

## 股票的最大利润

假设把某股票的价格按照时间先后顺序存储在数组中，请问买卖该股票一次可能获得的最大利润是多少？

示例 1:

输入: [7,1,5,3,6,4]

输出: 5

解释: 在第 2 天（股票价格 = 1）的时候买入，在第 5 天（股票价格 = 6）的时候卖出，最大利润 = 6-1 = 5 。

注意利润不能是 7-1 = 6，因为卖出价格需要大于买入价格。

示例 2:

输入: [7,6,4,3,1]

输出: 0

解释: 在这种情况下，没有交易完成，所以最大利润为 0。

```
class Solution {
public:
    /*
    能得到的最大利益就是未来能卖出的股票的最大值 减去现在你购买股票的成本
    而要得到未来能卖出的股票的最大值
    假设你现在处在索引 i 则你需要知道索引 i 之后的数组最大值
    那么可以反向遍历，从第 n - 1 个元素开始，始终维护一个最大值 rmax
    */
    int maxProfit(vector<int>& prices) {
        int resmax=0;
        int curmax=0;
        for(int i=prices.size()-1;i>=0;i--)
        {
            curmax=max(curmax,prices[i]); //更新未来（后方）局部价格最大值
            resmax=max(resmax,curmax-prices[i]); //更新利润最大值
        }
        return resmax;
    }
};
```