Problem 3712: Sum of Elements With Frequency Divisible by K

Problem Information

Difficulty: Easy
Acceptance Rate: 76.79%
Paid Only: No
Tags: Array, Hash Table, Counting

Problem Description

You are given an integer array `nums` and an integer `k`.

Return an integer denoting the **sum** of all elements in `nums` whose **frequency** is divisible by `k`, or 0 if there are no such elements.

Note: An element is included in the sum **exactly** as many times as it appears in the array if its total frequency is divisible by `k`.

Example 1:

Input: nums = [1,2,2,3,3,3,3,4], k = 2

Output: 16

Explanation:

* The number 1 appears once (odd frequency). * The number 2 appears twice (even frequency). * The number 3 appears four times (even frequency). * The number 4 appears once (odd frequency).

So, the total sum is `2 + 2 + 3 + 3 + 3 + 3 = 16`.

Example 2:

Input: nums = [1,2,3,4,5], k = 2

Output: 0

Explanation:

There are no elements that appear an even number of times, so the total sum is 0.

Example 3:

Input: nums = [4,4,4,1,2,3], k = 3

Output: 12

Explanation:

* The number 1 appears once. * The number 2 appears once. * The number 3 appears once.
* The number 4 appears three times.

So, the total sum is `4 + 4 + 4 = 12`.

Constraints:

* `1 <= nums.length <= 100` * `1 <= nums[i] <= 100` * `1 <= k <= 100`

Code Snippets

C++:

```
class Solution {
public:
int sumDivisibleByK(vector<int>& nums, int k) {

}
};
```

Java:

```
class Solution {
public int sumDivisibleByK(int[] nums, int k) {

}
}
```

Python3:

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