main.c

```c
#include <stdio.h>
#include <time.h>
void merge_sort(int i, int j, int a[], int aux[]) {
    if (j <= i) {
        return;
    }
    int mid = (i + j) / 2;
    merge_sort(i, mid, a, aux);
    merge_sort(mid + 1, j, a, aux);
    int pointer_left = i;
    int pointer_right = mid + 1;
    int k;
    for (k = i; k <= j; k++) {
        if (pointer_left == mid + 1) {
            aux[k] = a[pointer_right];
            pointer_right++;
        } else if (pointer_right == j + 1) {
            aux[k] = a[pointer_left];
            pointer_left++;
        } else if (a[pointer_left] < a[pointer_right]) {
            aux[k] = a[pointer_left];
            pointer_left++;
        } else {
            aux[k] = a[pointer_right];
            pointer_right++;
        }
    }
    for (k = i; k <= j; k++) {
        a[k] = aux[k];
```

```
            aux[k] = a[pointer_left];
            pointer_left++;
        } else {
            aux[k] = a[pointer_right];
            pointer_right++;
        }
    }
    for (k = i; k <= j; k++) {
        a[k] = aux[k];
    }
}
int main() {
   int a[100], aux[100], n, i, d, swap;
   printf("Enter number of elements in the array:\n");
   scanf("%d", &n);
   printf("Enter %d numbers\n", n);
   for (i = 0; i < n; i++)
     scanf("%d", &a[i]);
     clock_t begin = clock();
   merge_sort(0, n - 1, a, aux);
   printf("Printing the sorted array:\n");
   for (i = 0; i < n; i++){
      printf("%d\t", a[i]);
 }
   clock_t end = clock();
    double time_spent = (double)(end - begin) / CLOCKS_PER_SEC;
    printf("\nExecution Time : %.10fseconds\n", time_spent);
    return 0;
```