EXERCISE 17

Write a C program to arrange a series of numbers using Merge Sort

Aim:

To write a C program to sort a series of numbers using the Merge Sort algorithm.

Algorithm:

1. Divide the array into two halves.
2. Recursively sort both halves using Merge Sort.
3. Merge the two sorted halves into one sorted array.
4. Repeat until the entire array is sorted.

Program:

```
#include <stdio.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++];
```

```
        else
            arr[k++] = R[j++];
    }
    // Copy remaining elements
    while (i < n1)
        arr[k++] = L[i++];
    while (j < n2)
        arr[k++] = R[j++];
}
void mergeSort(int arr[], int left, int right) {
    if (left < right) {
        int mid = (left + right) / 2;
        mergeSort(arr, left, mid);
        mergeSort(arr, mid + 1, right);
        merge(arr, left, mid, right);
    }
}
void display(int arr[], int size) {
    printf("Sorted array:\n");
    for (int i = 0; i < size; i++)
        printf("%d ", arr[i]);
    printf("\n");
}
int main() {
    int arr[50], n;
    printf("Enter number of elements: ");
```

```
    scanf("%d", &n);
    printf("Enter %d elements:\n", n);
    for (int i = 0; i < n; i++)
        scanf("%d", &arr[i]);
    mergeSort(arr, 0, n - 1);
    display(arr, n);
    return 0;
}
```

Input and output:

```
Enter number of elements: 4
Enter 4 elements:
45 12 72 2
Sorted array:
2 12 45 72


=== Code Execution Successful ===
```

Result:

The series of numbers has been successfully sorted using the Merge Sort method.