**CAP539: ALGORITHM DESIGN AND ANALYSIS - LABORATORY**

**CONTINUOUS ASSESSMENTS (C.A)-1**

**ST\_NAME : - EKHLAKH AHMAD**

**REG NO. : - 12209166**

**ROLL NO. : - 6**

**GROUP : - 1**

#include<iostream>

using namespace std;

void merge(int arr[], int left, int mid, int right){

    int a1 = mid - left + 1;

    int a2 = right - mid;

    int L[a1], R[a2];

    for(int i = 0; i < a1; i++)

        L[i] = arr[left + i];

    for (int j = 0; j < a2; j++)

        R[j] = arr[mid + 1 + j];

    int i = 0;

    int j = 0;

    int k = left;

    while(i < a1 && j < a2){

        if(L[i] <= R[j]){

            arr[k] = L[i];

            i++;

        }

        else{

            arr[k] = R[j];

            j++;

        }

        k++;

    }

    while (i < a1){

        arr[k] = L[i];

        i++;

        k++;

    }

    while (j < a2){

        arr[k] = R[j];

        j++;

        k++;

    }

}

void mergeSort(int arr[], int left, int right){

    if(left < right){

        int mid = left + (right - left) / 2;

    mergeSort(arr, left, mid);

    mergeSort(arr, mid + 1, right);

    merge(arr, left, mid, right);

    }

}

void printArray(int arr[], int size){

    for (int i = 0; i < size; i++)

        cout << arr[i] << " ";

    cout << endl;

}

int main(){

    int arr[] = {21, 1, 15, 3, 51, 16, 27, 20};

    int arr\_size = sizeof(arr) / sizeof(arr[0]);

    cout << "Given array is \n";

    printArray(arr , arr\_size);

    mergeSort(arr , 0, arr\_size - 1);

    cout << "Sorted array is \n";

    printArray(arr , arr\_size);

    return 0;

}

