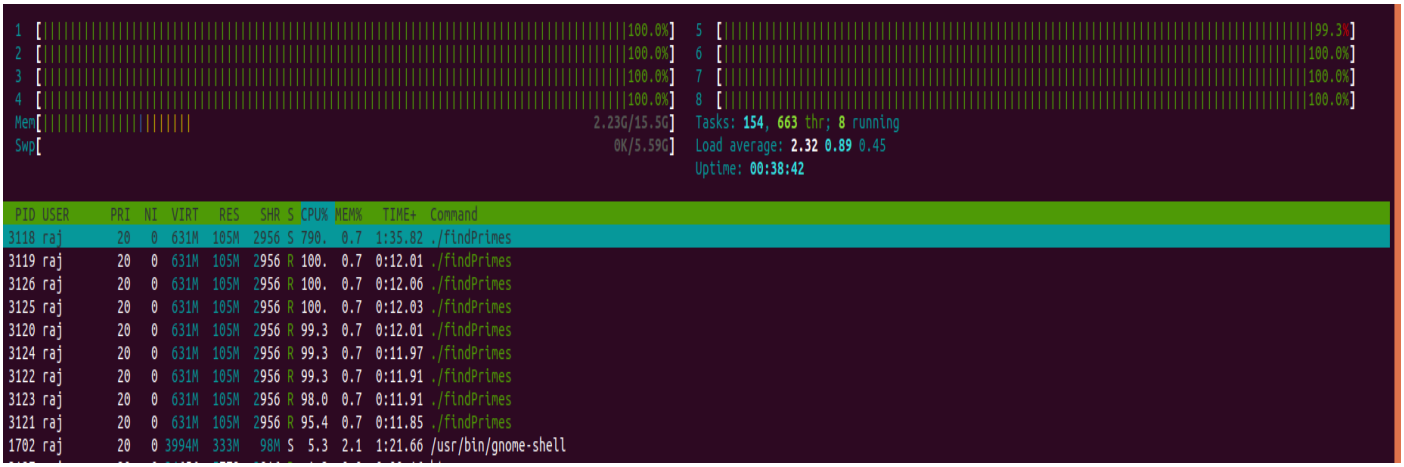
Programming Assignment 1

Description:

- This program finds the total number of prime numbers between 1 to 10^8 numbers using 8 concurrent threads. Once the the process of finding prime numbers is complete, it also prints total execution time in seconds, sum of the prime numbers and top 10 maximum prime numbers found.
- The program discards 0, 1 and all of the even numbers except for 2. Hence, processing only odd values.
- It also makes use of atomic counter to safely cycle through all of the numbers from 1 to 10^8. When saving the prime number, the program makes use of locking mechanism.
- Program does not involve any I/O access when threads are processing the numbers and only prints the required output (i.e. execution time, total number of primes, sum of primes, top 10 primes) once process completes.
- The prime number is found by calculating modulus till square root of a particular number.
- The experiments are run 6 times to provide an analysis of execution times. Screenshots of output are provided below as well as in file primes.txt. Execution time graph is also provided in the last section.
- Below is a screenshot of CPU usage which shows 8 threads running when findPrimes executable is run.



Step in finding PrimeNumber:

1. Program sets a variable flag = “true” at the start of the process
2. Then each thread increments the atomic counter by 1 before calculating prime
3. the value is then calculated to find if it is a Prime number or not
4. if the value is not prime number flag value is changed to “false”
5. prime numbers are stored into a vector if flag = “true”

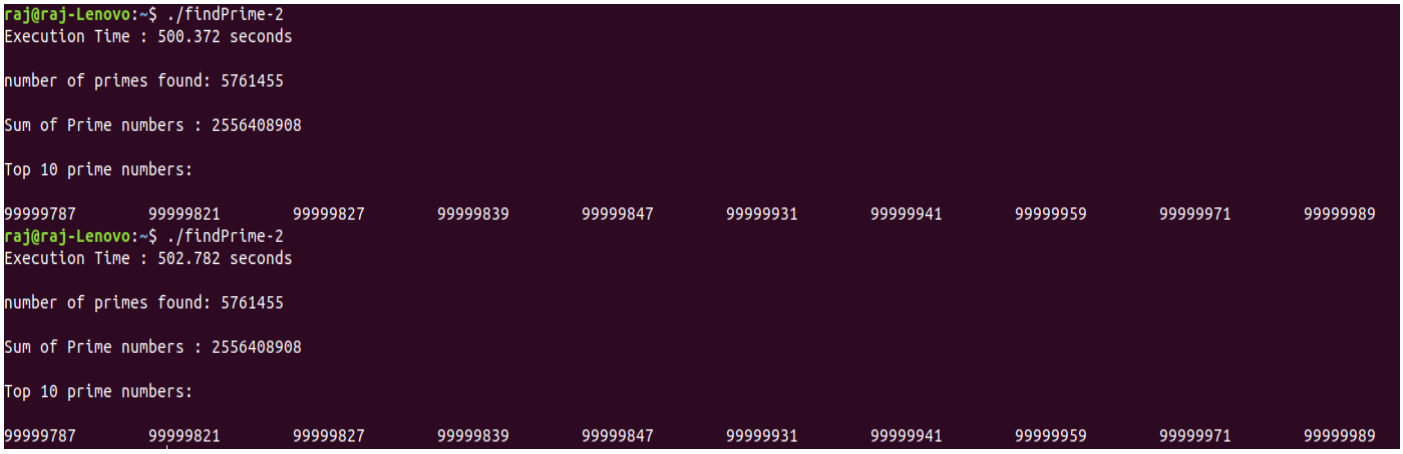
How to run:

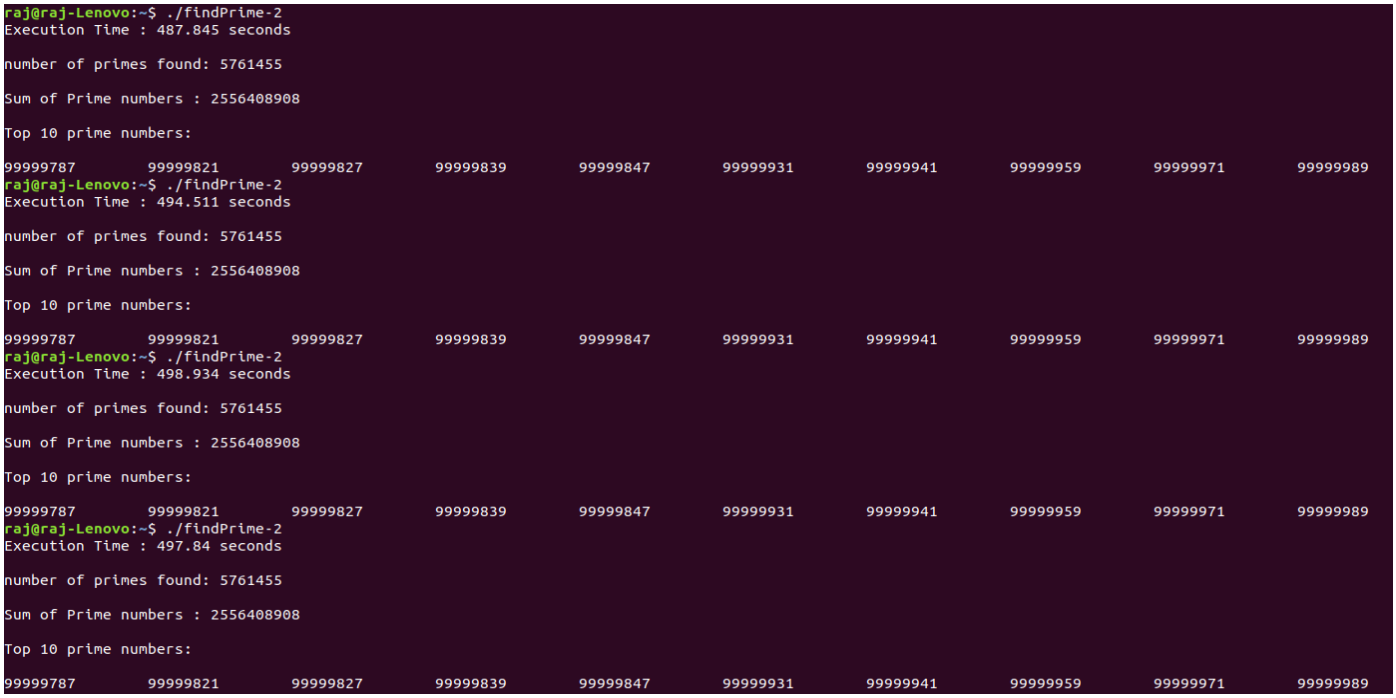
- program is written in c++ programming language with c++11 version and executed on Linux (Ubuntu) system.
- First unzip files. To run the code simply copy and paste following commands in terminal.

```
g++ -std=c++11 -pthread findPrimes.cpp -o findPrime
./findPrime
```

- First commnand will create an executable called “findPrimes ” and second command will run this execuable.

Output Screenshots:





- It’s clear from output that program finds 5761455 prime numbers which is the correct number of primes between 1 to 10^8. The process takes approximately 495 seconds.

Execution time graph: