Mehdi Merali

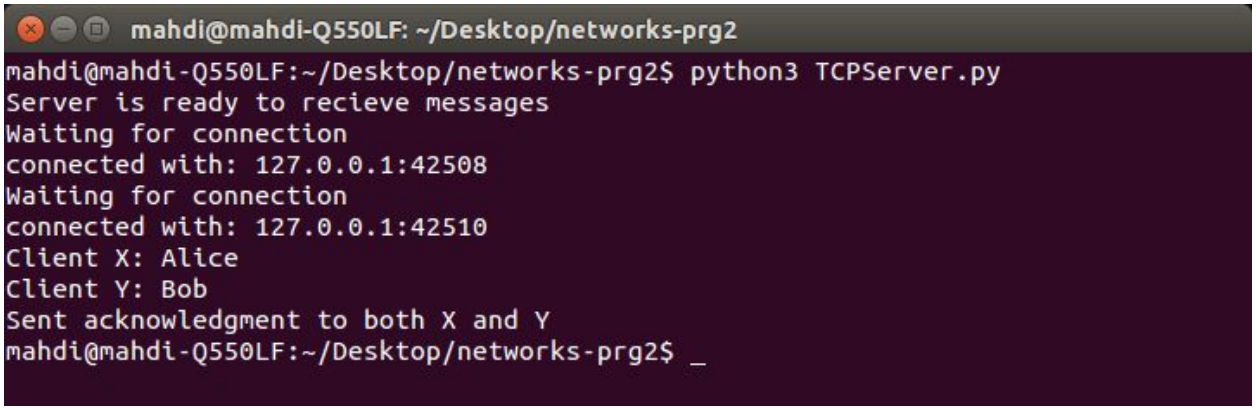
Program 2 Design Document

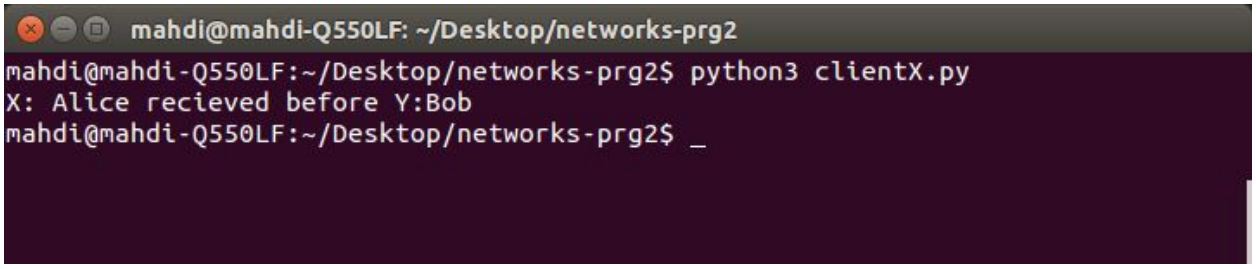
- Created client in Python because it was easy to implement and I wanted to play with pythons socket library.
- Server code will create server socket and loop through connections. Once 2 connections are received, it will send acknowledgements and print messages received. Then, it breaks its loop and terminates
    - Server includes extra print statements when it is connecting. I thought they were helpful so I left them in the program
- ClientX and ClientY code are essentially identical besides their messages
- Clients create socket, then send message to server and wait for ACK message. Once ACK is recieved, they print ACK message and set flag to break loop and terminate.
- Improvements:
    - Style could be changed to be cleaner
    - More exception handling
- NOTE: Must run program using Python 3 or will have compiler errors

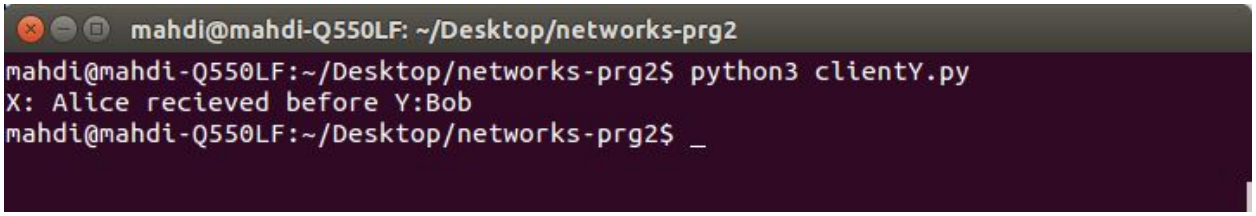
Running Program:

- Navigate to correct directory
- Start server in backgraound using *python3 TCPServer.py*
- Run clients using the following in any order:
    - *$ python3 clientX.py*
    - *$ python3 clientY.py*

Below is a sample output of the program


mahdi@mahdi-Q550LF: ~/Desktop/networks-prg2
mahdi@mahdi-Q550LF:~/Desktop/networks-prg2$ python3 TCPServer.py
Server is ready to recieve messages
Waiting for connection
connected with: 127.0.0.1:42508
Waiting for connection
connected with: 127.0.0.1:42510
Client X: Alice
Client Y: Bob
Sent acknowledgment to both X and Y
mahdi@mahdi-Q550LF:~/Desktop/networks-prg2$ _


mahdi@mahdi-Q550LF: ~/Desktop/networks-prg2
mahdi@mahdi-Q550LF:~/Desktop/networks-prg2$ python3 clientX.py
X: Alice recieved before Y:Bob
mahdi@mahdi-Q550LF:~/Desktop/networks-prg2$ _


mahdi@mahdi-Q550LF: ~/Desktop/networks-prg2
mahdi@mahdi-Q550LF:~/Desktop/networks-prg2$ python3 clientY.py
X: Alice recieved before Y:Bob
mahdi@mahdi-Q550LF:~/Desktop/networks-prg2$ _