

COMP 2401 -- Tutorial 10

Sockets and Threads

Learning Objectives

After this tutorial, you will become familiar with:

- **sockets, and**
- **threads**

Tutorial Steps

1. Speeding up prime.c

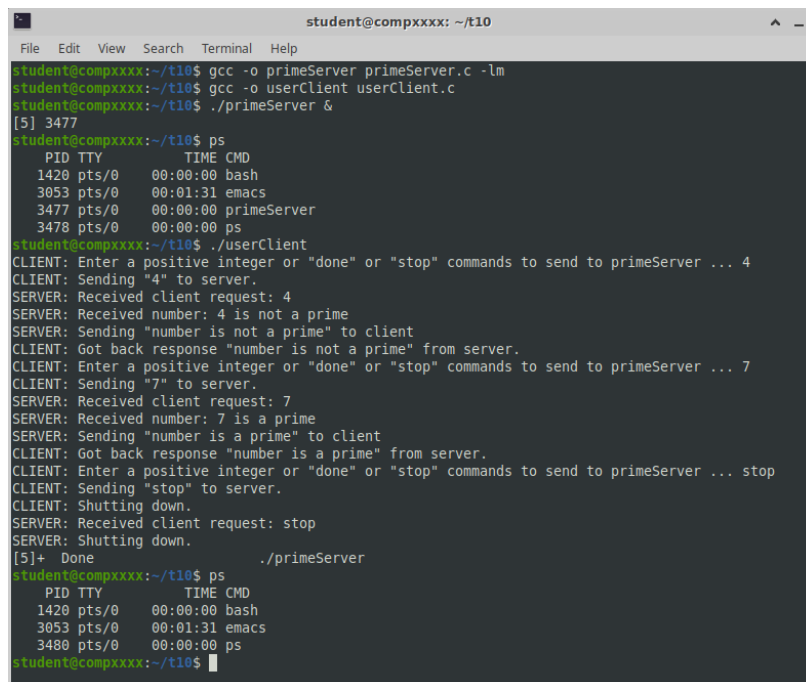
Get `prime.c` from the T10 folder. Speed it up by making each prime call a thread and call the new program `primeThreads.c`

Note the difference in execution time by using the `time` command from the shell:

```
> gcc -o prime prime.c -lm
> time ./prime
    (Note the execution time)
> gcc -o primeThreads primeThreads.c -lm -lpthread
> time ./primeThread
    (Is there a difference with the time above and if so, why?)
```

- #### 2. Transform `udpServer.c` (Section 5.3 of course notes, p220-221) into `primeServer.c` by using the `prime.c` from above. Also make a client, call it `userClient.c`, similar to `udpClient.c` (Section 5.3 of course notes, p221-222) where
- `userClient.c` gets a number from the user,
 - sends it to `primeServer.c` which finds out if it's a prime number or not and responds to the `userClient` with the result,
 - `userClient.c` prints the server's reply to the user

The interaction would be similar (not identical) to this



```
student@compxxxx: ~/t10
File Edit View Search Terminal Help
student@compxxxx:~/t10$ gcc -o primeServer primeServer.c -lm
student@compxxxx:~/t10$ gcc -o userClient userClient.c
student@compxxxx:~/t10$ ./primeServer &
[5] 3477
student@compxxxx:~/t10$ ps
  PID TTY          TIME CMD
 1420 pts/0    00:00:00 bash
 3053 pts/0    00:01:31 emacs
 3477 pts/0    00:00:00 primeServer
 3478 pts/0    00:00:00 ps
student@compxxxx:~/t10$ ./userClient
CLIENT: Enter a positive integer or "done" or "stop" commands to send to primeServer ... 4
CLIENT: Sending "4" to server.
SERVER: Received client request: 4
SERVER: Received number: 4 is not a prime
SERVER: Sending "number is not a prime" to client
CLIENT: Got back response "number is not a prime" from server.
CLIENT: Enter a positive integer or "done" or "stop" commands to send to primeServer ... 7
CLIENT: Sending "7" to server.
SERVER: Received client request: 7
SERVER: Received number: 7 is a prime
SERVER: Sending "number is a prime" to client
CLIENT: Got back response "number is a prime" from server.
CLIENT: Enter a positive integer or "done" or "stop" commands to send to primeServer ... stop
CLIENT: Sending "stop" to server.
CLIENT: Shutting down.
SERVER: Received client request: stop
SERVER: Shutting down.
[5]+ Done
student@compxxxx:~/t10$ ./primeServer
  PID TTY          TIME CMD
 1420 pts/0    00:00:00 bash
 3053 pts/0    00:01:31 emacs
 3480 pts/0    00:00:00 ps
student@compxxxx:~/t10$
```

Tar all the files you worked on into `t10.tar` and submit it.