Experiment. No. 06

Mini project 1
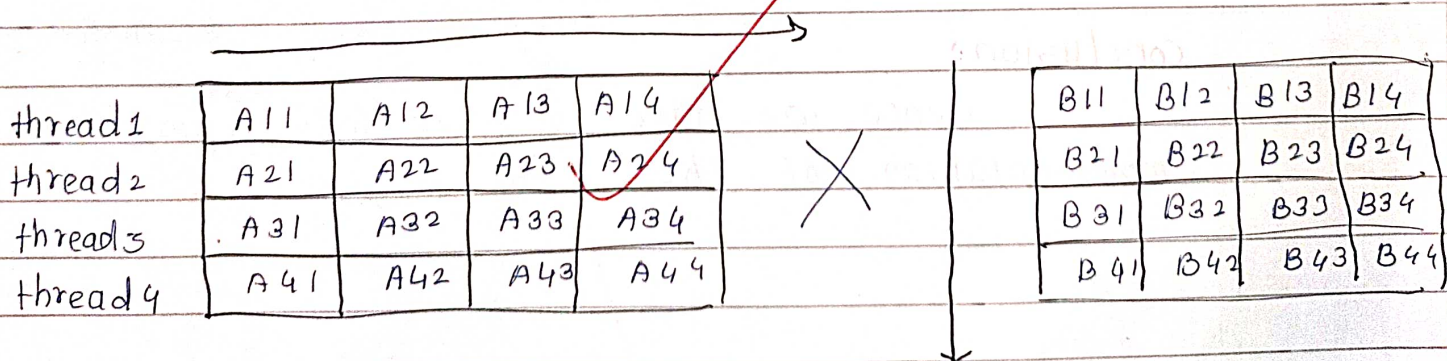
Title: write a program to implement matrix multiplication Also implement multithread matrix multiplication with either one thread per row or one thread per cell. Analyze and compare their performance.

objective: To implement and analyze matrix multiplication and multithread matrix application with either one thread per row or one thread per cell.

Theory:

multiplication of matrix does take time surely. Time complexity of matrix multiplication is $O(n^3)$ using normal matrix multiplication And strassen algorithm improves it and its name its time complexity is $O(n^{2.8074})$.

Depending upon the no. of uses your processor has, you can create the no. of thread required. Although you can create or many thread on you and need a better way is to create each thread for one use.

thread1	A11	A12	A13	A14
thread2	A21	A22	A23	A24
thread3	A31	A32	A33	A34
thread4	A41	A42	A43	A44

X

B11	B12	B13	B14
B21	B22	B23	B24
B31	B32	B33	B34
B41	B42	B43	B44

methods :

1. Matrix size (minip) : In this example, we
matrices of size 1000 x 1000 for A & B
matrices may not demand.

2. multithreading overhead — The use of multi
introduces overhead in terms of thread

3. Number of cpu thread — The number of
available can affect performance.

4. cache & memory — The efficiency of cache
can play a role in performance

5. optimized Libraries — specialized libraries
Numpy have highly optimized matrix mu
routine that can out perform custom multi
implementations.

6. Hardware — The performance can vary bas
the specific hardware.

Conclusion :
        Hence we have successfully
implementation of this mini project completed