Best time to buy stock

121. Best Time to Buy and Sell Stock

Easy 24.6K 769

Companies

You are given an array `prices` where `prices[i]` is the price of a given stock on the i^{th} day.

You want to maximize your profit by choosing a **single day** to buy one stock and choosing a **different day in the future** to sell that stock.

Return *the maximum profit you can achieve from this transaction*. If you cannot achieve any profit, return `0`.

Example 1:

```
Input: prices = [7,1,5,3,6,4]
Output: 5
Explanation: Buy on day 2 (price = 1) and sell on day 5 (price = 6), profit = 6-1 = 5.
Note that buying on day 2 and selling on day 1 is not allowed because you must buy before you sell.
```

Example 2:

```
Input: prices = [7,6,4,3,1]
Output: 0
Explanation: In this case, no transactions are done and the max profit = 0.
```

Constraints:

- `1 <= prices.length <= 10^5`
- `0 <= prices[i] <= 10^4`

Brute force: Using 2 loop

→ use loop of i from 0 to n
→ use loop of j from i+1 to n
→ If arr[j] > arr[i] do ans= max(ans, arr[j]-arr[i]
→ Return ans

[7, 1, 5, 3, 6, 4] ans=0 T.C-O(n²) S.C-O(1)
i i i i i i
j j>i

[7, 1, 5, 3, 6, 4]
i
j j>i
max(ans, 5-1) | 3>1 True ans=4 | 6>1 max(ans, 6-1) ans=5
ans = 4

return ans;// 5

Optimal solution:

→ Create varible maxProfit = 0
→ Create a varible minPrices = INT_MAX
→ Run loop 0 to n
→ update minPrice if arr[i] is smaller than current minPrices

→ check maxprofit if it greater than current max profit

Code

```
minPrice = INT_MAX
maxprofit = 0
loop 0 -> n
    minPrice = min(minprices, arr[i])
    maxprofit = max(maxprofit, arr[i] - minprice)
```

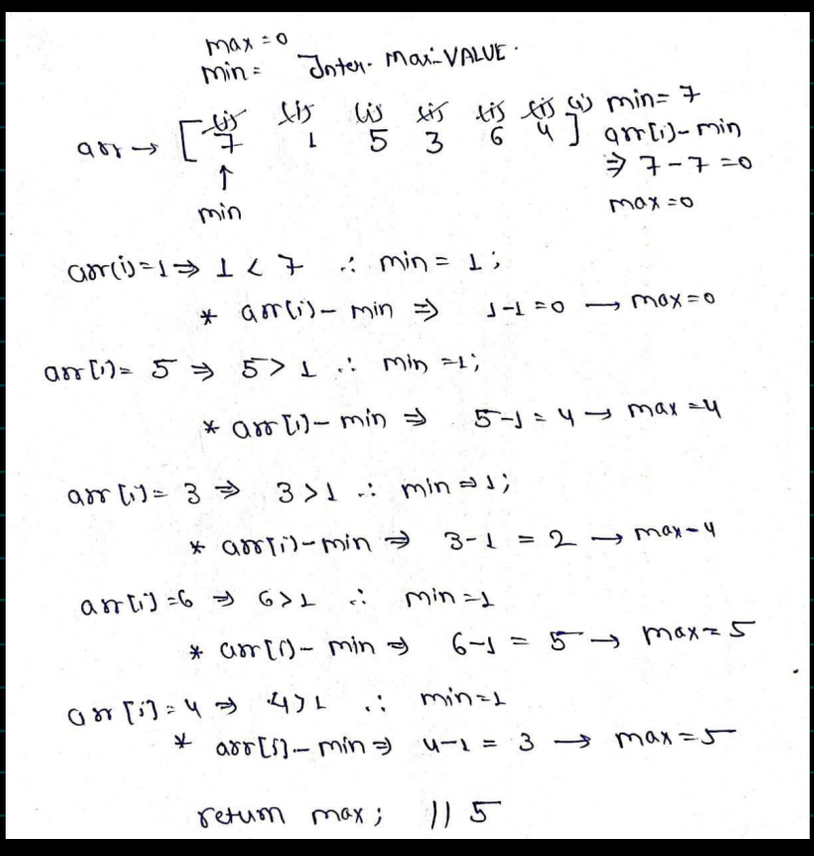
T.C-O(n)
S.C-O(1)

Code

Optimal Code

```cpp
class Solution {
public:
    int maxProfit(vector<int>& p) {
        int maxprofit=0,buy=p[0];
        int n=p.size();
        for(int i=1;i<n;i++)
        {
            if(buy>p[i]){
                buy=p[i];
            }
            maxprofit=max(maxprofit,p[i]-buy);
        }
        // cout<<"buy "<<buy<<"  max "<<maxprofit;
        return maxprofit;
    }

};
```



Taken from
Take you forward