

Assignment 2

Thamshree Alg.
211047004

①

Using a sorting algorithm; sort the input elements.

Solⁿ Algorithm :-

MERGE-SORT (arr, beg, end)

if beg < end

 let mid = (beg + end) / 2

 MERGE-SORT (arr, beg, mid)

 MERGE (arr, beg, mid, end)

end of if

END MERGE-SORT

Program :-

void merge (int a[], int beg, int mid, int end)

{

 int i, j, k;

 int n1 = mid - beg + 1;

 int n2 = end - mid;

 int LeftArray[n1], RightArray[n2];

 / temporary arrays

// copy data to temp arrays

for (int i = 0; i < n1; i++)

LeftArray[i] = a[beg + i];

for (int j = 0; j < n2; j++)

RightArray[j] = a[mid + 1 + j];

i = 0; // initial index of first sub-array

j = 0; // initial index of second sub-array.

k = beg; // initial index of merged sub-array.

while (i < n1 && j < n2)

{
if (LeftArray[i] <= RightArray[j])

{
a[k] = LeftArray[i];

{
a[k] = LeftArray[i];

i++;

}

else

{
a[k] = RightArray[j];

j++;

}

②

```
k++;  
}  
while (i < n1)  
{  
    a[k] = LeftArray[i];  
    i++;  
    k++;  
}
```

```
while (j < n2)  
{  
    a[k] = RightArray[j];  
    j++;  
    k++;  
}
```

```
}  
  
void mergeSort (int a[], int beg, int end)
```

```
{  
    if (beg < end)
```

```
{  
    int mid = (beg + end) / 2;  
    mergeSort (a, beg, mid);  
    mergeSort (a, mid + 1, end);  
    merge (a, beg, mid, end);  
}
```

```
}  
}
```

// Function to print array.

void printArray(int a[], int n)

```
{  
    int i;  
    for (i=0; i<n; i++)  
        printf("%d", a[i]);  
    printf("\n");  
}
```

}

int main()

```
{  
    int a[] = {12, 31, 25, 8, 32, 17, 40, 42};  
    int n = sizeof(a) / sizeof(a[0]);  
    printf("Before sorting, the elements are :");  
    printArray(a, n);  
    mergeSort(a, 0, n-1);  
    printf("After sorting :");  
    printArray(a, n);  
    return 0;  
}
```

(3)

/* function to take user input

int a[100], i, j, k;

int n = sizeof(a) / sizeof(a[0]);

printf("Enter ¹⁰ elements to sort");

i = 0;

scanf("%d", &i)

for (j = 0; j < i; j++)

scanf("%d", &a[j]);

pf("The entered elements are")

for (j = 0; j < i; j++)

pf("%d", a[j]);

merge sort(a, 0, n - 1);

pf("After sorting:");

printArray(a, n);

return 0;

*/

3

Output :

Before sorting array elements are :

12 31 25 8 32 17 40 42

After sorting array elements are :

8 12 17 25 31 32 40 42

Sample o/p for user input array :

Enter the no of elements to sort : 5

Enter elements : 2 7 5 6 1

Sorted elements : 1 2 5 6 7