**Data Structures and Algorithms**

**Lab Report**

**Lab09**



|  |  |
| --- | --- |
| Group Members Name & Reg #: | **Muhammad Haris Irfan**  **(FA18-BCE-090)** |
|  |  |
| Class | Data Structures and Algorithms CSC211 (**BCE-3B**) |
| Instructor’s Name | Dilshad Sabir |

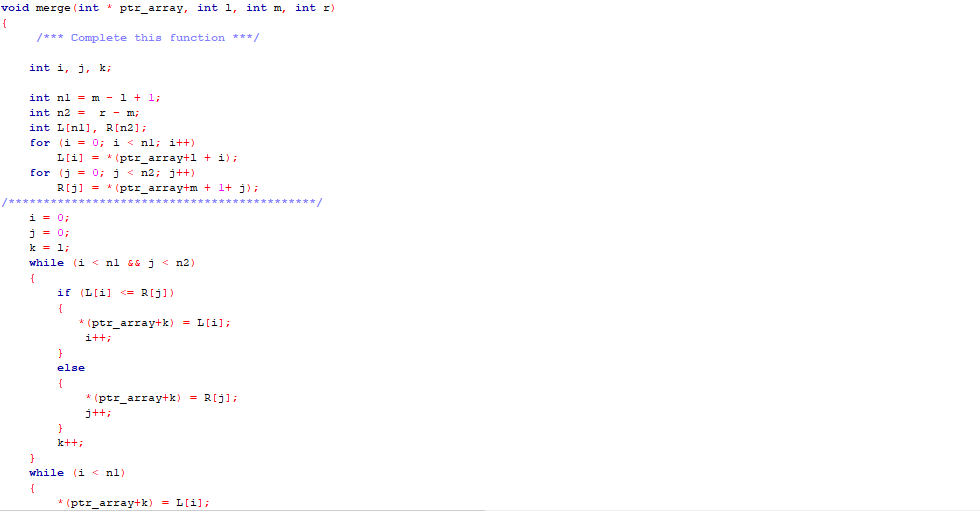
**In Lab Tasks**

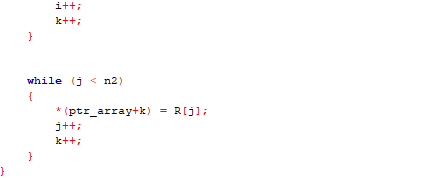
**Task:1**

**Complete the merge () function for Merge Sort.**

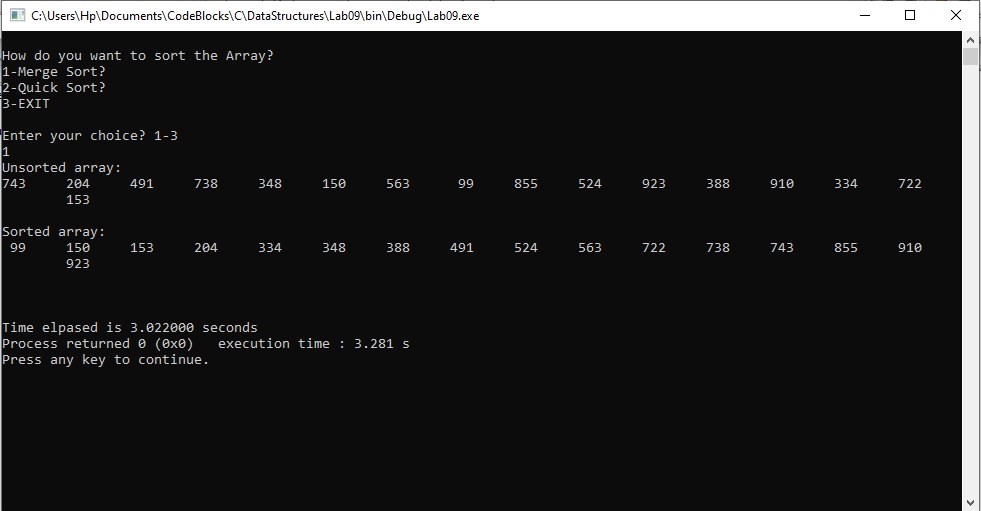
**Solution:**

The code is shown below,





The Result of the following code is attached below:



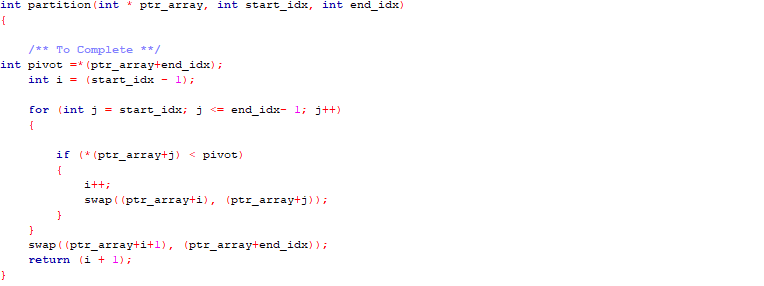
-------------------------------

**Task:2**

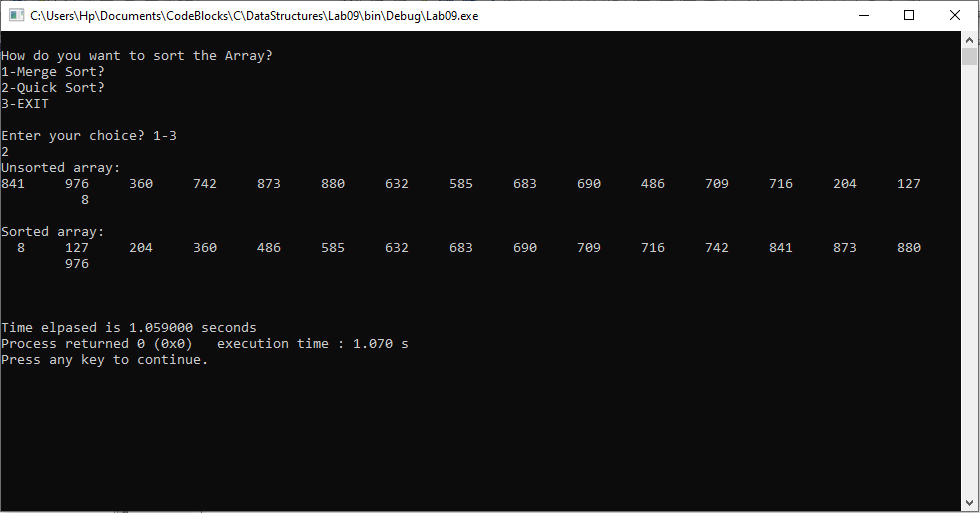
**Complete the partition () function Quick Sort.**

**Solution:**

The code is shown below,



The Result of the following code is attached below:



-------------------------------

**Post Lab Task.**

**Task 3:**

**Study and perform comparative analysis between different sorting algorithms we have implemented in current and previous Lab.**

Solution

In last lab we determined the time of each type of sort, and we reached the conclusion that Merge Sort was indeed the best type of sort as it took least time.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Data Size** | **Bubble Sort** | **Selection Sort** | **Insertion Sort** | **Merge Sort** |
| 16 | 1.14s | 1.74s | 0.90s | 0.74s |
| 128 | 0.90s | 0.84s | 0.58s | 0.47s |
| 1024 | 1.26s | 1.46s | 1.10s | 0.64s |
| 16384 | 2.86s | 2.55s | 2.43s | 1.78s |
| 131072 | 55.18s | 29.88s | 21.12s | 11.7s |

But in this lab we learnt about quick sort, it is similar to merge sort but it uses inplace sorting due to which is is abit faster then even merge sort, for the worst case, it has a quadratic running time given as **O(n2)**, however its efficient implementation can minimize the probability of requiring quadratic time.

Quick sort does not require additional memory space to store data, whereas Merge Sort algorithm requires **O(n) additional memory** for sorting in **O(nlogn)** time, hence we can safely conclude that Quick Sort is the best type of sorting keeping time barrier in mind.

------------------------------

THE END