

16 最接近的三数之和

题目描述

给定一个包括 n 个整数的数组 `nums` 和一个目标值 `target`。找出 `nums` 中的三个整数，使得它们的和与 `target` 最接近。返回这三个数的和。假定每组输入只存在唯一答案。

例如，给定数组 `nums = [-1, 2, 1, -4]`，和 `target = 1`。

与 `target` 最接近的三个数的和为 `2`。 ($-1 + 2 + 1 = 2$)。

代码

```
class Solution:
    def threeSumClosest(self, nums: List[int], target: int) -> int:
        closest = float('inf')
        nums.sort()

        for i in range(len(nums) - 2):
            j = i + 1
            k = len(nums) - 1

            while j < k:
                triple = nums[i] + nums[j] + nums[k]
                if triple == target:
                    return target
                if abs(triple - target) < abs(closest - target):
                    closest = triple

                if triple - target > 0:
                    k -= 1
                else:
                    j += 1

        return closest
```

成功 [显示详情](#) >

执行用时: **180 ms**, 在3Sum Closest的Python3提交中击败了**47.02%** 的用户

内存消耗: **12.8 MB**, 在3Sum Closest的Python3提交中击败了**98.95%** 的用户

进行下一个挑战:

较小的三数之和

炫耀一下:     

提交时间	状态	执行用时	内存消耗	语言
几秒前	通过	180 ms	12.8 MB	python3
几秒前	通过	180 ms	13.1 MB	python3
几秒前	通过	176 ms	13.3 MB	python3