

# Merge Sorted Array

Try to solve the Merge Sorted Array problem.

We'll cover the following ^

- Statement
- Example
- Understand the problem
- Figure it out!
- Try it yourself

## Statement

Given two sorted integer arrays, *nums1* and *nums2*, and the number of data elements in each array, *m* and *n*, implement a function that merges the second array into the first one. You have to modify *nums1* in place.

**Note:** Assume that *nums1* has a size equal to  $m + n$ , meaning it has enough space to hold additional elements from *nums2*.

### Constraints:

- `nums1.length` =  $m + n$
- `nums2.length` =  $n$
- $0 \leq m, n \leq 200$
- $1 \leq m + n \leq 200$
- $-10^3 \leq \text{nums1}[i], \text{nums2}[j] \leq 10^3$

## Example

The inputs will be two integer arrays and two integers representing the number of data elements in each array.

### Sample example 1

**Input**

nums1

3

4

9

0

0

0

nums2

1

2

7

m = 3

n = 3

**Output**

1

2

3

4

7

9

1 of 2

The zeroes at the end of *nums1* represent uninitialized integers. This additional space will be used to merge it with *nums2*.

## Understand the problem

Let's take a moment to make sure you've correctly understood the problem. The quiz below helps you check if you're solving the correct problem:

### Merge Sorted Array

1

What is the output if the following arrays and values of *m* and *n* are given as input?

*nums1* = [1, 2, 7, 92, 0, 0, 0], *m* = 4

*nums2* = [3, 4, 5], *n* = 3

A) [1, 2, 3, 4, 5, 7, 92]

B) [1, 2, 7, 92, 3, 4, 5]

C) [3, 4, 5, 1, 2, 7, 92]

Submit Answer



Question 1 of 2  
0 attempted



Reset Quiz ↺

## Figure it out!

We have a game for you to play. Rearrange the logical building blocks to develop a clearer understanding of how to solve this problem.

Drag and drop the cards to rearrange them in the correct sequence.

Initialize two pointers, *p1* and *p2*, that point to the last data elements in *nums1* and *nums2*, respectively.



Initialize a pointer  $p$ , that points to the last element of  $nums1$ .

If the value at  $p1$  is greater than the value at  $p2$ , set the value at  $p$  equal to  $p1$  and decrement  $p1$  and  $p$  by 1.

Else, if the value at  $p2$  is greater than the value at  $p1$ , set the value at  $p$  equal to  $p2$  and decrement  $p2$  and  $p$  by 1.

Continue the traversal until  $nums2$  is merged with  $nums1$ .

Reset

Show Solution

Submit

Try it yourself

Implement your solution in `MergeSorted.java` in the following coding playground. We have provided a useful code template in the other file that you may build on to solve this problem.

Java

MergeSorted.java

```
1 public class MergeSorted{
2     // Tip: You may use some of the code templates provided
3
4
5
6     // Write your code here
7
8     int[] list = new int[]{};
9     return list;
10 }
11 }
```

Powered by AI

Submit

Test Cases

Results

Case 1

Case 2

Case 3

Input #1

[1,2,3,0,0,0]

Input #2

3

?

Tt

Input #3

[4,5,6]

Input #4

3

Merge Sorted Array

← Back

Next →

K-way Merge: Introdu...

Solution: Merge Sorte...

☒ Mark as  
Completed

