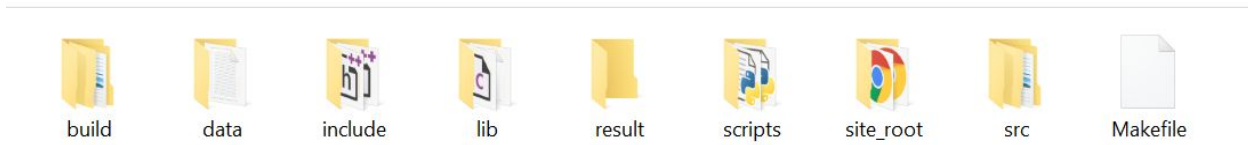


COL 701 : SSL- Assignment 4

Project Directory Structure :



build :

Obj : contains all object files.

Bin : contains all executables.

data : contains generated data (like data for plots)

include : contains all header files.

lib : contains a library for myThread.

result : contains plot.

scripts : scripts for testing web server.

site_root : root folder for webserver

src : contains source files.

Makefile : contains rules for build

Source File Description :

Name	Date modified	Type	Size
httpClient.c	25-12-2020 18:20	C Source	3 KB
HTTPHandler.c	25-12-2020 17:32	C Source	12 KB
Mime.c	20-12-2020 13:19	C Source	1 KB
ServerSignals.c	23-12-2020 18:02	C Source	1 KB
SocketHandler.c	25-12-2020 18:11	C Source	3 KB
ThreadQueue.c	23-12-2020 23:48	C Source	2 KB
webserver.c	25-12-2020 18:17	C Source	4 KB

Configuration :

```
#define PORT 8888
#define BUFFER_SIZE 1024*128
#define THREAD_POOL_SIZE 10
#define CHUNK_SIZE 256
```

Functions used from myThread.

myThread_create() : to create server thread for each client.

myThread_cancel() : to exit a thread when MAX_THREAD pool reached.

myThread_exit() : to exit a thread.

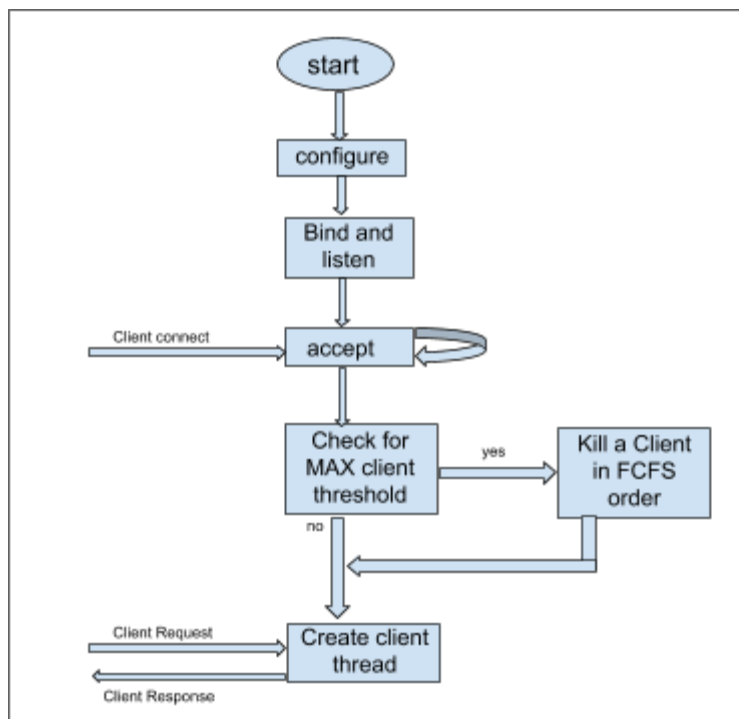
myThread_mutex : To achieve mutual exclusion between threads.

myThread_mutex_init : To initialise mutex.

myThread_mutex_lock : locking a mutex variable.

myThread_mutex_unlock : unlock a mutex variable.

Operations :

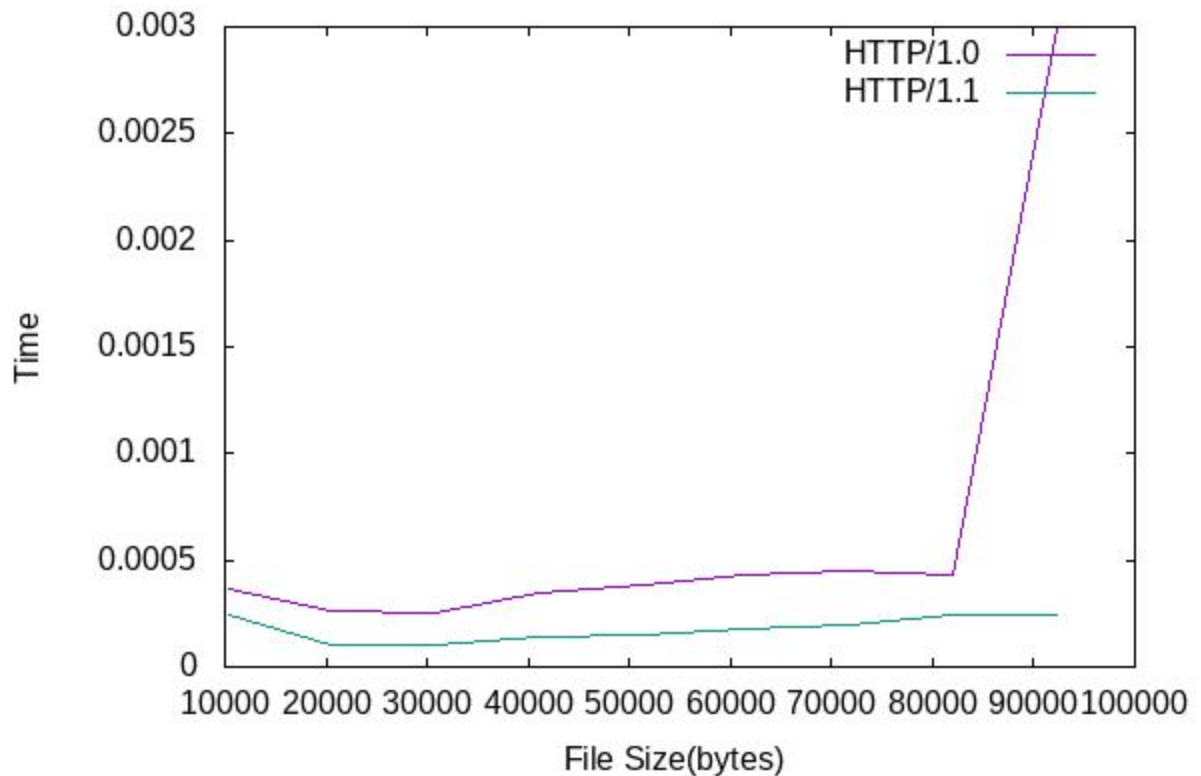


How to Run :

In Project root Dir : make webserver

Deliverables.

1. Test 1: Single client-server communication.
 - 1.1. A simple hello message with client ID from client to server.
make helloClient
 - 1.2. Client requests for index.html from server 2.
 - 1.2.1. Open any web browser and enter : <http://localhost:8888/index.html>
2. Test 2: Multiple client-server connection.
 - 2.1. Simultaneous communication on a different connection.
 - 2.1.1. make runmultipleclient
 - 2.2. Try to create more than Max_connection_threshold connection
 - 2.2.1. Max_Threadpool_size is 10. On max count reach client deleted in FCFS order.
3. Test 3: Plot the time taken to transfer files from the server using HTTP 1.0 and HTTP 1.1 and draw time vs. file size graph using GNUPlot.



4. Now that you have built a multi-threaded web server; extend this to build a multi Chat working on the browser. The server acts as an intermediate between clients and stores chat history as web server logs. This part is open-ended; you can use your creativity to make it a more realistic web chatting application.
Visit to <http://localhost:8888/chat/chat.html> from multiple client windows.
5. Once the chat server is built, the player wants to play a game. In-game, the server initiates a global integer variable, and other players want to update the value. However, only one player can update the value at a time. You need to use the synchronization method built-in assignment 2.
Visit to <http://localhost:8888/game/game.html> from multiple client and play.

Make sure web server running all time when these operations is performed.