Problem 1283: Find the Smallest Divisor Given a Threshold

Problem Information

Difficulty: Medium
Acceptance Rate: 64.94%
Paid Only: No
Tags: Array, Binary Search

Problem Description

Given an array of integers `nums` and an integer `threshold`, we will choose a positive integer `divisor`, divide all the array by it, and sum the division's result. Find the **smallest** `divisor` such that the result mentioned above is less than or equal to `threshold`.

Each result of the division is rounded to the nearest integer greater than or equal to that element. (For example: `7/3 = 3` and `10/2 = 5`).

The test cases are generated so that there will be an answer.

Example 1:

Input: nums = [1,2,5,9], threshold = 6 **Output:** 5 **Explanation:** We can get a sum to 17 (1+2+5+9) if the divisor is 1. If the divisor is 4 we can get a sum of 7 (1+1+2+3) and if the divisor is 5 the sum will be 5 (1+1+1+2).

Example 2:

Input: nums = [44,22,33,11,1], threshold = 5 **Output:** 44

Constraints:

* `1 <= nums.length <= 5 * 104` * `1 <= nums[i] <= 106` * `nums.length <= threshold <= 106`

Code Snippets

C++:

```
class Solution {
public:
int smallestDivisor(vector<int>& nums, int threshold) {

}
};
```

Java:

```
class Solution {
public int smallestDivisor(int[] nums, int threshold) {

}
}
```

Python3:

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