Lab Assignment: Merge Sort

Problem Statement:

Write a program to implement the Merge Sort algorithm to sort an array of integers in ascending order. Demonstrate the divide-and-conquer approach used in Merge Sort and ensure your program outputs the sorted array.

Example:

Input: [38, 27, 43, 3, 9, 82, 10]

Output: [3, 9, 10, 27, 38, 43, 82]

Explanation:

Merge Sort works by recursively dividing the array into halves, sorting them individually, and then merging them into a single sorted array.

Logic: Divide and Conquer

Merge Sort follows the divide-and-conquer paradigm:

- Divide the array into two halves.

- Conquer the subarrays by sorting them recursively.

- Combine the sorted subarrays to form a single sorted array.

Time complexity:

- Best / Average / Worst: O(n log n)

Space complexity:

- O(n) for temporary arrays used in merging.

Function Signature (C Language):

void mergeSort(int arr[], int left, int right);

- arr[]: The array to be sorted.

- left: The starting index of the array.

- right: The ending index of the array.

You will also need a helper function:

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

Task:

- Implement the mergeSort and merge functions in C.

- The array should be sorted in ascending order using Merge Sort.

- Demonstrate the recursive division and merging steps clearly.