**CSE 303**

**A2305219290 – NEEYATI SHIVIND JAUHAR**

**5CSE4Y**

**LAB 3 - 4TH AUGUST 2021**

MERGE SORT

#include<stdlib.h>

#include<stdio.h>

// func for merge

// l = index of first element (leftmost)

// r = index of last element (rightmost)

// if r=l , array sorted but if r>l , find m (middle)

// m = (l+r)/2 to divide array into two

// call margeSort for both halves separately ; continue step

// recursively till each array has only one element

void merge ( int arr[], int l , int m , int r)

{

int i , j , k;

int n1 = m-l+1;

int n2 = r - m;

//temp arrs

int L[n1] ;

int R[n2];

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

L[i] = arr[l + i ];

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

R[j] = arr[m + 1 + j ];

i = 0 ; j = 0 ; k = l;

while (i <n1 && j <n2){

if (L[i] <= R[j])

{

arr[k] = L[i];

i++;

}

else {

arr[k] = R[j];

j++;

}

k++;

}

while (i<n1){

arr[k] = L[i];

i++;

k++;

}

while (j < n2)

{

arr[k] = R[j];

j++;

k++;

}

}

//merge sort func

void mergeSort( int arr[] , int l , int r)

{

if (l<r){

int m = l + (r-l)/2;

mergeSort(arr , l , m);

mergeSort(arr, m + 1, r);

merge(arr , l , m , r);

}

}

// print array func

void print(int A[], int n)

{

int i ;

printf("Sorted array : \n");

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

printf("%d", A[i]);

printf("\n");

}

int main()

{

int n , i ;

int arr[100] ;

printf("enter number of elements: \n");

scanf("%d" , &n);

printf("enter values for %d elements: \n", n);

for ( i = 0 ; i <n ; i++){

scanf("%d", &arr[i]);

}

mergeSort(arr , 0 , n - 1);

print(arr , n );

return 0;

}

A screenshot of a computer

Description automatically generated