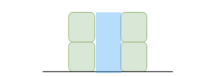
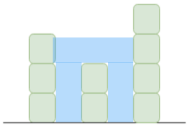
Problem: Trapping rain water on a 1D-terrain*

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.

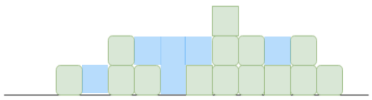
The goal is to compute the total amount of water that is accumulated in the terrain (aka volume of flood) when steady state is reached. You may consider that the elevations are all non-negative (i.e. ≥ 0), and the width of each "pixel" (ie the distance between two consecutive heights in the array) is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



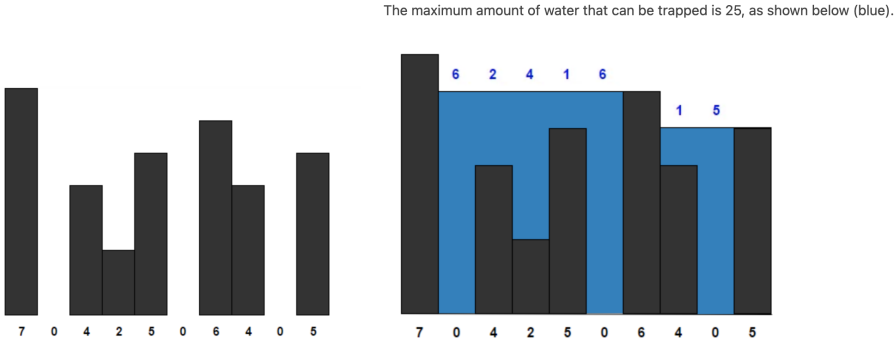
Input: A = [0, 2, 0, 2, 0]
Output: 2



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.

*Leetcode 42 (hard)