LeetCode – Best Time to Buy and Sell Stock II (Java)

Say you have an array for which the ith element is the price of a given stock on day i.

Design an algorithm to find the maximum profit. You may complete as many transactions as you like (ie, buy one and sell one share of the stock multiple times). However, you may not engage in multiple transactions at the same time (ie, you must sell the stock before you buy again).

Analysis

This problem can be viewed as finding all ascending sequences. For example, given {5, 1, 2, 3, 4}, buy at 1 & sell at 4 is the same as buy at 1 &sell at 2 & buy at 2& sell at 3 & buy at 3 & sell at 4.

We can scan the array once, and find all pairs of elements that are in ascending order.

Java Solution

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

Related Posts:

1. **LeetCode – Best Time to Buy and Sell Stock (Java)**
2. **LeetCode – Best Time to Buy and Sell Stock III (Java)**
3. **LeetCode – Best Time to Buy and Sell Stock IV (Java)**
4. **LeetCode – 3Sum Closest (Java)**

Category >> Algorithms

If you want someone to read your code, please put the code inside <pre><code> and </code></pre> tags. For example:

```
<pre><code>
String foo = "bar";
</code></pre>
```

5 Comments Program Creek prateek shekhar

Recommend Share Sort by Best

Join the discussion…

curiousguy • 3 years ago

12,41,54,12,20

This input returns 42 as the answer. Shouldn't it be 50?
54-12 = 42
20-12 = 8

so 42+8 = 50??

Reply • Share ›

Sivaramakrishnan Vaidyanathan ➚ curiousguy • 2 years ago

It returns 50. (41-12) + (54-41) + (20-12). Not sure if this is still a question for you.

Reply • Share ›

Puneet • 3 years ago

The above code doesn't take care of this case stockPrices = {1,5,4,7}, above code would return 7 ((5-1) + (7-4)), instead of ((5-1) + (7-5)) = 6... Stock has to be bought before being sold... So the code below keeps track of the price stock was bought at..

```
public int maxProfit(int[] prices) {
int profit = 0;
// to keep track of when the stock was bought
int boughtAt = stockPrices[0];
for (int i = 1; i 0) { // profit
// update stock bought at
boughtAt = stockPrices[i];
// update profit
profit += diff;
}
}
}
```

Reply • Share ›

Sivaramakrishnan Vaidyanathan ➚ Puneet • 2 years ago

The answer is 7 and you said it yourself: (5-1) + (7-4).

Reply • Share ›

Valentin Konovalov ➚ Puneet • 3 years ago

7 is the correct answer. You can buy at 5 instead 4 on second step, but why?

Reply • Share ›