5672. Check if Array Is Sorted and Rotated

My Submissions (/contest/weekly-contest-227/problems/check-if-array-is-sorted-and-rotated/submissions/) | Back to Contest (/contest/weekly-contest-227/)

User Accepted:	0
User Tried:	0
Total Accepted:	0
Total Submissions:	0
Difficulty:	Easy

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.
```

Example 4:

```
Input: nums = [1,1,1]
Output: true
Explanation: [1,1,1] is the original sorted array.
You can rotate any number of positions to make nums.
```

Example 5:

```
Input: nums = [2,1]
Output: true
Explanation: [1,2] is the original sorted array.
You can rotate the array by x = 5 positions to begin on the element of value 2: [2,1].
```

Constraints:

- `1 <= nums.length <= 100`
- `1 <= nums[i] <= 100`

JavaScript

```
/**
 * @param {number[]} nums
 * @return {boolean}
 */
const check = (nums) => {
    if (isAscending(nums)) return true;
    let A = [...nums].sort((a, b) => a - b);
    let n = A.length;
    for (let i = 0; i< n; i++) {
        let l = A.slice(0, i+ 1);
        let r = A.slice(i + 1);
        let ro = r.concat(l);
        if (ok(ro, nums)) return true;
    }
    return false;
};

const ok = (a, b) => {
    let n = a.length;
    for (let i = 0; i < n; i++) {
        if (a[i] != b[i]) return false;
    }
    return true;
};

const isAscending = (arr) => {
    return arr.every((x, i) => {
        return i === 0 || x >= arr[i - 1];
    });
};
```

☐ **Custom Testcase** Use Example Testcases

Run Submit

Submission Result: Accepted (/submissions/detail/453004652/) More Details (/submissions/detail/453004652/)

Share your acceptance!

1

Help Center (/support) | Jobs (/jobs) | Bug Bounty (/bugbounty) | Students (/student) | Terms (/terms) | Privacy Policy (/privacy)

United States (/region)