16. 3Sum Closest

Description

Given an integer array `nums` of length `n` and an integer `target`, find three integers in `nums` such that the sum is closest to `target`.

Return *the sum of the three integers*.

You may assume that each input would have exactly one solution.

Example 1:

```
Input: nums = [-1,2,1,-4], target = 1
Output: 2
Explanation: The sum that is closest to the target is 2. (-1 + 2 + 1 = 2).
```

Example 2:

```
Input: nums = [0,0,0], target = 1
Output: 0
Explanation: The sum that is closest to the target is 0. (0 + 0 + 0 = 0).
```

Constraints:

- `3 <= nums.length <= 500`
- `-1000 <= nums[i] <= 1000`
- `-10^4 <= target <= 10^4`

Solution 1: Sorting + Two Pointers

We sort the array first, then traverse the array. For each element $nums[i]$, we use pointers j and k to point to $i + 1$ and $n - 1$ respectively, calculate the sum of the three numbers. If the sum of the three numbers equals $target$, we directly return $target$. Otherwise, we update the answer based on the difference from $target$. If the sum of the three numbers is greater than $target$, we move k one place to the left, otherwise, we move j one place to the right.

The time complexity is $O(n^2)$, and the space complexity is $O(\log n)$. Here, n is the length of the array.