**Problem Name:**

**Topics:**

**Companies:**

**Level:** Easy

**Language:** C++

**Problem Statement**:

**Input Format:**

**Output Format:**

**Constraints:**

**Examples:**

**Brute force Solution:**

# Explanation:

**Code:**

**Time Complexity**:

**Space Complexity:**

**Optimized Solution:**

**Explanation:**

Algorithm: Double Pointer

The requirements of the problem are very detailed. It is known that the array is ordered, and there must be a unique solution. What needs to be asked is a pair of values, then we can consider using double pointers to solve it.

Use a pair of double pointers I, j For target, if target<0target<0 For convenience, consider target\*-1target∗−1, because num[i]-num[j] = -(num[j]-num[i])num[i]-num [j]=−(num[j]−num[i])

When i==j, j++, guarantee num[j]>num[i]num[j]>num[i]

When j < n && nums[j] - nums[i] < target, it means that the current value of num[j]num[j] is not large enough, and the pointer needs to be moved to the right

When j < n && nums[j] - nums[i] == target, it means that a matching value is found, and the answer can be returned directly

The current situation is judged ii The pointer moves right to find the answer

**Code:**

**Time Complexity**: O(n)

**Space Complexity:** O(1)