



codechef.com/signup



prajna123



praj nabhandary15@gmail.com

☒ Female ☐ Male ☐ Other

undefined, Karnataka, India

☒ Student ☐ Professional ☐ Other

Alvas Institute of Engineering and



2020 ▼

C(gcc 6.3) ▼

☒ Send me newsletter & contest invitations.☒ I abide by [CodeChef's Code Of Conduct](#).

Register

[CodeChef is a non-profit competitive](#)[bout CodeChef](#) | [CEO's Corner](#) | [Contact Us](#)odeChef uses SPOJ © by [Sphere Research Labs](#)order to report copyright violations of any kind, send in an email to copyright@codechef.com**[odeChef](#)** - A Platform for Aspiring Programmers

CodeChef was created as a platform

Code, Compile & Run

IDE

C (gcc 6.3)

```
1 #include <stdio.h>
2
3 #define max 10
4
5 int a[10] = { 10, 14, 19, 26, 27, 31, 33, 35, 42, 44, 0 };
6 int b[10];
7
8 void merging(int low, int mid, int high) {
9     int i, j, k;
10
11     for(i = low, j = mid + 1, k = low; i <= mid && j <= high; i++)
12     {
13         if(a[i] <= a[j])
14             b[k] = a[i++];
15         else
16             b[k] = a[j++];
17     }
18
19     while(i <= mid)
20         b[k++] = a[i++];
21
22     while(j <= high)
23         b[k++] = a[j++];
24
25     for(i = low; i <= high; i++)
26         a[i] = b[i];
27 }
28
29 void sort(int low, int high)
```

23.3



Open File

Custom Input

Run





Status Successfully executed Date 2020-06-17 05:34:12 Time 0 sec Mem 9.424 kB




Output

```
List before sorting
10 14 19 26 27 31 33 35 42 44 0
List after sorting
0 10 14 19 26 27 31 33 35 42 44
```

Code, Compile & Run

C (gcc 6.3)    

```
31 void mergeSort(int a[], int low, int high)
32 {
33     if (low < high)
34     {
35         int mid = (low + high) / 2;
36         mergeSort(a, low, mid);
37         mergeSort(a, mid + 1, high);
38         merge(a, low, mid, high);
39     }
40 }
41
42 int main()
43 {
44     int a[] = {10, 14, 19, 26, 27, 31, 33, 35, 42, 44};
45     printf("List before sorting\n");
46     for (int i = 0; i < sizeof(a) / sizeof(int); i++)
47         printf("%d ", a[i]);
48     printf("\n");
49     mergeSort(a, 0, sizeof(a) / sizeof(int) - 1);
50     printf("List after sorting\n");
51     for (int i = 0; i < sizeof(a) / sizeof(int); i++)
52         printf("%d ", a[i]);
53     printf("\n");
54 }
```

333 

[Open File](#)☐ Custom Input[Run](#)

Status: Successfully executed Date: 2020-06-17 05:34:12 Time: 0 sec Mem: 9.424 kB



Output

```
List before sorting
10 14 19 26 27 31 33 35 42 44
List after sorting
0 10 14 19 26 27 31 33 35 42 44
```

C Program to implement merge Sort

Algorithm:

Step 1: Start

Step 2: MergeSort (arr[], 1, n), where 1 is the index of the first element & n is the index of the last element.

Step 3: If $n > 1$

Step 4: Find the middle index of the array divide it in two halves.
 $m = (1 + n) / 2$

Step 5: Call Merge sort for second half:
merge sort (array, 1, m)

Step 6: Call mergesort for second half:
mergesort (array, m+1, n)

Step 7: Recursively, merge the two halves in a sorted manner, so that only one sorted array is left:
merge (array, 1, m, n)

Step 8: Moving on with this article

Step 9: Stop

Flowchart:

