

Merge Without Extra Space

Friday, December 24, 2021 10:28 PM

Merge Without Extra Space

Hard Accuracy: 36.41% Submissions: 96947 Points: 8

Given two sorted arrays `arr1[]` of size `N` and `arr2[]` of size `M`. Each array is sorted in non-decreasing order. Merge the two arrays into one sorted array in non-decreasing order without using any extra space.

Example 1:

Input:

`N = 4, M = 5`

`arr1[] = {1, 3, 5, 7}`

`arr2[] = {0, 2, 6, 8, 9}`

Output: `0 1 2 3 5 6 7 8 9`

Explanation: Since you can't use any extra space, modify the given arrays to form `arr1[] = {0, 1, 2, 3}`
`arr2[] = {5, 6, 7, 8, 9}`

Example 2:

Input:

`N = 2, M = 3`

`arr1[] = {10, 12}`

`arr2[] = {5, 18, 20}`

Output: `5 10 12 18 20`

Explanation: Since you can't use any extra space, modify the given arrays to form `arr1[] = {5, 10}`
`arr2[] = {12, 18, 20}`

Your Task:

You don't need to read input or print anything. Complete the function **`merge()`** which takes the two arrays `arr1[]`, `arr2[]` and their sizes `n` and `m`, as input parameters. The function does not return anything. Use the given arrays to sort and merge `arr1[]` and `arr2[]` in-place.

Note: The generated output will print all the elements of `arr1[]` followed by all the elements of `arr2[]`.

Expected Time Complexity: $O((n+m)*\log(n+m))$

Expected Auxiliary Space: $O(1)$

Constraints:

$1 \leq N, M \leq 5*10^4$

$0 \leq arr1_i, arr2_i \leq 10^6$

View Bookmarked Problems

Company Tags



- ☐ Amazon
- ☐ Amdocs
- ☐ Brocade
- ☐ FactSet
- ☐ Goldman Sachs
- ☐ Juniper Networks
- ☐ Linkedin
- ☐ Microsoft
- ☐ Quikr
- ☐ Snapdeal
- ☐ Synopsys
- ☐ Visa
- ☐ Zoho

Topic Tags



- ☐ Arrays
- ☐ Mathematical
- ☐ Sorting

Related Courses



- ☐ Microsoft SDE Test Series
- ☐ Must Do Coding Questions

```
import java.util.*;
import java.io.*;
public class mergeSortedArray {
    public static void merge(int arr1[], int arr2[], int n, int m) {
        /*
         * arr1[] = {1, 3, 5, 7}
         * arr2[] = {0, 2, 6, 8, 9}
         * Output: 0 1 2 3 5 6 7 8 9
         */
        int i = n - 1;
        int j = 0;
        while (i >= 0 && j < m) {
            if (arr2[j] < arr1[i]) {
                int temp = arr1[i];
```

```

        arr1[i]=arr2[j];

        arr2[j]=temp;

        i--;

        j++;

    }
    else
        j++;

}
System.out.println(" i: "+i +" j : "+j);

Arrays.sort(arr1);

Arrays.sort(arr2);

System.out.println("  --1--- : " + Arrays.toString(arr1) + "--- 2---: "
+ Arrays.toString(arr2));

}
public static void main(String[] args) throws IOException {
    BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
    System.out.println(" enter the n1 and n2 ");
    String lineOne[] = br.readLine().trim().split("\\s");

    int n1 = Integer.parseInt(lineOne[0]);
    int n2 = Integer.parseInt(lineOne[1]);

    System.out.println("enter the element for 1st array");
    String[] lineTwo = br.readLine().trim().split("\\s");
    int arr[] = new int[n1];

    for (int i = 0; i < arr.length; i++) {
        arr[i] = Integer.parseInt(lineTwo[i]);
    }

    System.out.println("enter the element for 2nd array");
    String[] lineThree = br.readLine().trim().split("\\s");
    int arr2[] = new int[n2];

    for (int i = 0; i < arr2.length; i++) {
        arr2[i] = Integer.parseInt(lineThree[i]);
    }

    System.out.println(" 1 : " + Arrays.toString(arr) + " 2: " +
Arrays.toString(arr2));
    merge(arr, arr2, n1, n2);
}
}

```