Problem 2110: Number of Smooth Descent Periods of a Stock

Problem Information

Difficulty: Medium
Acceptance Rate: 59.92%
Paid Only: No
Tags: Array, Math, Dynamic Programming

Problem Description

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

A **smooth descent period** of a stock consists of **one or more contiguous** days such that the price on each day is **lower** than the price on the **preceding day** by **exactly** `1`. The first day of the period is exempted from this rule.

Return _the number of**smooth descent periods**_.

Example 1:

Input: prices = [3,2,1,4] **Output:** 7 **Explanation:** There are 7 smooth descent periods: [3], [2], [1], [4], [3,2], [2,1], and [3,2,1] Note that a period with one day is a smooth descent period by the definition.

Example 2:

Input: prices = [8,6,7,7] **Output:** 4 **Explanation:** There are 4 smooth descent periods: [8], [6], [7], and [7] Note that [8,6] is not a smooth descent period as 8 - 6 ≠ 1.

Example 3:

Input: prices = [1] **Output:** 1 **Explanation:** There is 1 smooth descent period: [1]

Constraints:

* `1 <= prices.length <= 105` * `1 <= prices[i] <= 105`

Code Snippets

C++:

```
class Solution {
public:
long long getDescentPeriods(vector<int>& prices) {

}
};
```

Java:

```
class Solution {
public long getDescentPeriods(int[] prices) {

}
}
```

Python3:

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