Problem 1493: Longest Subarray of 1's After Deleting One Element

Problem Information

Difficulty: Medium
Acceptance Rate: 70.91%
Paid Only: No
Tags: Array, Dynamic Programming, Sliding Window

Problem Description

Given a binary array `nums`, you should delete one element from it.

Return _the size of the longest non-empty subarray containing only_`1` _' s in the resulting array_. Return `0` if there is no such subarray.

Example 1:

Input: nums = [1,1,0,1] **Output:** 3 **Explanation:** After deleting the number in position 2, [1,1,1] contains 3 numbers with value of 1's.

Example 2:

Input: nums = [0,1,1,1,0,1,1,0,1] **Output:** 5 **Explanation:** After deleting the number in position 4, [0,1,1,1,1,1,0,1] longest subarray with value of 1's is [1,1,1,1,1].

Example 3:

Input: nums = [1,1,1] **Output:** 2 **Explanation:** You must delete one element.

Constraints:

* `1 <= nums.length <= 105` * `nums[i]` is either `0` or `1`.

Code Snippets

C++:

```
class Solution {
public:
int longestSubarray(vector<int>& nums) {

}
};
```

Java:

```
class Solution {
public int longestSubarray(int[] nums) {

}
}
```

Python3:

```
class Solution:
def longestSubarray(self, nums: List[int]) -> int:
```