Problem 1063: Number of Valid Subarrays

Problem Information

Difficulty: Hard
Acceptance Rate: 79.52%
Paid Only: Yes
Tags: Array, Stack, Monotonic Stack

Problem Description

Given an integer array `nums`, return _the number of non-empty**subarrays** with the leftmost element of the subarray not larger than other elements in the subarray_.

A **subarray** is a **contiguous** part of an array.

Example 1:

Input: nums = [1,4,2,5,3] **Output:** 11 **Explanation:** There are 11 valid subarrays: [1],[4],[2],[5],[3],[1,4],[2,5],[1,4,2],[2,5,3],[1,4,2,5],[1,4,2,5,3].

Example 2:

Input: nums = [3,2,1] **Output:** 3 **Explanation:** The 3 valid subarrays are: [3],[2],[1].

Example 3:

Input: nums = [2,2,2] **Output:** 6 **Explanation:** There are 6 valid subarrays: [2],[2],[2],[2,2],[2,2],[2,2,2].

Constraints:

* `1 <= nums.length <= 5 * 104` * `0 <= nums[i] <= 105`

Code Snippets

C++:

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

}
};
```

Java:

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

}
}
```

Python3:

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