Problem 1365: How Many Numbers Are Smaller Than the Current Number

Problem Information

Difficulty: Easy
Acceptance Rate: 87.31%
Paid Only: No
Tags: Array, Hash Table, Sorting, Counting Sort

Problem Description

Given the array `nums`, for each `nums[i]` find out how many numbers in the array are smaller than it. That is, for each `nums[i]` you have to count the number of valid `j's` such that `j != i` **and** `nums[j] < nums[i]`.

Return the answer in an array.

Example 1:

Input: nums = [8,1,2,2,3] **Output:** [4,0,1,1,3] **Explanation:** For nums[0]=8 there exist four smaller numbers than it (1, 2, 2 and 3). For nums[1]=1 does not exist any smaller number than it. For nums[2]=2 there exist one smaller number than it (1). For nums[3]=2 there exist one smaller number than it (1). For nums[4]=3 there exist three smaller numbers than it (1, 2 and 2).

Example 2:

Input: nums = [6,5,4,8] **Output:** [2,1,0,3]

Example 3:

Input: nums = [7,7,7,7] **Output:** [0,0,0,0]

Constraints:

* `2 <= nums.length <= 500` * `0 <= nums[i] <= 100`

Code Snippets

C++:

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

}
};
```

Java:

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

}
}
```

Python3:

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