一、题目说明

题目309. Best Time to Buy and Sell Stock with Cooldown，计算买卖股票的最大利润，卖出股票后第二天不能买入，需要冷却一天。

二、我的解答

这个题目，我没做出来。看了高手的解答，才知道用dp。

```
class Solution{
    public:
        //sold[i] = hold[i-1] + price[i];
        //hold[i] = max(hold[i-1], rest[i-1] - price[i])
        //rest[i] = max(rest[i-1], sold[i-1])
        //最后一天max(sold，rest)
        int maxProfit(vector<int>& prices){
            int sold=0,rest=0,hold=INT_MIN;
            for(int p: prices){
                int pre_sold = sold;// sold[i-1]
                sold = hold + p; //sold[i]
                hold = max(hold,rest-p); // hold[i]
                rest = max(rest,pre_sold);
            }
            return max(sold,rest);
        }
};
```

性能如下：

```
Runtime: 0 ms, faster than 100.00% of C++ online submissions for Best Time to
Buy and Sell Stock with Cooldown.
Memory Usage: 8.7 MB, less than 55.56% of C++ online submissions for Best Time
to Buy and Sell Stock with Cooldown.
```

三、优化措施

dp状态方程怎么来？

每一天有3个状态：

持有hold：可以是前一天买入的继续持有；或者前一天冷却今天买入，取其最大值。

`max(hold[i-1], rest[i-1] - price[i])`

卖出sold: 卖出是前一天持有量 + 卖出的价格

`sold[i] = hold[i-1] + price[i]`

冷却rest：前一天冷却今天继续冷却，或者前一天卖出今天冷却，取其最大值。

`rest[i] = max(rest[i-1], sold[i-1])`