```
/*C Program to implement Merge Sort
Input  : 1. Size of the array
         2. Array elements
Output : Sorted array elements in ascending order
*/

#include<stdio.h>
#define MAX 50

void mergeSort(int arr[],int low,int mid,int high);
void partition(int arr[],int low,int high);

int main(){

    int merge[MAX],i,n;

    printf("\n Enter the size of the array: ");
    printf("\n");
    scanf("%d",&n);
    printf("\n");

    printf("\n Enter the array elements :\n ");
    for(i=0;i<n;i++){
        scanf("%d",&merge[i]);
    }
    printf("\n");
    partition(merge,0,n-1);

    printf(" Sorted elements are:\n\n ");
    for(i=0;i<n;i++){
        printf("%d ",merge[i]);
```

```
    }
     printf("\n\n ");
   return 0;
}

void partition(int arr[],int low,int high){
    int mid;
    if(low<high){
        mid=(low+high)/2;
        partition(arr,low,mid);
        partition(arr,mid+1,high);
        mergeSort(arr,low,mid,high);
    }
}

void mergeSort(int arr[],int low,int mid,int high){
    int i,m,k,l,temp[MAX];

    l=low;
    i=low;
    m=mid+1;

    while((l<=mid)&&(m<=high)){

        if(arr[l]<=arr[m]){
            temp[i]=arr[l];
            l++;
        }
        else{
            temp[i]=arr[m];
```

```
            m++;
        }
        i++;
    }

    if(l>mid){
        for(k=m;k<=high;k++){
            temp[i]=arr[k];
            i++;
        }
    }
    else{
        for(k=l;k<=mid;k++){
            temp[i]=arr[k];
            i++;
        }
    }

    for(k=low;k<=high;k++){
        arr[k]=temp[k];
    }
}
```

Sample Input and Output: