SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

FACULTY OF ENGINEERING AND TECHNOLOGY

SCHOOL OF COMPUTING

DEPARTMENT OF COMPUTING TECHNOLOGIES

21CSC203P ADVANCED PROGRAMMING PRACTICE

Week 12- Tutorial Assignment

Implement Python program - TCP/UDP program using Sockets

1. Develop a simple Python program of TCP, client that can connect to the server and client can send a **"Hello, Server!"** message to the server.
2. Develop a Python program that allows the TCP client to send a list of numbers to the server. The server should calculate and return the sum of the numbers to the client.
3. Create a Python UDP client that sends a "UDP Message" packet to a UDP server. Demonstrate the sending and receiving of the packet.
4. Create a Python UDP client that sends a random number to the UDP server. The server should check if the number is even or odd and send the result back to the client.
5. Write a Python program to create a UDP server that listens on port 54321. Ensure the server can receive UDP packets from clients.
6. Extend the UDP server to respond to the client's "UDP Message" packet with an acknowledgment message. Provide the code for the server-client interaction**.**
7. Implement a Python program that calculates and displays the time taken for a TCP client to connect to the server and receive a response. Measure the time elapsed in seconds.
8. Create a TCP server that echoes back any message it receives from a client. Develop a Python client to send messages to the server and display the echoed response.
9. Develop a simple Python program that sends a small text file from a TCP client to a TCP server. Confirm that the file is received and saved correctly.
10. Write a Python program to receive UDP packets and display their content. Simulate sending UDP packets from a separate client program.