[I2A] - Merge_sorted_arrays Function

Problem Statement

Implement the following function:

```
void MergeSortedArrays(int A[], int n, int B[], int m, int C[]);
```

Parameter Explanation:

- `A` and `B` are integer arrays with lengths `n` and `m`, respectively. Both arrays are already sorted in non-decreasing order.
- `C` is an array for storing the result, with a length of `n + m`.

Function Requirements:

- Merge the integers from arrays `A` and `B` into a single sorted sequence and store it in array `C`.

For example, if the function is called with array `A` being `1 3 5 7` and array `B` being `2 3 7`,
then after the function executes, the content of array `C` should be `1 2 3 3 5 7 7`.

Submission Instructions

This is a function implementation task. Your submitted code must include the following identifier:

```
/* probID: W0-C-MergeSortedArrays */
```

and must include the implementation of the `MergeSortedArrays` function (additional function declarations/implementations are allowed if necessary), but must not include a `main` function, as its presence will cause compilation errors.

When submitting, choose the language `C - function only`.

On the server side, the DomJudge system will compile and test your submitted code along with the following C code:

```
#include <stdio.h>

void MergeSortedArrays(int[], int, int[], int, int[]);

int A[200000], n;
int B[200000], m;
int C[400000];

int main()
{
    scanf("%d", &n);
    for (int i = 0; i < n; i++)
        scanf("%d", &A[i]);
    scanf("%d", &m);
    for (int i = 0; i < m; i++)
        scanf("%d", &B[i]);

    MergeSortedArrays(A, n, B, m, C);

    for (int i = 0; i < n + m; i++)
        printf("%d ", C[i]);
    printf("\n");

    return 0;
}
```

Example

Input:

```
4
1 3 5 7
3
2 3 7
```

Output:

1 2 3 3 5 7 7