf{7}), (\mathbf{23}, 13, \mathbf{29}), (\mathbf{23}, 4, \mathbf{28})\}$$

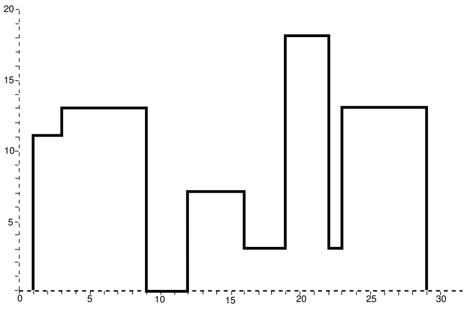
   is

   $$\{(\mathbf{1}, 11), (\mathbf{3}, 13), (\mathbf{9}, 0), (\mathbf{12}, 7), (\mathbf{16}, 3), (\mathbf{19}, 18), (\mathbf{22}, 3), (\mathbf{23}, 13), (\mathbf{29}, 0)\}$$

   (note that the x coordinates in a skyline are sorted).



buildings



skyline

   (a) Let the size of a skyline be the number of elements (tuples) in its list.
   Describe an algorithm for combining a skyline A of size n_1 and a skyline B of size n_2 into one skyline S of size $O(n_1 + n_2)$. Your algorithm should run in time $O(n_1 + n_2)$.

   (b) Describe an $O(n \log n)$ algorithm for finding the skyline of n buildings.

3. (CLRS 2-4) Let $A[1..n]$ be an array of n distinct numbers. If $i < j$ and $A[i] > A[j]$, then the pair (i, j) is called an inversion of A.

   (a) List the inversions of the array $< 2, 3, 8, 6, 1 >$.

   (b) What array with elements from the set $\{1, 2, ..., n\}$ has the most inversions? How many does it have?

   (c) Give an algorihm that determines the number of inversions in an array in $O(n^2)$ time.

   (d) Give an algorihm that determines the number of inversions in an array in $O(n \lg n)$ time worst-case (Hint: modify merge sort).