Assignment 6

csci2200, Algorithms

Instructions:

- Honor code: Work on this assignment alone, or with one partner. Between different teams, Collaboration is at level 1 [verbal collaboration only]
- Check out the Homework guidelines on class website.

---

1. **Finding k-quantiles:** Given an unsorted sequence S of n elements, and an integer k, we want to find $O(k)$ elements that have rank $\lceil n/k \rceil$, $2\lceil n/k \rceil$, $3\lceil n/k \rceil$, and so on. You may assume that k is a power of 2.

   (a) Describe the "naive" algorithm that works by repeated selection, and analyze its running time function of n and k (do not assume k to be a constant).

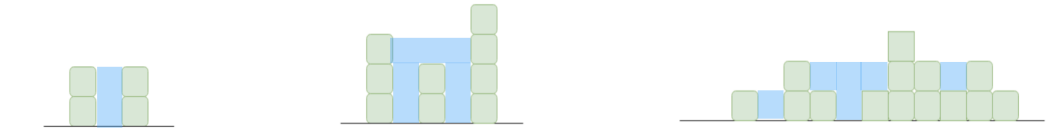
   (b) Describe an improved algorithm that runs in $O(n \lg k)$ time. After you describe it, argue why its running time is $O(n \lg k)$.

   We expect: pseudocode and an English description of your algorithm, and analysis of its running time.

2. **Flooding 1D-terrain**: You are given an array whose values represent heights of a one-dimensional terrain, sampled at 1m resolution. Imagine an arbitrarily large amount of water falling from the sky and flooding the terrain, and also imagine that the terrain is surrounded by a giant sink/ocean. Water will accumulate on top of a cell in the terrain unless it can find a path to the ocean. At some point flooding will reach steady state when further rain will no longer increase the flooding.

   The goal is to compute the total amount of water that is accumulated in the terrain (aka volume of flood) when steady state is reached. You may consider that the elevations are all non-negative (i.e. ≥ 0), and the width of each "pixel" (ie the distance between two consecutive heights in the array) is 1 unit.

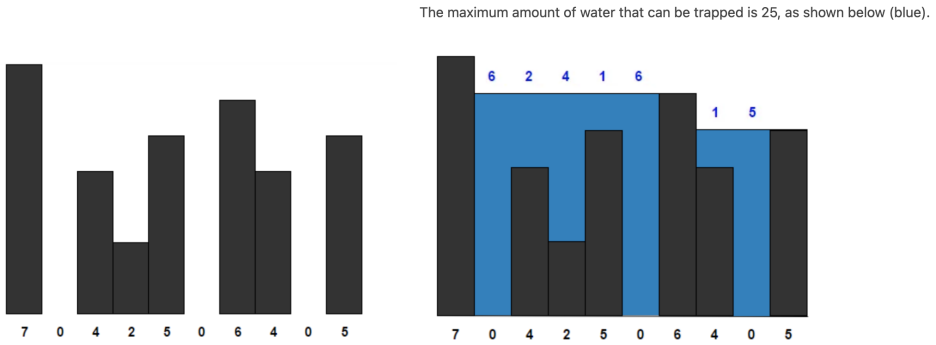
   Examples:



Input: A = [0, 2, 0, 2, 0]
Output: 2

Input: A = [0, 3, 0, 2, 0, 4, 0]
Output: 7

Input: A = [0, 1, 0, 2, 1, 0, 1, 3, 2, 1, 2, 1, 0]
Output: 6



Aim for a linear time algorithm.

We expect: (1) Formulate a claim on what is the height of water accumulated on $A[i]$, in terms of the elevation of the pixels to its left and right. Hint: consider min and max values. (2) high-level pseudocode ; (3) analysis of running time.

Evaluation

This assignment (and all subsequent assignments) will be evaluated along four general criteria:

(a) **Algorithm:** Is the algorithm clearly described ? Is the general idea included? Is high-level pseudocode included?[1]

(b) **Correctness**: Does the algorithm solve the problem?

(c) **Analysis**: Is the running time of your algorithm analysed?

(d) **Style**: Does it look professional and neat? Is the explanation written carefully in complete sentences, and well-organized logic? Is it easily human-readable? Is it compleete yet concise? Is it easy to understand? These kinds of questions do not affect correctness but greatly affect how readable the algorithm is.

[1]Pseudocode should be clear enough that a student who took 1101 can understand what your algorithm is doing and could implement it in a language of their choice, without thinking too hard. At the same time, pseudocode is **not actual code**, and should not include details that are straightforward and make the ideas hard to follow. For e.g.it is preffered to say "find the max element in the array" (basic straightforward process) rather then spell it out.