

Q17. Best time to buy and sell stock.

i/p = {7,91,5,3,6,43

0/b - 5 - maximum profit 6-1=5

0 = (1) 490184

15 - Simply octure profit.

Brute force

Take i=0 and j=i+1 and then just find the maximum profit possible · j here will represent the selling Index & i represents the buying index and we don't have to care about the thing of first buying & then selling as j will always be ahead of i.

Optimal solution + Dry run
mini = INT\_MAX;

profit = 0; man + it and ent bar upmen

1) 1°=0

brices [i] = 7

7< INT\_MAX => True & Rence mini = 7

lint max Profit (vector sint & brices)

2) 1°=1

prices [i] = 1 X AM TUIT = min

1<7 => Trye & hence mini = 1.

3) 1 = 2 + 1 () 9512 20017 - 1 () 9512 20017 - 1

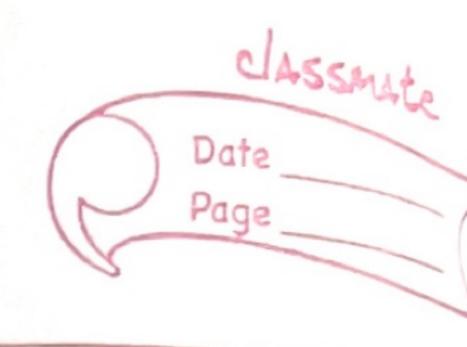
brices [i] = 5

5 <1 > False and hence

profit = max (0, 5-1) = max (0, 4) = 4

prices [i]

4) i'=30 ind 170 x pm = +170 md



prices TiJ = 3  $3 < 1 \Rightarrow$  False and hence profit = max(4, 3-1) = max(4, 2) = 4

- 5) i = 4 prices [i] = 6  $6 < 1 \Rightarrow False and hence$  profit = max (4, 6-1) = max (4,5) = 5
- 6) l=5prices [i]=4  $4 < l \Rightarrow False and hence$ profit = max(5, 4-1) = max(5, 3) = 5

Hence we have now fully traversed the array and the profit comes out to be 5. Simply return profit.

Code

int max Profit (vector <int > 4 prices) {

// Setting the initial values

int mini = INT-MAXi

int profit = 0;

// Traverse the whole array

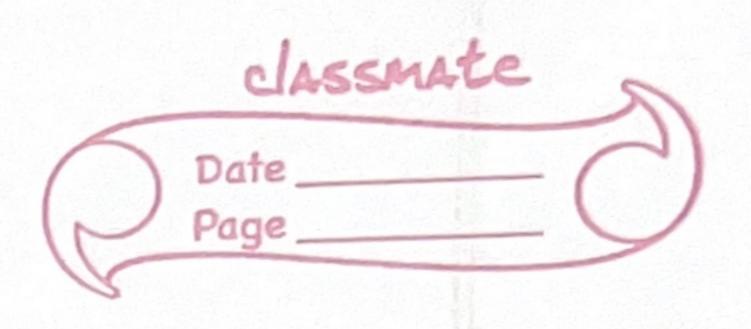
for (int i=0 ; i< prices Size(); i++)?

// Should buy if condition saished

if (prices [i] < mini) {

mini = prices [i];

else { // Should sell
profit = max (profit, prices [i]
mini



return profit j

Time complexity = 0(n) Space complexity = 0(1)