

Array - LeetCode

3Sum Closest - LeetCode

+

leetcode.com/problems/3sum-closest/?envType=problem-list-v2&envId=array

Array < > Run Submit

Description Editorial Solutions Submissions

All Solutions

```
def threeSumClosest(nums, target):
    closest_sum = float('inf')
    for i in range(len(nums) - 2):
        for j in range(i + 1, len(nums) - 1):
            for k in range(j + 1, len(nums)):
                current_sum = nums[i] + nums[j] + nums[k]
                if abs(current_sum - target) < abs(closest_sum - target):
                    closest_sum = current_sum
    return closest_sum
```

Approach 2: Brute Force

1. Iterate through all possible triplets in the array.
2. Calculate the sum for each triplet.
3. Track the closest sum to the target.

Python Solution (Approach 2):

```
class Solution(object):
    def threeSumClosest(self, nums, target):
        """
        :type nums: List[int]
        :type target: int
        :rtype: int
        """
        closest_sum = float('inf')
        for i in range(len(nums) - 2):
            for j in range(i + 1, len(nums) - 1):
                for k in range(j + 1, len(nums)):
                    current_sum = nums[i] + nums[j] + nums[k]
                    if abs(current_sum - target) < abs(closest_sum - target):
                        closest_sum = current_sum
        return closest_sum
```

Code

```
14         ++left;
15     } else if (current_sum > target) {
16         --right;
17     } else {
18         return current_sum;
19     }
20 }
21
22
23     return closest_sum;
24 }
25
26
```

Saved:Ln 26, Col 3

Testcase Test Result

Accepted Runtime: 0 ms

• Case 1 • Case 2

Input

nums =
[-1, 2, 1, -4]