# Rajalakshmi Engineering College

Name: Adithya Varman
Email: 240801010@rajalakshmi.edu.in
Roll no: 240801010
Phone: 8122197670
Branch: REC
Department: I ECE FA
Batch: 2028
Degree: B.E - ECE

Scan to verify results

## NeoColab_REC_CS23231_DATA STRUCTURES

### REC_DS using C_Week 6_COD_Question 5

Attempt : 1
Total Mark : 10
Marks Obtained : 10

### Section 1 : Coding

1. Problem Statement

Jose has an array of N fractional values, represented as double-point numbers. He needs to sort these fractions in increasing order and seeks your help.

Write a program to help Jose sort the array using the merge sort algorithm.

***Input Format***

The first line of input consists of an integer N, representing the number of fractions to be sorted.

The second line consists of N double-point numbers, separated by spaces, representing the fractions array.

***Output Format***

The output prints N double-point numbers, sorted in increasing order, and rounded to three decimal places.

Refer to the sample output for formatting specifications.

***Sample Test Case***

Input: 4
0.123 0.543 0.321 0.789
Output: 0.123 0.321 0.543 0.789

***Answer***

```
#include <stdio.h>
#include <stdlib.h>

int compare(double a, double b){



    return a > b;

}

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

{


    int n1 = m - l + 1;
    int n2 = r - m;
    double left[n1], right[n2];

    for (int i = 0; i < n1; i++)
        left[i] = arr[l + i];
    for (int j = 0; j < n2; j++)
        right[j] = arr[m + 1 + j];

    int i = 0, j = 0, k = l;
    while (i < n1 && j < n2)
```

```
{
        if (!compare(left[i], right[j]))
{
            arr[k++] = left[i++];
} else
{
            arr[k++] = right[j++];
}
}
    while (i < n1)
{
        arr[k++] = left[i++];
}
    while (j < n2)
{
        arr[k++] = right[j++];
}
```

```
}
void mergeSort(double arr[], int l, int r)
{
   if (l < r)
{
      int m = (l + r) / 2;
      mergeSort(arr, l, m);
      mergeSort(arr, m + 1, r);
      merge(arr, l, m, r);
}
}
int main() {
   int n;
   scanf("%d", &n);
   double fractions[n];
   for (int i = 0; i < n; i++) {
      scanf("%lf", &fractions[i]);
   }
   mergeSort(fractions, 0, n - 1);
   for (int i = 0; i < n; i++) {
      printf("%.3f ", fractions[i]);
   }
   return 0;
}
```

***Status :*** Correct ***Marks : 10/10***