

# Merge Without Extra Space

---

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[]`.

```
def merge(self, arr1, arr2, n, m):  
    # code here  
    #Approach 1  
    for i in range(n):  
        if arr1[i]>arr2[0]:  
            arr1[i],arr2[0] = arr2[0],arr1[i]
```

```
k = 1
temp = arr2[0]
while k<m and arr2[k]<temp:
    arr2[k-1] = arr2[k]
    k = k+1
arr2[k-1] = temp
```

*#Approach 2*

```
i = n-1
j = 0
while i>=0 and j<m:
    if arr1[i]>arr2[j]:
        arr1[i],arr2[j] = arr2[j],arr1[i]
    i = i-1
    j = j+1
arr1.sort()
arr2.sort()
```