Practical 12

AIM:

b) Implement chat client server using TCP/UDP sockets.

ALGORITHM:

**Chat Server**

1. **Start the Server:**

   o Create a socket (like a phone line).

   o Bind it to a specific address and port (set your phone number).

   o Listen for incoming connections (wait for calls).

2. **Accept Connections:**

   o When a new client connects:

      ▪ Add the client to a list of connected clients.

      ▪ Start a new process to talk to this client (like talking on a different line).

3. **Receive Messages:**

   o For each connected client:

      ▪ Keep checking for new messages.

      ▪ When a message is received, show it on the server screen.

      ▪ Send this message to all other clients (share the message with everyone).

4. **Handle Disconnections:**

   o If a client disconnects:

      ▪ Remove that client from the list.

      ▪ Stop talking to that client.

5. **Keep Running:**

   o Repeat the process until you stop the server.

**<u>Chat Client</u>**

1. **Connect to the Server:**

   o Create a socket (your phone).

   o Connect it to the server's address and port (call the server).

2. **Receive Messages:**

   o Start a process to listen for messages from the server:

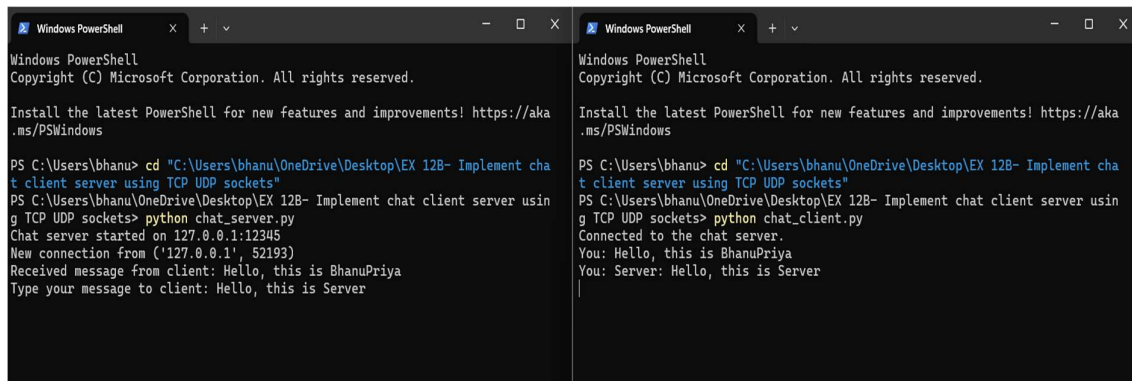     ▪ Whenever a message arrives, show it on your screen.

3. **Send Messages:**

   o Keep asking the user for new messages:

     ▪ If the user types a message, send it to the server.

     ▪ If the user types 'exit', disconnect from the server and close the client.

4. **Keep Running:**

   o Repeat the process until the user decides to exit.

<u>OUTPUT:</u>