Student ID: 7223791

Name: Jordan Cheong

Subject: CSCI301

Assignment 3

Packages Required:

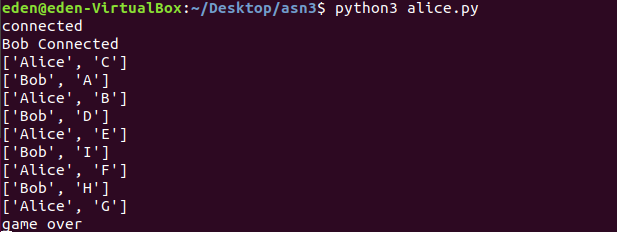
1. pubnub

Included files:

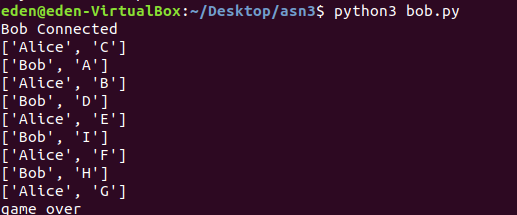
1. alice.py
2. bob.py
3. block0.json
   1. results from one game, block1-block9.json have also been included

Expected outcomes of python scripts:

1. When alice.py is run with command “python3 alice.py”, “connected” will be printed, and the game is ready to begin when Bob connects.
   1. Note that alice.py should be ran before bob.py
2. When Bob.py is then run with command “python3 bob.py”, the game will begin. Bob connected will be displayed on both terminals before all moves will be printed. Finally, the message “game over” will be printed in the terminal, and all blocks will have been generated in the same folder as alice.py and bob.py
   1. Alice and bob.py can be run in different folders, but both of them must have their own block0.json



*Example running of alice.py*

**

*Example running of bob.py*

1. Note that a win condition has not been added, as the brief states that the game will continue until all spaces in the board are taken

Code:

There are 4 main aspects to the code

1. The TicTacToe game, with no win condition. Instead, moves are chosen and spaces are removed from a predefined list.
2. Writing the blocks, including hashing
3. Messaging and Handling Messages
   1. Most of the framework of messaging is handled by PubNub
   2. Handling messages is a matter of parsing the blocks and dealing with information
4. Verifying the blocks sent

Of these, the tictactoe game is relatively simple. It just processes enemy moves and your own moves, removing spaces from the move pool(A list) until the game is over.

Hashing the block is performed with json dumps, and hashlib to encode the code. A nonce is added to lower the hash value as well.

Verification of the blocks is simply by checking the hash value against a new hash value generated by hashing the previous block.