121. Best Time to Buy and Sell Stock

June 23, 2016 | 424.1K views

Average Rating: 4.63 (169 votes)

Say you have an array for which the i^{th} element is the price of a given stock on day i.

If you were only permitted to complete at most one transaction (i.e., buy one and sell one share of the stock), design an algorithm to find the maximum profit.

Note that you cannot sell a stock before you buy one.

Example 1:

```
Input: [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.
             Not 7-1 = 6, as selling price needs to be larger than buying price.
```

Example 2:

```
Input: [7,6,4,3,1]
Output: 0
Explanation: In this case, no transaction is done, i.e. max profit = 0.
```

Solution

We need to find out the maximum difference (which will be the maximum profit) between two numbers in the given array. Also, the second number (selling price) must be larger than the first one (buying price).

In formal terms, we need to find $\max(\text{prices}[j] - \text{prices}[i])$, for every i and j such that $j > i$.

Approach 1: Brute Force

Java

Copy

```
public class Solution {
    public int maxProfit(int prices[]) {
        int maxprofit = 0;
        for (int i = 0; i < prices.length - 1; i++) {
            for (int j = i + 1; j < prices.length; j++) {
                int profit = prices[j] - prices[i];
                if (profit > maxprofit)
                    maxprofit = profit;
            }
        }
        return maxprofit;
    }
}
```

Complexity Analysis

- Time complexity : $O(n^2)$. Loop runs $\frac{n(n-1)}{2}$ times.
- Space complexity : $O(1)$. Only two variables - maxprofit and profit are used.

Approach 2: One Pass

Algorithm

Say the given array is:

[7, 1, 5, 3, 6, 4]

If we plot the numbers of the given array on a graph, we get:

The points of interest are the peaks and valleys in the given graph. We need to find the largest peak following the smallest valley. We can maintain two variables - minprice and maxprofit corresponding to the smallest valley and maximum profit (maximum difference between selling price and minprice) obtained so far respectively.

Java

Copy

```
public class Solution {
    public int maxProfit(int prices[]) {
        int minprice = Integer.MAX_VALUE;
        int maxprofit = 0;
        for (int i = 0; i < prices.length; i++) {
            if (prices[i] < minprice)
                minprice = prices[i];
            else if (prices[i] - minprice > maxprofit)
                maxprofit = prices[i] - minprice;
        }
        return maxprofit;
    }
}
```

Complexity Analysis

- Time complexity : $O(n)$. Only a single pass is needed.
- Space complexity : $O(1)$. Only two variables are used.

Rate this article:

Previous Next

Comments: 144

Sort By

Type comment here... (Markdown is supported)

Preview Post

Doombris ★600 June 26, 2018 9:45 PM

It's not a dynamic programming problem.

598 Share Reply

SHOW 27 REPLIES

momukhtar ★43 March 31, 2018 2:10 PM

I do not understand why is this a dynamic programming problem. The above solution looks simple. It just uses two simple variables. It does not use any memoization or tabulations as mentioned on this link.

This is the definition of dynamic programming as mentioned on this link:-

Read More

43 Share Reply

SHOW 3 REPLIES

IKnowWhatYouDidLastSummer ★31 August 26, 2018 12:28 AM

my code as follows,i write them by my intuition,
do u think my code is so short and clear ?

```
int maxProfit(int* prices, int pricesSize) {
```

Read More

30 Share Reply

SHOW 4 REPLIES

StayingFoolish ★58 February 9, 2019 12:59 PM

I wanna know whether approach 2 is DP and why

25 Share Reply

SHOW 9 REPLIES

kevin41 ★21 August 5, 2018 10:00 AM

I was thinking for the whole afternoon trying to come up with a DP solution, then cleared my mind when taking the shower and came up with sol2.

21 Share Reply

SHOW 4 REPLIES

terrible_whiteboard ★627 May 19, 2020 6:19 PM

I made a video if anyone is having trouble understanding the solution (clickable link)
https://youtu.be/mmIMpgh67vg

Read More

20 Share Reply

LittleOtterpop ★17 June 25, 2018 4:27 PM

Shouldn't it be maxprofit = max(maxprofit, prices[i] - minprice)?

Consider [4, 7, 1, 2]

i = 0: minprice = 4 maxprofit = 0

Read More

17 Share Reply

SHOW 5 REPLIES

CodingGeek123 ★108 August 23, 2018 9:41 AM

Kadane's Algorithm! :P

14 Share Reply

SHOW 6 REPLIES

jiawenzhong888 ★13 July 8, 2018 8:30 AM

Using java for approach 2

```
class Solution {
    public int maxProfit(int[] prices) {
```

Read More

13 Share Reply

hundanLi ★39 August 9, 2018 10:51 AM

My code:

```
public int maxProfit(int prices[]) {
        if (prices.length == 0 || prices.length == 1) {
```

Read More

23 Share Reply

SHOW 3 REPLIES

1 2 3 4 5 6 ... 14 15