Experiment 17: Merge Sort

Aim:
To write a C program to arrange a series of numbers using Merge Sort.

Algorithm:

1. Start the program.
2. If array has more than one element:
    - Divide it into two halves.
    - Recursively sort both halves.
    - Merge sorted halves.
3. Display the sorted array.
4. Stop.

Code:

```
#include <stdio.h>
void merge(int arr[], int l, int m, int r) {
   int n1 = m - l + 1, n2 = r - m;
   int L[n1], R[n2];
   for (int i = 0; i < n1; i++) L[i] = arr[l + i];
   for (int j = 0; j < n2; j++) R[j] = arr[m + 1 + j];
   int i = 0, j = 0, k = l;
   while (i < n1 && j < n2) {
      if (L[i] <= R[j]) arr[k++] = L[i++];
      else arr[k++] = R[j++];
   }
   while (i < n1) arr[k++] = L[i++];
   while (j < n2) arr[k++] = R[j++];
}
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);
    }
}
int main() {
    int arr[5] = {12, 11, 13, 5, 6}, n = 5;
    mergeSort(arr, 0, n - 1);
    printf("Sorted array: ");
    for (int i = 0; i < n; i++)
        printf("%d ", arr[i]);
    return 0;
}
```

Sample Output:

```
Sorted array: 5 6 11 12 13

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

Result:
The program successfully sorts numbers using Merge Sort.