

15 三数之和

题目描述

给定一个包含 n 个整数的数组 `nums`，判断 `nums` 中是否存在三个元素 a, b, c ，使得 $a + b + c = 0$ ？找出所有满足条件且不重复的三元组。

注意：答案中不可以包含重复的三元组。

例如，给定数组 `nums = [-1, 0, 1, 2, -1, -4]`，

满足要求的三元组集合为：

```
[
  [-1, 0, 1],
  [-1, -1, 2]
]
```

代码

```
class Solution:
    def threeSum(self, nums: List[int]) -> List[List[int]]:
        results = []
        nums.sort()

        i = 0
        while i < len(nums):
            j = i + 1
            k = len(nums) - 1

            while j < k:
                triple_sum = nums[i] + nums[j] + nums[k]

                if triple_sum == 0:
                    results.append([nums[i], nums[j], nums[k]])
                    k -= 1
                    while k > j and nums[k] == nums[k + 1]:
                        k -= 1
                    j += 1
                    while j < k and nums[j] == nums[j - 1]:
                        j += 1

                elif triple_sum > 0:
                    k -= 1
```

```
        while k > j and nums[k] == nums[k + 1]:
            k -= 1
        else:
            j += 1
            while j < k and nums[j] == nums[j - 1]:
                j += 1
            i += 1
        while i < len(nums) - 2 and nums[i] == nums[i - 1]:
            i += 1

    return results
```

成功 [显示详情](#) >

执行用时：1688 ms, 在3Sum的Python3提交中击败了35.84% 的用户

内存消耗：16.6 MB, 在3Sum的Python3提交中击败了90.13% 的用户

进行下一个挑战：

最接近的三数之和

四数之和

较小的三数之和

炫耀一下：[!\[\]\(3211b5d1d968fc1665909b34f9f16010_img.jpg\)](#) [!\[\]\(d47ad152ec3d86a04ad64c8049e1f17f_img.jpg\)](#) [!\[\]\(6b7fbb0b7bdb78cadf73d50851a4dfb1_img.jpg\)](#) [!\[\]\(cd0f39e2b8d76d7e84d5eed1ed02b2df_img.jpg\)](#) [!\[\]\(11e014e490252374320bfedb9a33c896_img.jpg\)](#)

提交时间	状态	执行用时	内存消耗	语言
几秒前	通过	1688 ms	16.6 MB	python3
几秒前	通过	2576 ms	16.7 MB	python3
几秒前	通过	2316 ms	16.8 MB	python3
2 分钟前	解答错误	N/A	N/A	python3