Problem 1862: Sum of Floored Pairs

Problem Information

Difficulty: Hard
Acceptance Rate: 30.38%
Paid Only: No
Tags: Array, Math, Binary Search, Prefix Sum

Problem Description

Given an integer array `nums`, return the sum of `floor(nums[i] / nums[j])` for all pairs of indices `0 <= i, j < nums.length` in the array. Since the answer may be too large, return it **modulo** `109 + 7`.

The `floor()` function returns the integer part of the division.

Example 1:

Input: nums = [2,5,9] **Output:** 10 **Explanation:** floor(2 / 5) = floor(2 / 9) = floor(5 / 9) = 0 floor(2 / 2) = floor(5 / 5) = floor(9 / 9) = 1 floor(5 / 2) = 2 floor(9 / 2) = 4 floor(9 / 5) = 1 We calculate the floor of the division for every pair of indices in the array then sum them up.

Example 2:

Input: nums = [7,7,7,7,7,7,7] **Output:** 49

Constraints:

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

Code Snippets

C++:

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

}
};
```

Java:

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

}
}
```

Python3:

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