**OPERATING SYSTEMS**

**Assignment 1 Report**

Threads assignment

**Name: Mostafa Nabil Mohamed Sayed Ahmed.**

**ID:**  **54**

**Problem Statement:**

It required to implement two popular algorithms as multi- threaded ones 1) Matrix Multiplication 2) Merge Sort

**Objectives:**

1. Introducing threads concepts and POSIX threads library.

2. Implementing popular algorithms as multi-threaded ones.

**Main Functions Of Merge sort:**

**Rread\_Array:**

In this method we read the input array from file “merge- sort-readme.txt” and put the values in the array.

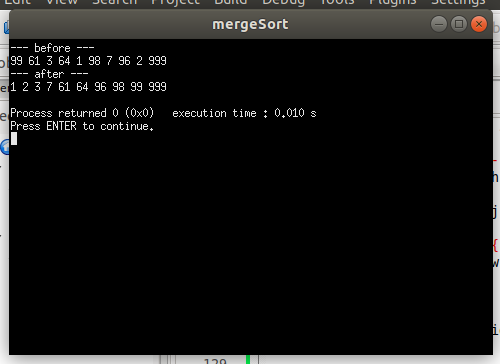
**Merge\_sort:**

we have two functions with the same name one for the main call and the other to be called by the thread as it must be of the type (void \*).  
 In this function we split the array into 2 halves and call the function again using threads until its only one element in the array then we merge them togeather.

**Merge:**

In this method we merge the two arrays we have in sorted order.

**Sample Runs:**





**Main Functions Of Matrix multiplication:**

**Rread\_matrix:**

In this method we read the input matrices from file “Matrix-readme.txt” and put the values in matrix

A and B.

**matrix\_multiplication\_row:**

In this method we create threads to calculate the value of the multiplication of each row and calculate time after it finish.

**multiply\_row:**

In this method we calculate the multiplication of the specified row in a single thread.

**matrix\_multiplication\_element:**

In this method we create threads to calculate the value of the multiplication of each element and calculate time after it finish.

**multiply\_element:**

In this method we calculate the multiplication of the specified element in a single thread.

**→ The time that the multiply\_row takes is less than the time that the multiply\_element takes because the threads that works in parallel is calculating nested for loop. But, in multiply\_element the nested loop is in the main thread waiting much more time than being in parallel .**

**Sample Runs:**

