## 链滴

# ［每日 LeetCode］122．Best Time to Buy a nd Sell Stock II 

作者：Hanseltu
原文链接：https：／／ld246．com／article／1553613382195
来源网站：链滴
许可协议：署名－相同方式共享 4.0 国际（CC BY－SA 4．0）

## Description：

Say you have an array for which the $i^{\wedge}\{$ th $\}$ 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 ou like（i．e．，buy one and sell one share of the stock multiple times）．

Note：You may not engage in multiple transactions at the same time（i．e．，you must sell the st ck before you buy again）．

## Example 1：

Input：［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 $=6-3=3$ ．

## Example 2：

Input：［1，2，3，4，5］
Output： 4
Explanation：Buy on day 1 （price $=1$ ）and sell on day 5 （price $=5$ ），profit $=5-1=4$ ．
Note that you cannot buy on day 1，buy on day 2 and sell them later，as you are engaging multiple transactions at the same time．You must sell before buying again．

## Example 3：

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

思路：本题在121 Best Time to Buy and Sell Stock上增加了可以多次买入卖出。这里存在一个技巧只需要把每个相邻数组元素的值加起来，即可得最大差值，即最大利润。

```
C++代码
class Solution {
public:
    int maxProfit(vector<int> & prices) {
        int res = 0, n = prices.size();
        for (int i = 0; i < n - 1; ++i) {
            if (prices[i] < prices[i + 1]) {
                res += prices[i + 1] - prices[i];
            }
        }
        return res;
    }
};
```

运行时间： 8 ms
运行内存： 9.4 M

