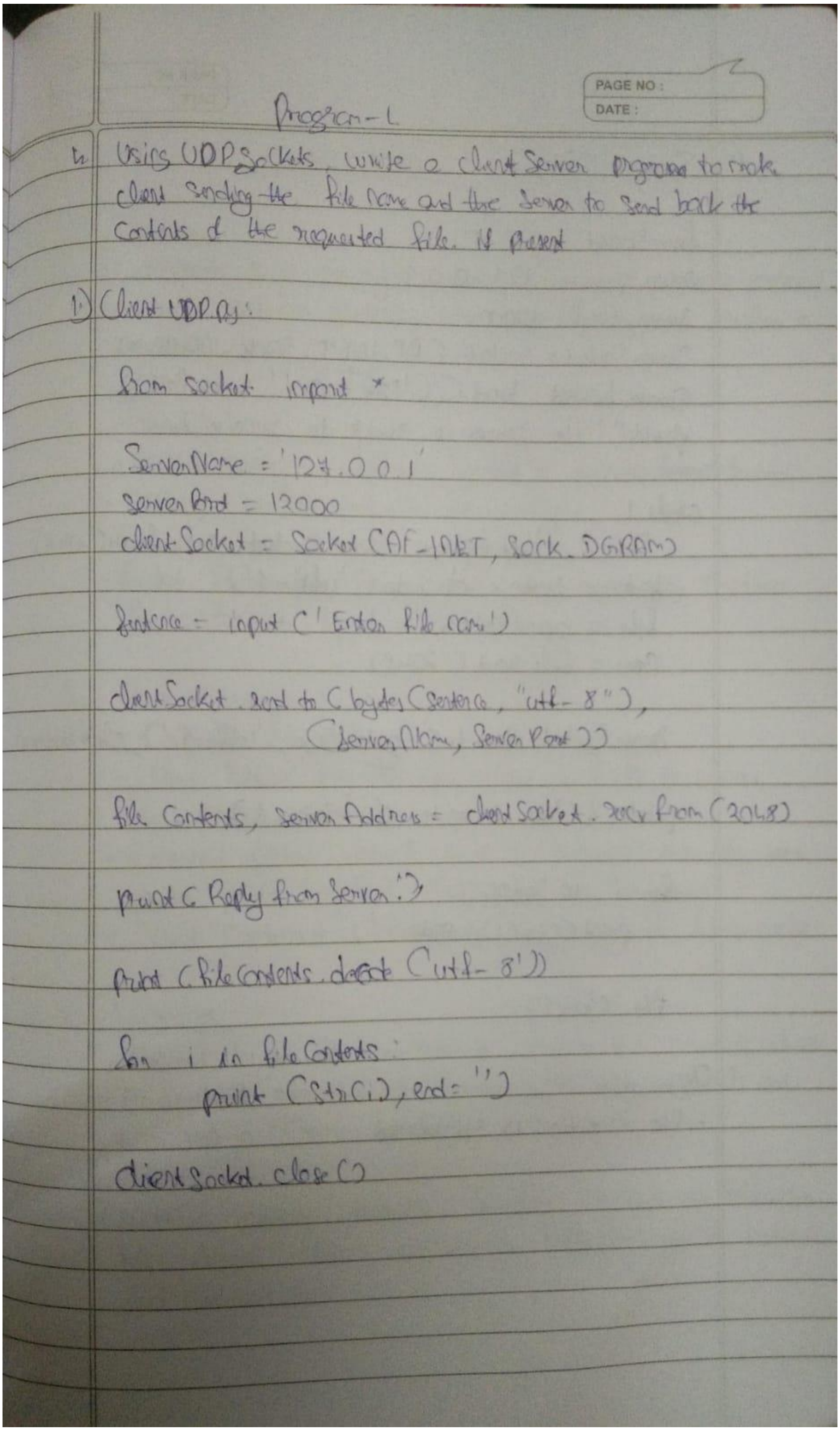
LABORATORY PROGRAM – 4

Using UDP sockets, 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.



PAGE NO :

DATE :

Program-4

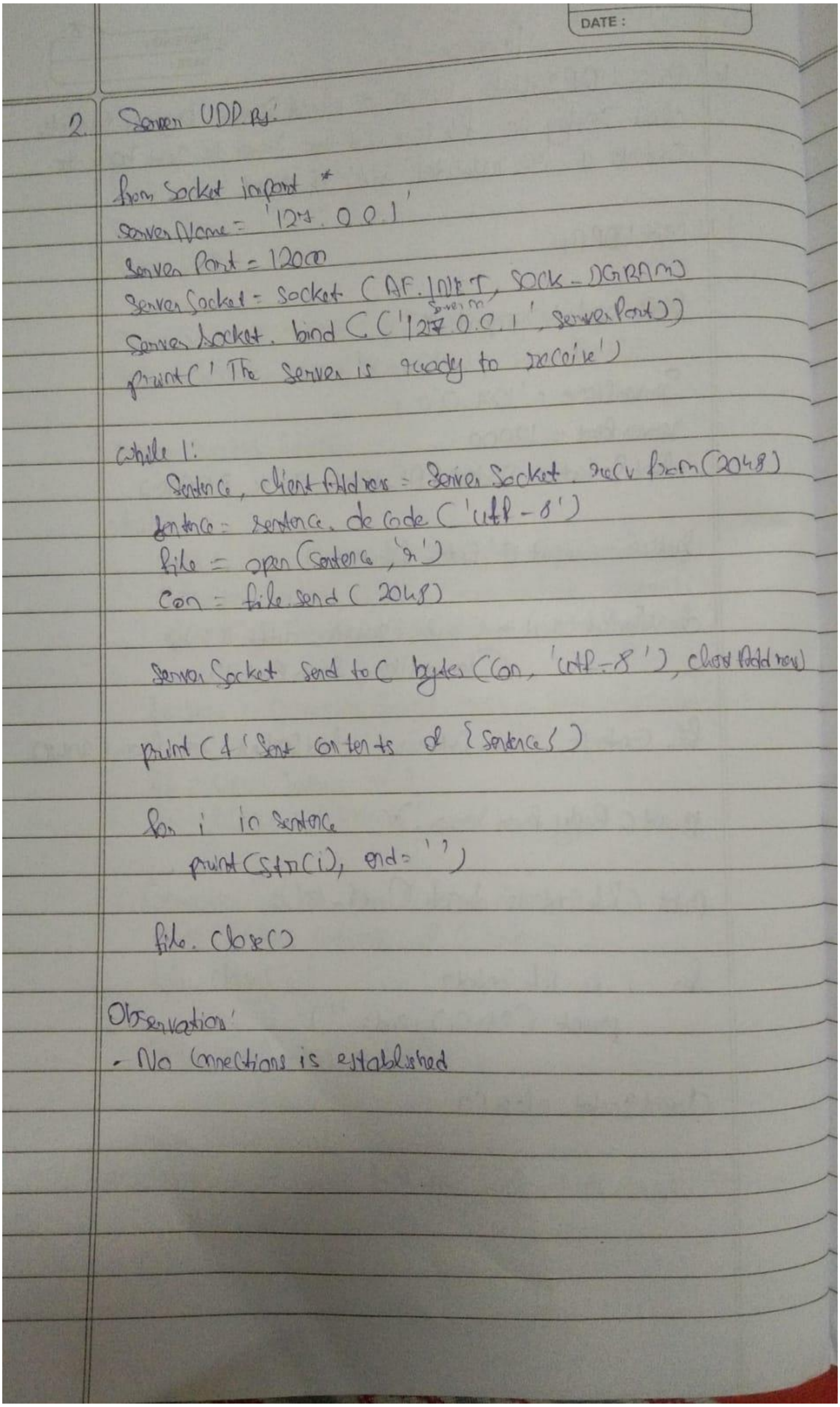
4. Using UDP Sockets, write a client Server program to make client sending the file name and the Server to send back the contents of the requested file. if present

1) Client UDP.py:

```
from socket import *

ServerName = '127.0.0.1'
serverPort = 12000
clientSocket = Socket (AF_INET, SOCK_DGRAM)

sentence = input ('Enter file name')

clientSocket.sendto (bytes (sentence, "utf-8"),
                     (ServerName, ServerPort))

file Contents, serverAddress = clientSocket.recvfrom (2048)

print ('Reply from Server:')

print (fileContents.decode ('utf-8'))

for i in fileContents:
        print (str(i), end='')

clientSocket.close ()
```

Code: ClientUDP.py

```
from socket import *
serverName = "127.0.0.1"  # Server address (localhost)
serverPort = 12000  # Port number where the server listens

# Create UDP socket
clientSocket = socket(AF_INET, SOCK_DGRAM)

# Ask user for file name to request
sentence = input("Enter file name: ")

# Send the file name to the server using UDP
clientSocket.sendto(sentence.encode("utf-8"), (serverName, serverPort))

# Receive file contents from the server
fileContents, serverAddress = clientSocket.recvfrom(2048)

# Print the file contents received from the server
print("From Server:", fileContents.decode())

# Close the UDP socket
clientSocket.close()
```



DATE :

2. Server UDP.py:

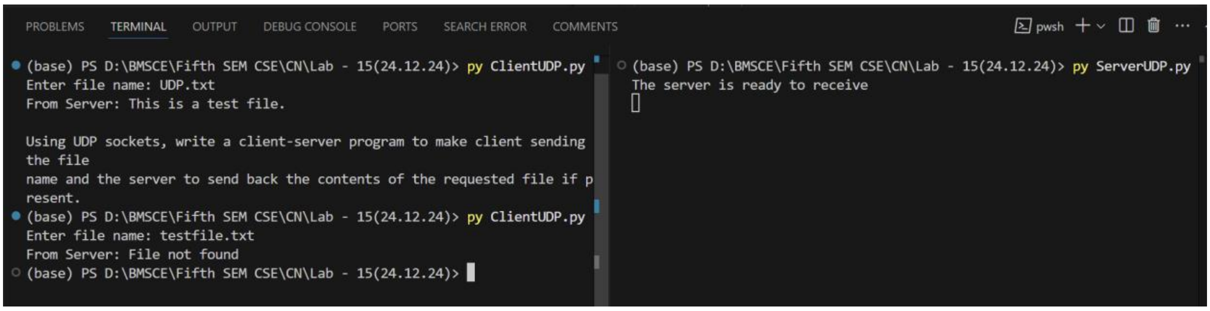
```
from Socket import *
serverName = '127.0.0.1'
Server Port = 12000
Server Socket = socket (AF.INET, SOCK_DGRAM)
                            serverN
Server Socket. bind (('127.0.0.1', serverPort))
print(' The server is ready to receive')

while 1:
   Sentence, client Address = Server Socket. recv from(2048)
  Sentence = sentence. decode ('utf-8')
  file = open(Sentence, 'r')
  Con = file.send ( 2048)

  Server Socket send to( bytes (Con, 'utf-8'), client Address)

  print (f 'Sent contents of {Sentence} )

  for i in Sentence
     print(str(i), end='')

  file. Close()
```

Observation:
- No Connections is established

Code: ServerUDP.py

```
from socket import *
serverPort = 12000  # Port number to listen on

# Create UDP socket
serverSocket = socket(AF_INET, SOCK_DGRAM)
serverSocket.bind(("127.0.0.1", serverPort))  # Bind the socket to the server address and port

print("The server is ready to receive")

while True:
    # Receive file name from the client
    sentence, clientAddress = serverSocket.recvfrom(2048)

    # Try opening the file
    try:
        file = open(sentence.decode(), "r")  # Open file in read mode
        fileContents = file.read(2048)  # Read file content (up to 2048 bytes)
        serverSocket.sendto(fileContents.encode("utf-8"), clientAddress)  # Send file contents to
client
        file.close()
    except FileNotFoundError:
        # Send error message if file not found
        serverSocket.sendto("File not found".encode("utf-8"), clientAddress)
```

Output



```
(base) PS D:\BMSCE\Fifth SEM CSE\CN\Lab - 15(24.12.24)> py ClientUDP.py
Enter file name: UDP.txt
From Server: This is a test file.

Using UDP sockets, 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 p
resent.
(base) PS D:\BMSCE\Fifth SEM CSE\CN\Lab - 15(24.12.24)> py ClientUDP.py
Enter file name: testfile.txt
From Server: File not found
(base) PS D:\BMSCE\Fifth SEM CSE\CN\Lab - 15(24.12.24)>
```

```
(base) PS D:\BMSCE\Fifth SEM CSE\CN\Lab - 15(24.12.24)> py ServerUDP.py
The server is ready to receive
```