

Find out the merge sort of the given array.

```
#include <stdio.h>
int c[50];
```

```
void merge (int arr[], int low, int mid,
            int high)
```

```
{ int i = low;
```

```
  int j = mid + 1;
```

```
  int k = 20;
```

```
  while (i <= mid & j <= high)
```

```
  { if (arr[i] <= arr[j])
```

```
    { c[k] = arr[i];
```

```
      i++;
```

```
      k++;
```

```
  } else
```

```
  { c[k] = arr[j];
```

```
    j++;
```

```
    k++;
```

```
  }
```

```
while (i <= mid)
```

```
{ c[k] = arr[i];
```

```
  i++;
```

```
  k++;
```

```
}
```

```
while (j <= high)
```

```
{ c[k] = arr[j];
```

```
  j++;
```

```
  k++;
```

```
}
```



```
for (k=0; k < (high - low - 1); k++)  
    arr[low+k] = c[k];
```

```
void mergesort (int arr[], int low, int high)  
{  
    if (low < high)  
    {  
        int mid = (low + high) / 2;
```

```
        mergesort (arr, low, mid);  
        mergesort (arr, mid + 1, high);  
        merge (arr, low, mid, high);  
    }  
}
```

```
int main ()
```

```
{  
    int size;
```

```
    printf ("Enter the number of elements : ");  
    scanf ("%d", &size);
```

```
    int arr[size];
```

```
    printf ("Enter the elements ");
```

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

```
        scanf ("%d", &arr[i]);
```

```
    printf ("Original array : ");
```

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

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

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

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

```
    printf ("Sorted array : ");
```

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

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

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


8 elements;

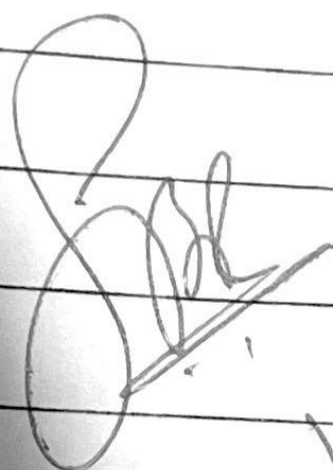
Output

Enter the number of elements: 4

Enter the elements: 8 4 6 9

Original array: 8 4 6 9

Sorted array: 4 6 8 9



13/7/23

Enter the number of elements: 4

Enter the elements:

8

4

6

9

Original array: 8 4 6 9

Sorted array: 4 6 8 9

Process returned 0 (0x0) execution time : 29.762 s

Press any key to continue.