Problem 3512: Minimum Operations to Make Array Sum Divisible by K

Problem Information

Difficulty: Easy
Acceptance Rate: 88.44%
Paid Only: No
Tags: Array, Math

Problem Description

You are given an integer array `nums` and an integer `k`. You can perform the following operation any number of times:

* Select an index `i` and replace `nums[i]` with `nums[i] - 1`.

Return the **minimum** number of operations required to make the sum of the array divisible by `k`.

Example 1:

Input: nums = [3,9,7], k = 5

Output: 4

Explanation:

* Perform 4 operations on `nums[1] = 9`. Now, `nums = [3, 5, 7]`. * The sum is 15, which is divisible by 5.

Example 2:

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

Output: 0

Explanation:

* The sum is 8, which is already divisible by 4. Hence, no operations are needed.

Example 3:

Input: nums = [3,2], k = 6

Output: 5

Explanation:

* Perform 3 operations on `nums[0] = 3` and 2 operations on `nums[1] = 2`. Now, `nums = [0, 0]`. * The sum is 0, which is divisible by 6.

Constraints:

* `1 <= nums.length <= 1000` * `1 <= nums[i] <= 1000` * `1 <= k <= 100`

Code Snippets

C++:

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

}
};
```

Java:

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

}
}
```

Python3:

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