```
//Merge Sort
//Om Dattatray Gavande
//Class:-SY CSE A
//Roll No:-CS2145


#include <stdio.h>
#include <stdlib.h>


void merge(int arr[], int l, int m, int r) {
  int i, j, k;
  int n1 = m - l + 1;
  int n2 = r - m;


  // Temporary arrays
  int L[n1], R[n2];


  // Copy data to temp arrays L[] and R[]
  for (i = 0; i < n1; i++)
      L[i] = arr[l + i];


  for (j = 0; j < n2; j++)
      R[j] = arr[m + 1 + j];


  i = 0; // Initial index of first subarray
  j = 0; // Initial index of second subarray
  k = l; // Initial index of merged subarray


  while (i < n1 && j < n2) {
      if (L[i] <= R[j]) {
          arr[k] = L[i];
          i++;
      }
      else {
          arr[k] = R[j];
          j++;
      }
      k++;
  }


  // Copy remaining elements of L[]
  while (i < n1) {
      arr[k] = L[i];
      i++;
      k++;
  }
  // Copy remaining elements of R[]
  while (j < n2) {
    arr[k] = R[j];
    j++;
    k++;
  }
}

void mergeSort(int arr[], int l, int r) {
  if (l < r) {
    int m = l + (r - l) / 2;

    mergeSort(arr, l, m);      // Left part
    mergeSort(arr, m + 1, r);  // Right part

    merge(arr, l, m, r);       // Merge both parts
  }
}

int main() {
  int arr[] = {38, 27, 43, 10};
  int arr_size = sizeof(arr) / sizeof(arr[0]);

  printf("Unsorted array: ");
  for (int i = 0; i < arr_size; i++)
    printf("%d ", arr[i]);

  mergeSort(arr, 0, arr_size - 1);

  printf("\nSorted array: ");
  for (int i = 0; i < arr_size; i++)
    printf("%d ", arr[i]);

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