# 题目

给你一个正整数数组 nums 。

元素和是 nums 中的所有元素相加求和。

数字和是 nums 中每一个元素的每一数位（重复数位需多次求和）相加求和。

返回元素和与数字和的绝对差。

注意：两个整数 x 和 y 的绝对差定义为 |x - y| 。

示例 1：

输入：nums = [1,15,6,3]

输出：9

解释：

nums 的元素和是 1 + 15 + 6 + 3 = 25 。

nums 的数字和是 1 + 1 + 5 + 6 + 3 = 16 。

元素和与数字和的绝对差是 |25 - 16| = 9 。

示例 2：

输入：nums = [1,2,3,4]

输出：0

解释：

nums 的元素和是 1 + 2 + 3 + 4 = 10 。

nums 的数字和是 1 + 2 + 3 + 4 = 10 。

元素和与数字和的绝对差是 |10 - 10| = 0 。

提示：

1 <= nums.length <= 2000

1 <= nums[i] <= 2000

# 分析

## 方法一：模拟

思路：

代码：

class Solution {

public:

int differenceOfSum(vector<int>& nums) {

int sum = 0, sum1 = 0, ans = 0;

for (auto &i : nums) {

sum += i;

while(i) {

sum1 += i % 10;

i /= 10;

}

ans = sum - sum1;

}

return abs(ans);

}

};