**Practical-4**

**AIM:** To implement merge sort in C programming.

**SOFTWARE REQUIRED:** Vs Code

**PSEUDO CODE:**

MergeSort(A, p, r):

if p > r

return

q = (p+r)/2

mergeSort(A, p, q)

mergeSort(A, q+1, r)

merge(A, p, q, r)

**CODE:**

#include <stdio.h>

#include <stdlib.h>

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

    int i, j, k;

    int n1 = mid - left + 1;

    int n2 = right - mid;

    int L[n1], R[n2];

    for (i = 0; i < n1; i++)

        L[i] = arr[left + i];

    for (j = 0; j < n2; j++)

        R[j] = arr[mid + 1 + j];

    i = 0;

    j = 0;

    k = left;

    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++;

    }

}

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);

    }

}

int main() {

    int n;

    printf("Name: Ananta Walli");

    printf("\nEnrollment No: A2305221322");

    printf("\nPlease enter the elements needed in array: ");

    scanf("%d", &n);

    int arr[n];

    printf("Please enter the elements of array:\n");

    for (int i = 0; i < n; i++) {

        scanf("%d", &arr[i]);

    }

    printf("The array will be: ");

    for (int i = 0; i < n; i++) {

        printf("%d ", arr[i]);

    }

    mergeSort(arr, 0, n - 1);

    printf("\nThe array after sorting will be: ");

    for (int i = 0; i < n; i++) {

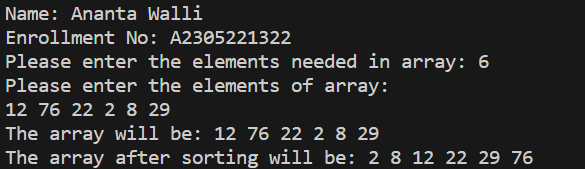
        printf("%d ", arr[i]);

    }

    return 0;

}

**OUTPUT:**

****

**TIME COMPLEXITY:** The time complexity should be: nlogn

**RESULT:** The above code implements the merge sort in C programming.