

## Assignment 9

**Name: Hitesh Tolani**

**Roll no: 73**

**Class: SY-AIDS-A**

**Title: Write a program using TCP socket for wired network for following:**

- 1. Say Hello to Each other**
- 2. File transfer**

### **Theory:**

Transmission Control Protocol (TCP) sockets form the backbone of reliable communication in networked systems. A TCP socket establishes a connection-oriented communication channel between two endpoints, providing a reliable, ordered, and error-checked stream of bytes. At the heart of TCP's functionality lies its three-way handshake mechanism, where a connection is established through a series of SYN (synchronize), SYN-ACK (synchronize-acknowledge), and ACK (acknowledge) messages exchanged between the client and server.

Once the connection is established, data transmission occurs in a stream-oriented manner, ensuring that data arrives in the same order it was sent and that lost or corrupted packets are retransmitted. TCP also implements flow control and congestion control mechanisms to manage the rate of data transmission and prevent network congestion. Through these features, TCP sockets facilitate robust, efficient, and dependable communication between networked applications, making them a cornerstone of modern internet protocols and enabling the seamless exchange of data across diverse computing environments.

### **Code:**

**hello\_server.py**

```
import socket

s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)

port = 123

s.bind(('', port))

s.listen(5)

while True:
    c, addr = s.accept()
```

```
c_msg = c.recv(1024).decode()

print(f"Received {c_msg} from client")

c.send("Hello".encode())

c.close()

break
```

### hello\_client.py

```
import socket

s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)

port = 123

s.connect(("127.0.0.1", port))

s.send("Hello".encode())

s_msg = s.recv(1024).decode()

print(f"Received {s_msg} from server")
s.close()
```

### file\_server.py

```
import socket

s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)

port = 123

s.bind(('', port))

s.listen(5)

while True:
    c, addr = s.accept()

    data = c.recv(1024).decode()

    c_msg = 'File transferred successfully!'
    if data:
```

```
        print("File received from client")
        with open("output.txt","w") as file:
            file.write(data)
    else:
        c_msg = "File transfer failed no data received"

    c.send(c_msg.encode())

    c.close()

    break
```

### **file\_client.py**

```
import socket
file_path = input("Enter path of the file\n")

s = socket.socket(socket.AF_INET,socket.SOCK_STREAM)

port = 123

s.connect(("127.0.0.1",port))

data = ''
with open(file_path,"r") as file:
    data = file.read()

if data:
    s.send(data.encode())
else:
    print("Empty File!")

s_msg = s.recv(1024).decode()

print(s_msg)
s.close()
```

### **Output:**

The image displays four terminal windows arranged in a 2x2 grid, showing the execution of Python programs for TCP/IP communication. The top-left window shows a client program (hello\_client.py) receiving a 'Hello' message from a server. The top-right window shows a server program (hello\_server.py) receiving a 'Hello' message from a client. The bottom-left window shows a client program (file\_client.py) receiving a file from a server. The bottom-right window shows a server program (file\_server.py) receiving a file path from a client and successfully transferring the file.

```
C:\Windows\system32\cmd.exe
P:\College\Python\4th Lab\Assignment 5\python hello_client.py
Received Hello from server
P:\College\Python\4th Lab\Assignment 5_

C:\Windows\system32\cmd.exe
P:\College\Python\4th Lab\Assignment 5\python hello_server.py
Received Hello from client
P:\College\Python\4th Lab\Assignment 5_

C:\Windows\system32\cmd.exe
P:\College\Python\4th Lab\Assignment 5\python file_server.py
File received from client
P:\College\Python\4th Lab\Assignment 5_

C:\Windows\system32\cmd.exe
P:\College\Python\4th Lab\Assignment 5\python file_client.py
Enter path of the file:
test.txt
File transferred successfully
P:\College\Python\4th Lab\Assignment 5_
```

## Conclusion:

In this assignment, we have successfully learned about TCP IP protocol and implement a program in python for sending hello message and transferring a file.