

```

import Foundation
/// 剑指 Offer 63. 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 {
    func maxProfit(_ prices: [Int]) -> Int {
        /// 前 i 日的最大利润 & 前 i 日的最低价格
        var (profit, minPrice): (Int, Int) = (0, Int.max)
        /// 找准状态转移方程
        /// **前 i 日的最大利润 = MAX[前 i-1 日的最大利润, 第 i 日的价格 - 前 i 日的最低价格]**
        for price in prices {
            minPrice = min(price, minPrice)
            profit = max(profit, price - minPrice)
        }
        return profit
    }
}

let solution = Solution()
print(solution.maxProfit([7,1,5,3,6,4])) // 5
print(solution.maxProfit([7,6,4,3,1])) // 0

```