mergeArrays

Write a C function mergeArrays() that accepts two arrays ***a*** and ***b*** as parameters, which contain integer numbers **sorted in ascending order**. The parameters ***n1*** and ***n2*** indicate the size of the two arrays ***a*** and ***b*** respectively. The function merges the two array elements in ascending order and stores them into an array ***c*** which is passed in to the function as a parameter. For example, if the sizes for ***a*** and ***b*** are 3 (i.e., ***n1***) and 5 (i.e., ***n2***) respectively, ***a***[3]={5,9,19}, ***b***[5]={12,24,26,37,48}, then ***c*** will be {5,9,12,19,24,26,37,48} after function execution. The function will also return the size of the merged array ***c*** (i.e. 8) to the calling function. In the program, the size of the merged array will be limited to 80. There is no need to check user input errors in your program.

A sample program template is given below:

```
#include <stdio.h>
#define M 80
int mergeArrays(int a[M], int b[M], int c[M], int n1, int n2);
int main()
{
   int a[M],b[M],c[M],i,k=0,n1,n2;

   printf("Enter the size of array a: \n");
   scanf("%d", &n1);
   printf("Enter the size of array b: \n");
   scanf("%d", &n2);
   printf("Enter array a[%d]: \n", n1);
   for (i=0; i<n1; i++)
      scanf("%d",&a[i]);
   printf("Enter array b[%d]: \n", n2);
   for (i=0; i<n2; i++)
      scanf("%d",&b[i]);
   k=mergeArrays(a,b,c,n1,n2);
   printf("mergeArrays(): \n");
   for (i=0;i<k;i++)
      printf("%d ",c[i]);
   return 0;
}
int mergeArrays(int a[M], int b[M], int c[M], int n1, int n2)
{
   /* Write code here */
}
```

Some sample input and output sessions are given below:

(1) Test Case 1:

```
Enter the size of array a:
3
Enter the size of array b:
5
Enter array a[3]:
3 9 19
Enter array b[5]:
12 24 26 37 48
mergeArrays():
3 9 12 19 24 26 37 48
```

(2) Test Case 2:

```
Enter the size of array a:
3
Enter the size of array b:
5
Enter array a[3]:
1 2 3
Enter array b[5]:
1 2 3 4 5
mergeArrays():
1 1 2 2 3 3 4 5
```

(3) Test Case 3:

```
Enter the size of array a:
6
Enter the size of array b:
4
Enter array a[6]:
2 4 6 8 10 12
Enter array b[4]:
1 3 5 7
mergeArrays():
1 2 3 4 5 6 7 8 10 12
```

```c
#include <stdio.h>
#define M 80
int mergeArrays(int a[M], int b[M], int c[M], int n1, int n2);
int main()
{
  int a[M],b[M],c[M],i,k=0,n1,n2;

  printf("Enter the size of array a: \n");
  scanf("%d", &n1);
  printf("Enter the size of array b: \n");
  scanf("%d", &n2);
  printf("Enter array a[%d]: \n", n1);
  for (i=0; i<n1; i++)
    scanf("%d",&a[i]);
  printf("Enter array b[%d]: \n", n2);
  for (i=0; i<n2; i++)
    scanf("%d",&b[i]);
  k=mergeArrays(a,b,c,n1,n2);
  printf("mergeArrays(): \n");
  for (i=0;i<k;i++)
    printf("%d ",c[i]);
  return 0;
}
int mergeArrays(int a[M], int b[M], int c[M], int n1, int n2)
{
  int i=0,j=0,h=0;

  while(i<n1&&j<n2)        when both arrays not finished
  {
     if(a[i]>b[j])
     {
        c[h] = b[j];
        h++;
        j++;
     }
     else
     {
        c[h] = a[i];
        h++;
        i++;
     }
  }

  while(i<n1)
  {
     c[h] = a[i];
     i++;
     h++;
  }

  while(j<n2)
  {
     c[h] = b[j];
     j++;
     h++;
  }

return (n1+n2);
}
```

when one array finished and the other is not