

Problem 3351: Sum of Good Subsequences

Problem Information

Difficulty: Hard

Acceptance Rate: 30.45%

Paid Only: No

Tags: Array, Hash Table, Dynamic Programming

Problem Description

You are given an integer array `nums`. A **good** subsequence is defined as a subsequence of `nums` where the absolute difference between any **two** consecutive elements in the subsequence is **exactly** 1.

Return the **sum** of all **possible** **good subsequences** of `nums`.

Since the answer may be very large, return it **modulo** $10^9 + 7$.

Note that a subsequence of size 1 is considered good by definition.

Example 1:

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

Output: 14

Explanation:

* Good subsequences are: `[1]`, `[2]`, `[1]`, `[1,2]`, `[2,1]`, `[1,2,1]`. * The sum of elements in these subsequences is 14.

Example 2:

Input: `nums = [3,4,5]`

****Output:**** 40

****Explanation:****

* Good subsequences are: `[3]`, `[4]`, `[5]`, `[3,4]`, `[4,5]`, `[3,4,5]`. * The sum of elements in these subsequences is 40.

****Constraints:****

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

Code Snippets

C++:

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

    }
};
```

Java:

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

    }
}
```

Python3:

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