

Description

Solution

Discuss (999+)

Submissions

88. Merge Sorted Array

Easy26084439Add to ListShare

Given two sorted integer arrays *nums1* and *nums2*, merge *nums2* into *nums1* as one sorted array.

Note:

- The number of elements initialized in *nums1* and *nums2* are *m* and *n* respectively.
- You may assume that *nums1* has enough space (size that is **equal** to *m* + *n*) to hold additional elements from *nums2*.

Example:

Input:

nums1 = [1,2,3,0,0,0], m = 3

nums2 = [2,5,6], n = 3

Output: [1,2,2,3,5,6]

Constraints:

- 10^9 <= nums1[i], nums2[i] <= 10^9
- nums1.length == m + n
- nums2.length == n

Accepted 667,266Submissions 1,679,565

Seen this question in a real interview before?

Yes

No

Companies

Related Topics

Similar Questions

Show Hint 1

Show Hint 2

Merge Sorted Array - LeetCode

i

C++

Autocomplete

i

{ }

↺

⚙

⌵

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

```
class Solution {
public:
    void merge(vector<int>& nums1, int m, vector<int>& nums2, int n) {
        int p1=m-1;
        int p2=n-1;
        int pi=(m+n)-1;
        while(p1>=0 && p2>=0 && pi>p1){
            if(nums1[p1]>nums2[p2]){
                nums1[pi]=nums1[p1];
                p1-=1;
                pi-=1;
            }
            else{
                nums1[pi]=nums2[p2];
                p2-=1;
                pi-=1;
            }
        }
        while(p1>=0 && pi>=0){
            nums1[pi]=nums1[p1];
            p1-=1;
            pi-=1;
        }
        while(p2>=0 && pi>=0){
            nums1[pi]=nums2[p2];
            p2-=1;
            pi-=1;
        }
    }
};
```