

27/12/24

Experiment - 15

TCP/IP

- a) using ~~UDP~~ socket, write a client-server program to make client sending the file name and the server to send back the contents of the requested file if present.

clientTCP.py

```
from socket import *
serverName = '127.0.0.1'
serverPort = 12000
clientSocket = socket(AF_INET, SOCK_STREAM)
clientSocket.connect((serverName,
                      serverPort))
sentence = input("Enter file name:")
clientSocket.send(sentence.encode())
fileContents = clientSocket.recv(1024).decode()
print(fileContents)
clientSocket.close()
```

serverTCP.py

```
from socket import *
serverName = "127.0.0.1"
serverPort = 12000
serverSocket = socket(AF_INET, SOCK_STREAM)
serverSocket.bind((serverName, serverPort))
serverSocket.listen(1)
while 1:
    print("The server is ready to receive")
    connectionSocket, addr = serverSocket.accept()
    sentence = connectionSocket.recv(1024).decode()
    file = open(sentence, "r")
```

```
file = open(sentence, "r")
l = file.read(1024)
connectionSocket.send(l.encode())
print('In socket contents of ' +
      sentence)
file.close()
connectionSocket.close()
```

~~tcp.py~~ TCP.py [client]

```
from socket import *
serverName = '127.0.0.1'
serverPort = 12000
serverSocket = socket(AF_INET, SOCK_STREAM)
serverSocket.bind((serverName,
                  serverPort))
serverSocket.listen(1)
while 1:
    print("The server is ready to
          receive")
    connectionSocket, addr = serverSocket.
                             accept()
    sentence = connectionSocket.recv(1024).
               decode()
    file = open(sentence, "r")
    l = file.read(1024)
    connectionSocket.send(l.encode())
    print('In sent contents of ' + sentence)
    file.close()
    connectionSocket.close()
```

~~quiz~~ TCP.py [Server side]

The server is ready to receive.