OS Assignment 3

Ritwik Saha, B16110
Hrushikesh Sudam Sarode, B16032
Mukul Jangid, B16103

Q1.

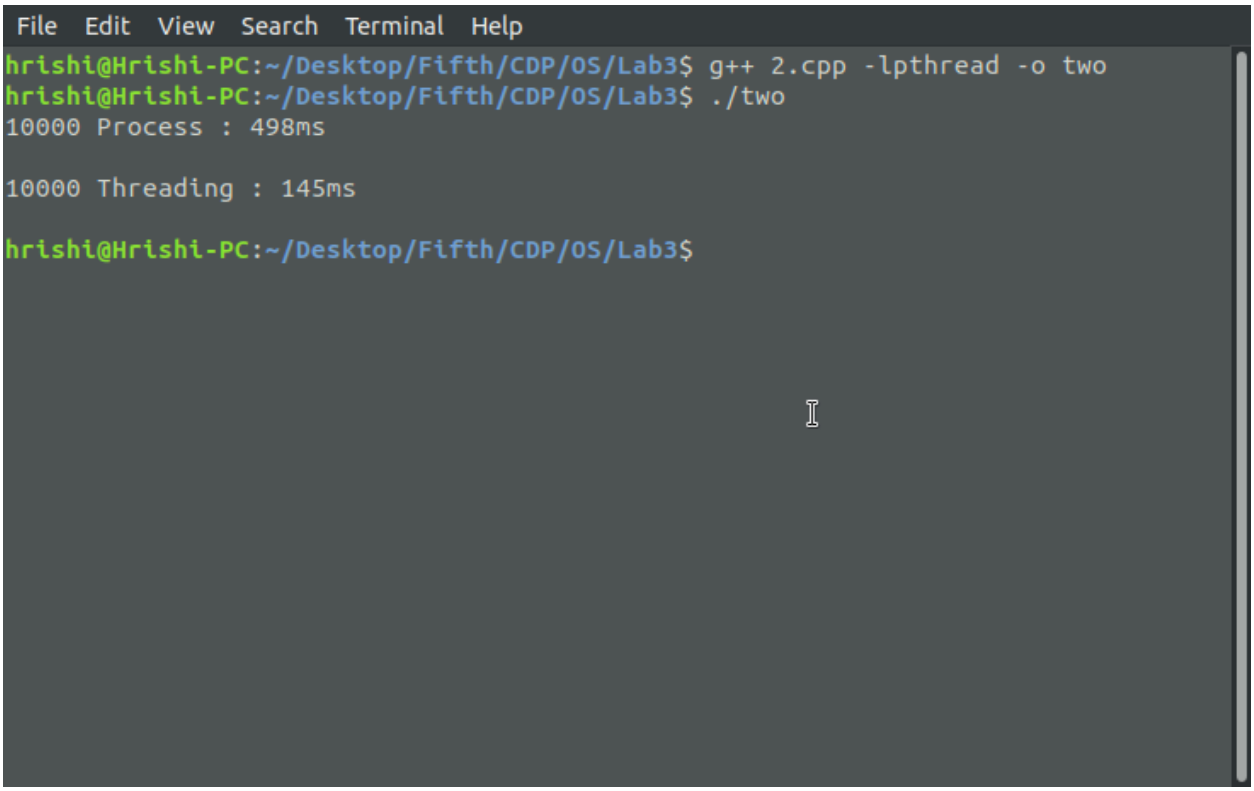
Source Code : `1.cpp`
Compile : `g++ 1.cpp -o one`
Usage : `./one`
Output: `Standard Output`

```
File  Edit  View  Search  Terminal  Help
hrishi@Hrishi-PC:~/Desktop/Fifth/CDP/OS/Lab3$ g++ 1.cpp -o one
hrishi@Hrishi-PC:~/Desktop/Fifth/CDP/OS/Lab3$ ./one
START!!

Process ID : 30360      level #0
Process ID : 30361      level #1
Process ID : 30362      level #1
Process ID : 30363      level #2
Process ID : 30364      level #2
Process ID : 30365      level #2
Process ID : 30366      level #2
hrishi@Hrishi-PC:~/Desktop/Fifth/CDP/OS/Lab3$
```

Q2.

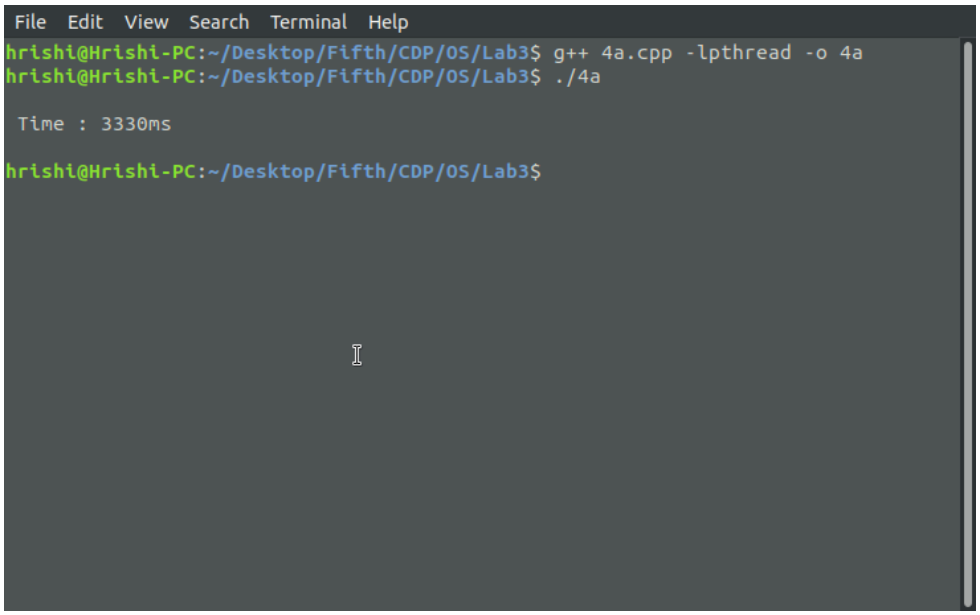
Source Code : `2.cpp`
Compile : `g++ 2.cpp -lpthread -o two`
Usage : `./two`
Output: `Standard Output`



Spawning new threads is much faster than spawning new processes as there is much less overhead associated with threads. This is because new process requires allocation of separate heap from the main program.

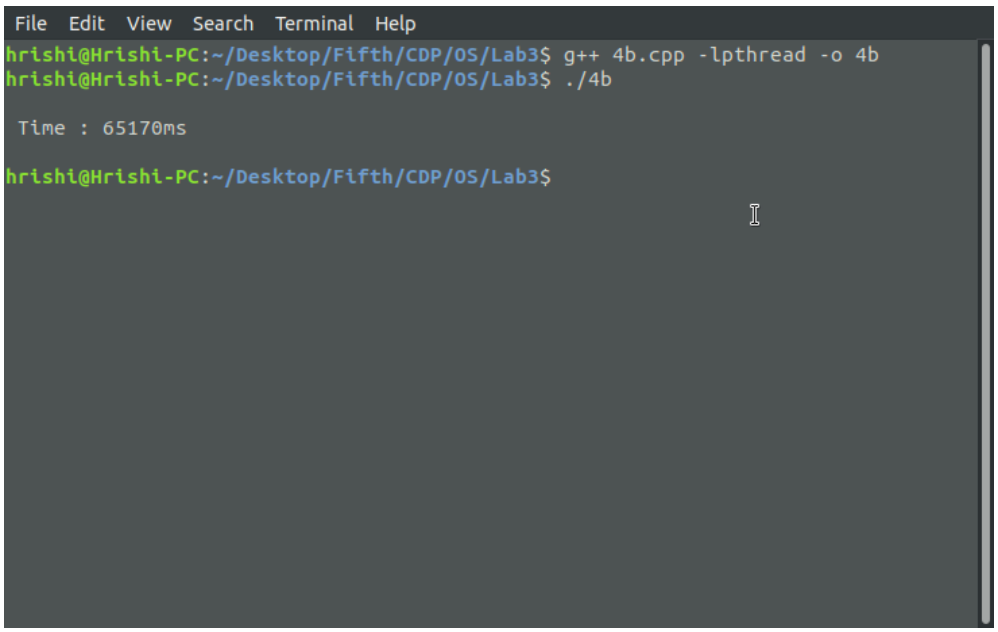
Q4. a)

Source Code : `4a.cpp`
Compile : `g++ 4a.cpp -lpthread -o 4a`
Usage : `./4a`
Output: `Standard Output`



b)

Source Code : `4b.cpp`
Compile : `g++ 4b.cpp -lpthread -o 4b`
Usage : `./4b`
Output: `Standard Output`



In the first part, calculation of each column of the resultant matrix was taken care by a single thread whereas in the second part, calculation of each value of the resultant matrix was done by separate threads. This means a much higher number of threads were spawned in the second case. The huge time difference is there between execution of first and second part as spawning new threads requires large overhead. Furthermore, as the number of processor cores are limited on the machine, spawning large number of threads do not necessarily increase the performance but may increase the execution time. The above mentioned reasons contribute to the fact that the second program is much slower than the first program.