

[Explore](#)[Problems](#)[Interview](#)

New

[Contest](#)[Discuss](#)[Store](#)[Premium](#)[Description](#)[Solution](#)[Discuss \(999+\)](#)[Submissions](#)[C++](#)

15. 3Sum

Medium[19001](#)[1826](#)[Add to List](#)[Share](#)

Given an integer array `nums`, return all the triplets `[nums[i], nums[j], nums[k]]` such that `i != j`, `i != k`, and `j != k`, and `nums[i] + nums[j] + nums[k] == 0`.

Notice that the solution set must not contain duplicate triplets.

Example 1:

Input: `nums = [-1,0,1,2,-1,-4]`

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

Example 2:

Input: `nums = []`

Output: `[]`

Example 3:

Input: `nums = [0]`

Output: `[]`

Constraints:

- `0 <= nums.length <= 3000`
- `-105 <= nums[i] <= 105`

Accepted 2,054,574

Submissions 6,528,652

Seen this question in a real interview before?

Companies

Related Topics

Similar Questions

```
1 class Solution {
2 public:
3     vector<vector<int>>>
4     threeSum(vector<int>& nums) {
5         sort(nums.begin(),
6              nums.end());
7         int n = nums.size();
8         for (int i = 0; i < n; i++) {
9             if (i > 0 && nums[i] ==
10                nums[i-1]) continue;
11             int j = i + 1, k = n - 1;
12             while (j < k) {
13                 if (nums[i] + nums[j] +
14                    nums[k] == 0) {
15                     ans.push_back({
16                         nums[i],
17                         nums[j],
18                         nums[k]});
19                     while (j < k &&
20                          nums[j] ==
21                          nums[j+1]) j++;
22                     while (j < k &&
23                          nums[k] ==
24                          nums[k-1]) k--;
25                     j++; k--;
26                 } else if (nums[i] +
27                    nums[j] < -nums[k])
28                     j++;
29                 else if (nums[i] +
30                    nums[k] < -nums[j])
31                     k--;
32             }
33         }
34         return ans;
35     }
36 }
```

Your previous code was

[Problems](#)[Pick One](#)[Prev](#)

15/2324

[Next](#)[Contribute](#)[Run Code](#)