Name: Gunin Jain

Roll. No : IMT2019034

GAME SHOW USING PYTHON-3

ASSUMPTIONS

1. There are exactly 3 players.
2. The host and all the 3 players are connected on the same local area network.
3. No two players press the buzzer at the same time.
4. The game ends with no winner if all the questions are exhausted.

DESCRIPTION OF THE CODE

Server side

1. We create a server socket and bind it to the server's IP address and an unused port.
2. We listen for connections.
3. We accept connections and store the player sockets in a list till the total number of connections are 3.
4. After connecting with all the players the host starts the game.
5. Each time a question is sent to all the players and they are given 20 seconds to press the buzzer. The select function receives all the buzzers from players within a time of 20 seconds. If no one presses the buzzer, all the players are sent a message that no one has pressed the buzzer.
6. The first player to send the buzzer is sent a message to answer the question within 20 seconds .If the player answers the question within 20 seconds, the server checks if the answer matches the correct answer or not .Corresponding reply is sent to that player. If the player doesn't answer the question, the host sends a timeout message to that player and moves to the next question.
7. Points are given accordingly for correct and wrong answers. The scores list is checked if there is any player with points greater than or equal to 5.If yes, that player is declared the winner and others are sent a consolation message. Game ends at that point.
8. To avoid sending back to back messages to the players in quick succession, a sleep of 5s is given.

Client side

1. We create a client socket and connect to the server using the server's IP address and port number.
2. When we receive a question, we call the Thread() function which runs the send_buzzer () function for at most 20 seconds as a separate thread.

3. Send_buzzer() ensures that it stays there till the thread returns by doing time.sleep() for the remainder of the time(which is left after giving the input).This is done to ensure that all players receive the questions at the same time, irrespective of who has answered the previous question.
4. If we are the first to press the buzzer, we get a message from the server to answer the question within 20 seconds. Again, we call the Thread() function to run the send_ans() function for at most 20 seconds. send_ans() also has a similar functionality as send_buzzer().
5. Messages sent by server (other than questions) are handled accordingly.
6. When the game ends, the client receives a message from server and closes the connection

Instructions to run the game

1. Install python 3.7 from www.python.org
2. Open the command prompt (Windows)/terminal (Unix and Mac OS X) and go to the respective parent directory of the server.py and client.py files.
3. Open one command prompt for server and 3 command prompts for clients if you are using a single machine.
4. If you are using different end systems on the same LAN, remember to change the server IP address in the client.py to the local IP of your server.
5. Type the command "python server.py" and "python client.py" and press ENTER key to run the server and client respectively.

Instructions to the play the game

1. To press the buzzer on receiving a question, press "y" and press ENTER.
2. To answer the question type the answer using keyboard and press ENTER.