Merge Sort By Divide and Conquer

```
#include <stdio.h>
#include <stdlib.h>

void merge(int arr[],int left,int mid,int right);
void merge_sort (int arr[], int left, int right)
{
   if(left >=right)
   {return;}
   int mid = left+(right-left)/2;
   merge_sort(arr,left,mid);
   merge_sort(arr,mid+1,right);
   merge(arr,left,mid,right);
}


void merge(int arr[],int left,int mid,int right)
{

   int index_a, index_l=0, index_r=0;

   int size_left = mid - left + 1;

   int size_right = right - mid;

   int L[size_left], R[size_right];

   for(int i=0;i<size_left;i++)
   {
      L[i]=arr[left+i];
   }

   for(int i=0;i<size_right;i++)
   {
      R[i]=arr[mid+1+i];
   }

   for(index_a = left; index_l < size_left && index_r < size_right; index_a++)
   {
      if( L[index_l]< R[index_r] )
      {
         arr[index_a] = L[index_l];
         index_l += 1;
      }

      else
      {
         arr[index_a] = R[index_r];
         index_r += 1;
      }

   }

      while (index_l < size_left)
   {
      arr[index_a] = L[index_l];
      index_l += 1;
      index_a += 1;
```

```
    }

     while (index_r < size_right)
    {
       arr[index_a] = R [index_r];
       index_r += 1;
       index_a += 1;
    }

}


int main()
{
    int i, n, *arr;

    printf("Enter the number of elements you want to enter: \t");

    scanf("%d",&n);
    arr=(int*) malloc(n*sizeof(int));

    for(int i=0;i<n;i++){scanf("%d",&arr[i]);}

    printf("After sorting:");
    merge_sort(arr,0,n);

    for(i=1; i<n+1; i++)
   {

    printf("%d\n",arr[i]);

   }
    return 0;

}
```