121. Best Time to Buy and Sell Stock

Easy 15642 527 Add to List Share

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.
```

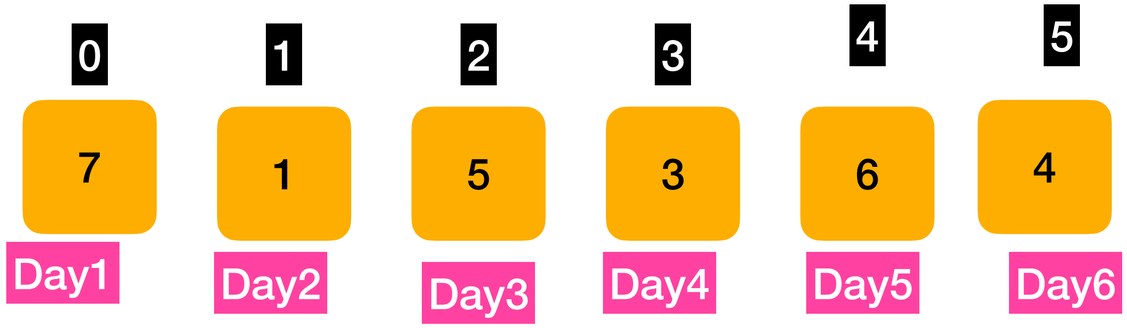
Example 2:

```
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`

Return the max Profit



Buy On Day2 — Sell it on Day3 — Profit = 5-1 =4

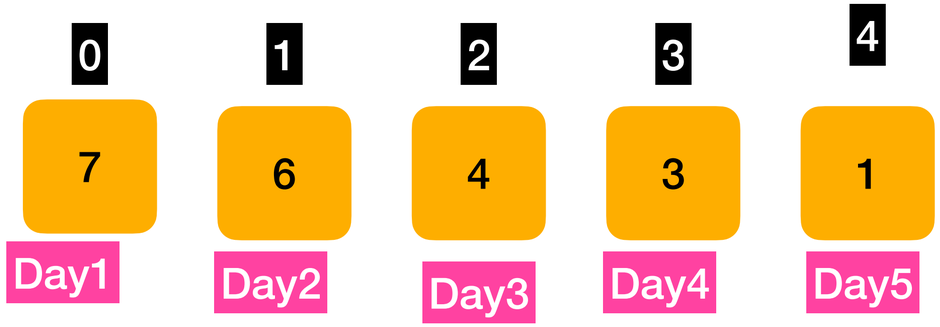
Buy On Day2 — Sell it on Day4 — Profit = 3-1 =2

Buy On Day2 — Sell it on Day5 — Profit = 6-1 =5 ✓

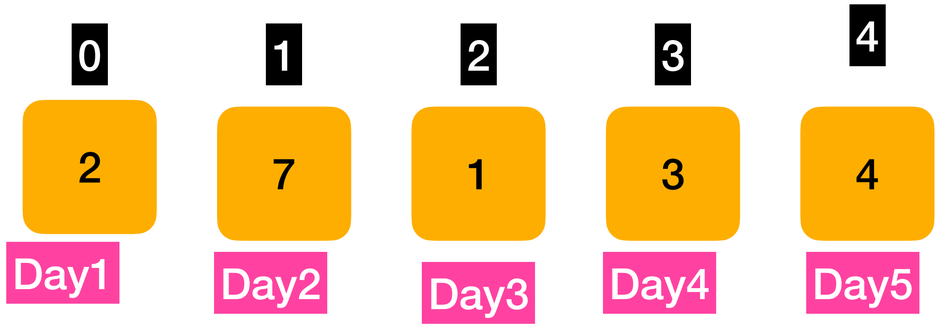
Buy On Day2 — Sell it on Day6 — Profit = 4-1 =1

Buy On Day4 — Sell it on Day5 — Profit = 6-3 =3

Buy On Day4 — Sell it on Day6 — Profit = 6-4 =2



Profit = 0



Sell it on Day2

Buy On Day1

Profit = 5

Sell it on Day5

Buy On Day4

Profit = 1

Sell it on Day4

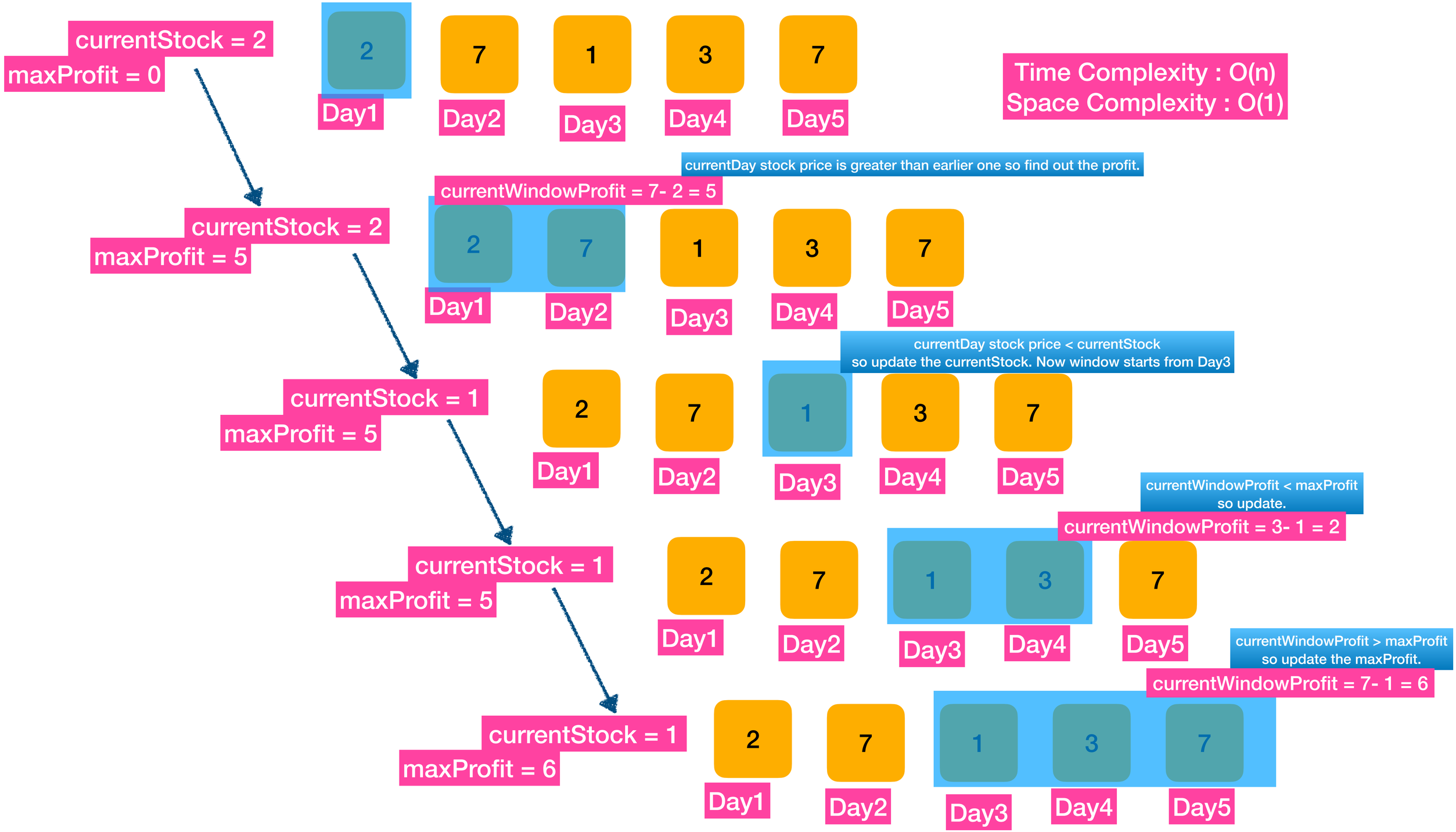
Buy On Day3

Profit = 2

Sell it on Day5

Buy On Day4

Profit = 3


currentStock = 2
maxProfit = 0
2
7
1
3
7
Day1
Day2
Day3
Day4
Day5
Time Complexity : O(n)
Space Complexity : O(1)
currentDay stock price is greater than earlier one so find out the profit.
currentWindowProfit = 7- 2 = 5
currentStock = 2
maxProfit = 5
currentStock = 1
maxProfit = 5
currentDay stock price < currentStock
so update the currentStock. Now window starts from Day3
currentWindowProfit < maxProfit
so update.
currentWindowProfit = 3- 1 = 2
currentStock = 1
maxProfit = 5
currentWindowProfit > maxProfit
so update the maxProfit.
currentWindowProfit = 7- 1 = 6
currentStock = 1
maxProfit = 6

122. Best Time to Buy and Sell Stock II

Medium 7367 2359 Add to List Share

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

On each day, you may decide to buy and/or sell the stock. You can only hold **at most one** share of the stock at any time. However, you can buy it then immediately sell it on the **same day**.

Find and return *the **maximum** profit you can achieve*.

Example 1:

```
Input: prices = [7,1,5,3,6,4]
Output: 7
Explanation: Buy on day 2 (price = 1) and sell on day 3 (price = 5), profit = 5-1 = 4.
Then buy on day 4 (price = 3) and sell on day 5 (price = 6), profit = 6-3 = 3.
Total profit is 4 + 3 = 7.
```

Example 2:

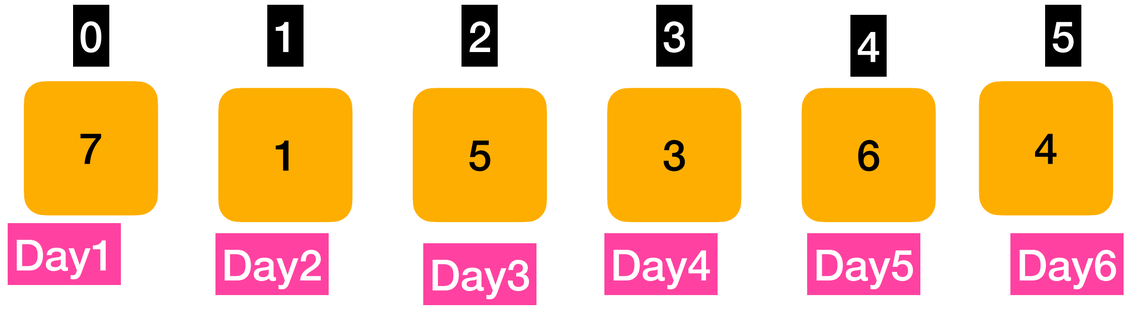
```
Input: prices = [1,2,3,4,5]
Output: 4
Explanation: Buy on day 1 (price = 1) and sell on day 5 (price = 5), profit = 5-1 = 4.
Total profit is 4.
```

Example 3:

```
Input: prices = [7,6,4,3,1]
Output: 0
Explanation: There is no way to make a positive profit, so we never buy the stock to achieve the maximum profit of 0.
```

Constraints:

- `1 <= prices.length <= 3 * 10^4`
- `0 <= prices[i] <= 10^4`



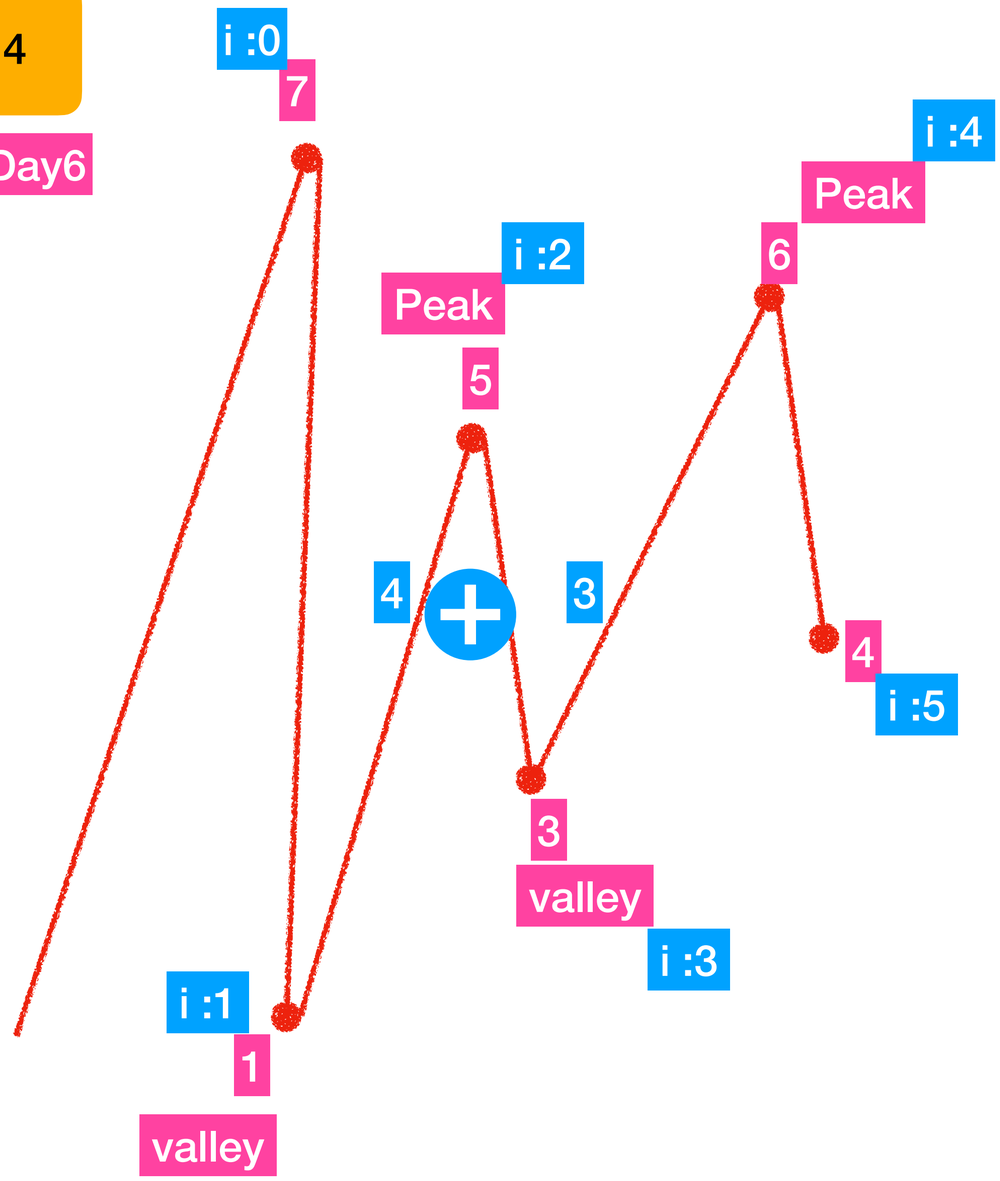
Buy On Day2 Sell on Day3 = 5-1 = 4

+

Buy On Day4 Sell on Day5 = 6-3 = 3

Total Profit = 7

FindOut each occurrence of valley and peak.
Summation of each difference
between peak & valley is the total profit.



7	1	0	5	7	4	8

Time Complexity : O(n)
Space Complexity : O(1)

FindOut each occurrence of valley and peak.
Summation of each difference
between peak & valley is the total profit