Problem 309: Best Time to Buy and Sell Stock with Cooldown

Problem Information

Difficulty: Medium
Acceptance Rate: 61.32%
Paid Only: No
Tags: Array, Dynamic Programming

Problem Description

You are given an array `prices` where `prices[i]` is the price of a given stock on the `ith` day.

Find the maximum profit you can achieve. You may complete as many transactions as you like (i.e., buy one and sell one share of the stock multiple times) with the following restrictions:

* After you sell your stock, you cannot buy stock on the next day (i.e., cooldown one day).

Note: You may not engage in multiple transactions simultaneously (i.e., you must sell the stock before you buy again).

Example 1:

Input: prices = [1,2,3,0,2] **Output:** 3 **Explanation:** transactions = [buy, sell, cooldown, buy, sell]

Example 2:

Input: prices = [1] **Output:** 0

Constraints:

* `1 <= prices.length <= 5000` * `0 <= prices[i] <= 1000`

Code Snippets

C++:

```cpp
class Solution {
public:
int maxProfit(vector<int>& prices) {

}
};
```

Java:

```java
class Solution {
public int maxProfit(int[] prices) {

}
}
```

Python3:

```python
class Solution:
def maxProfit(self, prices: List[int]) -> int:
```