E - 1752. Check if Array Is Sorted and Rotated

Given an array nums, return true if the array was originally sorted in non-decreasing order, then rotated some number of positions (including zero). Otherwise, return false.
There may be duplicates in the original array.

Note: An array A rotated by x positions results in an array B of the same length such that A[i] == B[(i+x) % A.length], where % is the modulo operation.

Example 1:

Input: nums = [3,4,5,1,2]
Output: true

Explanation: [1,2,3,4,5] is the original sorted array.
You can rotate the array by x = 3 positions to begin on the the element of value 3: [3,4,5,1,2].

Example 2:

Input: nums = [2,1,3,4]
Output: false

Explanation: There is no sorted array once rotated that can make nums.

Example 3:

Input: nums = [1,2,3]
Output: true

Explanation: [1,2,3] is the original sorted array.
You can rotate the array by x = 0 positions (i.e. no rotation) to make nums.

Solution:

```
class Solution {
    public boolean check(int[] nums) {
       int count = 0;
        for(int i=1 ; i<nums.length ; i++){
            if(nums[i-1]>nums[i]){
                count++;
            }
        }
        if(nums[nums.length-1]>nums[0]){
            count++;
        }
        return count<=1;

    }
}
```