## 31. Next Permutation

Medium 11917 3616 Add to List Share

A **permutation** of an array of integers is an arrangement of its members into a sequence or linear order.

- For example, for `arr = [1,2,3]`, the following are considered permutations of `arr`: `[1,2,3]`, `[1,3,2]`, `[3,1,2]`, `[2,3,1]`.

The **next permutation** of an array of integers is the next lexicographically greater permutation of its integer. More formally, if all the permutations of the array are sorted in one container according to their lexicographical order, then the **next permutation** of that array is the permutation that follows it in the sorted container. If such arrangement is not possible, the array must be rearranged as the lowest possible order (i.e., sorted in ascending order).

- For example, the next permutation of `arr = [1,2,3]` is `[1,3,2]`.
- Similarly, the next permutation of `arr = [2,3,1]` is `[3,1,2]`.
- While the next permutation of `arr = [3,2,1]` is `[1,2,3]` because `[3,2,1]` does not have a lexicographical larger rearrangement.

Given an array of integers `nums`, *find the next permutation of* `nums`.

The replacement must be **in place** and use only constant extra memory.

**Example 1:**

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

**Example 2:**

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

**Example 3:**

```
Input: nums = [1,1,5]
Output: [1,5,1]
```

**Constraints:**

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

Accepted 866,553 Submissions 2,359,374

Seen this question in a real interview before? Yes No

Companies

Related Topics

Similar Questions

```cpp
class Solution
{
public:
    void nextPermutation(vector<int>& nums) {
        int n = nums.size();

        if(n == 1)

 return;

        int idx1;

 for(int i=n-2; i>=0; i--){

 if(nums[i] < nums[i+1]){

 idx1 = i;

 break;
            }
        }


 if(idx1 < 0)

 reverse(nums.begin(), nums.end());
        else{

 int idx2 = 0;

 for(int i=n-1; i>=0; i--){

 if(nums[i] > nums[idx1]){

     idx2 = i;

     break;

}
            }


 swap(nums[idx2], nums[idx1]);


 sort(nums.begin()+idx1+1, nums.end());
        }
    }
};
```

Te... Run Cod... De...

Accepted Runtime: 3 ms

Your input [1,2,3

Output

Expected [1,3,2

Di

Could not connect to the reCAPTCHA service. Please check your internet connection and reload to get a reCAPTCHA challenge.