Problem 1608: Special Array With X Elements Greater Than or Equal X

Problem Information

Difficulty: Easy
Acceptance Rate: 66.73%
Paid Only: No
Tags: Array, Binary Search, Sorting

Problem Description

You are given an array `nums` of non-negative integers. `nums` is considered **special** if there exists a number `x` such that there are **exactly** `x` numbers in `nums` that are **greater than or equal to** `x`.

Notice that `x` **does not** have to be an element in `nums`.

Return `x` _if the array is**special** , otherwise, return _`-1`. It can be proven that if `nums` is special, the value for `x` is **unique**.

Example 1:

Input: nums = [3,5] **Output:** 2 **Explanation:** There are 2 values (3 and 5) that are greater than or equal to 2.

Example 2:

Input: nums = [0,0] **Output:** -1 **Explanation:** No numbers fit the criteria for x. If x = 0, there should be 0 numbers >= x, but there are 2. If x = 1, there should be 1 number >= x, but there are 0. If x = 2, there should be 2 numbers >= x, but there are 0. x cannot be greater since there are only 2 numbers in nums.

Example 3:

Input: nums = [0,4,3,0,4] **Output:** 3 **Explanation:** There are 3 values that are greater than or equal to 3.

Constraints:

* `1 <= nums.length <= 100` * `0 <= nums[i] <= 1000`

Code Snippets

C++:

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

}
};
```

Java:

```
class Solution {
public int specialArray(int[] nums) {

}
}
```

Python3:

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