

```

1  //MERGE SORT ALGORITHM IMPLEMENTATION
2
3  #include<stdio.h>
4
5  void Merge(int array[], int p, int q, int r)
6  {
7      int length1 = q-p+1;
8      int length2 = r-q;
9      int i,j;
10     int left_array[length1];
11     int right_array[length2];
12
13     for(i = 0; i < length1; i++)
14     {
15         left_array[i] = array[p+i];
16     }
17     for(j = 0; j < length2; j++)
18     {
19         right_array[j] = array[q+j+1];
20     }
21
22     left_array[length1] = 2147483647;
23     right_array[length2] = 2147483647;
24     i = j = 0;
25
26     for(int k = p; k <= r; k++)
27     {
28         if(left_array[i] <= right_array[j])
29         {
30             array[k] = left_array[i];
31             i++;
32         }
33         else
34         {
35             array[k] = right_array[j];
36             j++;
37         }
38     }
39 }
40
41 void MergeSort(int array[], int p, int r)
42 {
43     if(p<r)
44     {
45         int q = (p+r)/2;
46         MergeSort(array, p, q);
47         MergeSort(array, q+1, r);
48         Merge(array, p, q, r);
49     }
50 }
51
52 void display(int array[], int length)
53 {
54     int i;
55     for(i = 0; i < length; i++)
56     {
57         printf("%d ", array[i]);
58     }
59 }
60
61 int main()
62 {
63     int length, i;
64     printf("##### MERGE SORT ALGORITHM TESTING #####\n");
65     printf("\n=> Enter array size to create an array = ");
66     scanf("%d", &length);

```

```
67     int array[length];
68     printf("\n=> Enter %d array element:\n", length);
69
70     for(i = 0; i < length; i++)
71     {
72         scanf("%d", &array[i]);
73     }
74
75     printf("\n\n=> Before sort array elements are: ");
76     display(array, length);
77
78     int p = 0;
79     int r = length - 1;
80
81     MergeSort(array, p, r);
82
83     printf("\n\n=> After sort array elements are: ");
84     display(array, length);
85     printf("\n\n");
86
87     return 0;
88 }
```