Problem 259: 3Sum Smaller

Problem Information

Difficulty: Medium
Acceptance Rate: 51.24%
Paid Only: Yes
Tags: Array, Two Pointers, Binary Search, Sorting

Problem Description

Given an array of `n` integers `nums` and an integer `target`, find the number of index triplets `i`, `j`, `k` with `0 <= i < j < k < n` that satisfy the condition `nums[i] + nums[j] + nums[k] < target`.

Example 1:

Input: nums = [-2,0,1,3], target = 2 **Output:** 2 **Explanation:** Because there are two triplets which sums are less than 2: [-2,0,1] [-2,0,3]

Example 2:

Input: nums = [], target = 0 **Output:** 0

Example 3:

Input: nums = [0], target = 0 **Output:** 0

Constraints:

* `n == nums.length` * `0 <= n <= 3500` * `-100 <= nums[i] <= 100` * `-100 <= target <= 100`

Code Snippets

C++:

```
class Solution {
public:
int threeSumSmaller(vector<int>& nums, int target) {

}
};
```

Java:

```
class Solution {
public int threeSumSmaller(int[] nums, int target) {

}
}
```

Python3:

```
class Solution:
def threeSumSmaller(self, nums: List[int], target: int) -> int:
```