

24-8-23

Date _____
Page _____

Q) C/C++

(1) Using TCP/IP sockets, write a client-server program to make a client making a client sending a file name and the server to send back the content of the requested file if present.

ClientTCP.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()
```

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")  
data = file.read(1024)  
connection socket.send(data.encode())  
print("\n Sent content of " + sentence)  
file.close()  
connection socket.close()
```

Output

Python 3.7.3 (v3.7.3:ef4e612, Mar 25 2019, 21:26:53)
Type "help", "copyright", "credits" or "license()" for more info

The server is ready to receive

When we run client TCP.py

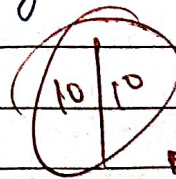
Enter file name: server TCP.py
from server:

(The files from server TCP.py will be
and displayed ~~at~~ here)

In server TCP.py

The server is ready to receive

Sent content of server TCP.py



8.30

Q) Using UDP sockets write client-server program to make client sending the file name and server to send back the contents of the requested file if present.

A) client UDP.py

```
from socket import *
serverName = "127.0.0.1"
serverPort = 12000
clientSocket = socket(AF_INET, SOCK_DGRAM)
sentence = input("\n Enter file name: ")
clientSocket.sendto(bytes(sentence, "utf-8"), (serverName, serverPort))
fileContent, serverAddress = clientSocket.recvfrom(2048)

print("\n Reply from server: \n")
print(fileContent.decode("utf-8"))
# for i in fileContent:
#     print(i, end=" ")
clientSocket.close()
clientSocket.close()
```

2. server UDP.py

```
from socket import *
serverPort = 12000
serverSocket = socket(AF_INET, SOCK_DGRAM)
serverSocket.bind(("127.0.0.1", serverPort))
print("The server is ready to receive")
while 1:
```



```

sentence = client Address = serverSocket.recvfrom(2048)
sentence = sentence.decode("utf-8")
file = open(sentence, "r")
con = file.read(2048)

```

```

serverSocket.sendto (bytes (con, "utf" ),
client Address)

```

```

printf ("In sent content of", end = '')
print (sentence)
file.close()

```

Output :

Run serverUDP.py

The server is ready to receive

Run clientUDP.py : Enter file name: serverUDP.py
Reply from server

In serverUDP.py

The server is ready to receive

10/10

28/8/28

file 3 - 7