Famous — 1-D arrays → VVI Interviews

Q→ Given an array of all non-negative integers where n is the length of the array.

A person is standing at the 0^{th} index of the array. And each element of the array represents the maximum jump a person can take from that index. Determine if the person standing on 0^{th} idx, can reach the last one.

[2, 3, 1, 1, 4] → true

[3, 2, 1, 0, 4] → false

$n \le 10^7$

$A \le 10^4$

src last_pos dest

(1) [2, 3, 1, 1, 4]

0 1 2 (3) 4

[3, 2, 1, 0, 4]

src dest

$\boxed{\text{last\_pos}}$

$\boxed{1 + arr[1] \geq 2}$

$0 + arr[0] \geq 1$

it doesn't matter

last_idx

$3 + 1 \Rightarrow 4$

denotes that we can make a jump of either 1, 2 or 3

src → $\boxed{\text{follow Sreeker}}$ → dest

(1)

If I am able to anyhow reach index 3, we can for sure reach index 4.

(jumps of 1)

TC $O(n)$, SC $O(1)$

last_pos

last_pos

3	3	1	0	4
0	1	2	3	4

$$\boxed{idx + arr[idx] \geq last\_pos}$$

$2 + arr[2] \geq 4$

NO

$1 + arr[1] \geq 4$

Yes

$0 + arr[0] \geq last\_pos$

$0 + 3 \geq 1$ Yes

0 1 2 3 4

[2, 3, 1, 1, 4]

idx + arr[idx] ≥ last

3 + arr[3] ≥ 4

4th → dest

for the dest index x

if anyhow we can reach to x, from x-1

then our problem reduces to reach

x-1 from source.

Q. Given an array of non negatives, you are present at the 0th idx, & every value of array represent max jump you can take from that idx.

Find the min no. of jump required to reach the last idx when you can assume, that you will get i/p such that you always reach destination

[2, 3, 1, 1, 4] ⟶ ans → 2

$N \le 10^7$

≤ i   max_pos   min jumps

2, 3, 1, 1, 4

→ what are jumps you can take? → max step

→ what is the max pos we can reach → ≤ i

curr max step

max_pos   ≤ i

max_step → 2
max_pos → ~~2~~ 4

2, 3, 1, 1, 4

i

jumps = 2

1 + 2 = 3

2 3 1 2 4 3 . . .

coverage
max_~~pos~~ → maximum pos, one could reach from current-ind or i or before. Anyhow we arrive at index i, what's the max coverable pos.

max-jump → maximum pos reachable after current jump

?

2, 3, 1, 1, 4
i

coverage
max ~~pos~~ = ~~2~~ 4
max jump = 2

jump = jump + 1

when (i == max-jump) we should have beyond one more jump at i or before i, to progress further

2, 3, 0 1, 4

4

max - coverage =

last - max - jump = 4

3 + 1 = 4

jump = 1 + 1

if required we will take a jump from idx that can carry till idx = 4

8, 2, 4, 4, 4, 9, 5, 2, 5, 8, 8, 0, 8, 6, 9, 1, 1, 6, 3

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18

i

last_max_jump = ~~14~~ 23

last_max_coverage = ~~20~~ 23

3 + 4

5 + 4

9 + 5 = 14

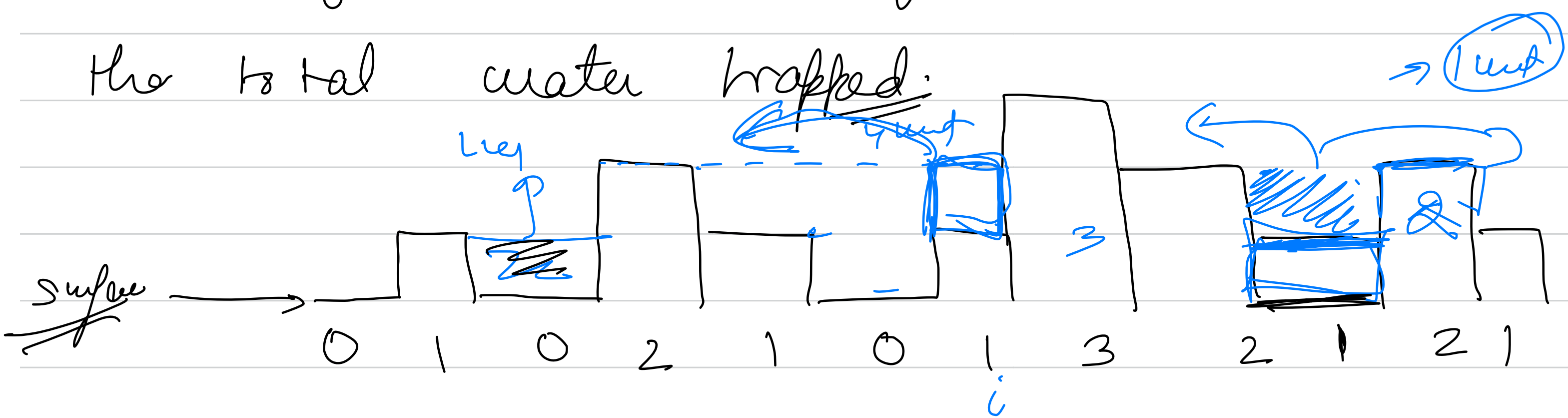
0

→ I can cover x index from here. Let's go to next index, but also keep exploring, the next best candidate index to take a jump from.

Rain water harvesting problem

Q: Given an array of size n, non-negative numbers, that represents heights of building when width of all buildings is 1 unit. If rain occurs, what is the total water trapped.

$n \le 10^6$



$m(2,3) - 1$

$2 - 1 = 1$

$(3,2)$

$2 - 1 = 1$

min (max height building at left, max height building at right) − height of curr building

0 1 2

	0	1	0	2	1	0	1	3	2	1	2	1
left max	0	1	1	2	2	2	2	3	3	3	3	3
right max	3	3	3	3	3	3	3	3	2	2	2	1

global left

global right

1

$(1,3) \to 1 - 1 = 0$

$2 - 1 = 1$

$2 - 0 = \ldots$

$2 - 1 = 1$

$1 - 0 = 1$

O(n)

O(n)

water = 0

2 pointers

$i \to$ left max[i]

$i \to$ right max[i]