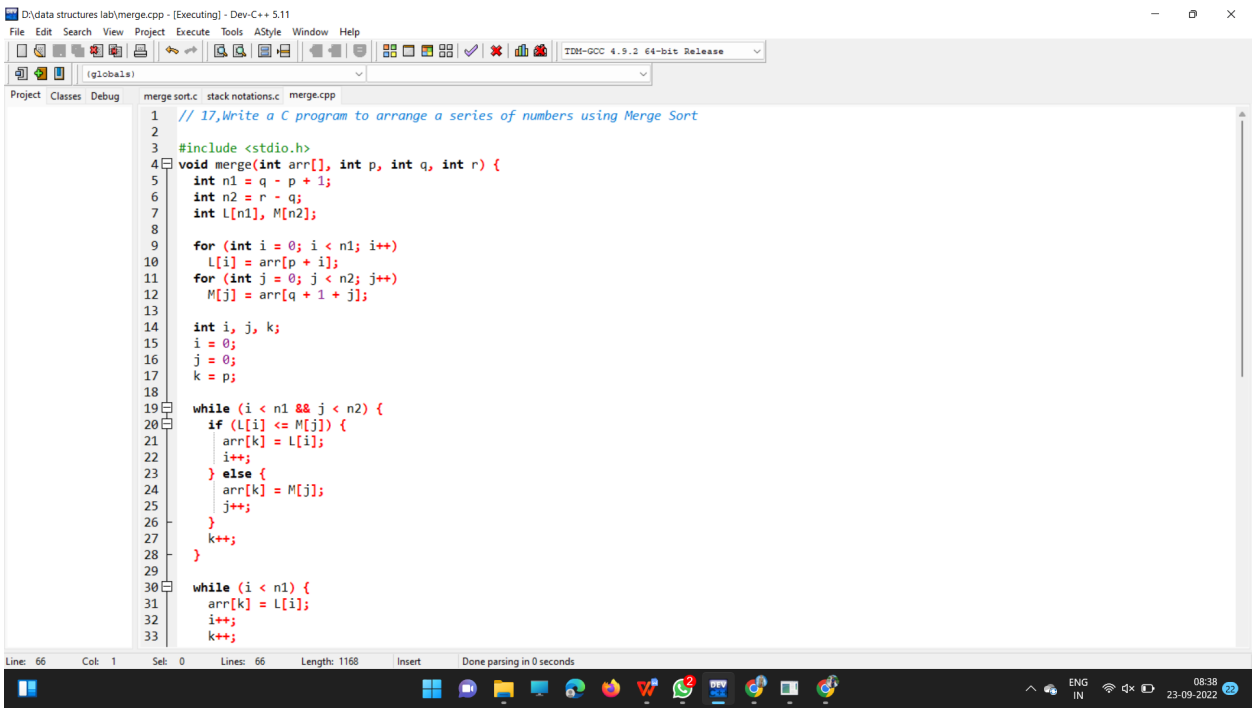
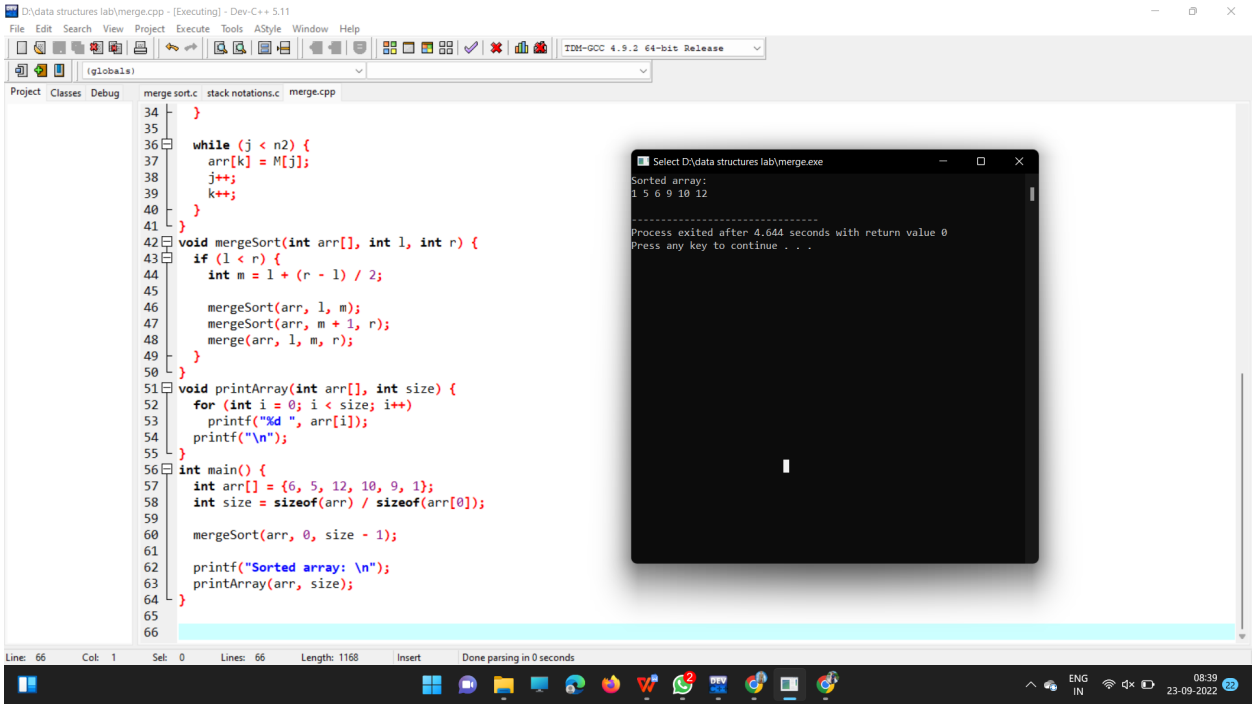
// 17,Write a C program to arrange a series of numbers using Merge Sort

```
#include <stdio.h>
void merge(int arr[], int p, int q, int r) {
  int n1 = q - p + 1;
  int n2 = r - q;
  int L[n1], M[n2];

  for (int i = 0; i < n1; i++)
    L[i] = arr[p + i];
  for (int j = 0; j < n2; j++)
    M[j] = arr[q + 1 + j];

  int i, j, k;
  i = 0;
  j = 0;
  k = p;

  while (i < n1 && j < n2) {
    if (L[i] <= M[j]) {
      arr[k] = L[i];
      i++;
    } else {
      arr[k] = M[j];
      j++;
    }
    k++;
  }

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

  while (j < n2) {
    arr[k] = M[j];
    j++;
    k++;
  }
}
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);
  }
}
void printArray(int arr[], int size) {
  for (int i = 0; i < size; i++)
    printf("%d ", arr[i]);
  printf("\n");
}
int main() {
  int arr[] = {6, 5, 12, 10, 9, 1};
  int size = sizeof(arr) / sizeof(arr[0]);

  mergeSort(arr, 0, size - 1);

  printf("Sorted array: \n");
  printArray(arr, size);
}
```




D:\data structures lab\merge.cpp - [Executing] - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 4.9.2 64-bit Release
(globals)
Project Classes Debug
merge sort.c stack notations.c merge.cpp
34 }
35
36 while (j < n2) {
37 arr[k] = M[j];
38 j++;
39 k++;
40 }
41 }
42 void mergeSort(int arr[], int l, int r) {
43 if (l < r) {
44 int m = l + (r - l) / 2;
45
46 mergeSort(arr, l, m);
47 mergeSort(arr, m + 1, r);
48 merge(arr, l, m, r);
49 }
50 }
51 void printArray(int arr[], int size) {
52 for (int i = 0; i < size; i++)
53 printf("%d ", arr[i]);
54 printf("\n");
55 }
56 int main() {
57 int arr[] = {6, 5, 12, 10, 9, 1};
58 int size = sizeof(arr) / sizeof(arr[0]);
59
60 mergeSort(arr, 0, size - 1);
61
62 printf("Sorted array: \n");
63 printArray(arr, size);
64 }
65
66
Select D:\data structures lab\merge.exe
Sorted array:
1 5 6 9 10 12
Process exited after 4.644 seconds with return value 0
Press any key to continue . . .
Line: 66 Col: 1 Sel: 0 Lines: 66 Length: 1168 Insert Done parsing in 0 seconds
ENG
IN
08:39
23-09-2022