Your goal is to create a single multi-threaded server that can handle multiple clients at once. Since this is operating systems: the implementation language must be C or C++.

My suggested order of getting things working:

1. Study the provided client and server files to understand how sockets work,
2. Make the server and client able to send/receive messages ("chat") until some disconnect sent by the client (e.g., "bye"),
3. Update the server to accept a new client over and over
4. Make the server accept and service each new connection on a new thread instead of on the same thread over and over
5. See if you can get multiple clients connected and chatting with the server at once
6. Get the messages from each server thread forwarded to other chat threads so clients can see each other's messages

You can (optionally) go above and beyond here by building up a GUI for the chat client, adding the ability to send more than text (like images) and having them render properly in each client window, and whatever other functionality you think interesting (old-man Harrison says: the "Discord" should provide you with some good ideas!)

Be prepared to demonstrate your chat server by the final week of class.