8:44 AM Tue Aug 3 100%

leetcode.com

LeetCode Explore Day 2 Problems Interview New Contest Discuss Store 🚀 August LeetCoding Challenge 2021 🚀

Description | Solution | Discuss (221) | Submissions

< Back **proof for C++/BFS/priority queue**

0 mz1007 ★ 1 Last Edit: December 9, 2020 11:10 AM 302 VIEWS

Solution:
https://leetcode.com/problems/trapping-rain-water-ii/discuss/89476/concise-C++-priority_queue-solution/94152

Proof:
Suppose v is the point we popped from the priority queue, w is its adjacent point, now we need to show res(w) == max(0, H-h(w)), where res(w) is the amount of water that should be accumulated at point w, h(w) is the bar height at w. Let H' be the max height before we update H = max(H', h(v)).

Note that (1) based on how we updated the visited point, we can always find a path connecting w and the border such that every point along this path (excluding w) is visited and its height is less or equal than H; (2) for all points to be visited, if it can accumulate water, the water level should be greater or equal than H'. (Because there exist a time when H is updated to H' due to a point popped from the priority queue, and **all the rest of the points in the priority queue, i.e., points that surround the rest of the to-be-visited points**, have height greater or equal than H'). weird, I think it is H

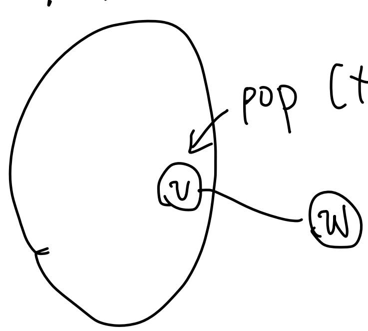
Case 1: if H <= h(w), due to (1), no water can be accumulated at w, hence res(w) = 0 = max(0, H-h(w)).
Case 2: if H>h(w), due to (1), res(w)<=H-h(w). Now we only need to show res(w)>=H-h(w). Since H = max(H', h(v)), if h(v)<=H', due to (2), res(w)>=H-h(w). Otherwise if h(v)>H', since v is the point popped from the priority queue, points that surround the to-be-visited points have height >= h(v), hence res(w)>=h(v)-h(w)=H-h(w). Q.E.D.

Comments: 0 Best | Most Votes | Newest to Oldest | Oldest to Newest

Type comment here... (Markdown is supported)

PQ H' $(H = \max(H', h(v)))$

pop (the smallest height)



Show: $\underbrace{res}_{\text{amount of water}}(w) = \max(0, H - h(w))$

Claim 1: $w \overset{\text{path}}{\rightsquigarrow}$ outside, max(height in path) $\le H$

Proof: If we follow

$$w \to v \rightsquigarrow \text{outside}$$

because H records the highest height so far

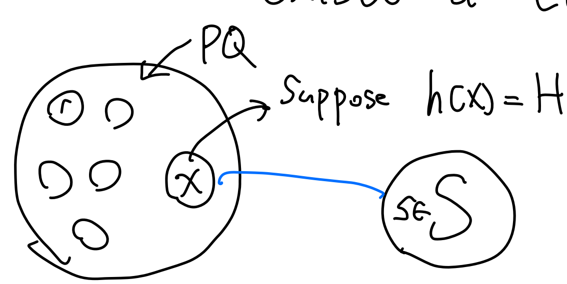
$$w \to v \rightsquigarrow \text{Outside}$$

$$h(v) \le H, \ h(v\text{'s parent}) \le H \ldots$$

Claim 2: Let S be the points we haven't visited yet.

The water level at $s \in S \ge H$

Proof: If H is updated to the current value. That means there exists a time that



At that time, the border height is at least $\ge H$ because $h(x)$ is the smallest.

There are two cases we will meet:

Case 1: $h(w) \geq H$:

By claim 1, there exists a path from $w \to v \rightsquigarrow$ outside where heights $\leq H$

$\Rightarrow$ water at $h(w)$ will leak

Thus $res(w) = \max(0, H - h(w))$
$= 0$

Case 2: $h(w) < H$

By claim 1:

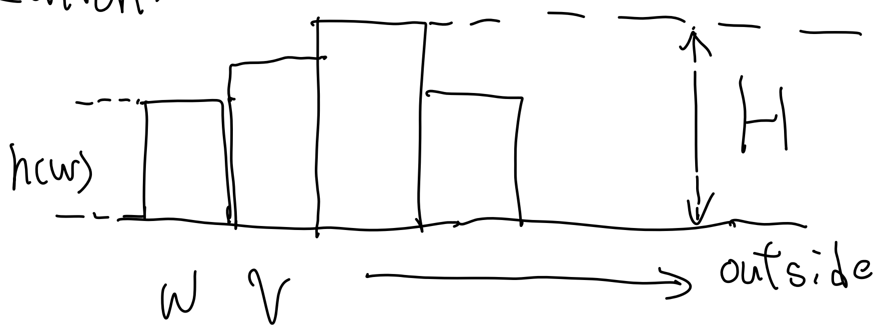
$w \to v \rightsquigarrow$ outside

The heights along this path $\leq H$

so the amount of water w can store is $\leq H - h(w)$

(i.e. $res(w) \leq H - h(w)$)

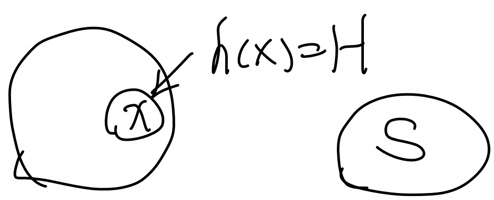
Visualization:



And we still need to prove $res(w) \geq H - h(w)$

(i) Assume $h(v) < H$:

by Claim 2, at some time, there is a cell x contributing this H, at that time, all cells in the PQ has $\geq H$ because $h(x)$ is the smallest

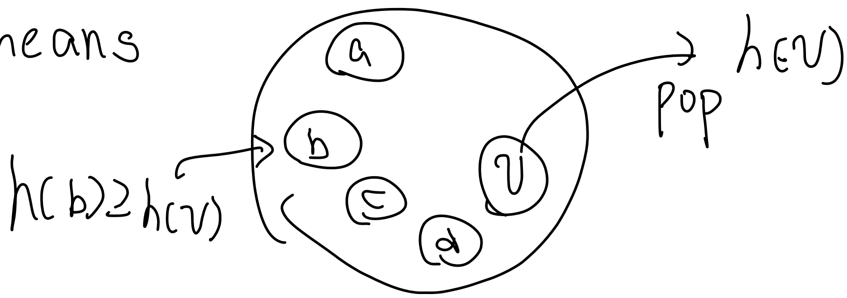


Thus: $res(X) \geq H - h(w)$

(ii) Assume $h(v) = H$

($h(v)$ contribute this H)

This means



all other cells' height $\geq H$

$\Rightarrow \quad res(X) \geq H - h(w)$

∎