


```
1  #include <iostream>
2  using namespace std;
3
4  int partition(int arr[], int low, int high){
5      int pivot = arr[high];
6
7      int a = low-1;
8      for(int i =low;i<=high;i++){
9          if(arr[i]<pivot){
10             a++;
11             int temp;
12             temp = arr[i];
13             arr[i] = arr[a];
14             arr[a] = temp;
15         }
16     }
17     int temp = arr[a+1];
18     arr[a+1] = arr[high];
19     arr[high] = temp;
20     return (a+1);
21 }
22
23 void quickSort(int arr[],int low,int high)
24 {
25     if(low<high)
26     {
27         int pi=partition(arr,low,high);
28         quickSort(arr,low,pi-1);
29         quickSort(arr,pi+1,high);
30     }
31 }
32
33 void Merge(int arr[], int start, int mid, int end){
34     int a,b,c;
35     int m1 = mid - start +1;
36     int m2 = end - mid;
37     int arr1[m1], arr2[m2];
38     for(int i = 0;i< m1; i++){
39         arr1[i] = arr[start+i];
40     }
41
42     for(int j = 0;j< m2; j++){
43         arr2[j] = arr[mid+1+j];
44     }
45     a = 0;
46     b=0;
47     c=start;
48     while(a<m1 && b<m2){
49         if(arr1[a]<=arr2[b]){
50             arr[c] = arr1[a];
51             a++;
52         } else {
```

```

53         arr[c]=arr2[b];
54         b++;
55     }
56     c++;
57 }
58 while(a<m1){
59     arr[c] = arr1[a];
60     a++;
61     c++;
62 }
63 while(b<m2){
64     arr[c] = arr2[b];
65     b++;
66     c++;
67 }
68 }
69
70
71 void MergeSort(int arr[], int start, int end){
72     if (start < end){
73         int mid = (start + end)/2;
74         MergeSort(arr, start, mid);
75         MergeSort(arr, mid+1,end);
76         Merge(arr, start, mid, end);
77     }
78 }
79
80 void printArr(int arr[], int num){
81     for(int i = 0; i < num; i++){
82         printf("%d\n", arr[i]);
83     }
84 }
85
86 int main(){
87     int arr[] = {67, 77, 87, 21, 53, 36, 98, 22, 56, 78};
88     int num = sizeof(arr)/sizeof(arr[0]);
89     MergeSort(arr, 0, num-1);
90     quickSort(arr,0,num-1);
91     printArr(arr, num);
92     return 0;
93 }
94
95
96
97
98

```

```
● ishadp@pop-os:~/Documents/Code/Practical-DAA$ ./a.out
Merge Sort
21
22
36
53
56
67
77
78
87
98
```

```
● ishadp@pop-os:~/Documents/Code/Practical-DAA$ ./a.out
Quick Sort
21
22
36
53
56
67
77
78
87
98
```