/*merge sort*/

```
#include<stdio.h>

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

int main() {
   int n, i;
   printf("Enter the array size: ");
   scanf("%d", &n);
   int arr[n];
   printf("Enter the array elements:\n");
   for (i = 0; i < n; i++) {
      scanf("%d", &arr[i]);
   }
   merge_sort(arr, 0, n - 1);
   printf("\nAfter sorting the array: ");
   for (i = 0; i < n; i++) {
      printf("%d ", arr[i]);
   }
   return 0;
}

void merge_sort(int arr[], int low, int high) {
   if (low < high) {
      int mid = (low + high) / 2;
      merge_sort(arr, low, mid);
      merge_sort(arr, mid + 1, high);
      merge(arr, low, mid, high);
   }
}
```

```
void merge(int arr[], int low, int mid, int high) {
   int i, j, k, n1, n2;
   n1 = mid - low + 1;
   n2 = high - mid;
   int l1[n1], l2[n2];

   for (i = 0; i < n1; i++) {
      l1[i] = arr[low + i];
   }
   for (j = 0; j < n2; j++) {
      l2[j] = arr[mid + 1 + j];
   }

   i = j = 0;
   k = low;

   while (i < n1 && j < n2) {
      if (l1[i] <= l2[j]) {
         arr[k] = l1[i];
         i++;
      } else {
         arr[k] = l2[j];
         j++;
      }
      k++;
   }

   while (i < n1) {
      arr[k] = l1[i];
      i++;
```

```
        k++;
    }

    while (j < n2) {
        arr[k] = l2[j];
        j++;
        k++;
    }
}
```