Lab 1
Name : Omar Ahmed Mohamed Kassem
ID : 43

(1) How the code is organized:

The Main File is (matmult.c) in which the program :

- ➔ open the file given from the user or open the default one.
- ➔ extract the dimension from the file which is should be in this format (row=xx col=yy)
- ➔ read the matrix from the files.
- ➔ start the first method following by the second one.

(2)The Main Functions:

- **method1:**
  - ✔ This function generates array of threads (size of array equals to number of rows of the first matrix).
  - ✔ Each thread will call a function to calculate the elements of the row.
  - ✔ The index of the thread in the array is the number of the row it executed .
  - ✔ The answer is filled in an matrix after joining each thread and then the matrix will copied to a file.
- **Method2:**
  - ✔ This function generates array of threads (size of array equals to number of rows of the first

matrix * number of columns of the second matrix).

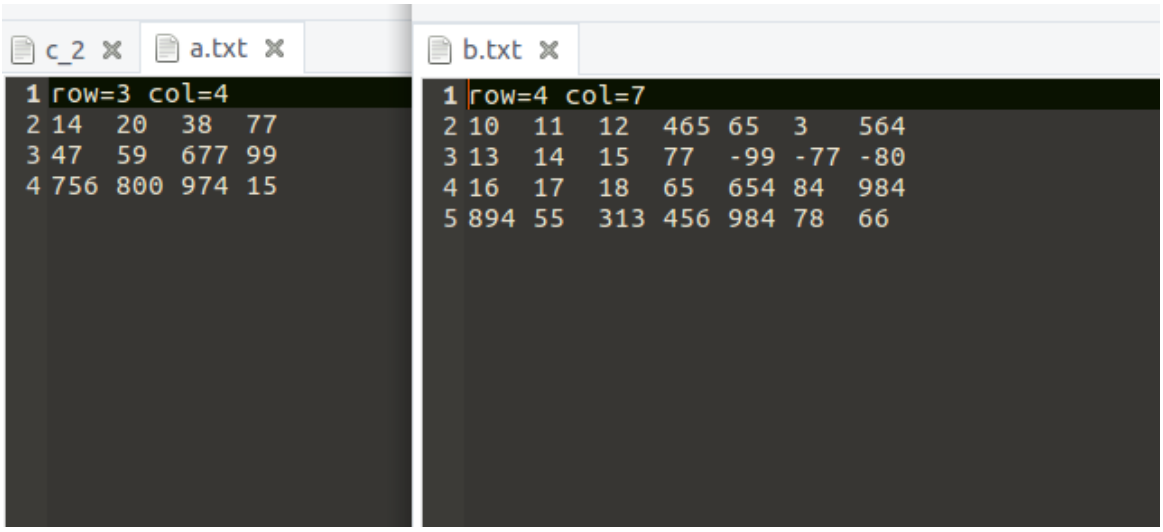
- ✔ Each thread will call a function to calculate an element in a given row and column.
- ✔ Information about the element is passed in a strusct.
- ✔ The answer is filled in an matrix after joining each thread and then the matrix will copied to a file.

(3)How to compile:

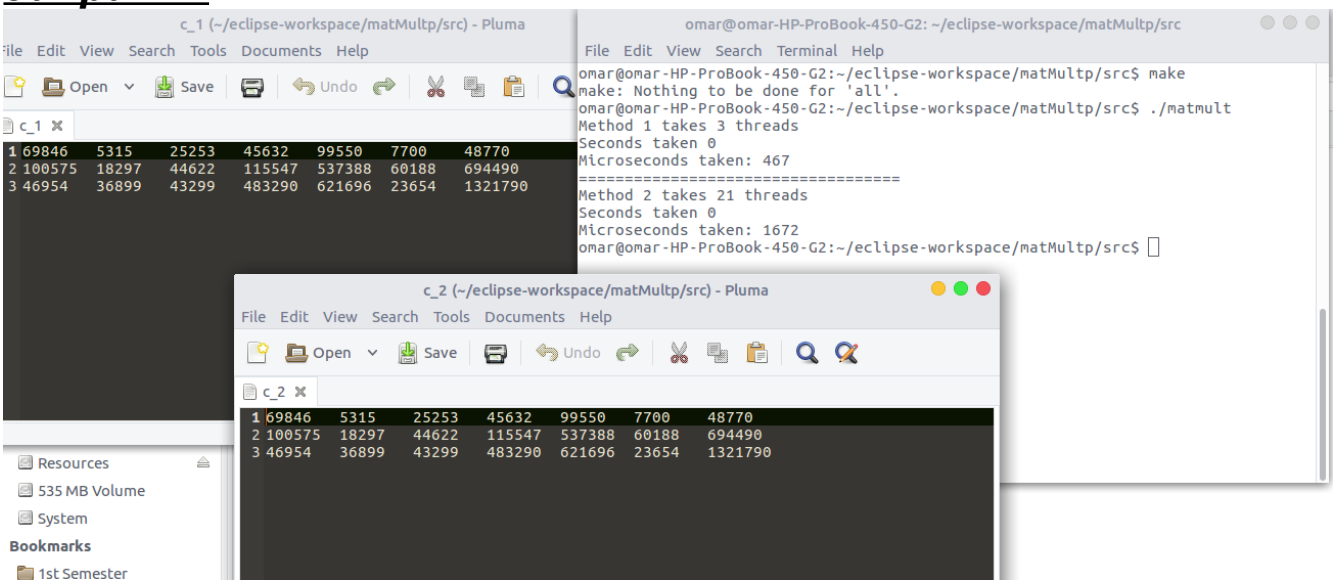
Simply after changing the directory to project directory , you will type in terminal (make) then after compiling type (./matmult) and if the matrices are in different directory then write their pass and write a name to the output file (which will generated in the same folder of the project) else if the matrices in the same file they should be named a.txt and b.txt for input matrices A and B, respectively

(4)sample run:

Input:-



Output:-



(5)Comparison between the two methods of matrix multiplication

For most cases the first method is faster than the second one.
Thats because the system calls for thread creation and joining in the second method take time greater than the computational function do while in the fist method the number of system calls are fewer.