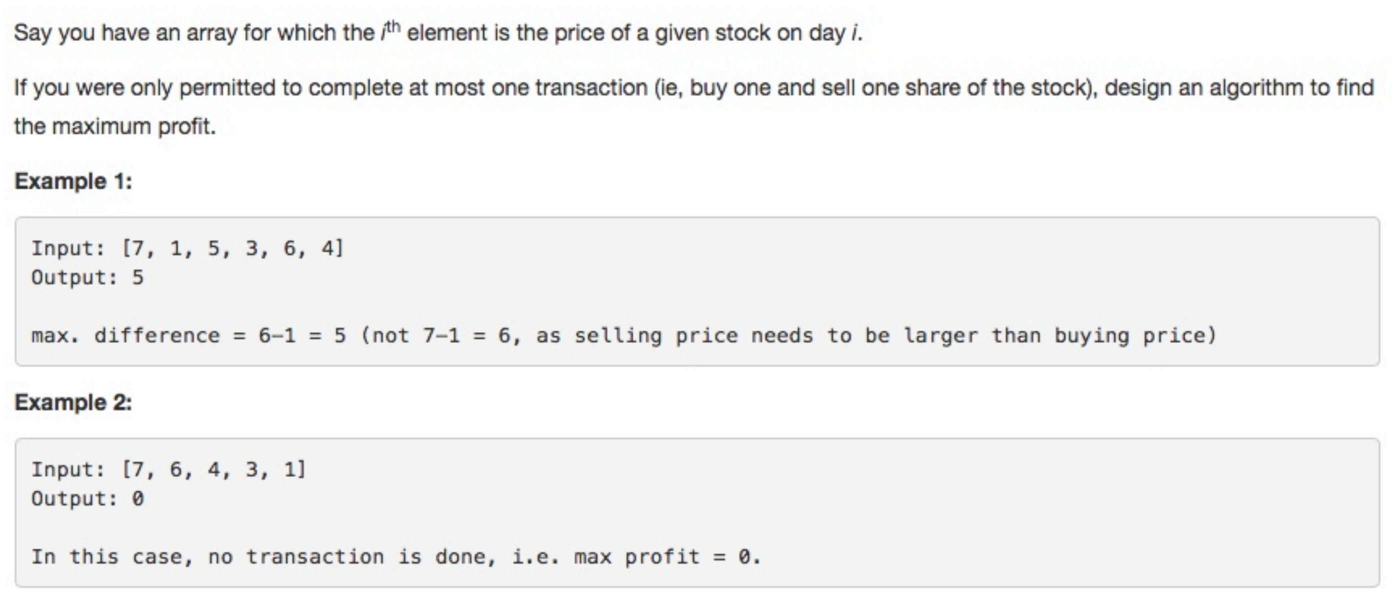
121. Best Time to Buy and Sell Stock

121 Best Time to Buy and Sell Stock

- **Dynamic Programming** + array

Description



Say you have an array for which the i^{th} element is the price of a given stock on day *i*.

If you were only permitted to complete at most one transaction (ie, buy one and sell one share of the stock), design an algorithm to find the maximum profit.

Example 1:

```
Input: [7, 1, 5, 3, 6, 4]
Output: 5

max. difference = 6-1 = 5 (not 7-1 = 6, as selling price needs to be larger than buying price)
```

Example 2:

```
Input: [7, 6, 4, 3, 1]
Output: 0

In this case, no transaction is done, i.e. max profit = 0.
```

1. Thought line

- buy is always before sell
- looking for the minimum of buy price
- looking for the maximum of sell price

2. Dynamic Programming + array

```
class Solution {
public:
    int maxProfit(vector<int>& prices) {
        if (prices.size() <= 1) return 0;
        int result = 0, buy = prices[0];
        for (int i = 1; i <= prices.size()-1; ++i){
            // possible to buy
            if (prices[i] < buy) buy = prices[i];
            // possible to sell (sell after buy)
            else result = (prices[i]-buy > result)?prices[i]-buy:result;
        }
        return result;
    }
};
```