# Documentation CSC 645 Computer Networks Final Project

Rohan Rawat
Student ID: 917018484

## Summary of Project:

The point of this project was to create a client server protocol where we can communicate what we want from the server to the client or the user. The menu is sent from the server to the client and the client is the one that chooses what option they want to select. I designed my own protocol where I created a dictionary that would store headers such as the message, if an input is needed, and if we are using another protocol that is already on the client side like udp. This way I know how to process a response from the server properly if I need input from the user or if I just need to print something out on the client side.

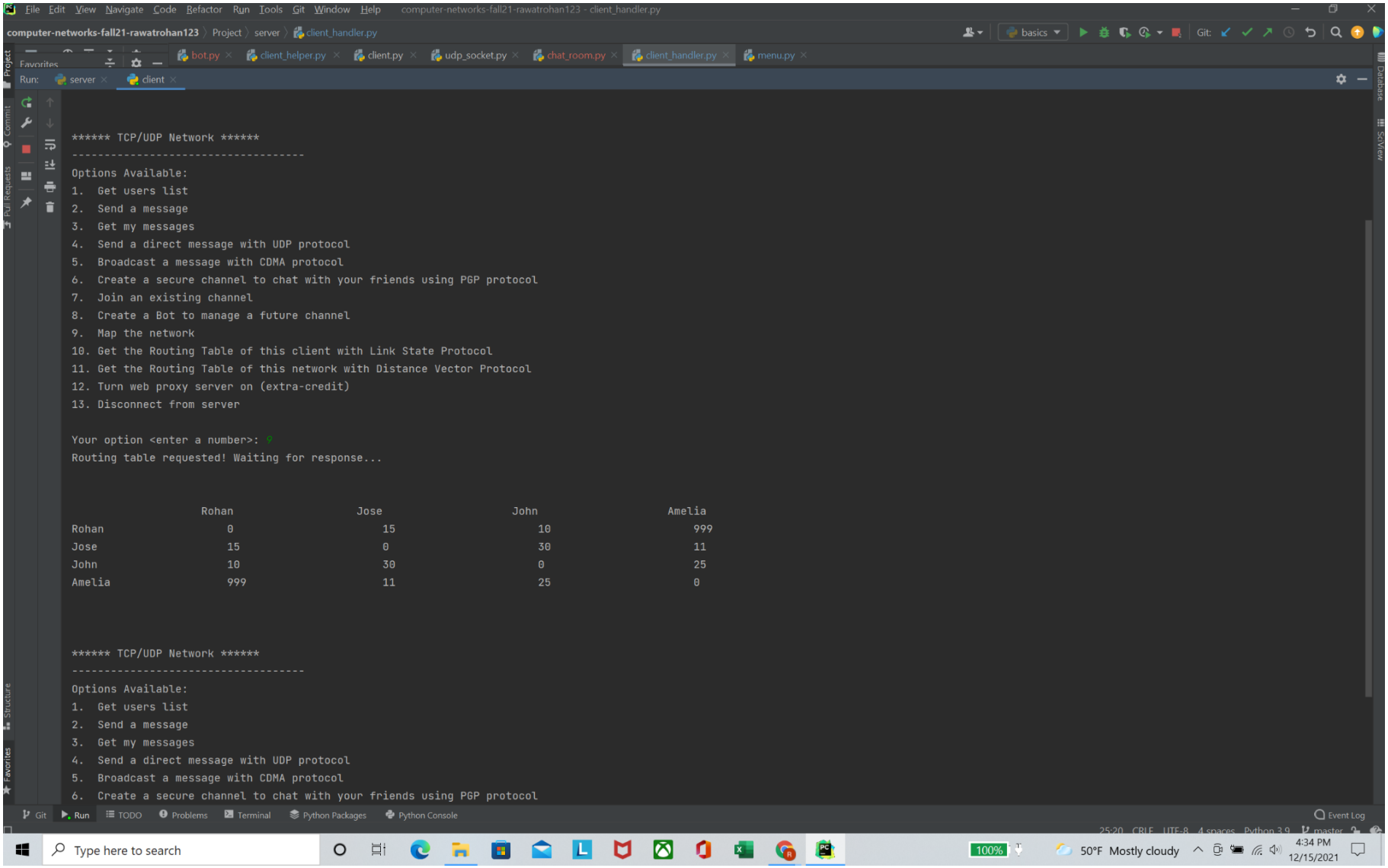
For the second part of this project, I had to implement a real-time chat using the PGP protocol. I was able to get it working but not get the bot working, however. Then, I had to implement mapping the network, the link-state protocol, and the distance vector protocol which I got working. Disconnecting the client from the server was very simple.

## Challenges Faced:

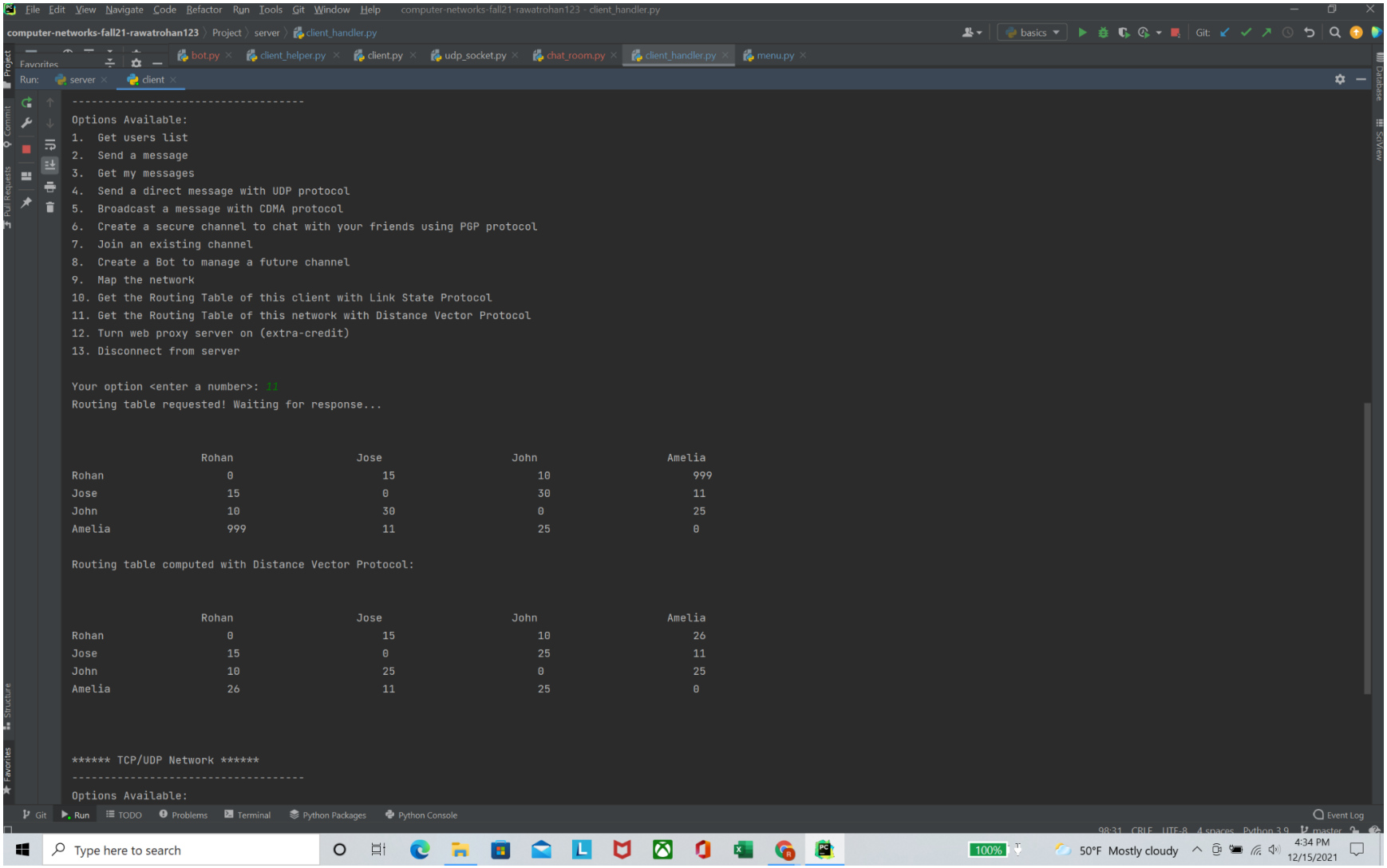
The second part of this project was a lot harder than the first. I struggled a lot with this project because I could not figure out how to shorten the message integer first of all. With the professor's help, I found out about the double modulo rule and implemented that into my logic. Then I was able to get PGP working for a certain length of messages. After, this I had a lot of trouble with the bot. I could not figure out how to implement some of the options but I had an idea for the first three. Then, I had to implement all the distance protocols. Mapping the network was extremely difficult and I spend a lot of time looking into this. I tried my best to find a way to calculate the distance between two nodes but could not figure it out. I looked into several things such as HZ to calculate distance but I always ended up back to square one. I asked the professor and he told me just hard code the values and focus on the link-state protocol and the distance vector protocol. I was able to implement these successfully. I definitely felt as if I could have done a lot better on this part of the project but it just required so much research on my part and I could not find the time for it this semester. But it definitely could have been doable by me.

## Screenshots:

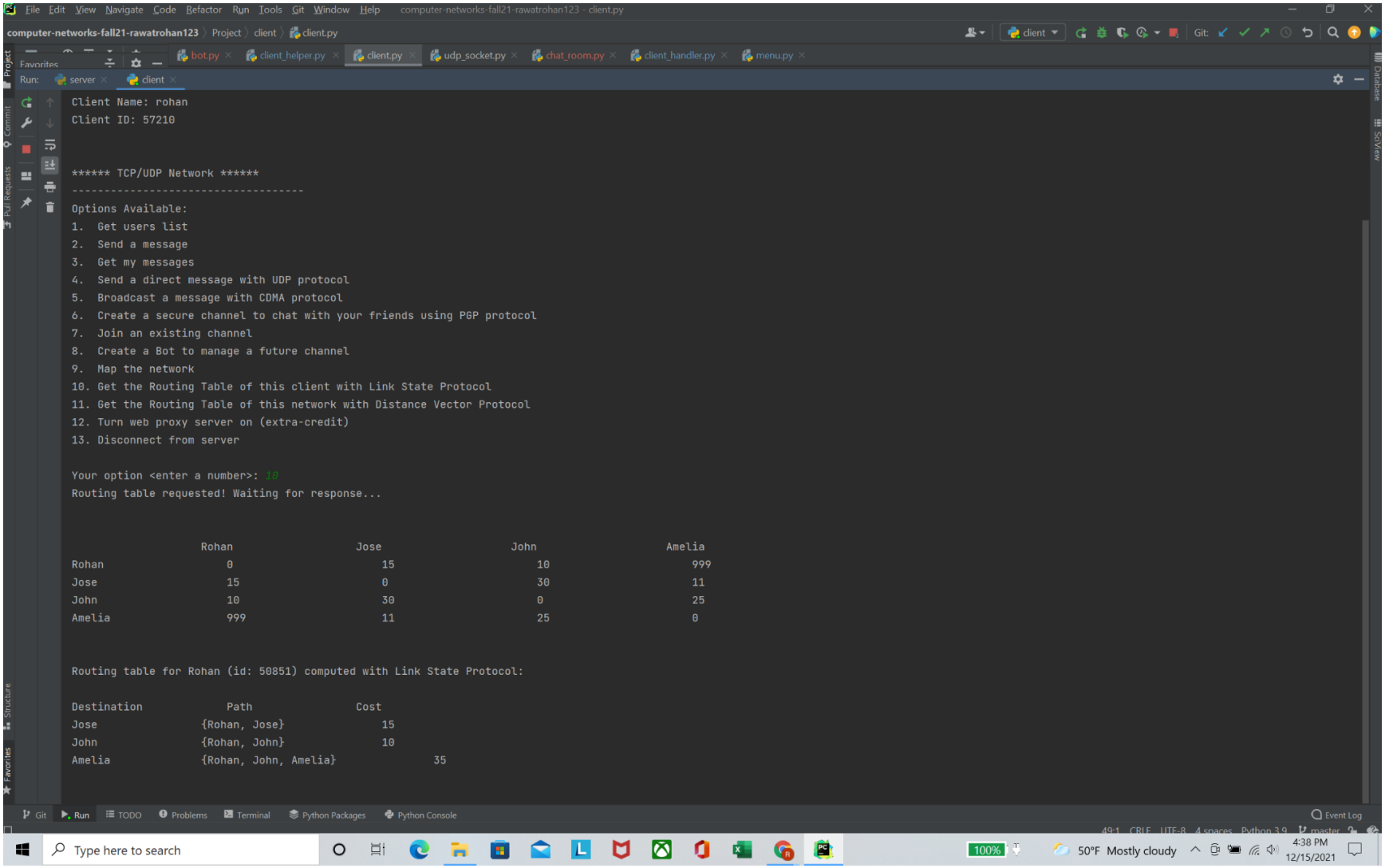
Mapping the Network:



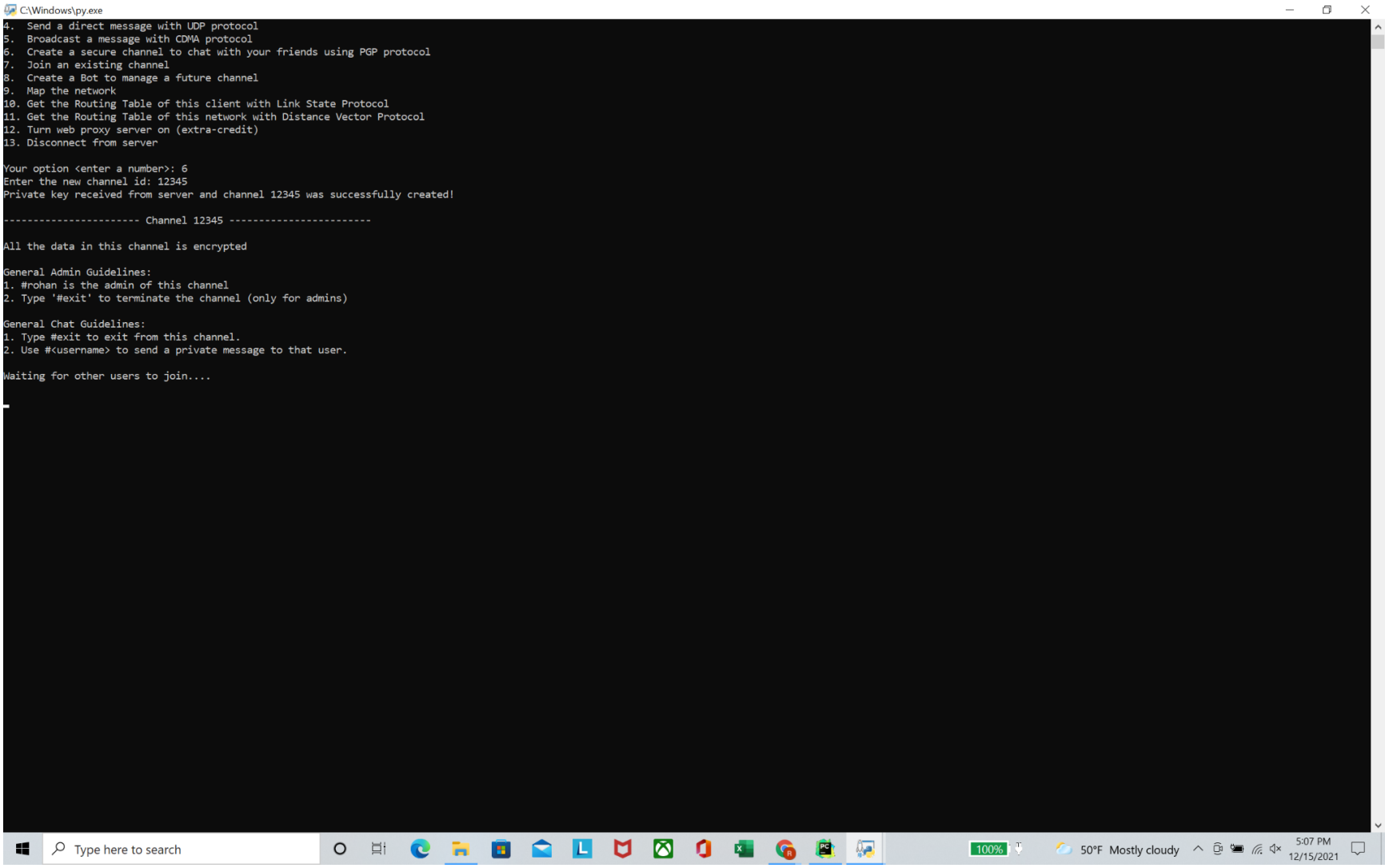
Distance Vector:



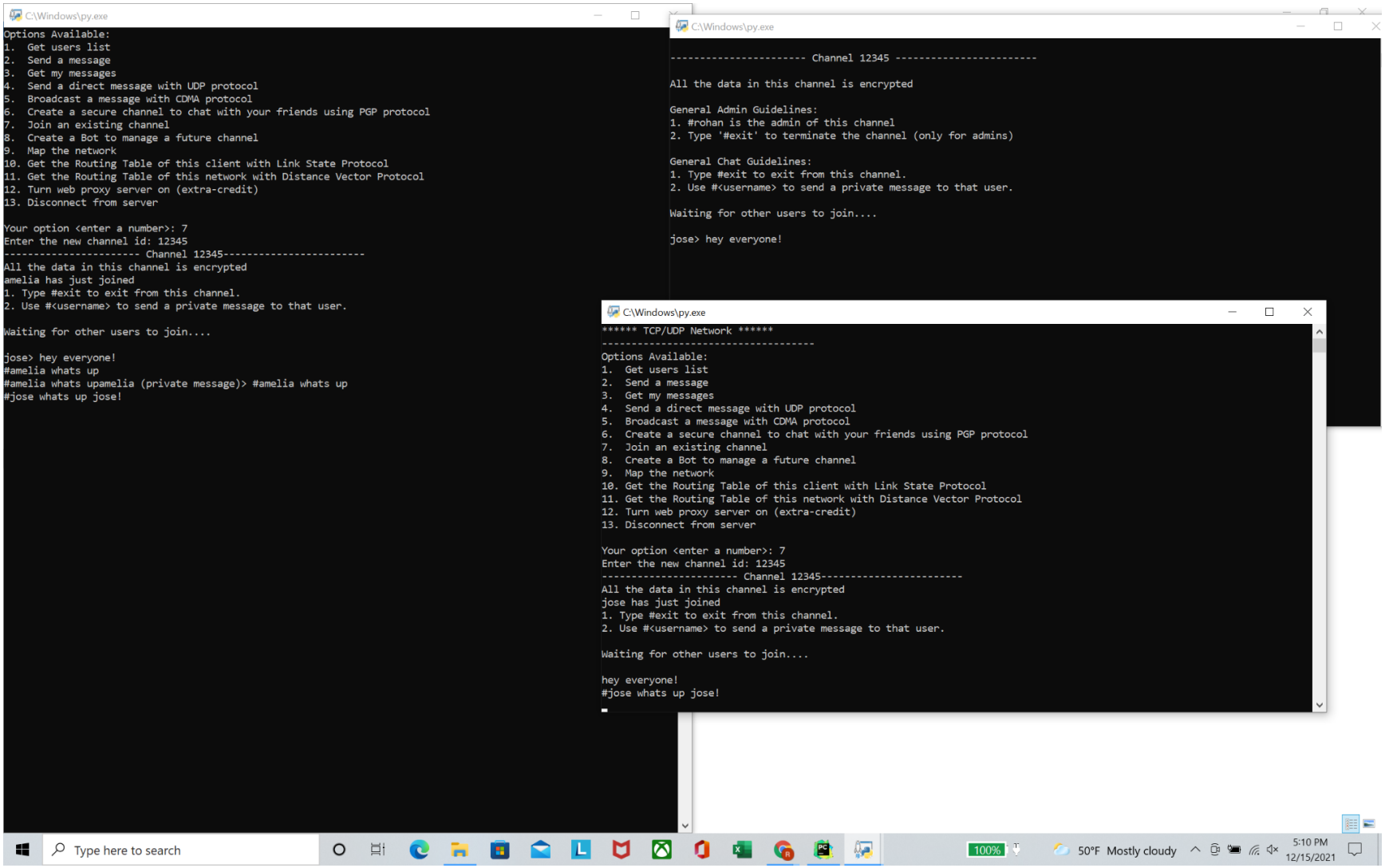
Link State:


File Edit View Navigate Code Refactor Run Tools Git Window Help computer-networks-fall21-rawatrohan123 - client.py
computer-networks-fall21-rawatrohan123 Project client client.py
bot.py client_helper.py client.py udp_socket.py chat_room.py client_handler.py menu.py
Run: server client
Client Name: rohan
Client ID: 57210
****** TCP/UDP Network ******
Options Available:
1. Get users list
2. Send a message
3. Get my messages
4. Send a direct message with UDP protocol
5. Broadcast a message with CDMA protocol
6. Create a secure channel to chat with your friends using PGP protocol
7. Join an existing channel
8. Create a Bot to manage a future channel
9. Map the network
10. Get the Routing Table of this client with Link State Protocol
11. Get the Routing Table of this network with Distance Vector Protocol
12. Turn web proxy server on (extra-credit)
13. Disconnect from server
Your option <enter a number>: 10
Routing table requested! Waiting for response...
Rohan Jose John Amelia
Rohan 0 15 10 999
Jose 15 0 30 11
John 10 30 0 25
Amelia 999 11 25 0
Routing table for Rohan (id: 50851) computed with Link State Protocol:
Destination Path Cost
Jose {Rohan, Jose} 15
John {Rohan, John} 10
Amelia {Rohan, John, Amelia} 35
Git Run TODO Problems Terminal Python Packages Python Console
Event Log
Type here to search
100%
50°F Mostly cloudy
4:38 PM
12/15/2021

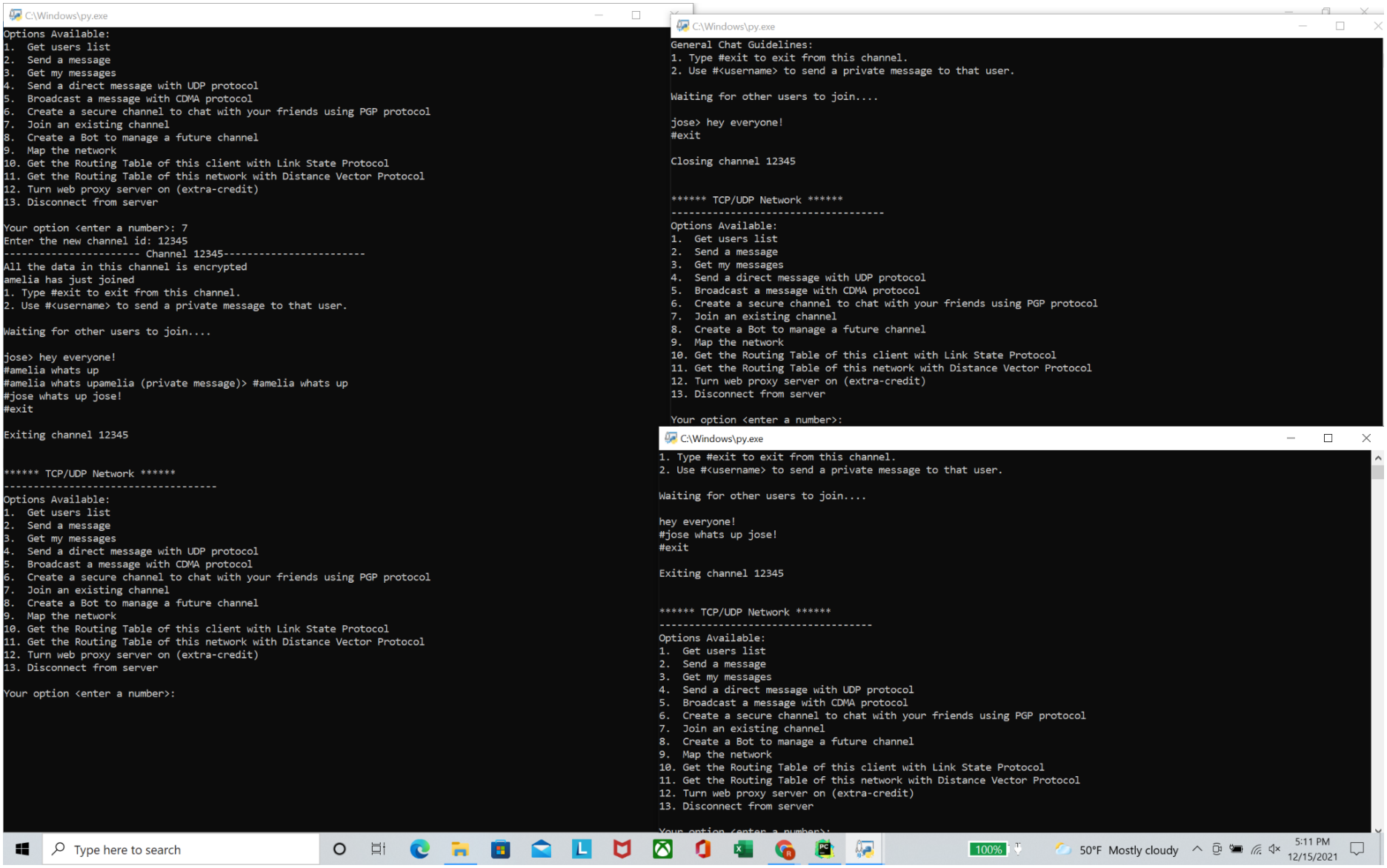
## Creating the Chat:



Joining the Chat + Sending Messages:
Also Demo of Private messages and Broadcast messages:



Exiting Channels:


C:\Windows\py.exe
Options Available:
1. Get users list
2. Send a message
3. Get my messages
4. Send a direct message with UDP protocol
5. Broadcast a message with CDMA protocol
6. Create a secure channel to chat with your friends using PGP protocol
7. Join an existing channel
8. Create a Bot to manage a future channel
9. Map the network
10. Get the Routing Table of this client with Link State Protocol
11. Get the Routing Table of this network with Distance Vector Protocol
12. Turn web proxy server on (extra-credit)
13. Disconnect from server
Your option <enter a number>: 7
Enter the new channel id: 12345
---------- Channel 12345----------
All the data in this channel is encrypted
amelia has just joined
1. Type #exit to exit from this channel.
2. Use #<username> to send a private message to that user.
Waiting for other users to join....
jose> hey everyone!
#amelia whats up
#amelia whats upamelia (private message)> #amelia whats up
#jose whats up jose!
#exit
Exiting channel 12345
****** TCP/UDP Network ******
Options Available:
1. Get users list
2. Send a message
3. Get my messages
4. Send a direct message with UDP protocol
5. Broadcast a message with CDMA protocol
6. Create a secure channel to chat with your friends using PGP protocol
7. Join an existing channel
8. Create a Bot to manage a future channel
9. Map the network
10. Get the Routing Table of this client with Link State Protocol
11. Get the Routing Table of this network with Distance Vector Protocol
12. Turn web proxy server on (extra-credit)
13. Disconnect from server
Your option <enter a number>:
C:\Windows\py.exe
General Chat Guidelines:
1. Type #exit to exit from this channel.
2. Use #<username> to send a private message to that user.
Waiting for other users to join....
jose> hey everyone!
#exit
Closing channel 12345
****** TCP/UDP Network ******
Options Available:
1. Get users list
2. Send a message
3. Get my messages
4. Send a direct message with UDP protocol
5. Broadcast a message with CDMA protocol
6. Create a secure channel to chat with your friends using PGP protocol
7. Join an existing channel
8. Create a Bot to manage a future channel
9. Map the network
10. Get the Routing Table of this client with Link State Protocol
11. Get the Routing Table of this network with Distance Vector Protocol
12. Turn web proxy server on (extra-credit)
13. Disconnect from server
Your option <enter a number>:
C:\Windows\py.exe
1. Type #exit to exit from this channel.
2. Use #<username> to send a private message to that user.
Waiting for other users to join....
hey everyone!
#jose whats up jose!
#exit
Exiting channel 12345
****** TCP/UDP Network ******
Options Available:
1. Get users list
2. Send a message
3. Get my messages
4. Send a direct message with UDP protocol
5. Broadcast a message with CDMA protocol
6. Create a secure channel to chat with your friends using PGP protocol
7. Join an existing channel
8. Create a Bot to manage a future channel
9. Map the network
10. Get the Routing Table of this client with Link State Protocol
11. Get the Routing Table of this network with Distance Vector Protocol
12. Turn web proxy server on (extra-credit)
13. Disconnect from server
Type here to search
100%
50°F Mostly cloudy
5:11 PM
12/15/2021