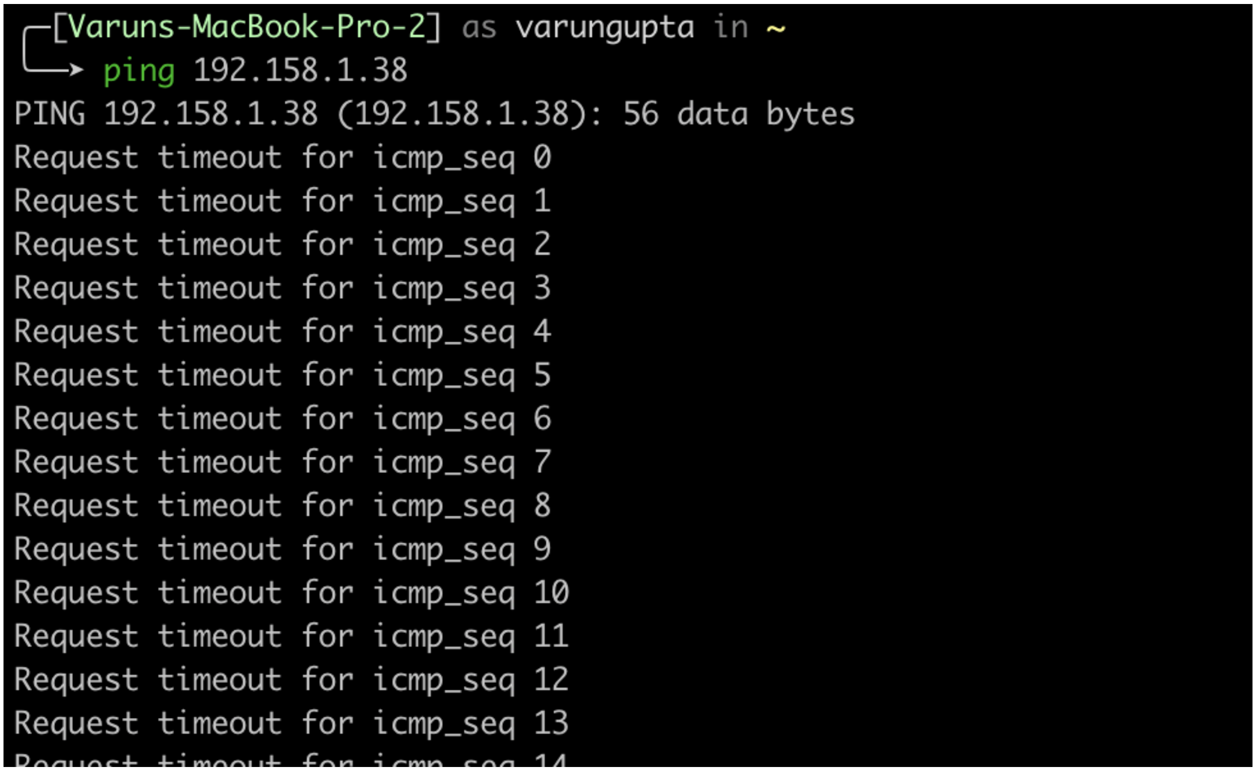
Computer Network

Name: Varun Gupta
Reg No.: 21BCE0361
Subject: computer network

Aim:

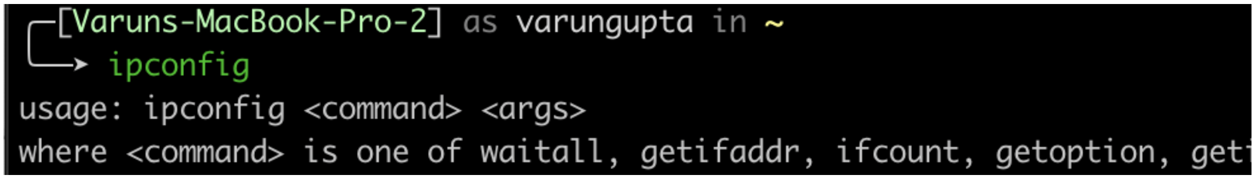
a)To study the Basic Network Commands
b)To Study all networking hardware and functions

1.Ping



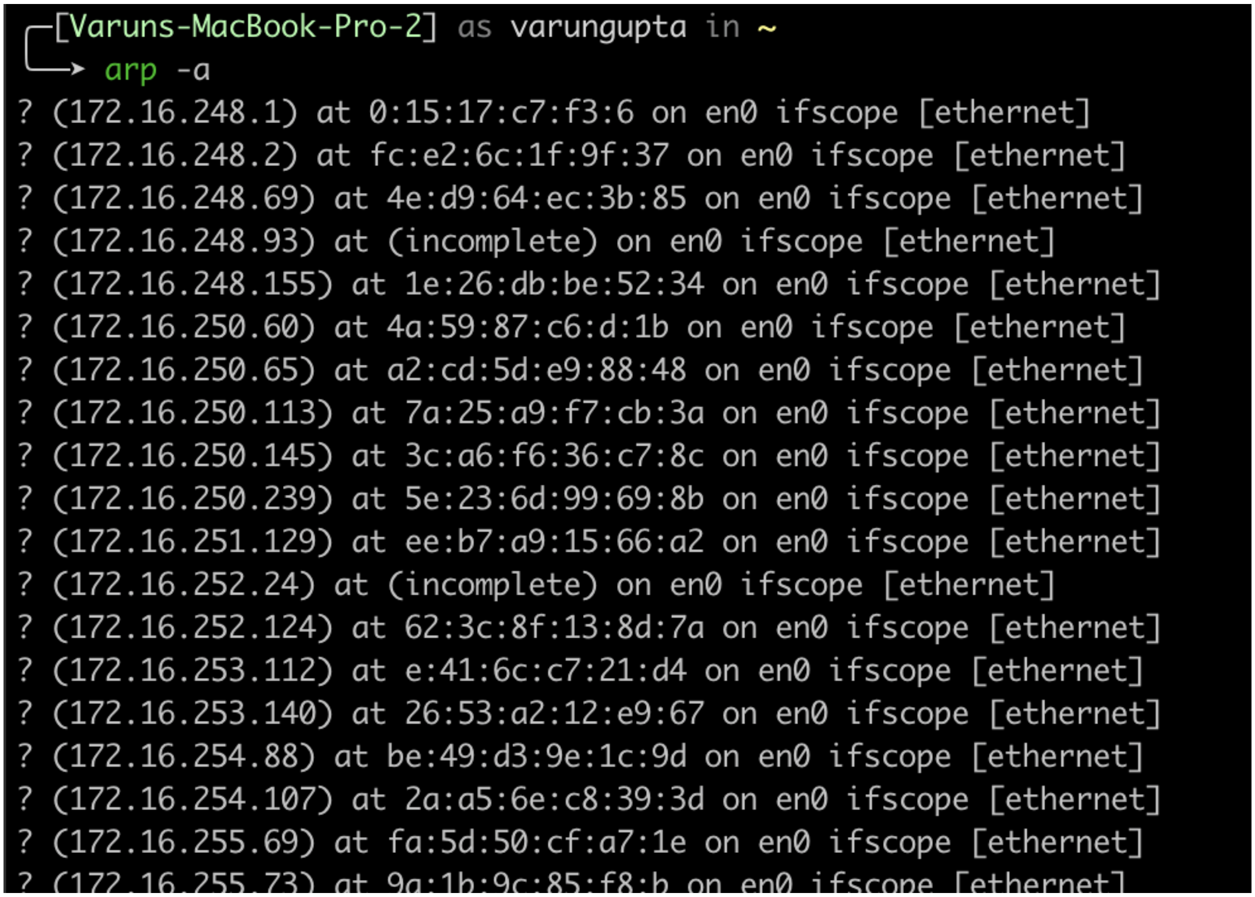
```
┌[Varuns-MacBook-Pro-2] as varungupta in ~
└─➤ ping 192.158.1.38
PING 192.158.1.38 (192.158.1.38): 56 data bytes
Request timeout for icmp_seq 0
Request timeout for icmp_seq 1
Request timeout for icmp_seq 2
Request timeout for icmp_seq 3
Request timeout for icmp_seq 4
Request timeout for icmp_seq 5
Request timeout for icmp_seq 6
Request timeout for icmp_seq 7
Request timeout for icmp_seq 8
Request timeout for icmp_seq 9
Request timeout for icmp_seq 10
Request timeout for icmp_seq 11
Request timeout for icmp_seq 12
Request timeout for icmp_seq 13
Request timeout for icmp_seq 14
```

2. ipconfig



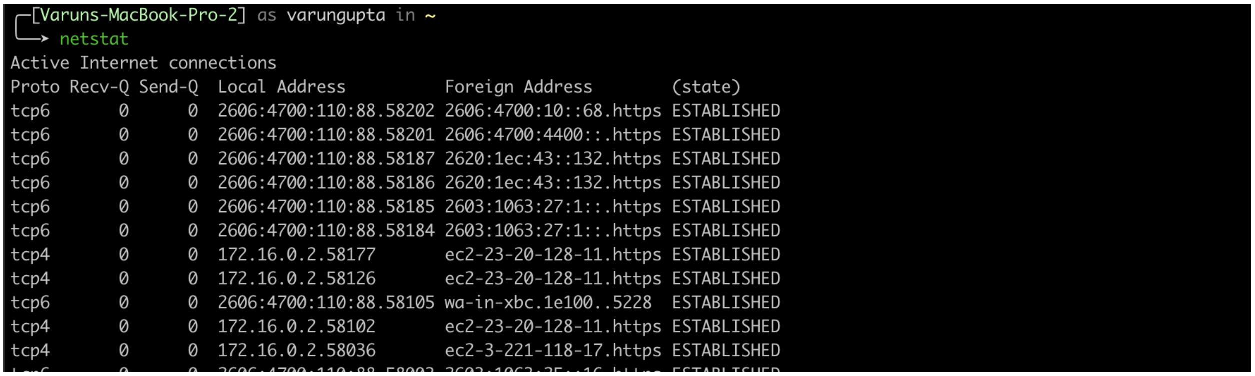
```
┌[Varuns-MacBook-Pro-2] as varungupta in ~
└─➤ ipconfig
usage: ipconfig <command> <args>
where <command> is one of waitall, getifaddr, ifcount, getoption, get
```

3.Arp



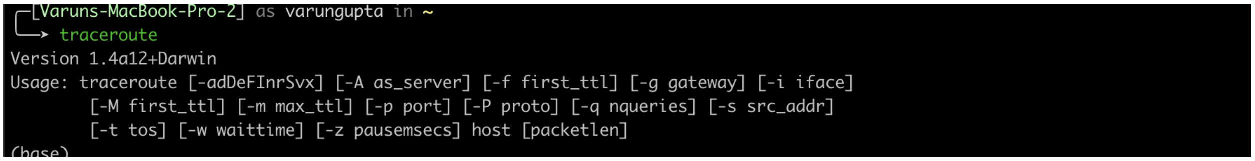
```
┌[Varuns-MacBook-Pro-2] as varungupta in ~
└─➤ arp -a
? (172.16.248.1) at 0:15:17:c7:f3:6 on en0 ifscope [ethernet]
? (172.16.248.2) at fc:e2:6c:1f:9f:37 on en0 ifscope [ethernet]
? (172.16.248.69) at 4e:d9:64:ec:3b:85 on en0 ifscope [ethernet]
? (172.16.248.93) at (incomplete) on en0 ifscope [ethernet]
? (172.16.248.155) at 1e:26:db:be:52:34 on en0 ifscope [ethernet]
? (172.16.250.60) at 4a:59:87:c6:d:1b on en0 ifscope [ethernet]
? (172.16.250.65) at a2:cd:5d:e9:88:48 on en0 ifscope [ethernet]
? (172.16.250.113) at 7a:25:a9:f7:cb:3a on en0 ifscope [ethernet]
? (172.16.250.145) at 3c:a6:f6:36:c7:8c on en0 ifscope [ethernet]
? (172.16.250.239) at 5e:23:6d:99:69:8b on en0 ifscope [ethernet]
? (172.16.251.129) at ee:b7:a9:15:66:a2 on en0 ifscope [ethernet]
? (172.16.252.24) at (incomplete) on en0 ifscope [ethernet]
? (172.16.252.124) at 62:3c:8f:13:8d:7a on en0 ifscope [ethernet]
? (172.16.253.112) at e:41:6c:c7:21:d4 on en0 ifscope [ethernet]
? (172.16.253.140) at 26:53:a2:12:e9:67 on en0 ifscope [ethernet]
? (172.16.254.88) at be:49:d3:9e:1c:9d on en0 ifscope [ethernet]
? (172.16.254.107) at 2a:a5:6e:c8:39:3d on en0 ifscope [ethernet]
? (172.16.255.69) at fa:5d:50:cf:a7:1e on en0 ifscope [ethernet]
? (172.16.255.73) at 9a:1b:9c:85:f8:b on en0 ifscope [ethernet]
```

4.Netstat



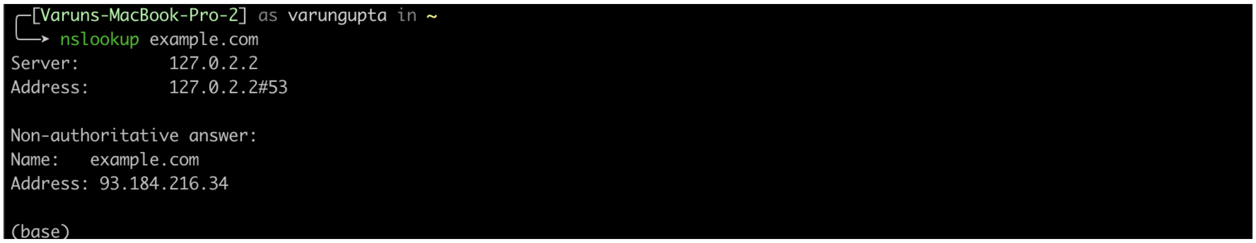
```
┌─[Varuns-MacBook-Pro-2] as varungupta in ~
└──→ netstat
Active Internet connections
Proto Recv-Q Send-Q  Local Address          Foreign Address        (state)
tcp6       0      0  2606:4700:110:88.58202 2606:4700:10::68.https ESTABLISHED
tcp6       0      0  2606:4700:110:88.58201 2606:4700:4400::.https ESTABLISHED
tcp6       0      0  2606:4700:110:88.58187 2620:1ec:43::132.https ESTABLISHED
tcp6       0      0  2606:4700:110:88.58186 2620:1ec:43::132.https ESTABLISHED
tcp6       0      0  2606:4700:110:88.58185 2603:1063:27:1::.https ESTABLISHED
tcp6       0      0  2606:4700:110:88.58184 2603:1063:27:1::.https ESTABLISHED
tcp4       0      0  172.16.0.2.58177       ec2-23-20-128-11.https ESTABLISHED
tcp4       0      0  172.16.0.2.58126       ec2-23-20-128-11.https ESTABLISHED
tcp6       0      0  2606:4700:110:88.58105 wa-in-xbc.1e100..5228  ESTABLISHED
tcp4       0      0  172.16.0.2.58102       ec2-23-20-128-11.https ESTABLISHED
tcp4       0      0  172.16.0.2.58036       ec2-3-221-118-17.https ESTABLISHED
tcp6       0      0  2606:4700:110:88.58003 2603:1063:35::16.https ESTABLISHED
```

5.Tracert



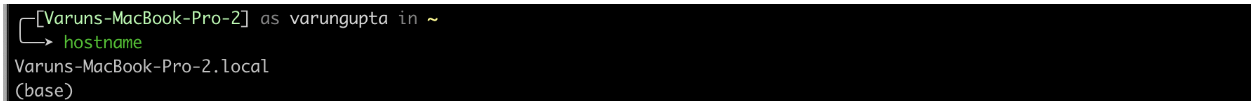
```
┌─[Varuns-MacBook-Pro-2] as varungupta in ~
└──→ traceroute
Version 1.4a12+Darwin
Usage: traceroute [-adDeFInrSvx] [-A as_server] [-f first_ttl] [-g gateway] [-i iface]
        [-M first_ttl] [-m max_ttl] [-p port] [-P proto] [-q nqueries] [-s src_addr]
        [-t tos] [-w waittime] [-z pausemsecs] host [packetlen]
(base)
```

6.NSLOOKUP



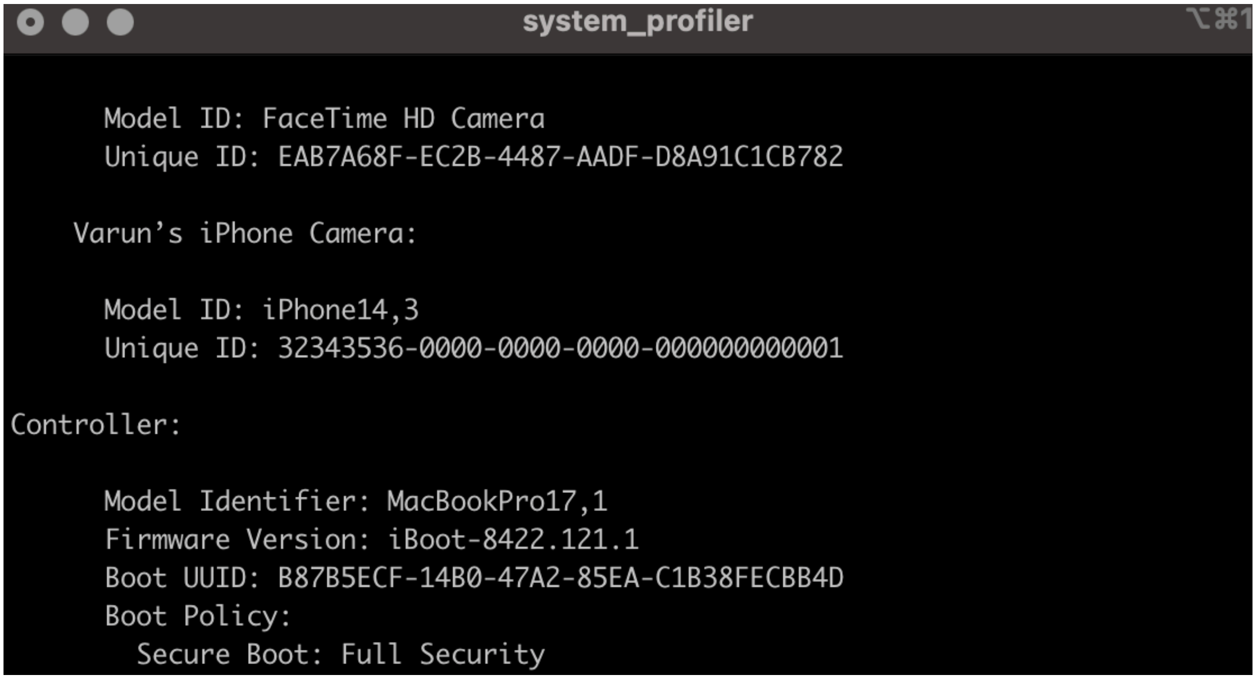
```
┌─[Varuns-MacBook-Pro-2] as varungupta in ~
└──→ nslookup example.com
Server:         127.0.2.2
Address:        127.0.2.2#53

Non-authoritative answer:
Name:   example.com
Address: 93.184.216.34

(base)
```

7.HOSTNAME



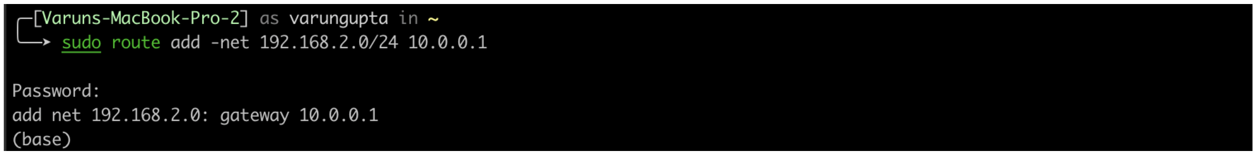
```
┌[Varuns-MacBook-Pro-2] as varungupta in ~
└─➤ hostname
Varuns-MacBook-Pro-2.local
(base)
```

8.SYSTEMINFO



```
      Model ID: FaceTime HD Camera
      Unique ID: EAB7A68F-EC2B-4487-AADF-D8A91C1CB782

    Varun’s iPhone Camera:

      Model ID: iPhone14,3
      Unique ID: 32343536-0000-0000-0000-000000000001

Controller:

      Model Identifier: MacBookPro17,1
      Firmware Version: iBoot-8422.121.1
      Boot UUID: B87B5ECF-14B0-47A2-85EA-C1B38FECBB4D
      Boot Policy:
        Secure Boot: Full Security
```

9.Route Print



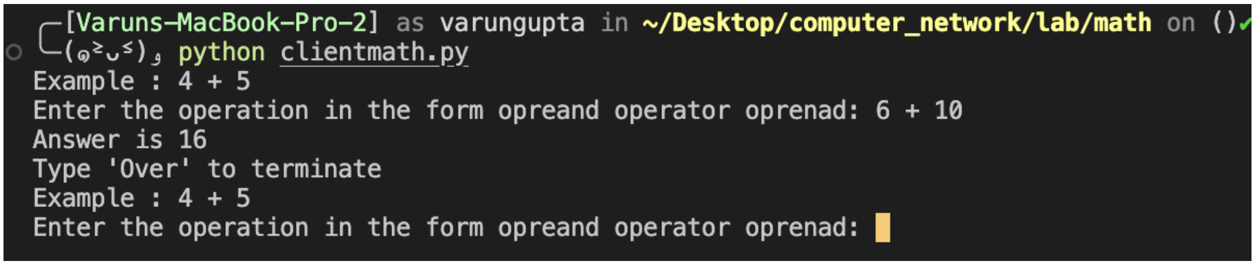
```
┌[Varuns-MacBook-Pro-2] as varungupta in ~
└─→ sudo route add -net 192.168.2.0/24 10.0.0.1

Password:
add net 192.168.2.0: gateway 10.0.0.1
(base)
```

Math - client server

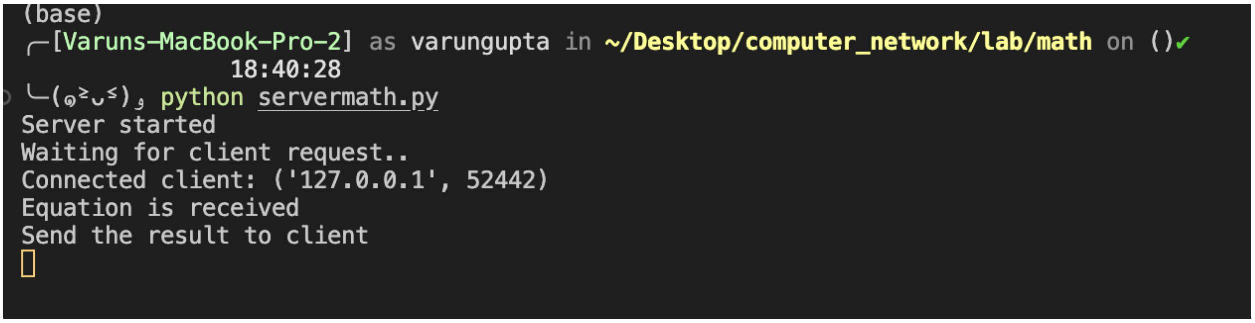
```
Import socket module
import socket

# In this Line we define our local host
# address with port number
SERVER = "127.0.0.1"
PORT = 8080
# Making a socket instance
client = socket.socket(socket.AF_INET,
                       socket.SOCK_STREAM)
# connect to the server
client.connect((SERVER, PORT))
# Running a infinite loop
while True:
    print("Example : 4 + 5")
    # here we get the input from the user
    inp = input("Enter the operation in \
the form opreand operator oprenad: ")
    # If user wants to terminate
    # the server connection he can type Over
    if inp == "Over":
        break
    # Here we send the user input
    # to server socket by send Method
    client.send(inp.encode())

    # Here we received output from the server socket
    answer = client.recv(1024)
    print("Answer is "+answer.decode())
    print("Type 'Over' to terminate")

client.close
```

server.py

```
# Import socket module
import socket

# Here we use localhost ip address
# and port number
LOCALHOST = "127.0.0.1"
PORT = 8080
# calling server socket method
server = socket.socket(socket.AF_INET,
                  socket.SOCK_STREAM)
server.bind((LOCALHOST, PORT))
server.listen(1)
print("Server started")
print("Waiting for client request..")
# Here server socket is ready for
# get input from the user
clientConnection, clientAddress = server.accept()
print("Connected client :", clientAddress)
msg = "
# Running infinite loop
while True:
  data = clientConnection.recv(1024)
  msg = data.decode()
  if msg == 'Over':
    print("Connection is Over")
    break
  print("Equation is received")
  result = 0
  operation_list = msg.split()
  oprnd1 = operation_list[0]
  operation = operation_list[1]
  oprnd2 = operation_list[2]

  # here we change str to int conversion
  num1 = int(oprnd1)
  num2 = int(oprnd2)
  # Here we are perform  basic arithmetic operation
  if operation == "+":
    result = num1 + num2
  elif operation == "-":

    result = num1 - num2
  elif operation == "/":
    result = num1 / num2
  elif operation == "*":
    result = num1 * num2

  print("Send the result to client")
  # Here we change int to string and
  # after encode send the output to client
  output = str(result)
  clientConnection.send(output.encode())
clientConnection.close()
```



```
 ┌─[Varuns-MacBook-Pro-2] as varungupta in ~/Desktop/computer_network/lab/math on ()✓
○└─(๑˃ᴗ˂)ﻭ python clientmath.py
 Example : 4 + 5
 Enter the operation in the form opreand operator oprenad: 6 + 10
 Answer is 16
 Type 'Over' to terminate
 Example : 4 + 5
 Enter the operation in the form opreand operator oprenad: 
```



```
(base)
╭─[Varuns-MacBook-Pro-2] as varungupta in ~/Desktop/computer_network/lab/math on ()✔
               18:40:28
╰─(๑˃ᴗ˂)ﻭ python servermath.py
Server started
Waiting for client request..
Connected client: ('127.0.0.1', 52442)
Equation is received
Send the result to client
```

Reverse the string :

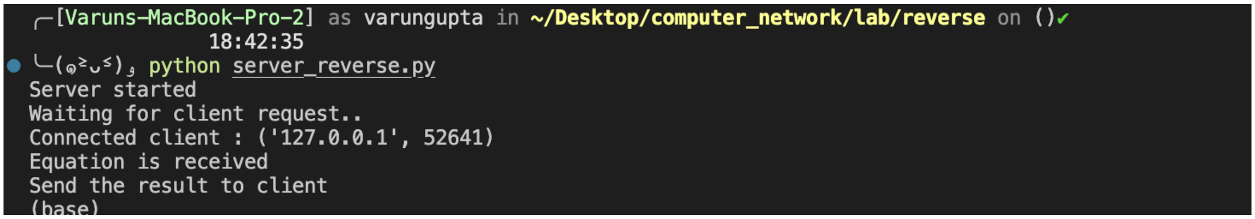
Code :

Sender Side (Server):

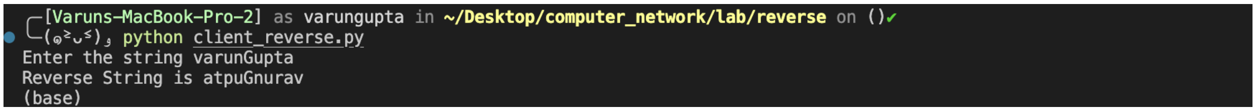
```
import socket

# Sender side (Server)
server_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
host = "127.0.0.1"
port = 12345
server_socket.bind((host, port))
server_socket.listen(1)
print(f"Sender side (Server) is listening on {host}:{port}")
connection, address = server_socket.accept()
print(f"Connection received from {address}")
input_string = "Hello, World!"
connection.sendall(input_string.encode())
print(f"Sent: {input_string}")
connection.close()
server_socket.close()
```

Receiver Side (Client):

```
import socket


#Receiver side (Client)
client_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
host = "127.0.0.1"
port = 12345
client_socket.connect((host, port))
received_data = client_socket.recv(1024).decode()
print(f"Received: {received_data}")
reversed_string = received_data[::-1]
print(f"Reversed: {reversed_string}")


client_socket.close()
```



```
 ╭─[Varuns-MacBook-Pro-2] as varungupta in ~/Desktop/computer_network/lab/reverse on ()✔
                 18:42:35
● ╰─(๑˃ᴗ˂)ﻭ python server_reverse.py
 Server started
 Waiting for client request..
 Connected client : ('127.0.0.1', 52641)
 Equation is received
 Send the result to client
 (base)
```



```
╭─[Varuns-MacBook-Pro-2] as varungupta in ~/Desktop/computer_network/lab/reverse on ()✔
╰─(๑˃ᴗ˂)ﻭ python client_reverse.py
Enter the string varunGupta
Reverse String is atpuGnurav
(base)
```