

*I certify that this assignment/report is my own work, based on my personal study and/or research and that I have acknowledged all material and sources used in its preparation, whether they be books, articles, reports, lecture notes, and any other kind of document, electronic or personal communication. I also certify that this assignment/report has not previously been submitted for assessment in any other course, except where specific permission has been granted from all course instructors involved, or at any other time in this course, and that I have not copied in part or whole or otherwise plagiarised the work of other students and/or persons. I pledge to uphold the principles of honesty and responsibility at CSE@IITH. In addition, I understand my responsibility to report honour violations by other students if I become aware of it.*

Name: K. Havya

Date: 22 Jan, 2020

Signature: K.H.S

## **REPORT**

### **CODE :**

For single threaded function , I calculated the multiplication of the 2 matrices using 3 for loops.

In multi-process function, I calculated the resulting matrix using two processes, I calculated one half of the resulting matrix (i.e, calculated the elements in first (number of rows)/2 rows) in the child process and the other half in parent process. I created the child process using fork(). (I tried calculating it by creating a separate process for each row but it took more time than doing it in 2 processes).

In multi-threaded process, I calculated the resultant matrix by creating 'z' number of threads and distributing total number of rows equally among threads (i.e, each thread calculates the elements in  $(1/z)$ th part of the matrix). It took less time for 8 threads in my PC .

### **SAMPLE OUTPUT :**

### **INTERACTIVE :**

```
./matmul --ar 3 --ac 3 --br 3 --bc 3 --interactive
```

Enter A:

2 3 4

4 3 4

5 3 7

Enter B:

1 0 0

0 1 0

0 0 1

Result:

2 3 4

4 3 4

5 3 7

Enter A:

2 3 4

4 3 4

5 3 7

Enter B:

1 0 0

0 1 0

0 0 1

Result:

2 3 4

4 3 4

5 3 7

Enter A:

2 3 4

4 3 4

5 3 7

Enter B:

1 0 0

0 1 0

0 0 1

Result:

2 3 4

4 3 4

5 3 7

Time taken for single threaded: 3 us

Time taken for multi process: 748 us  
Time taken for multi threaded: 1334 us  
Speedup for multi process : 0.00 x  
Speedup for multi threaded : 0.00 x

### **NON - INTERACTIVE :**

```
./matmul --ar 500 --ac 500 --br 500 --bc 500
```

Time taken for single threaded: 587370 us  
Time taken for multi process: 403511 us  
Time taken for multi threaded: 150692 us  
Speedup for multi process : 1.46 x  
Speedup for multi threaded : 3.90 x

### **OBSERVATIONS :**

When the size of the matrices to be multiplied is small, single threaded process takes more time than multi threaded and multi process because when the matrix size is small the time taken to create more processes and threads will be dominant. Hence, in the above interactive sample output, since 8 threads are to be created, more time is taken for multi-threaded process.

As the size of the matrix increases, multi threaded takes the least time and then multi process and then single threaded because calculating the resultant matrix dominates the time taken to create threads or processes. Hence, in the above non-interactive sample output, less time is taken by multi-threaded process and then multi-process and at last single threaded process.

In a multi threaded process, in my PC speedup is maximum for creating 8 threads (sharing the job equally among the 8 threads) because the number of threads in my PC is 8.