Problem 2971: Find Polygon With the Largest Perimeter

Problem Information

Difficulty: Medium
Acceptance Rate: 65.49%
Paid Only: No
Tags: Array, Greedy, Sorting, Prefix Sum

Problem Description

You are given an array of **positive** integers `nums` of length `n`.

A **polygon** is a closed plane figure that has at least `3` sides. The **longest side** of a polygon is **smaller** than the sum of its other sides.

Conversely, if you have `k` (`k >= 3`) **positive** real numbers `a1`, `a2`, `a3`, ..., `ak` where `a1 <= a2 <= a3 <= ... <= ak` **and** `a1 + a2 + a3 + ... + ak-1 > ak`, then there **always** exists a polygon with `k` sides whose lengths are `a1`, `a2`, `a3`, ..., `ak`.

The **perimeter** of a polygon is the sum of lengths of its sides.

Return _the**largest** possible **perimeter** of a **polygon** whose sides can be formed from_ `nums`, _or_ `-1` _if it is not possible to create a polygon_.

Example 1:

Input: nums = [5,5,5] **Output:** 15 **Explanation:** The only possible polygon that can be made from nums has 3 sides: 5, 5, and 5. The perimeter is 5 + 5 + 5 = 15.

Example 2:

Input: nums = [1,12,1,2,5,50,3] **Output:** 12 **Explanation:** The polygon with the largest perimeter which can be made from nums has 5 sides: 1, 1, 2, 3, and 5. The perimeter is 1 + 1 + 2 + 3 + 5 = 12. We cannot have a polygon with either 12 or 50 as the longest side because it is not possible to include 2 or more smaller sides that have a greater sum than

either of them. It can be shown that the largest possible perimeter is 12.

Example 3:

Input: nums = [5,5,50] **Output:** -1 **Explanation:** There is no possible way to form a polygon from nums, as a polygon has at least 3 sides and 50 > 5 + 5.

Constraints:

* `3 <= n <= 105` * `1 <= nums[i] <= 109`

Code Snippets

C++:

```
class Solution {
public:
long long largestPerimeter(vector<int>& nums) {

}
};
```

Java:

```
class Solution {
public long largestPerimeter(int[] nums) {

}
}
```

Python3:

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