Problem 3153: Sum of Digit Differences of All Pairs

Problem Information

Difficulty: Medium
Acceptance Rate: 42.82%
Paid Only: No
Tags: Array, Hash Table, Math, Counting

Problem Description

You are given an array `nums` consisting of **positive** integers where all integers have the **same** number of digits.

The **digit difference** between two integers is the _count_ of different digits that are in the **same** position in the two integers.

Return the **sum** of the **digit differences** between **all** pairs of integers in `nums`.

Example 1:

Input: nums = [13,23,12]

Output: 4

Explanation: We have the following: \- The digit difference between **1** 3 and **2** 3 is 1. \- The digit difference between 1**3** and 1**2** is 1. \- The digit difference between **23** and **12** is 2. So the total sum of digit differences between all pairs of integers is `1 + 1 + 2 = 4`.

Example 2:

Input: nums = [10,10,10,10]

Output: 0

Explanation: All the integers in the array are the same. So the total sum of digit differences between all pairs of integers will be 0.

Constraints:

* `2 <= nums.length <= 105` * `1 <= nums[i] < 109` * All integers in `nums` have the same number of digits.

Code Snippets

C++:

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

}
};
```

Java:

```
class Solution {
public long sumDigitDifferences(int[] nums) {

}
}
```

Python3:

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