Name: Gaurav Kumar Singh

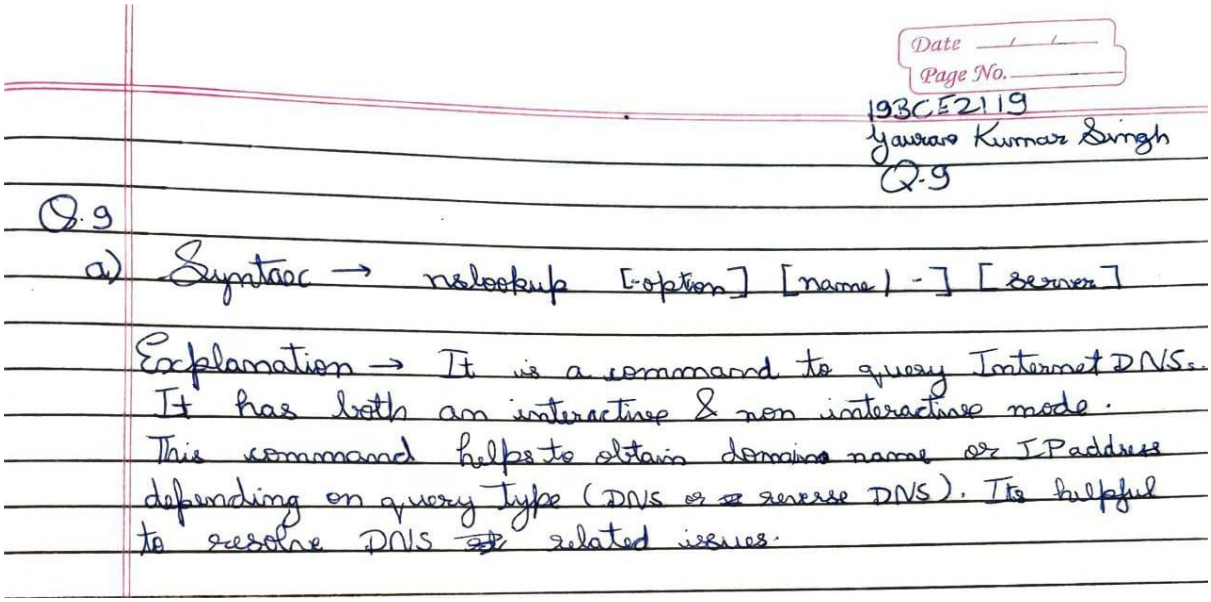
Registration number: 19BCE2119

Course: Network and Communication

Final Assessment Test

Question no. 9

a)



OUTPUT

```
gaurav1020@DESKTOP-R0RPIEK: ~
gaurav1020@DESKTOP-R0RPIEK:~$ man nslookup
gaurav1020@DESKTOP-R0RPIEK:~$ nslookup google.com
Server:         117.96.122.14
Address:        117.96.122.14#53

Non-authoritative answer:
Name:   google.com
Address: 142.250.77.238
Name:   google.com
Address: 2404:6800:4002:80a::200e

gaurav1020@DESKTOP-R0RPIEK:~$
```

b)

Date
Page No.

19BCE2119
Yuvraaj Kumar Singh
Q.9

Title

b) TCP/IP dependency on OSI Model

Aim :-

To implement a model of TCP/IP protocol similar to that of OSI Model

Algorithm :-

① Identify the layers of TCP/IP as :
   (i) Application (Application, Presentation, Session in OSI)
   (ii) Transport
   (iii) Network
   (iv) Data link
   (v) Physical

② After passing through each layer, the message recieves a header but when passing through data link layer, message recieves both header & footer as in frames.

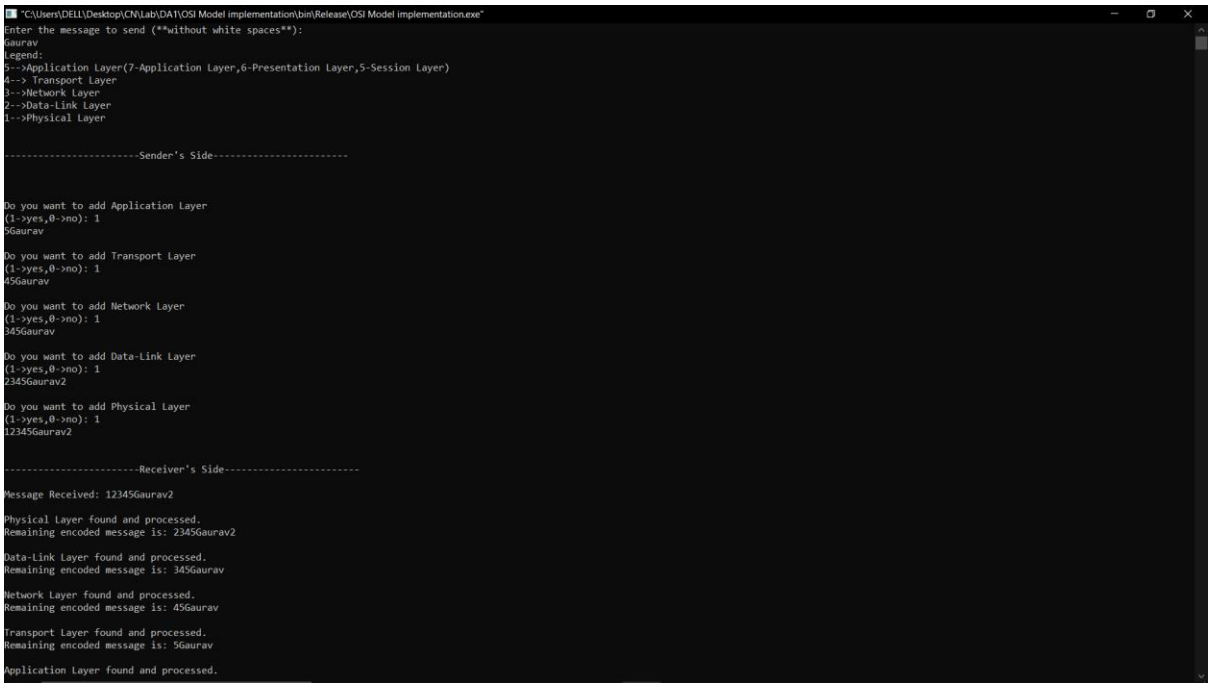
③ The message can be sent after passing through each layer.

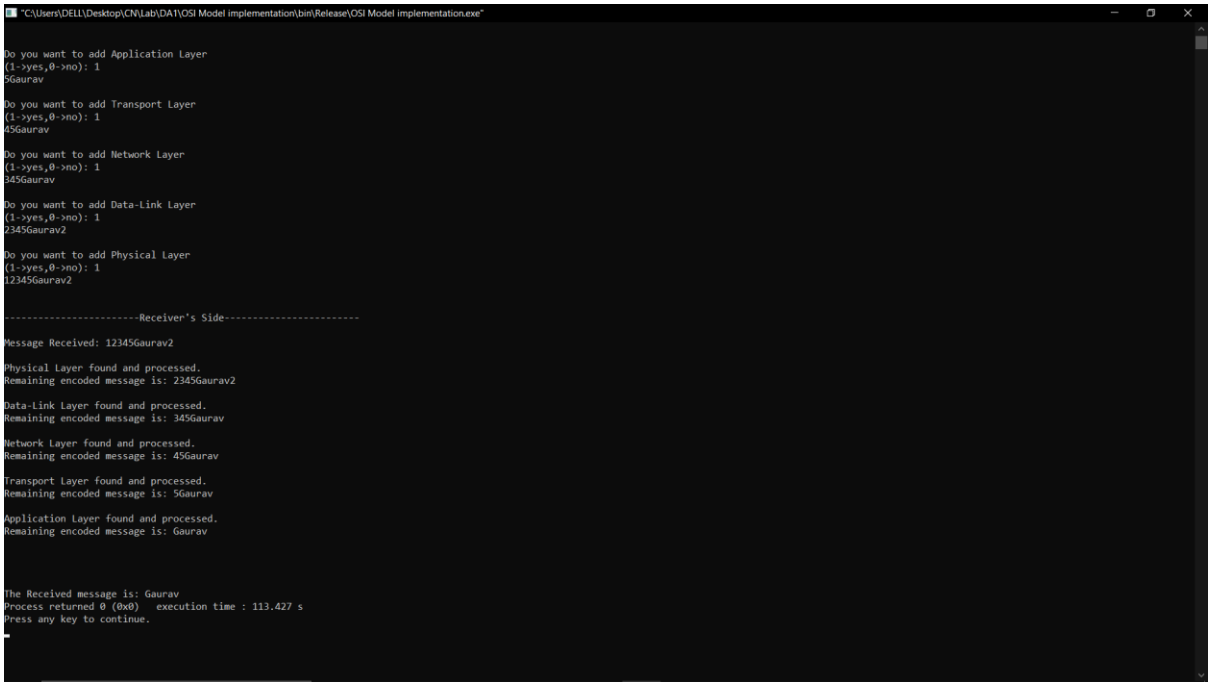
④ Recieved message's header is read & directed as to which layer is present.

⑤ If data link layer is found, both its header & tailer are removed.

⑥ The original message is restored.

OUTPUT


"C:\Users\DELL\Desktop\CN\Lab\DA1\OSI Model implementation\bin\Release\OSI Model implementation.exe"
Enter the message to send (**without white spaces**):
Gaurav
Legend:
5-->Application Layer(7-Application Layer,6-Presentation Layer,5-Session Layer)
4--> Transport Layer
3-->Network Layer
2-->Data-Link Layer
1-->Physical Layer
----------Sender's Side----------
Do you want to add Application Layer
(1->yes,0->no): 1
5Gaurav
Do you want to add Transport Layer
(1->yes,0->no): 1
45Gaurav
Do you want to add Network Layer
(1->yes,0->no): 1
345Gaurav
Do you want to add Data-Link Layer
(1->yes,0->no): 1
2345Gaurav2
Do you want to add Physical Layer
(1->yes,0->no): 1
12345Gaurav2
----------Receiver's Side----------
Message Received: 12345Gaurav2
Physical Layer found and processed.
Remaining encoded message is: 2345Gaurav2
Data-Link Layer found and processed.
Remaining encoded message is: 345Gaurav
Network Layer found and processed.
Remaining encoded message is: 45Gaurav
Transport Layer found and processed.
Remaining encoded message is: 5Gaurav
Application Layer found and processed.


"C:\Users\DELL\Desktop\CN\Lab\DA1\OSI Model implementation\bin\Release\OSI Model implementation.exe"
Do you want to add Application Layer
(1->yes,0->no): 1
5Gaurav
Do you want to add Transport Layer
(1->yes,0->no): 1
45Gaurav
Do you want to add Network Layer
(1->yes,0->no): 1
345Gaurav
Do you want to add Data-Link Layer
(1->yes,0->no): 1
2345Gaurav2
Do you want to add Physical Layer
(1->yes,0->no): 1
12345Gaurav2
----------Receiver's Side----------
Message Received: 12345Gaurav2
Physical Layer found and processed.
Remaining encoded message is: 2345Gaurav2
Data-Link Layer found and processed.
Remaining encoded message is: 345Gaurav
Network Layer found and processed.
Remaining encoded message is: 45Gaurav
Transport Layer found and processed.
Remaining encoded message is: 5Gaurav
Application Layer found and processed.
Remaining encoded message is: Gaurav
The Received message is: Gaurav
Process returned 0 (0x0)   execution time : 113.427 s
Press any key to continue.

c) Using the concept of classful addressing to manage file/folder system in central storage HUB to store data.

Date / /
Page No.

18BCE2115
Yaurav Kumar Singh
Q.9

c) Title :-
Using the concept of classful addressing to manage file/folder system in central storage HUB to store data.

Explanation :-
Every user can be allotted some fixed ~~data~~ storage space in the HUB & partitions can be made in the HUB. HUB have multiple disks & we can use its count as first part of our classful addressing & the starting & ending location of a particular users storage space in the HUB as the next two parts of our identification string. Whenever a user is registered the system automatically creates a hash table with username as key & disk no. + start address + last address as its value.
When a user logs in, its corresponding value in hash table is retrieved & by using classful addressing, disk no, start address & last address are ~~set~~ unlocked for user to access.

Example