

## 31. Next Permutation

[Add to List ▾](#)[Question](#)[Editorial Solution](#)[My Submissions \(/problems/next-permutation/submissions/\)](#)

Total Accepted: **95485** Total Submissions: **338861** Difficulty: **Medium** Contributors: **Admin**

Implement next permutation, which rearranges numbers into the lexicographically next greater permutation of numbers.

If such arrangement is not possible, it must rearrange it as the lowest possible order (ie, sorted in ascending order).

The replacement must be in-place, do not allocate extra memory.

Here are some examples. Inputs are in the left-hand column and its corresponding outputs are in the right-hand column.

1,2,3 → 1,3,2

3,2,1 → 1,2,3

1,1,5 → 1,5,1

[Hide Company Tags](#)[Google \(/company/google/\)](#)[Hide Tags](#)[Array \(/tag/array/\)](#)[Hide Similar Problems](#)[\(M\) Permutations \(/problems/permutations/\)](#)[\(M\) Permutations II \(/problems/permutations-ii/\)](#)[\(M\) Permutation Sequence \(/problems/permutation-sequence/\)](#)[\(M\) Palindrome Permutation II \(/problems/palindrome-permutation-ii/\)](#)

Have you met this question in a real interview?

[Discuss \(https://discuss.leetcode.com/category/39\)](https://discuss.leetcode.com/category/39)[Top Solutions](#)[Pick One \(/problems/random-one-question/\)](#)[C++ ▾](#)

```
1 class Solution {
2 public:
3     void nextPermutation(vector<int>& nums) {
4         int j,l;
5         int n = nums.size();
6         if(n<=1) return;
7         j = n-2;
8         while(j>=0&&nums[j]>=nums[j+1]) j--;
9         if(j<0){
10             reverse(nums,0,n);
11             return;
12         }
13         l = n-1;
14         while(nums[j]>=nums[l]) l--;
15         swap(nums[j],nums[l]);
16         reverse(nums,j+1,n);
17     }
18
19     void reverse(vector<int>& nums, int begin, int end){
20         for(end--;begin<end;begin++,end--){
21             swap(nums[begin],nums[end]);
22         }
23     }
24 };
```

[Notes](#)

Custom Testcase ☐

Contribute Testcase

[Run Code](#)[Submit Solution](#)

[Frequently Asked Questions \(/faq/\)](#) | [Terms of Service \(/tos/\)](#)

[Privacy](#)

Copyright © 2017 LeetCode