Problem 1052: Grumpy Bookstore Owner

Problem Information

Difficulty: Medium
Acceptance Rate: 63.95%
Paid Only: No
Tags: Array, Sliding Window

Problem Description

There is a bookstore owner that has a store open for `n` minutes. You are given an integer array `customers` of length `n` where `customers[i]` is the number of the customers that enter the store at the start of the `ith` minute and all those customers leave after the end of that minute.

During certain minutes, the bookstore owner is grumpy. You are given a binary array grumpy where `grumpy[i]` is `1` if the bookstore owner is grumpy during the `ith` minute, and is `0` otherwise.

When the bookstore owner is grumpy, the customers entering during that minute are not **satisfied**. Otherwise, they are satisfied.

The bookstore owner knows a secret technique to remain **not grumpy** for `minutes` consecutive minutes, but this technique can only be used **once**.

Return the **maximum** number of customers that can be _satisfied_ throughout the day.

Example 1:

Input: customers = [1,0,1,2,1,1,7,5], grumpy = [0,1,0,1,0,1,0,1], minutes = 3

Output: 16

Explanation:

The bookstore owner keeps themselves not grumpy for the last 3 minutes.

The maximum number of customers that can be satisfied = 1 + 1 + 1 + 1 + 7 + 5 = 16.

Example 2:

Input: customers = [1], grumpy = [0], minutes = 1

Output: 1

Constraints:

* `n == customers.length == grumpy.length` * `1 <= minutes <= n <= 2 * 104` * `0 <= customers[i] <= 1000` * `grumpy[i]` is either `0` or `1`.

Code Snippets

C++:

```
class Solution {
public:
int maxSatisfied(vector<int>& customers, vector<int>& grumpy, int minutes) {

}
};
```

Java:

```
class Solution {
public int maxSatisfied(int[] customers, int[] grumpy, int minutes) {

}
}
```

Python3:

```
class Solution:
def maxSatisfied(self, customers: List[int], grumpy: List[int], minutes: int)
-> int:
```