

NAME – RAJDEEP JAISWAL

DATE – 05 Oct 2021

BRANCH – BTECH CSE

SEC = 13 A

UID -20BCS2761

SUB- DS LAB

Q - Write a program to sort an array of integers in descending order using merge sort?

SOLUTION -

ALGORITHM -

Step 1: [INITIALIZE] SET I = BEG, J = MID + 1, INDEX = 0

Step 2: Repeat while (I <= MID) AND (J <= END)

IF ARR[I] < ARR[J]

SET TEMP[INDEX] = ARR[I]

SET I = I + 1

ELSE

SET TEMP[INDEX] = ARR[J]

SET J = J + 1

[END OF IF]

SET INDEX = INDEX + 1

[END OF LOOP]

Step 3: [Copy the remaining elements of right sub-array, if any]

IF I > MID

Repeat while J <= END

SET TEMP[INDEX] = ARR[J]

SET INDEX = INDEX + 1, SET J = J + 1

[END OF LOOP]

[Copy the remaining elements of left sub-array, if any]

ELSE

Repeat while I <= MID

SET TEMP[INDEX] = ARR[I]

SET INDEX = INDEX + 1, SET I = I + 1

[END OF LOOP]

[END OF IF]

Step 4: [Copy the contents of TEMP back to ARR] SET K = 0

Step 5: Repeat while K < INDEX

SET ARR[K] = TEMP[K]

SET K = K + 1

[END OF LOOP]

Step 6: Exit

MERGE_SORT(ARR, BEG, END)

Step 1: IF BEG < END

SET MID = (BEG + END)/2

CALL MERGE_SORT (ARR, BEG, MID)

CALL MERGE_SORT (ARR, MID + 1, END)

MERGE (ARR, BEG, MID, END)

[END OF IF]

Step 2: END

CODE IN COMPILER-

```
1 #include<stdio.h>
2 void mergeSort(int[], int, int);
3 void merge(int[], int, int, int);
4 int main()
5 {
6     int a[10]= {101, 91, 7, 1012, 2, 4, 18, 98, 34, 22};
7     int i;
8     mergeSort(a,0,9);
9     printf("printing the sorted elements");
10    for(int i=10;i>=0;i--)
11    {
12        printf("\n%d\n",a[i]);
13    }
14 }
15 }
16 void mergeSort(int a[], int beg, int end)
17 {
18     int mid;
19     if(beg<end)
20     {
21         mid = (beg+end)/2;
22         mergeSort(a,beg,mid);
23         mergeSort(a,mid+1,end);
24         merge(a,beg,mid,end);
25     }
26 }
27 void merge(int a[], int beg, int mid, int end)
28 {
29     int i=beg,j=mid+1,k,index = beg;
30     int temp[10];
```

```
30     int temp[10];
31     while(i<=mid && j<=end)
32     {
33         if(a[i]<a[j])
34         {
35             temp[index] = a[i];
36             i = i+1;
37         }
38         else
39         {
40             temp[index] = a[j];
41             j = j+1;
42         }
43         index++;
44     }
45     if(i>mid)
46     {
47         while(j<=end)
48         {
49             temp[index] = a[j];
50             index++;
51             j++;
52         }
53     }
54     else
55     {
56         while(i<=mid)
57         {
58             temp[index] = a[i];
59             index++;
```

```
57         {
58             temp[index] = a[i];
59             index++;
60             i++;
61         }
62     }
63     k = beg;
64     while(k<index)
65     {
66         a[k]=temp[k];
67         k++;
68     }
69 }
```

CODE IN TEXT –

```
#include<stdio.h>

void mergeSort(int[],int,int);
void merge(int[],int,int,int);
int main()
{
    int a[10]= {101, 91, 7, 1012, 2, 4, 18, 98, 34, 22};
    int i;
    mergeSort(a,0,9);
    printf("printing the sorted elements");
    for(int i=10;i>=0;i--)

    {
        printf("\n%d\n",a[i]);
    }
}

void mergeSort(int a[], int beg, int end)
{
    int mid;
    if(beg<end)
    {
        mid = (beg+end)/2;
        mergeSort(a,beg,mid);
        mergeSort(a,mid+1,end);
        merge(a,beg,mid,end);
    }
}

void merge(int a[], int beg, int mid, int end)
{
    int i=beg,j=mid+1,k,index = beg;
    int temp[10];
    while(i<=mid && j<=end)
    {
        if(a[i]<a[j])
```

```
{
    temp[index] = a[i];
    i = i+1;
}
else
{
    temp[index] = a[j];
    j = j+1;
}
index++;
}
if(i>mid)
{
    while(j<=end)
    {
        temp[index] = a[j];
        index++;
        j++;
    }
}
else
{
    while(i<=mid)
    {
        temp[index] = a[i];
        index++;
        i++;
    }
}
k = beg;
while(k<index)
{
    a[k]=temp[k];
    k++;
}
}
```

OUTPUT –

```
HN cd "/Volumes/RAJ 2/CODE/" && gcc kam.c -o kam && "/Volumes/RAJ 2/CODE/"kam
rajdeepjaiswal@Rajdeeps-MacBook-Air CODE % cd "/Volumes/RAJ 2/CODE/" && gcc kam.c -o kam && "/Volumes/RAJ 2/CODE/"kam
printing the sorted elements
1012
101
98
91
34
22
18
7
4
rajdeepjaiswal@Rajdeeps-MacBook-Air CODE %
```