```
#include <stdio.h>

void merge(int arr[], int left, int right, int mid)
{
    int Lsize = mid - left + 1;
    int Rsize = right - mid;

    int leftArr[Lsize], rightArr[Rsize];

    for (int i = 0; i < Lsize; i++)
        leftArr[i] = arr[i + left];

    for (int i = 0; i < Rsize; i++)
        rightArr[i] = arr[i + mid + 1];

    int i = 0, j = 0, k = left;

    while (i < Lsize && j < Rsize)
    {
        if (leftArr[i] < rightArr[j])
            arr[k++] = leftArr[i++];
        else
            arr[k++] = rightArr[j++];
    }

    while (i < Lsize)
        arr[k++] = leftArr[i++];
```

```
    while (j < Rsize)
        arr[k++] = rightArr[j++];
}

void mergeSort(int arr[], int left, int right)
{
    if (left >= right) // base condition to exit recursion
        return;

    int mid = left + (right - left) / 2;
    mergeSort(arr, left, mid);
    mergeSort(arr, mid + 1, right);
    merge(arr, left, right, mid);
}

int main()
{
    int arr[100], size;
    printf("\nEnter size ");
    scanf("%d", &size);
    printf("\nEnter array\n");
    for (int i = 0; i < size; i++)
        scanf("%d", &arr[i]);

    mergeSort(arr, 0, size - 1);
```

```
    printf("Sorted array\n");

    for (int i = 0; i < size; i++)

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


    return 0;

}
```

```
Enter size 7

Enter array
12
90
45
43
26
99
44
Sorted array
12 26 43 44 45 90 99
```