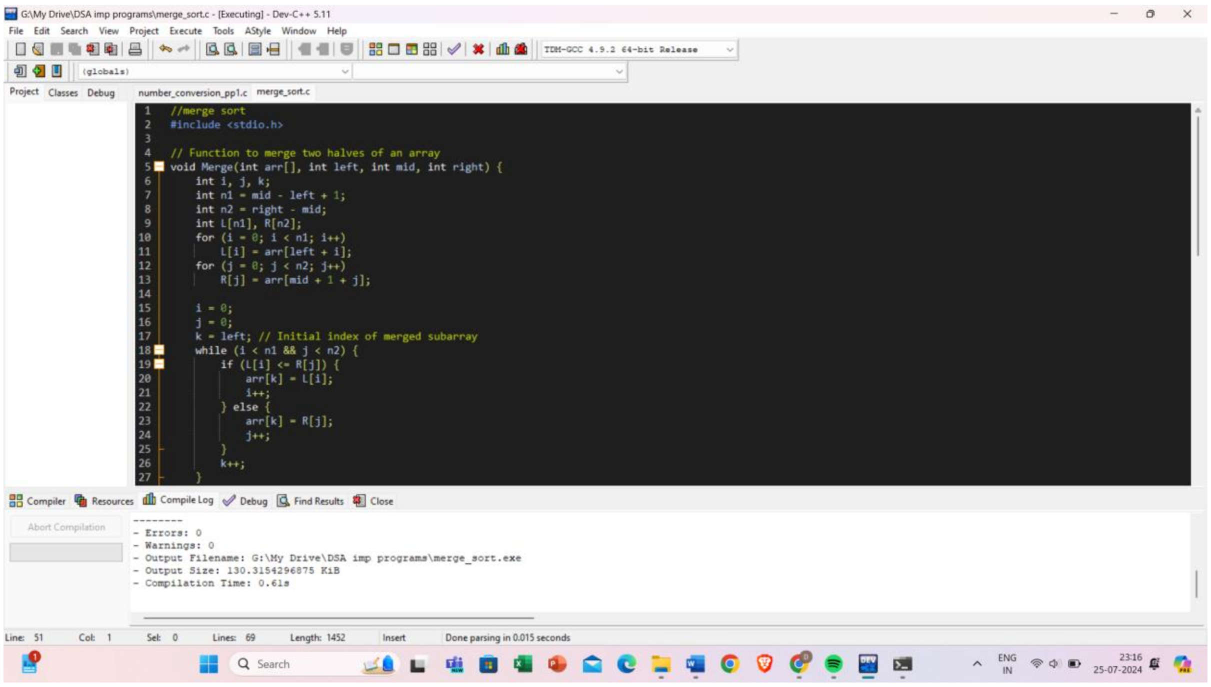
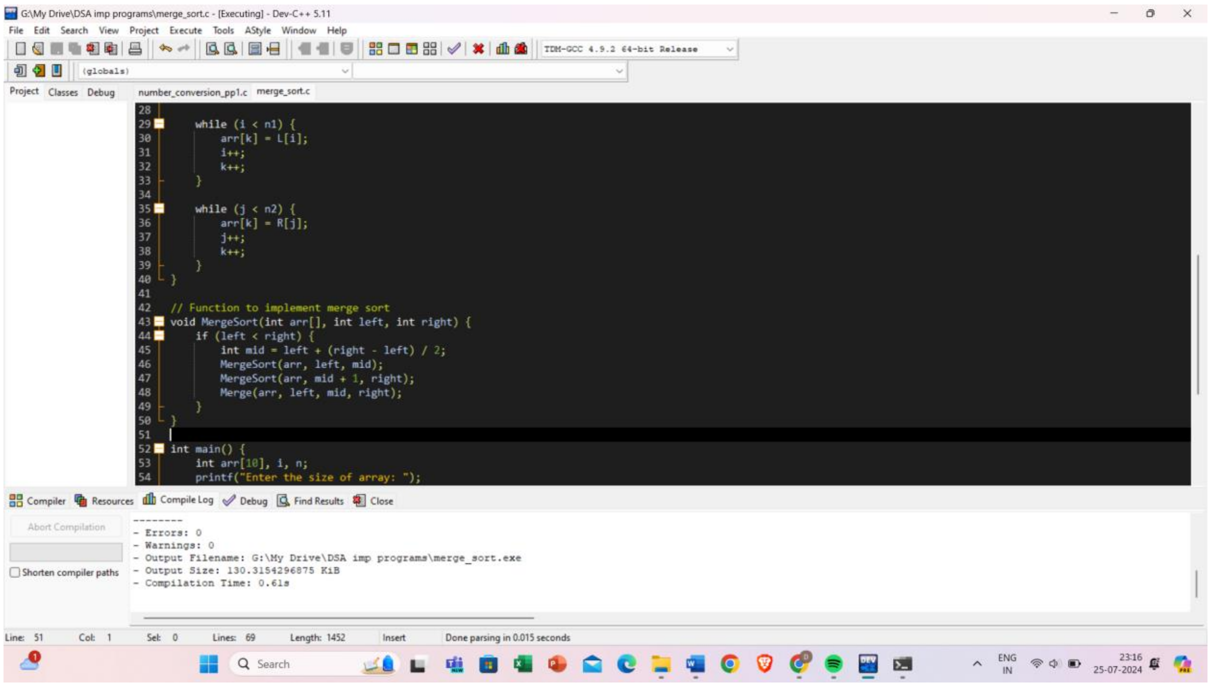
DS assignment:

Source code:



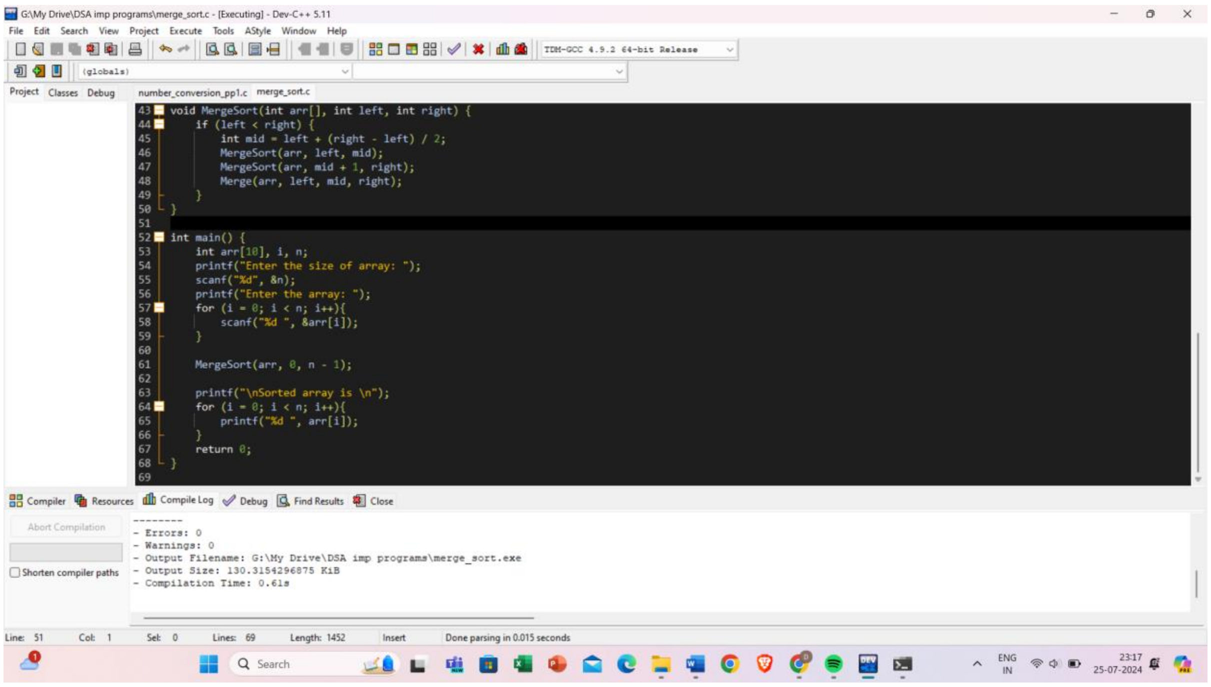
```c
//merge sort
#include <stdio.h>

// Function to merge two halves of an array
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; // Initial index of merged subarray
    while (i < n1 && j < n2) {
        if (L[i] <= R[j]) {
            arr[k] = L[i];
            i++;
        } else {
            arr[k] = R[j];
            j++;
        }
        k++;
    }
```

```
--------
- Errors: 0
- Warnings: 0
- Output Filename: G:\My Drive\DSA imp programs\merge_sort.exe
- Output Size: 130.3154296875 KiB
- Compilation Time: 0.61s
```



```c

    while (i < n1) {
        arr[k] = L[i];
        i++;
        k++;
    }

    while (j < n2) {
        arr[k] = R[j];
        j++;
        k++;
    }
}

// Function to implement merge sort
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 arr[10], i, n;
    printf("Enter the size of array: ");
```

```
--------
- Errors: 0
- Warnings: 0
- Output Filename: G:\My Drive\DSA imp programs\merge_sort.exe
- Output Size: 130.3154296875 KiB
- Compilation Time: 0.61s
```

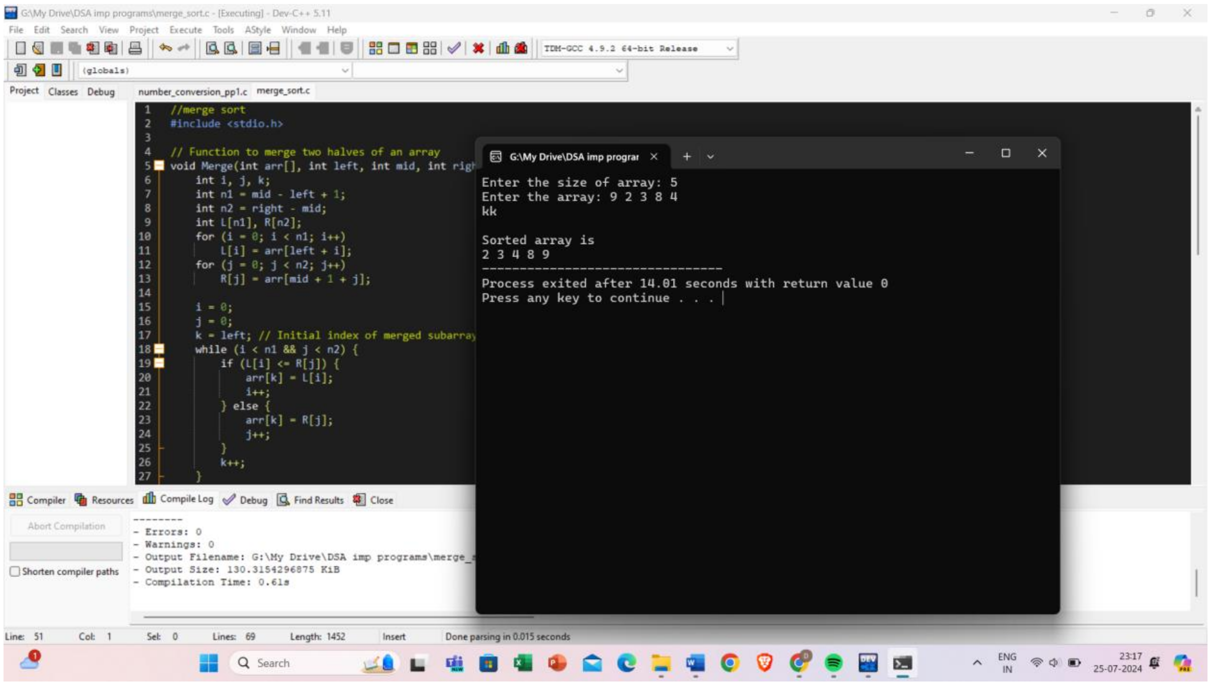

G:\My Drive\DSA imp programs\merge_sort.c - [Executing] - Dev-C++ 5.11

```c
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 arr[10], i, n;
    printf("Enter the size of array: ");
    scanf("%d", &n);
    printf("Enter the array: ");
    for (i = 0; i < n; i++){
        scanf("%d ", &arr[i]);
    }

    MergeSort(arr, 0, n - 1);

    printf("\nSorted array is \n");
    for (i = 0; i < n; i++){
        printf("%d ", arr[i]);
    }
    return 0;
}
```

```
---------
- Errors: 0
- Warnings: 0
- Output Filename: G:\My Drive\DSA imp programs\merge_sort.exe
- Output Size: 130.3154296875 KiB
- Compilation Time: 0.61s
```

Line: 51 Col: 1 Sel: 0 Lines: 69 Length: 1452 Insert Done parsing in 0.015 seconds

Output:


G:\My Drive\DSA imp programs\merge_sort.c - [Executing] - Dev-C++ 5.11

```c
//merge sort
#include <stdio.h>

// Function to merge two halves of an array
void Merge(int arr[], int left, int mid, int righ
    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; // Initial index of merged subarray
    while (i < n1 && j < n2) {
        if (L[i] <= R[j]) {
            arr[k] = L[i];
            i++;
        } else {
            arr[k] = R[j];
            j++;
        }
        k++;
    }
```

```
Enter the size of array: 5
Enter the array: 9 2 3 8 4
kk

Sorted array is
2 3 4 8 9
--------------------------------
Process exited after 14.01 seconds with return value 0
Press any key to continue . . .
```

```
---------
- Errors: 0
- Warnings: 0
- Output Filename: G:\My Drive\DSA imp programs\merge_
- Output Size: 130.3154296875 KiB
- Compilation Time: 0.61s
```