

⇒ Points to Remember

① Revision

[at least
3
times]

Dead Zone

→ after class
→ At weekend
→ After month

② Never have backlog.

- Improper revision leads to backlog as you will end up taking more time to solve problems.
- If done properly then delight outcome is waiting.

After DSA

Mock Interviews [Replica of actual interview]

- Actual Industry experts will conduct
- 1 hour
- Post DSA Curriculum
- if clear the Scaler will open up placement opportunities for you.

Q1. Given an integer array A, find the maximum subarray sum out of all the subarrays.

Ex1 $\rightarrow [-2, \underline{3, 4, -1, 5}, -10, 7]$ ↴ 11 ans.

Ex2 $\underline{[-3, 4, 6, 8, -10, 2, 7]}$ ↴ 18 ans

$$\begin{aligned} \text{Quiz 1} &= [4, 5, 2, 1, 6] \\ &= 18 \end{aligned}$$

$$\begin{aligned} \text{Quiz 2} &= [-4, -3, -6, -9, \underline{-2}] \\ &\text{all arr negative} \quad \text{(-2)} \end{aligned}$$

Brute Force

3 loops

iterate each subarray
get maxm sum.

Pseudo code

$$\text{ans} = A[0]$$

for ($i=0$; $i < N$; $i++$) {

 for ($j=i$; $j < N$; $j++$) {

 for ($k=i$; $\overset{\text{start}}{k} \leq \overset{\text{end}}{j}$; $k++$) {

$$\text{sum} += A[k];$$

$$\text{ans} = \max(\text{ans}, \text{sum});$$

$$\text{sum} = 0;$$

}

return ans;

T.C :- $O(N^3)$

S.C :- $O(1)$.



Optim ↑

Prefix sum \rightarrow $O(N^2)$

S.C: $O(N)$ → fixed sum array.

Opt 2

Carry forward :

Pseudocell

$$\text{ans} = A[0];$$

for $i=0$ to $N-1$) ✓

$$\text{Sum} = 0$$

for ($j = i + N - 1$) {

Sum + = A[j];

ans₂ = max(ans, sum);

۳

۴

returnans.

optimize y

Kadane's Algorithm, ✓

Cos 1

[4, 2, 1, 6, 7]

↳ sum of m arrays .

Card 2

$$[-v_1 \quad -v_2 \quad -v_3 \quad -v_4 \quad -v_5 \quad -v_6]$$

max number

\Rightarrow Case 3

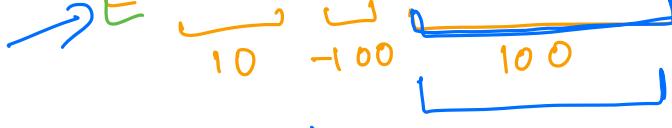
$$3 \xrightarrow{\quad} \left[\begin{array}{ccccccccc} + & + & - & + & + & + & + & - \\ \textcolor{orange}{\boxed{-1}} & & \textcolor{orange}{\boxed{-1}} & & \textcolor{orange}{\boxed{10}} & & \textcolor{orange}{\boxed{0}} & \end{array} \right] \xrightarrow{\quad} \boxed{1 = 10g}$$

$+ve + +ve = \text{greater positive}$

$\rightarrow v\varphi + \neg v\varphi = \text{less possible}$

$$\text{sh} \quad \begin{array}{c} \diagdown \\ \text{---} \end{array} \quad = 10^g$$

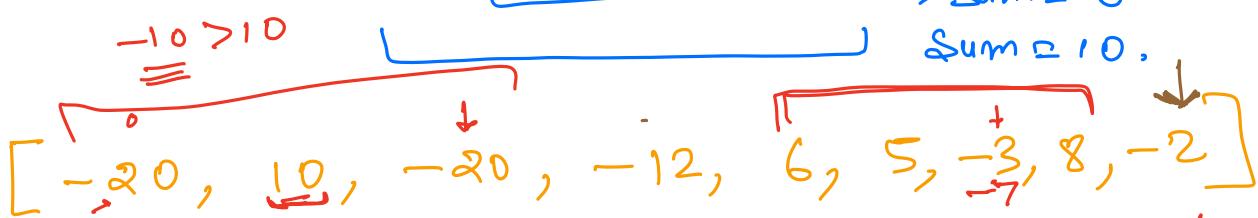
+ + - + + + + -] \quad \text{red}



$$\underline{\text{Sum1} = 0}$$

$$\underline{\text{Sum2} = 0}$$

$$\underline{\text{Sum} = 10.}$$



$$\underline{\text{run sum}} =$$

$$-20 \quad \underline{10}$$

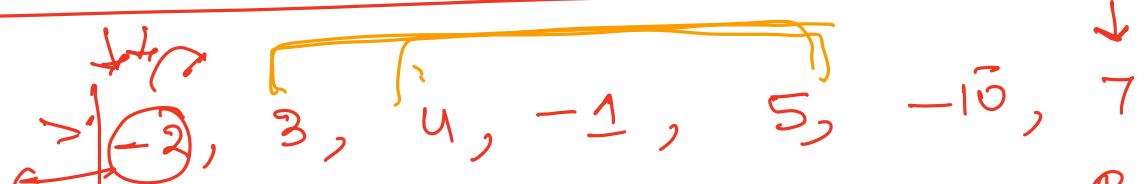
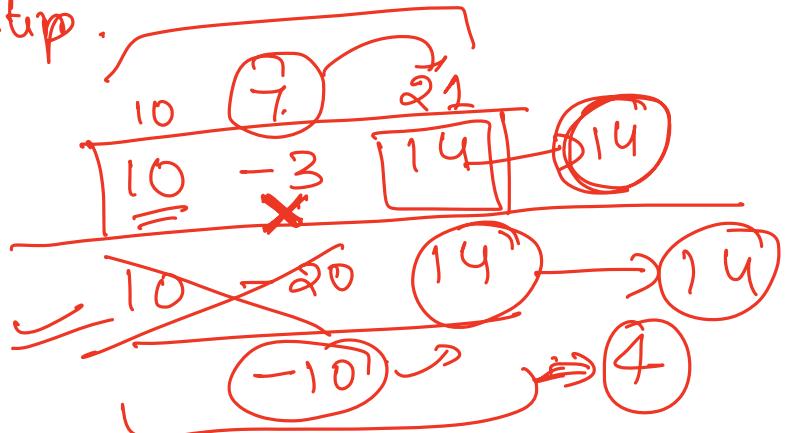
$$\underline{\text{max-sum}} = -20 \quad \underline{10} \quad 0 \quad 0 \quad 6 \quad 11 \quad 8 \quad 16 \quad 14$$

$$\underline{\text{Ans}} = 16.$$

calculate the running sum

if at any given point
run sum is $-ve$ then make it 0

calculate max-sum at each step.



$$\underline{\text{run-sum}}$$

$$\underline{\text{ans}}$$

$$\underline{\text{max-sum}} = -2 \quad 3 \quad ? \quad ? \quad ? \quad 11 \quad 11 \quad 11$$

4 -1 5

$\text{sum} = 4 \quad 3 \quad 8$
 $\text{max} = 4 \quad 4 \quad 8$

ANSWER

int arr[]
int max-sum = INT-MINVAL;

int sum-sum = 0

[10, -11, -5] for (i=0; i < arr.length; i++)

i	runsum	INT-MIN maxsum
0	-100	-10
1	-110	-10
2	-150	-5

→ sum-sum ≠ arr[i]

if (sum-sum > max-sum)
max-sum = sum-sum;

if (sum-sum < 0)
sum-sum = 0;

return max-sum;

[10, 5, 6]

T.C: O(n)

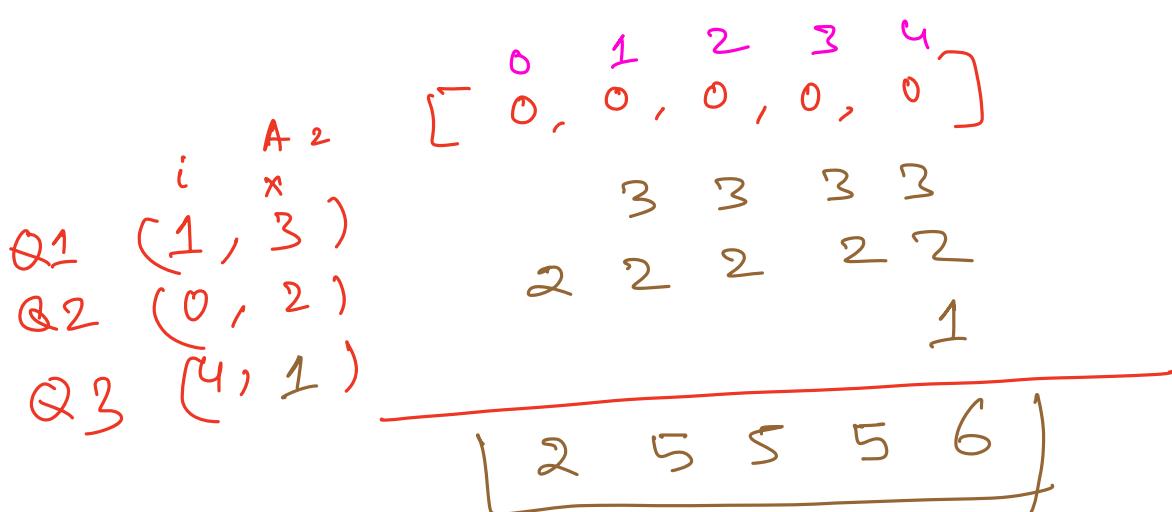
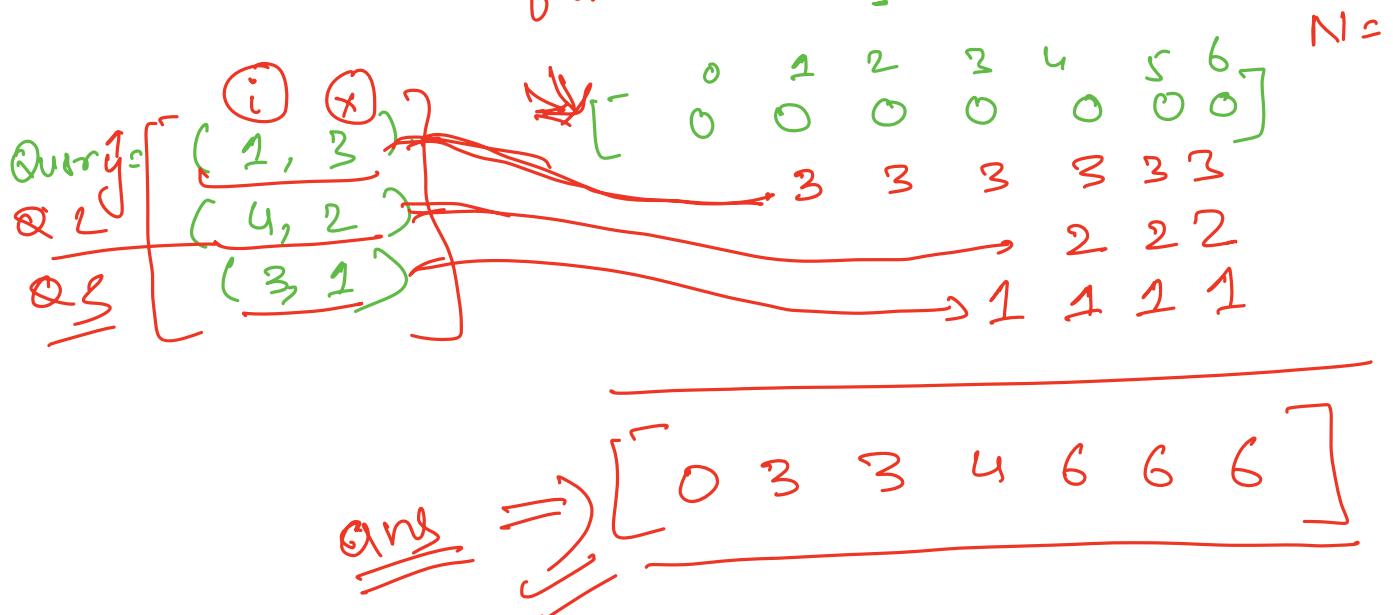
S.C: O(1)

i	sum-sum	INT-MIN max-sum
0	-100	-10
1	5	5
2	11	11

10:18 pm

Ques Given an integer array A where every element is 0. return the final array after performing multiple queries.

Query (i, x) = Add x to all the numbers from index i to $N-1$



Brute form

→ For each query iterate from $(i, n-1)$ and add x

$$T.C = O(Q \times n)$$

↓
no. of query.

length of array.

$$S.C = O(1)$$

Optimizd

0	1	2	3	4	5	6
0	0	0	0	0	0	0

- Q1(1, 3) +3
- Q2(0, 2) +2
- Q3(4, 1) +1

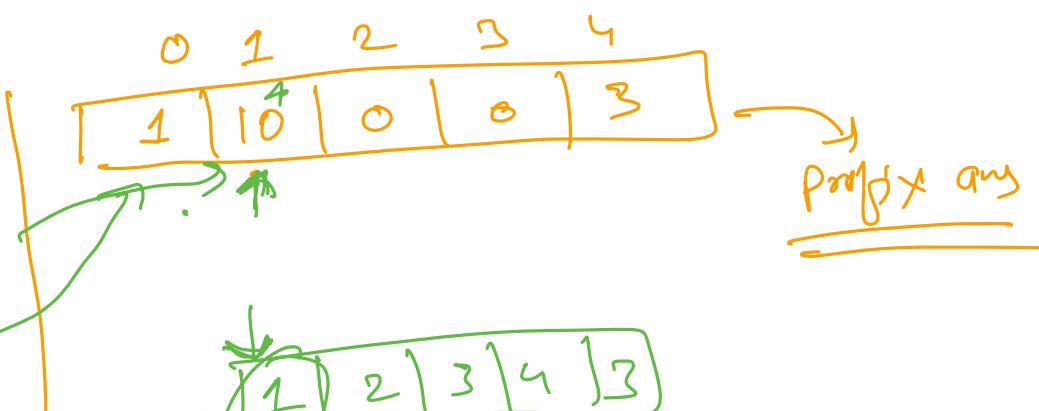
0	1	2	3	4	5
2	3	0	0	+1	0

0	1	2	3
2	5	5	5

$$\begin{aligned} T.C. &= O(Q+n) \\ S.C. &= O(1) \end{aligned}$$

① → iterate through all the queries and add x on i^{th} index

② → find prefix sum of the array.



$\rightarrow Q_3(4, 3)$

$Q_4(1, 6)$

int arr₂[]

for (i=0; i < Q.length; i++)

{

 int index = Q[i][0];

 int x = Q[i][1];

 arr[index] += x;

}

Query
are
processed.

for (i=1; i < arr.length; i++)

 arr[i] = arr[i] + arr[i-1];

}

return arr



Given an integer array A such that all the elements in the array are 0. Return the final array after performing multiple queries

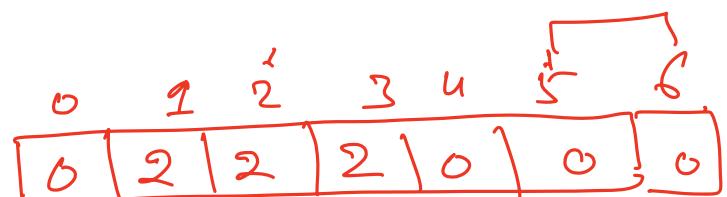
Query: (i, j, x): Add x to all the elements from index i to j

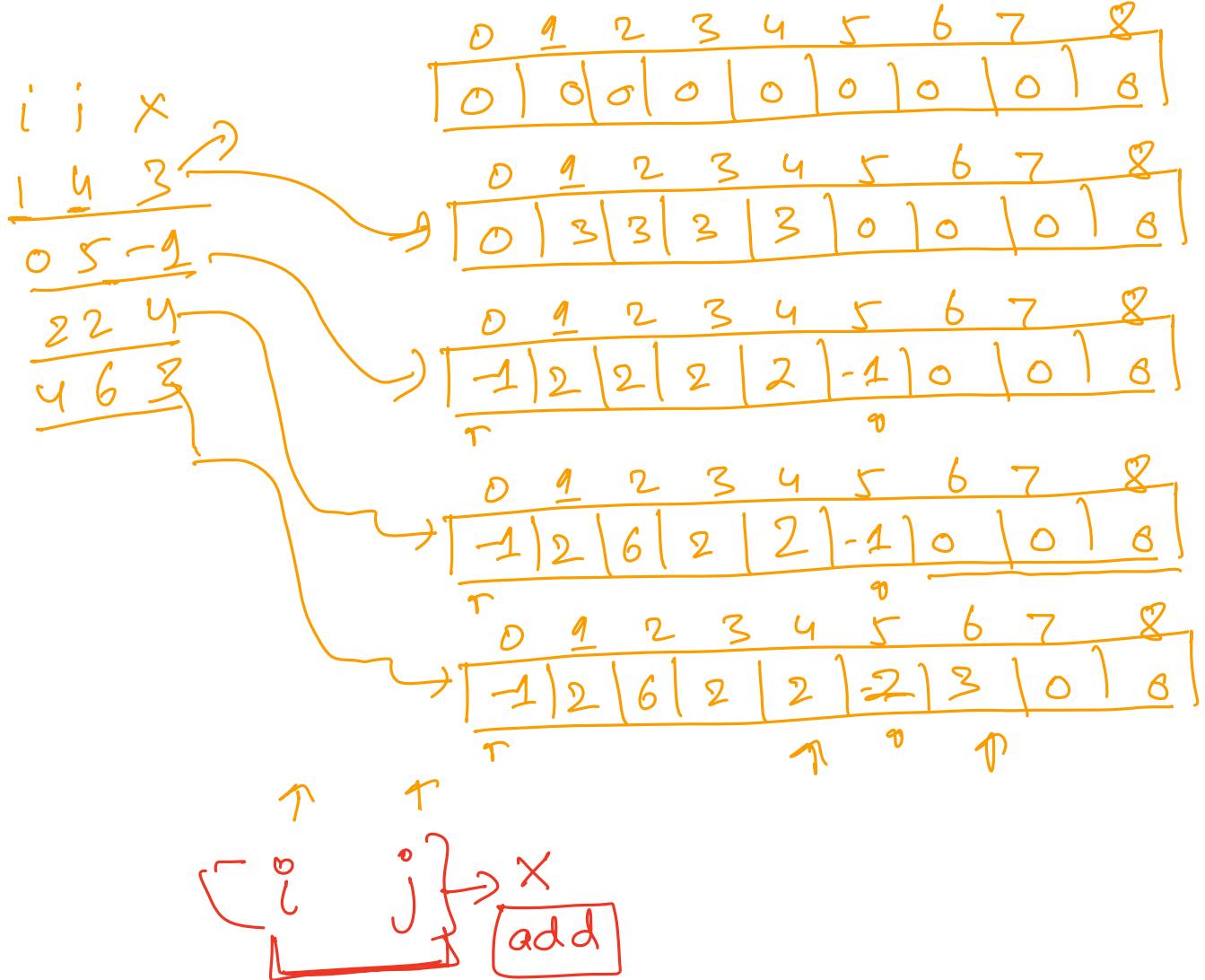
Given that $i \leq j$

$Q_1 = (1, 3, 2)$

$Q_2 = (2, 5, 3) \rightarrow$

$Q_3 = (5, 6, -1) \rightarrow$



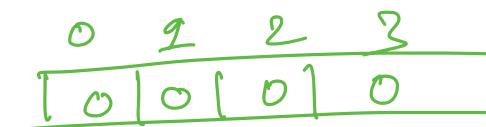
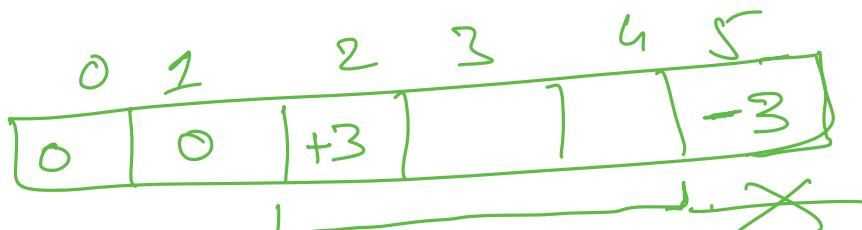


Brute force

iterate from i to j and add x

Optimizing \Rightarrow

$i \ i \ x$
 $\underline{2 \ 4 \ 3}$



1, 2, 1



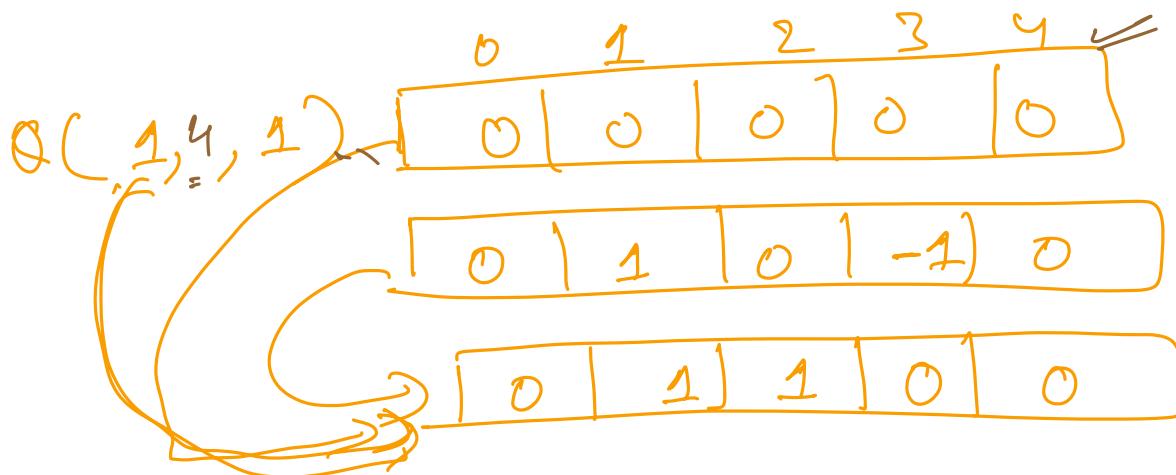
0, 1, 2



i, j
 $\boxed{-\nabla p}$

2	1	-2	-1
---	---	----	----

2	3	1	0
---	---	---	---



int n = 8

int arr = [0, 0, 0]

for (int i = 0; i < Q; i++)

T.C: $O(n+Q)$

S.C: $O(1)$

- start = $Q[i][0]$;
 - end = $Q[i][1]$;
 - x = $Q[i][2]$;
 {

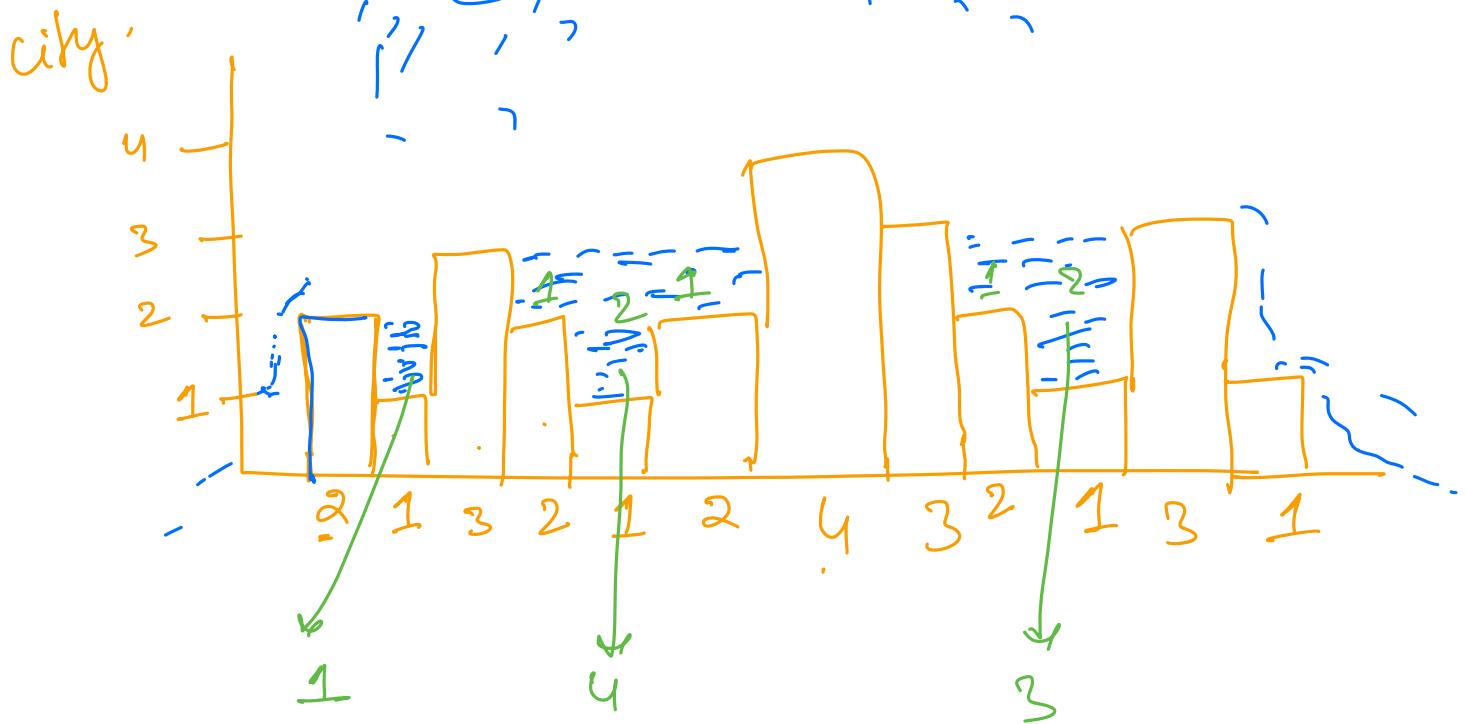
arr[start] += x;
 if (end + 1 < n) {
 arr[end + 1] += (-x);
 }

// Postfix Sum code

S

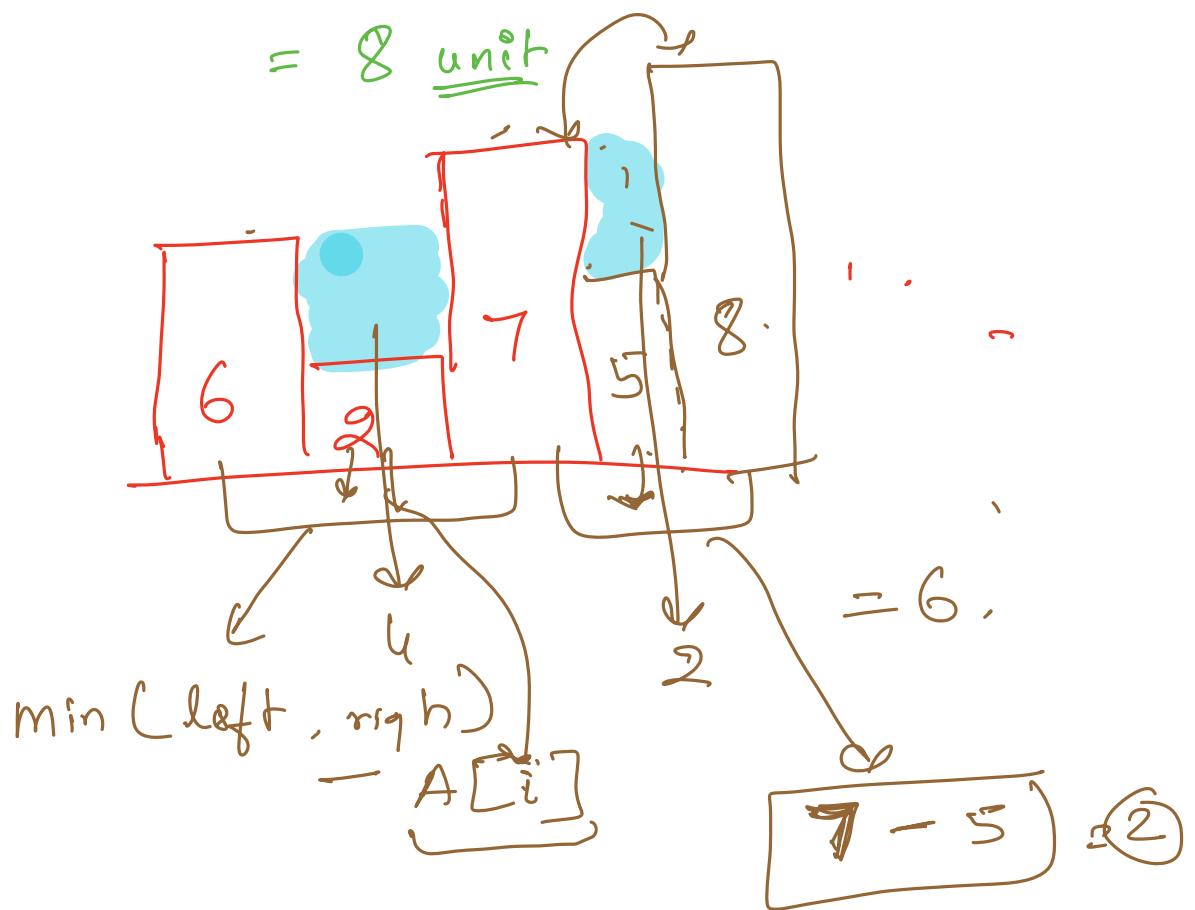
3

Q8

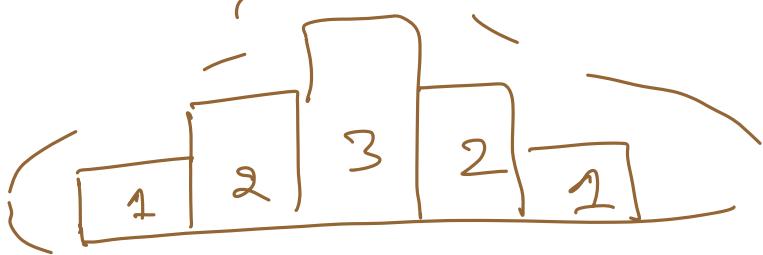


$$= 8 \text{ unit}$$

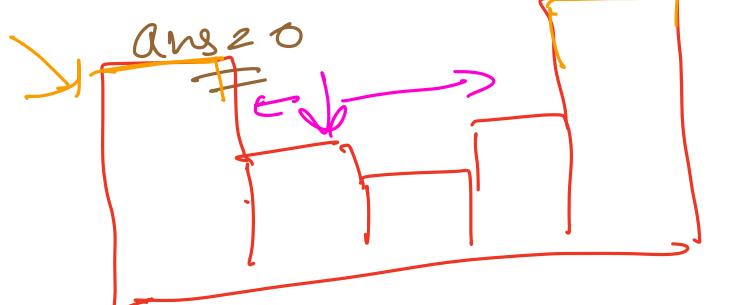
Q8s 1



1, 2, 3, 2, 1



Brute force



max(blue)
found
left

max
found
right

ans = 0

for (i = 1; i < N - 1; ++)

Lmax

Rmax

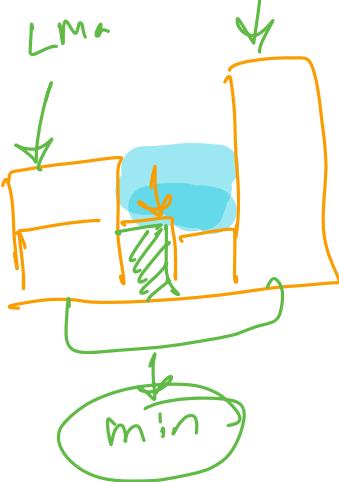
maxL = max(0 to i-1))

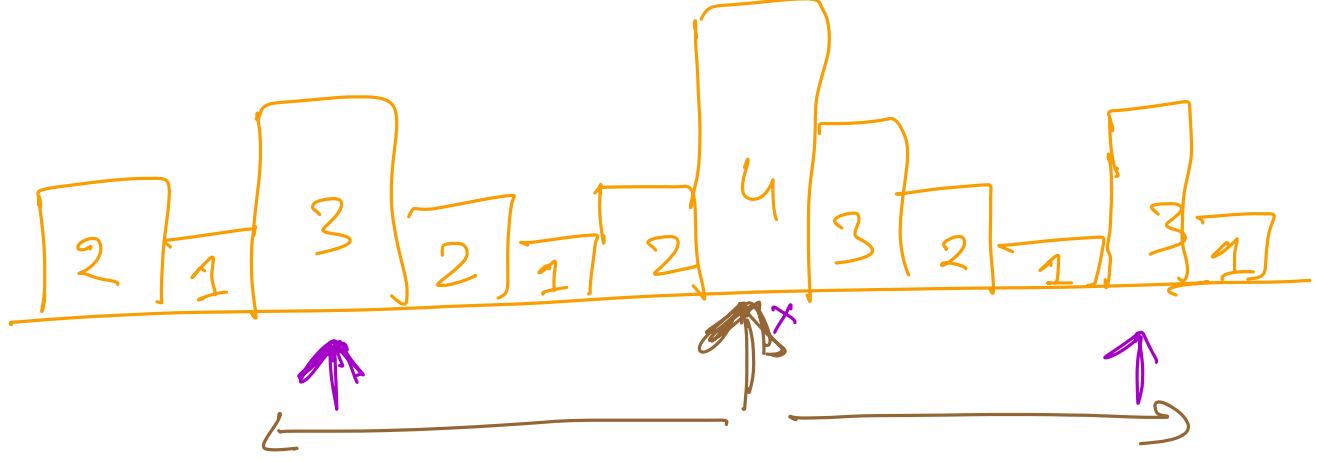
maxR = max(i+1 to N-1)

water = min(maxL, maxR)

= A[i];

if (water > 0)
ans += water



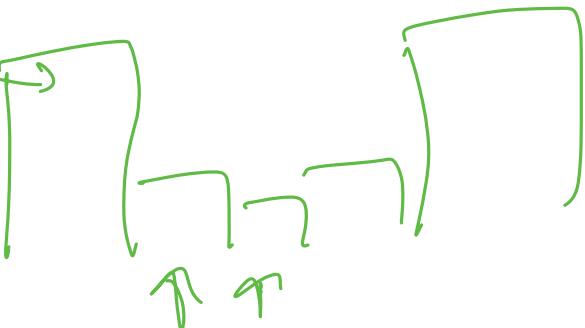


$$\text{water} = \min(3, 3) - A[3]$$

$$= 3 - 4$$

$$= -\frac{1}{1}$$

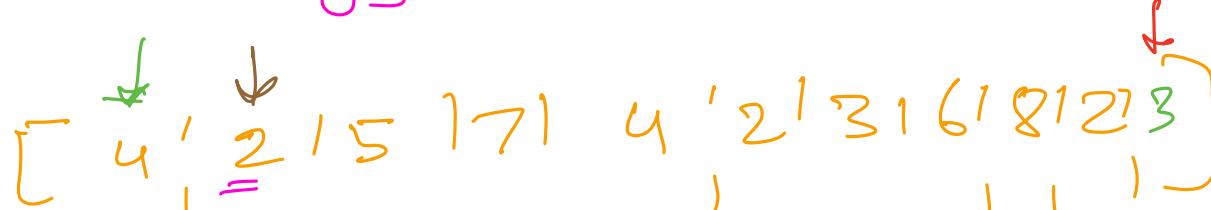
T.C. = $O(n^2)$
 S.C. = $O(1)$



Optimize it

Prefix max array]

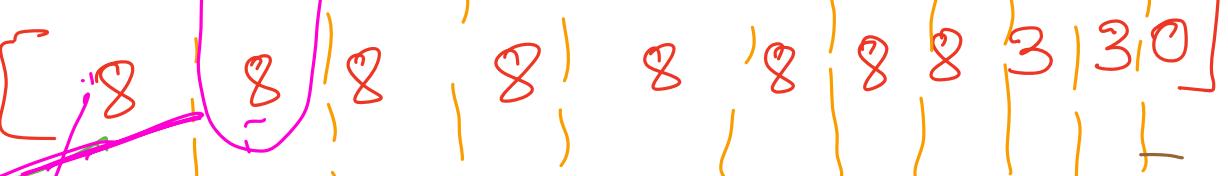
Suffix max array]



Left Max =



Right Max =



$$\min(\underbrace{\text{left max}, \text{right max}}_{\text{L.M.}, \text{R.M.}}) - A[i] = \underline{\underline{0}}$$

if no condition

$$\min(4, 8) - A[1]$$

$$4 - 2 = 2$$

// calculate left max prefix array
 // calculate right max suffix array

$\ell_{\max}[i] \rightarrow$

$\tau_{\max}[i] \rightarrow$

T.C. $O(n)$
 S.C. $O(n)$

for ($i = 1$; $i < n - 1$; $i++$)

{
 $\text{water} = \min(\ell_{\max}[i], \tau_{\max}[i]) - A[i]$

if ($\text{water} > 0$)

$\text{ans} += \text{water}$

}