Aaryan Shenoy
CSCE 438 - 500

MP1 Report

The program's objective is to create chat room servers that client processes can send messages into after client-server interaction. A main server will continuously run in order to accept multiple client connections and handle their respective commands. The main server also has the ability to create servers for chat rooms.

In order to accomplish this, I utilized a multithreaded and multiple process approach. Servers create one thread per client and run a function to process client commands or chat room messages. The servers send Reply structs to the corresponding client and the clients can read the structs for information about the success of the command and other important chat room information. If a client wanted to create a chat room, the main server would fork to execute a new server process for the chat room. Keeping in mind that the processes will not share memory, there are some global variables in the server code that are used for either the main server or the chat rooms. When a client joins a chat room server after processing their Reply structs, client processes set up two threads, one to listen to chat room messages from other clients and another to write messages to other chat room clients. In summary, the idea of this approach was to set up new processes whenever an entity like a chat room server needed to be created and to set up threads whenever an entity had to perform simultaneous tasks.