Merge 2 sorted subarrays in C++

```
#include <iostream>
#include <vector>
using namespace std;

// Function to merge two sorted subarrays within array 'a'
vector<int> mergeTwoSortedSubArray(vector<int>& a, int s, int m, int e) {
    vector<int> temp(e - s + 1);
    int p1 = s;
    int p2 = m + 1;
    int p3 = 0;

    // Merge elements from two subarrays into temp array
    while (p1 <= m && p2 <= e) {
        if (a[p1] < a[p2]) {
            temp[p3] = a[p1];
            p3++;
            p1++;
        } else {
            temp[p3] = a[p2];
            p3++;
            p2++;
        }
    }

    // Copy remaining elements of the first subarray, if any
    while (p1 <= m) {
        temp[p3] = a[p1];
        p3++;
        p1++;
    }

    // Copy remaining elements of the second subarray, if any
    while (p2 <= e) {
        temp[p3] = a[p2];
        p3++;
        p2++;
    }

    // Copy sorted elements from temp back to original array 'a'
    for (int i = 0; i < temp.size(); i++) {
        a[s + i] = temp[i];
    }

    return a;
}

int main() {
    // Hard-coded input
    vector<int> A = {1, 3, 5, 7, 2, 4, 6, 8};
    int s = 0;
    int m = 3; // Middle index of the first sorted subarray
    int e = 7; // End index of the second sorted subarray

    // Merging the two sorted subarrays
```

using the input:

A = {1, 3, 5, 7, 2, 4, 6, 8}
s = 0, m = 3, e = 7

This means:

- First sorted subarray = A[0..3] = {1, 3, 5, 7}
- Second sorted subarray = A[4..7] = {2, 4, 6, 8}

🔄 Dry Run Table:

Step	p1	p2	temp[] (after step)	Comment
1	0	4	{1}	1 < 2, so copy 1 from left
2	1	4	{1, 2}	2 < 3, so copy 2 from right
3	1	5	{1, 2, 3}	3 < 4, so copy 3 from left
4	2	5	{1, 2, 3, 4}	4 < 5, so copy 4 from right
5	2	6	{1, 2, 3, 4, 5}	5 < 6, so copy 5 from left
6	3	6	{1, 2, 3, 4, 5, 6}	6 < 7, so copy 6 from right
7	3	7	{1, 2, 3, 4, 5, 6, 7}	7 < 8, so copy 7 from left
8	4	7	{1, 2, 3, 4, 5, 6, 7, 8}	only 8 left, copy from right

Now the merged array looks like:

A = {1, 2, 3, 4, 5, 6, 7, 8}

✅ Final Output:

Merged array: 1 2 3 4 5 6 7 8

```
    vector<int> result = mergeTwoSortedSubArray(A,
s, m, e);

    // Print the result
    cout << "Merged array: ";
    for (int num : result) {
        cout << num << " ";
    }
    cout << endl;

    return 0;
}
```

```
Merged array: 1 2 3 4 5 6 7 8
```