

Problem 2963: Count the Number of Good Partitions

Problem Information

Difficulty: Hard

Acceptance Rate: 48.33%

Paid Only: No

Tags: Array, Hash Table, Math, Combinatorics

Problem Description

You are given a **0-indexed** array `nums` consisting of **positive** integers.

A partition of an array into one or more **contiguous** subarrays is called **good** if no two subarrays contain the same number.

Return _the**total number** of good partitions of _`nums`_.

Since the answer may be large, return it **modulo** `10⁹ + 7`.

Example 1:

Input: nums = [1,2,3,4] **Output:** 8 **Explanation:** The 8 possible good partitions are: ([1], [2], [3], [4]), ([1], [2], [3,4]), ([1], [2,3], [4]), ([1], [2,3,4]), ([1,2], [3], [4]), ([1,2], [3,4]), ([1,2,3], [4]), and ([1,2,3,4]).

Example 2:

Input: nums = [1,1,1,1] **Output:** 1 **Explanation:** The only possible good partition is: ([1,1,1,1]).

Example 3:

Input: nums = [1,2,1,3] **Output:** 2 **Explanation:** The 2 possible good partitions are: ([1,2,1], [3]) and ([1,2,1,3]).

****Constraints:****

`* `1 <= nums.length <= 105` * `1 <= nums[i] <= 109``

Code Snippets

C++:

```
class Solution {
public:
    int numberOfGoodPartitions(vector<int>& nums) {
        ...
    }
};
```

Java:

```
class Solution {
    public int numberOfGoodPartitions(int[] nums) {
        ...
    }
}
```

Python3:

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