Given a **0-indexed** integer array nums, return *the number of****distinct****quadruplets* (a, b, c, d) *such that:*

* nums[a] + nums[b] + nums[c] == nums[d], and
* a < b < c < d

**Example 1:**

**Input:** nums = [1,2,3,6]

**Output:** 1

**Explanation:** The only quadruplet that satisfies the requirement is (0, 1, 2, 3) because 1 + 2 + 3 == 6.

**Example 2:**

**Input:** nums = [3,3,6,4,5]

**Output:** 0

**Explanation:** There are no such quadruplets in [3,3,6,4,5].

**Example 3:**

**Input:** nums = [1,1,1,3,5]

**Output:** 4

**Explanation:** The 4 quadruplets that satisfy the requirement are:

- (0, 1, 2, 3): 1 + 1 + 1 == 3

- (0, 1, 3, 4): 1 + 1 + 3 == 5

- (0, 2, 3, 4): 1 + 1 + 3 == 5

- (1, 2, 3, 4): 1 + 1 + 3 == 5

**Constraints:**

* 4 <= nums.length <= 50
* 1 <= nums[i] <= 100