- **Program's Execution Time**

Because matrix multiplication is a central operation in many numerical algorithms including scientific computing and pattern recognition. Thus, different algorithms are designed for multiplying matrices on different types of hardware, including parallel and distributed systems, where the computational work is spread over multiple processors or threads.

However, in this assignment, you are requested to directly applying the mathematical definition of matrix multiplication that takes (Θ(n3) in big O notation). Thus, you need to write a C program that would allow you to measure the time that can be spent during the multiplications of two matrices of various sizes. You need to pass the size of these arrays to your program using the command line arguments (`./a.out n`, where `n` is an integer that represents the size of the two square matrices in which: $R_{n \times n} = A_{n \times n} \times B_{n \times n}$). Note: your program should be written in C (not C++), and compiled with the (gcc) compiler under Linux.

- **Specifications /Algorithm:**

In this assignment, only one thread should be used (the main thread or the master thread). Then, you should measure the multiplication time; the time used in reading/filling the two arrays A and B and writing out the results R should not be measured as part of the programs execution time.

- **Experimentation**

You should make your programs dynamic and configurable from the command line arguments. Dynamically building different matrices, each time with different sizes, these matrices can be filed with random numbers of type `float` that are generated automatically at each run. The C function `malloc()` or `cmalloc()` can be used.

a. You need to experiment and find the multiplication time for 5 different square matrices of sizes:
   i. A_{10x10} X B_{10x10}
   ii. $A_{100x100}$ X $B_{100x100}$
   iii. $A_{500x500}$ X $B_{500x500}$
   iv. $A_{1000x1000}$ X $B_{1000x1000}$
   v. $A_{2000x2000}$ X $B_{2000x2000}$

b. You should record the time for the sequential approach for the execution of each program. Note, you may use the Linux `time` command, but this will not be precise, because it will not allow you to separate the time used in filling the array from the multiplication time. Only the multiplication time is required.

c. Provide a column diagram for your experiments, with all 5 runs.

- **What to submit:**

1. Upload your programs along with your documentations (data & figures) into the JUST e-learning system as one compressed file (*.zip or *.rar). After submitting your work, you should schedule an appointment to discuss your submission. Your grade will be given based on your discussion.