



Προγραμματισμός Συστήματος Εργασία 3

Ονοματεπώνυμο: Μπαρμπάρ Ηλίας-Ελιάς

ΑΜ: 1115201200118

Files

server.cpp:

Our server starts by checking the command line arguments, then proceeds to create/bind/list to the main socket. Using select our server accepts connections from clients where it answers to incoming requests (in string form) which can be “LOG ON <IP add, portNum>” which also serves the GET_CLIENTS function, and LOG_OFF.

client.cpp:

Our client sets it's listening port and then proceeds to log in to the server to LOG_ON and get the client list, which it creates and then adds the nodes to our buffer. The main thread listens to incoming requests (in string form) while threads of ‘workerThreads’ number are created to read from our ring queue buffer and complete two distinct tasks: A GET_FILE_LIST task which gets a list of files that another client is able to share with the current client and then, a GET_FILE task which requests one of the given files from the previous procedure.

list.cpp/hpp:

This list is the client list that our server uses and it has basic linked list functionality along with some of the functionality needed to inform other clients about LOG_ON and LOG_OFF messages that come to the server.

list2.cpp/hpp:

This is the client list that our client and it's threads use. It has a basic linked list functionality.

threadFunctions.cpp/hpp:

(Performs the two tasks mentioned above).

readCmd.cpp/hpp:

A simple cmdl assignment function.

buffer.cpp/hpp:

A simple ring buffer, that we use with pthread_mutex and pthread_cond to add/remove/initialize.

Compilation and Running the program

We compile using a makefile so all the user has to do is type 'make' which has the following form:

```
all: dropbox_server dropbox_client
```

dropbox_server:

```
g++ server.cpp list.cpp basicFunctions.cpp readCmd.cpp -o dropbox_server
```

dropbox_client:

```
g++ -pthread client.cpp list2.cpp buffer.cpp readCmd.cpp threadFunctions.cpp  
basicFunctions.cpp -o dropbox_client
```

```
.PHONY: dropbox_server dropbox_client
```

We run the program using the following format:

```
./dropbox_server -p portNum
```

```
./dropbox_client -d dirName -p portNum -w workerThreads -b bufferSize -sp serverPort -sip  
serverIP
```

Resources

<https://stackoverflow.com/questions/19017651/how-to-send-files-in-chunk-using-socket-c-c>

<https://www.linuxquestions.org/questions/programming-9/tcp-file-transfer-in-c-with-socket-server-client-on-linux-help-with-code-4175413995/>

<https://www.geeksforgeeks.org/mutex-lock-for-linux-thread-synchronization/>

<https://en.cppreference.com/w/c/string/byte/strtok>

http://maxim.int.ru/bookshelf/PthreadsProgram/htm/r_27.html

<https://stackoverflow.com/questions/15910749/thread-synchronization-at-c>

<https://www.geeksforgeeks.org/sum-array-using-pthreads/>

<https://stackoverflow.com/questions/1352749/multiple-arguments-to-function-called-by-pthread-create>