714. Best Time to Buy and Sell Stock with Transaction Fee

DescriptionHintsSubmissionsDiscussSolution

- Difficulty:Medium
- Total Accepted:5.3K
- Total Submissions:13.4K
- Contributor: Jiachen_

Your are given an array of integers `prices`, for which the `i`-th element is the price of a given stock on day `i`; and a non-negative integer `fee` representing a transaction fee.

You may complete as many transactions as you like, but you need to pay the transaction fee for each transaction. You may not buy more than 1 share of a stock at a time (ie. you must sell the stock share before you buy again.)

Return the maximum profit you can make.

Example 1:

```
Input: prices = [1, 3, 2, 8, 4, 9], fee = 2
Output: 8
Explanation: The maximum profit can be achieved by:
Buying at prices[0] = 1Selling at prices[3] = 8Buying at prices[4] = 4Selling at prices[5] = 9The total profit is ((8 - 1) - 2) + ((9 - 4) - 2) = 8.
```

Note:

- `0 < prices.length <= 50000`.
- `0 < prices[i] < 50000`.
- `0 <= fee < 50000`.

```
class Solution {

public:

   int maxProfit(vector<int>& prices, int fee) {

      int l=prices.size();

      vector<int> notHold(l+1,0);

      vector<int> hold(l+1,0);

      hold[0]=INT_MIN;

      for(int i=1;i<=l;++i)

      {

         // we buy the stock at day i-1

         hold[i] = max(hold[i-1],notHold[i-1]-prices[i-1]-fee);
```

```
            notHold[i] = max(notHold[i-1],hold[i-1]+prices[i-1]);
        }
        return notHold[l];
    }
};
```