Problem 3542: Minimum Operations to Convert All Elements to Zero

Problem Information

Difficulty: Medium
Acceptance Rate: 52.76%
Paid Only: No
Tags: Array, Hash Table, Stack, Greedy, Monotonic Stack

Problem Description

You are given an array `nums` of size `n`, consisting of **non-negative** integers. Your task is to apply some (possibly zero) operations on the array so that **all** elements become 0.

In one operation, you can select a subarray `[i, j]` (where `0 <= i <= j < n`) and set all occurrences of the **minimum** **non-negative** integer in that subarray to 0.

Return the **minimum** number of operations required to make all elements in the array 0.

Example 1:

Input: nums = [0,2]

Output: 1

Explanation:

* Select the subarray `[1,1]` (which is `[2]`), where the minimum non-negative integer is 2. Setting all occurrences of 2 to 0 results in `[0,0]`. * Thus, the minimum number of operations required is 1.

Example 2:

Input: nums = [3,1,2,1]

Output: 3

Explanation:

* Select subarray `[1,3]` (which is `[1,2,1]`), where the minimum non-negative integer is 1. Setting all occurrences of 1 to 0 results in `[3,0,2,0]`. * Select subarray `[2,2]` (which is `[2]`), where the minimum non-negative integer is 2. Setting all occurrences of 2 to 0 results in `[3,0,0,0]`. * Select subarray `[0,0]` (which is `[3]`), where the minimum non-negative integer is 3. Setting all occurrences of 3 to 0 results in `[0,0,0,0]`. * Thus, the minimum number of operations required is 3.

Example 3:

Input: nums = [1,2,1,2,1,2]

Output: 4

Explanation:

* Select subarray `[0,5]` (which is `[1,2,1,2,1,2]`), where the minimum non-negative integer is 1. Setting all occurrences of 1 to 0 results in `[0,2,0,2,0,2]`. * Select subarray `[1,1]` (which is `[2]`), where the minimum non-negative integer is 2. Setting all occurrences of 2 to 0 results in `[0,0,0,2,0,2]`. * Select subarray `[3,3]` (which is `[2]`), where the minimum non-negative integer is 2. Setting all occurrences of 2 to 0 results in `[0,0,0,0,0,2]`. * Select subarray `[5,5]` (which is `[2]`), where the minimum non-negative integer is 2. Setting all occurrences of 2 to 0 results in `[0,0,0,0,0,0]`. * Thus, the minimum number of operations required is 4.

Constraints:

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

Code Snippets

C++:

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

```
}
};
```

Java:

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

}
}
```

Python3:

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