Problem 1955: Count Number of Special Subsequences

Problem Information

Difficulty: Hard
Acceptance Rate: 52.44%
Paid Only: No
Tags: Array, Dynamic Programming

Problem Description

A sequence is **special** if it consists of a **positive** number of `0`s, followed by a **positive** number of `1`s, then a **positive** number of `2`s.

* For example, `[0,1,2]` and `[0,0,1,1,1,2]` are special. * In contrast, `[2,1,0]`, `[1]`, and `[0,1,2,0]` are not special.

Given an array `nums` (consisting of **only** integers `0`, `1`, and `2`), return _the**number of different subsequences** that are special_. Since the answer may be very large, **return it modulo**`109 + 7`.

A **subsequence** of an array is a sequence that can be derived from the array by deleting some or no elements without changing the order of the remaining elements. Two subsequences are **different** if the **set of indices** chosen are different.

Example 1:

Input: nums = [0,1,2,2] **Output:** 3 **Explanation:** The special subsequences are bolded [**_0_** ,**_1_** ,**_2_** ,2], [**_0_** ,**_1_** ,2,**_2_**], and [**_0_** ,**_1_** ,**_2_** ,**_2_**].

Example 2:

Input: nums = [2,2,0,0] **Output:** 0 **Explanation:** There are no special subsequences in [2,2,0,0].

Example 3:

Input: nums = [0,1,2,0,1,2] **Output:** 7 **Explanation:** The special subsequences are bolded: - [**_0_** ,**_1_** ,**_2_** ,0,1,2] - [**_0_** ,**_1_** ,2,0,1,**_2_**] - [**_0_** ,**_1_** ,**_2_** ,0,1,**_2_**] - [**_0_** ,**_1_** ,2,0,**_1_** ,**_2_**] - [**_0_** ,1,2,**_0_** ,**_1_** ,**_2_**] - [**_0_** ,1,2,0,**_1_** ,**_2_**] - [0,1,2,**_0_** ,**_1_** ,**_2_**]

Constraints:

* `1 <= nums.length <= 105` * `0 <= nums[i] <= 2`

Code Snippets

C++:

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

}
};
```

Java:

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

}
}
```

Python3:

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