

VISVESVARAYATECHNOLOGICALUNIVERSITY

“JnanaSangama”, Belgaum -590014, Karnataka.



LAB RECORD

Computer Network Lab (23CS5PCCON)

Submitted by

Gautam Gandhani (1BM22CS101)

in partial fulfillment for the award of the degree of

**BACHELOROFENGINEERING
in
COMPUTER SCIENCE AND ENGINEERING**



**B.M.S. COLLEGE OF ENGINEERING
(Autonomous Institution under VTU)**

**BENGALURU-560019
Academic Year 2024-25 (odd)**

B.M.S. College of Engineering

Bull Temple Road, Bangalore 560019

(Affiliated To Visvesvaraya Technological University, Belgaum)

Department of Computer Science and Engineering



CERTIFICATE

This is to certify that the Lab work entitled “ Computer Network (23CS5PCCON)” carried out by **Gautam Gandhani (1BM22CS101)**, who is a bonafide student of **B.M.S. College of Engineering**. It is in partial fulfilment for the award of **Bachelor of Engineering in Computer Science and Engineering** of the Visvesvaraya Technological University, Belgaum. The Lab report has been approved as it satisfies the academic requirements of the above-mentioned subject and the work prescribed for the said degree.

Dr. Lata N.R Associate Professor Department of CSE, BMSCE	Dr. Kavitha Sooda Professor & HOD Department of CSE, BMSCE
---	--

Index

Sl. No.	Date	Experiment Title	Page No.
1	09-10-24	Create a topology and simulate sending a simple PDU from source to destination using hub and switch as connecting devices and demonstrate ping messages.	5-8
2	09-10-24	Configure IP address to routers in packet tracer. Explore the following messages: ping responses, destination unreachable, request timed out, reply.	9-11
3	16-10-24	Configure default route, static route to the Router (Part 1).	12-15
4	23-10-24	Configure default route, static route to the Router (Part 2).	16-20
5	13-11-24	Configure DHCP within a LAN and outside LAN.	21-26
6	20-11-24	Configure RIP routing Protocol in Routers .	28-30
7	20-11-24	Demonstrate the TTL/ Life of a Packet.	31-34
8	27-11-24	Configure OSPF routing protocol.	35-40
9	18-12-24	Configure Web Server, DNS within a LAN.	41-42
10	18-12-24	To construct a simple LAN and understand the concept and operation of Address Resolution Protocol (ARP).	43-45
11	18-12-24	To understand the operation of TELNET by accessing the router in the server room from a PC in the IT office.	46-48
12	18-12-24	To construct a VLAN and make the PC's communicate among a VLAN.	49-53
13	18-12-24	To construct a WLAN and make the nodes communicate wirelessly.	54-57
14	18-12-24	Write a program for error detecting code using CRC-CCITT (16-bits).	58-59
15	18-12-24	Write a program for congestion control using Leaky bucket algorithm.	60-61

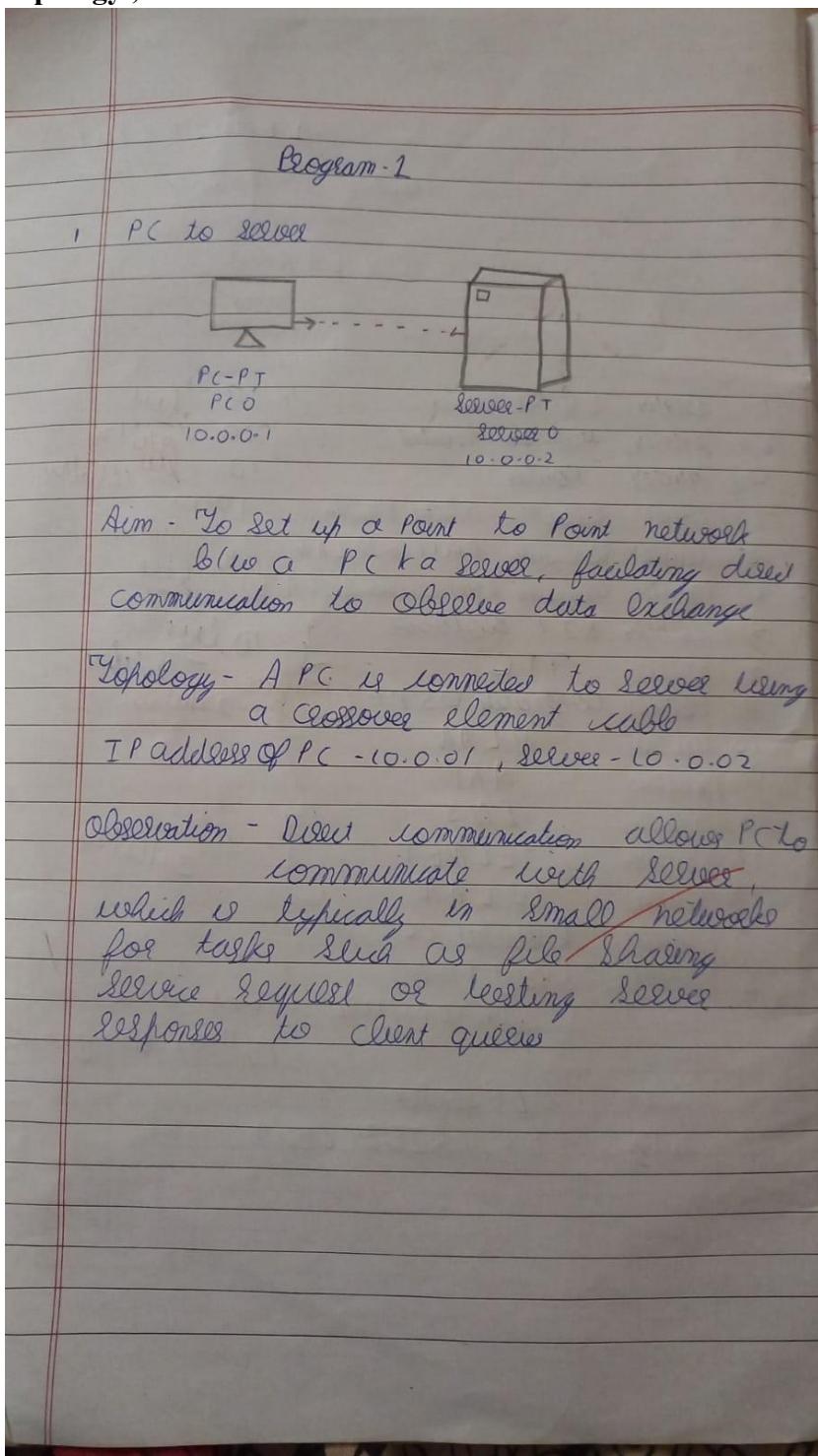
16	18-12-24	Using TCP/IP sockets, write a client-server program to make the client send the file name and the server to send back the contents of the requested file if present.	62
17	18-12-24	Using UDP sockets, write a client-server program to make client sending the file name and the server to send back the contents of the requested file if present.	63-64

Github Link: [link](#)

Program 1

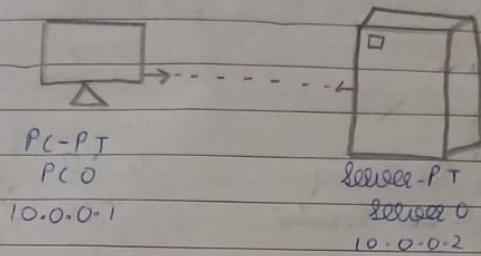
Aim: Create a topology and simulate sending a simple PDU from source to destination using hub and switch as connecting devices and demonstrate ping messages.

Topology , Procedure and Observation:



Program - 1

1. PC to server

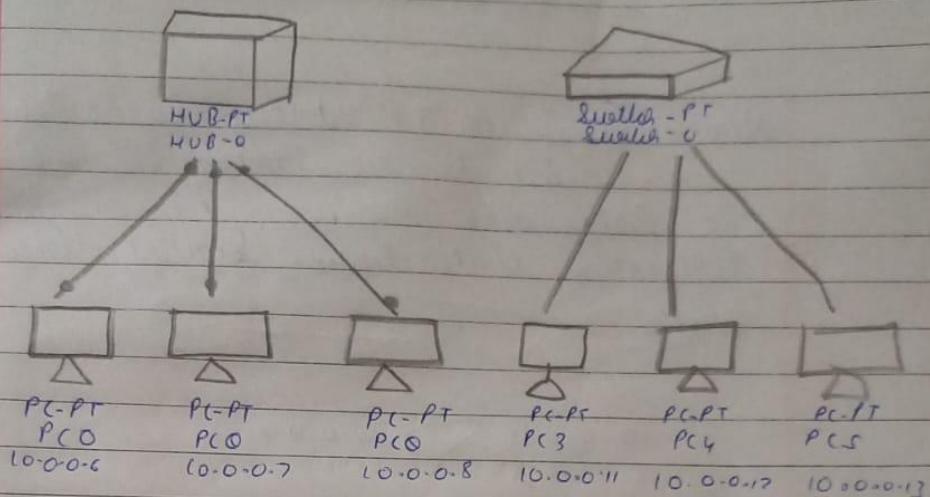


Aim - To set up a Point to Point network
b/w a PC & a server, facilitating direct communication to observe data exchange

Topology - A PC is connected to server using a crossover element cable
IP address of PC - 10.0.0.1, server - 10.0.0.2

Observation - Direct communication allows PC to communicate with server, which is typically in small networks for tasks such as file sharing, service request or testing service responses to client queries

2) Hub & Switch

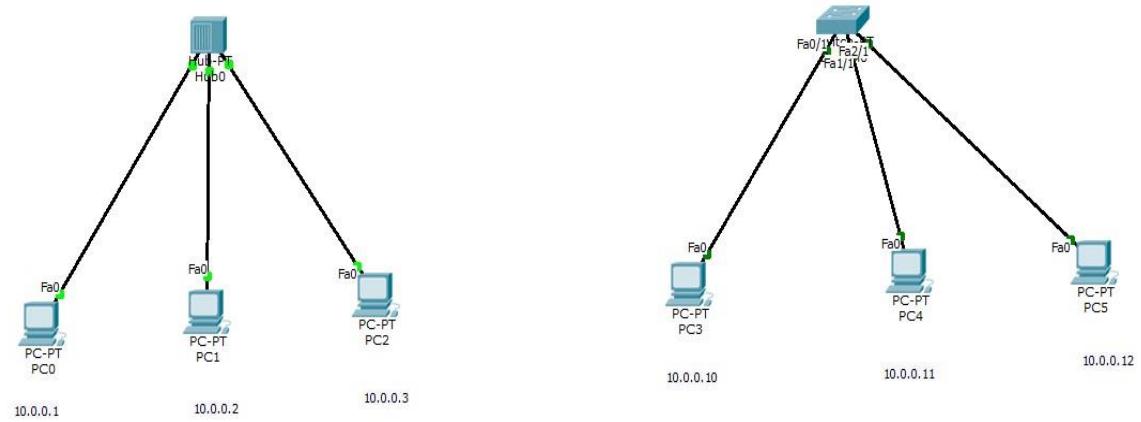
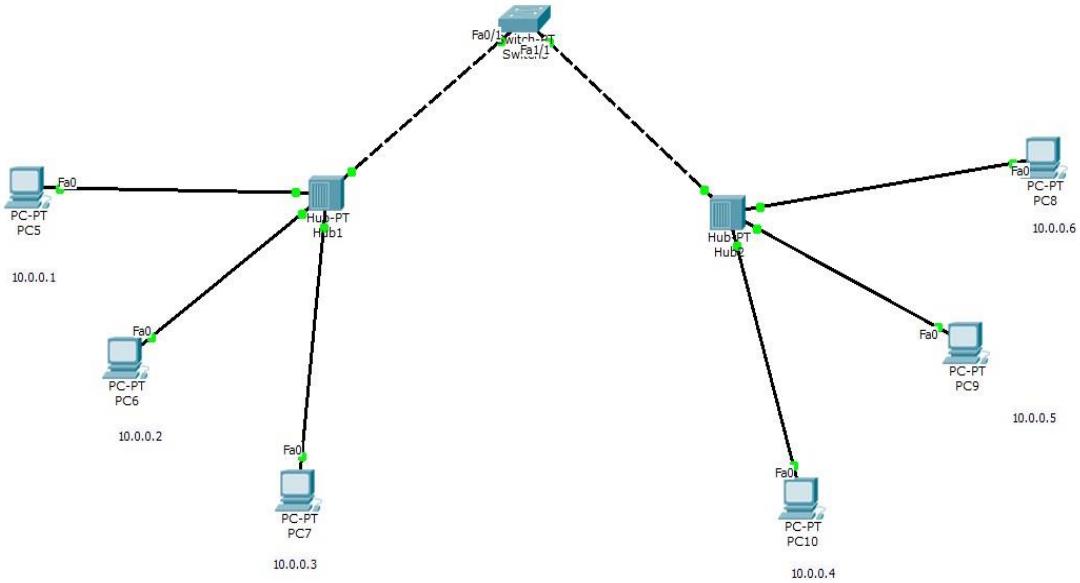


Aim - To create simple network consisting of 3 PCs connected to a central hub and a network with 3 PCs connected to switch. This connection will help observe of data transmission using hub & switch device.

~~Topology - 3 PCs are connected to hub & switch using straight through ethernet cable~~

Observations - Hub broadcast packets to all devices which may cause unnecessary traffic. Switch forwards packet only to appropriate device by learning mac address, making it more efficient in reducing traffic.

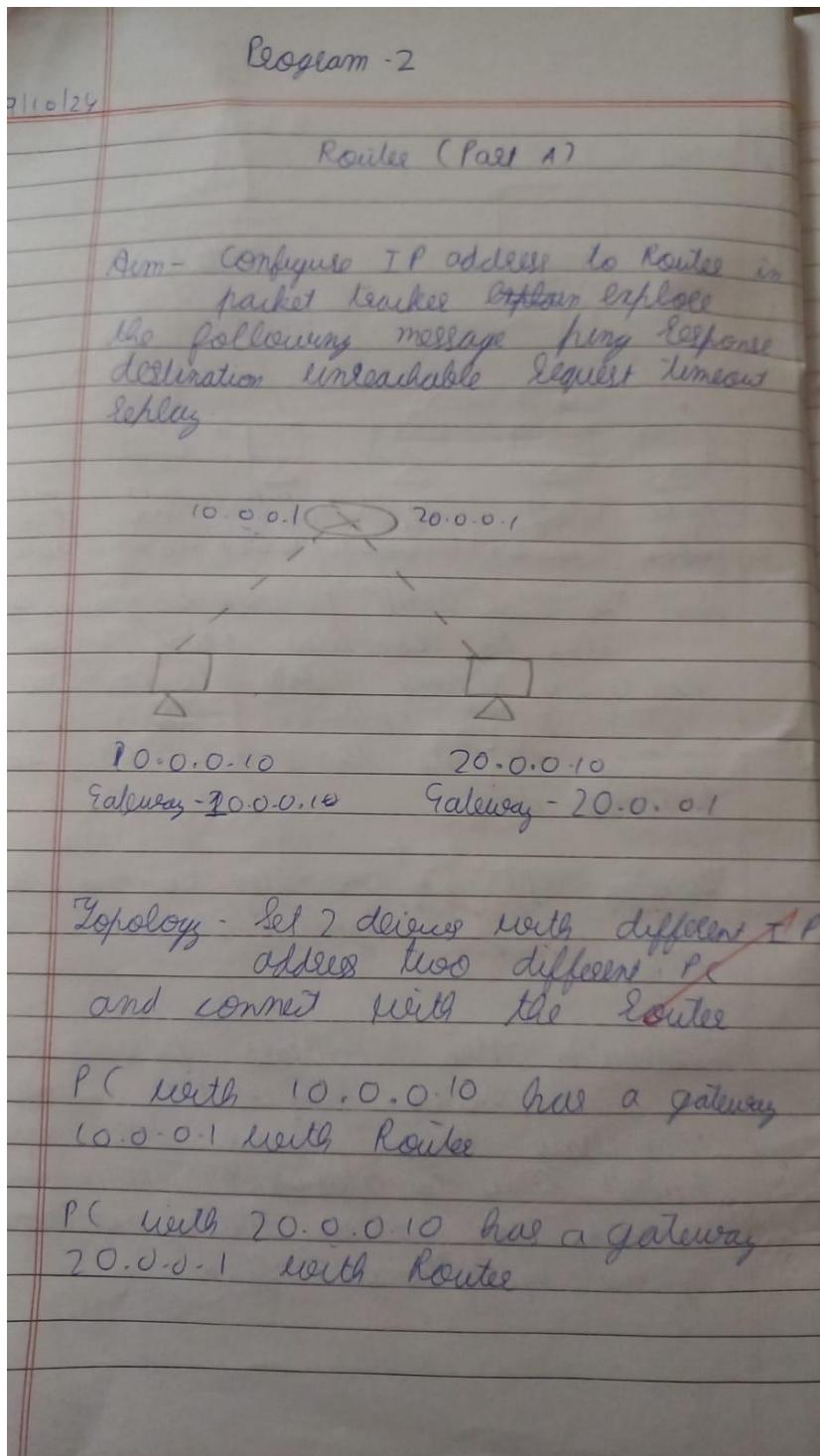
Screen Shots:



Program 2

Aim: Configure IP address to routers in packet tracer. Explore the following messages: ping responses, destination unreachable, request timed out, reply.

Topology , Procedure and Observation:



Observation :-

+0.0

Procedure

Router - enable

Router#1 - config terminal

Router Config - interface fastethernet 0/0
ip address 10.0.0.1 255.0.0.0
no shutdown

interface fastethernet 1/0
ip address 20.0.0.1 255.0.0.0
no shutdown

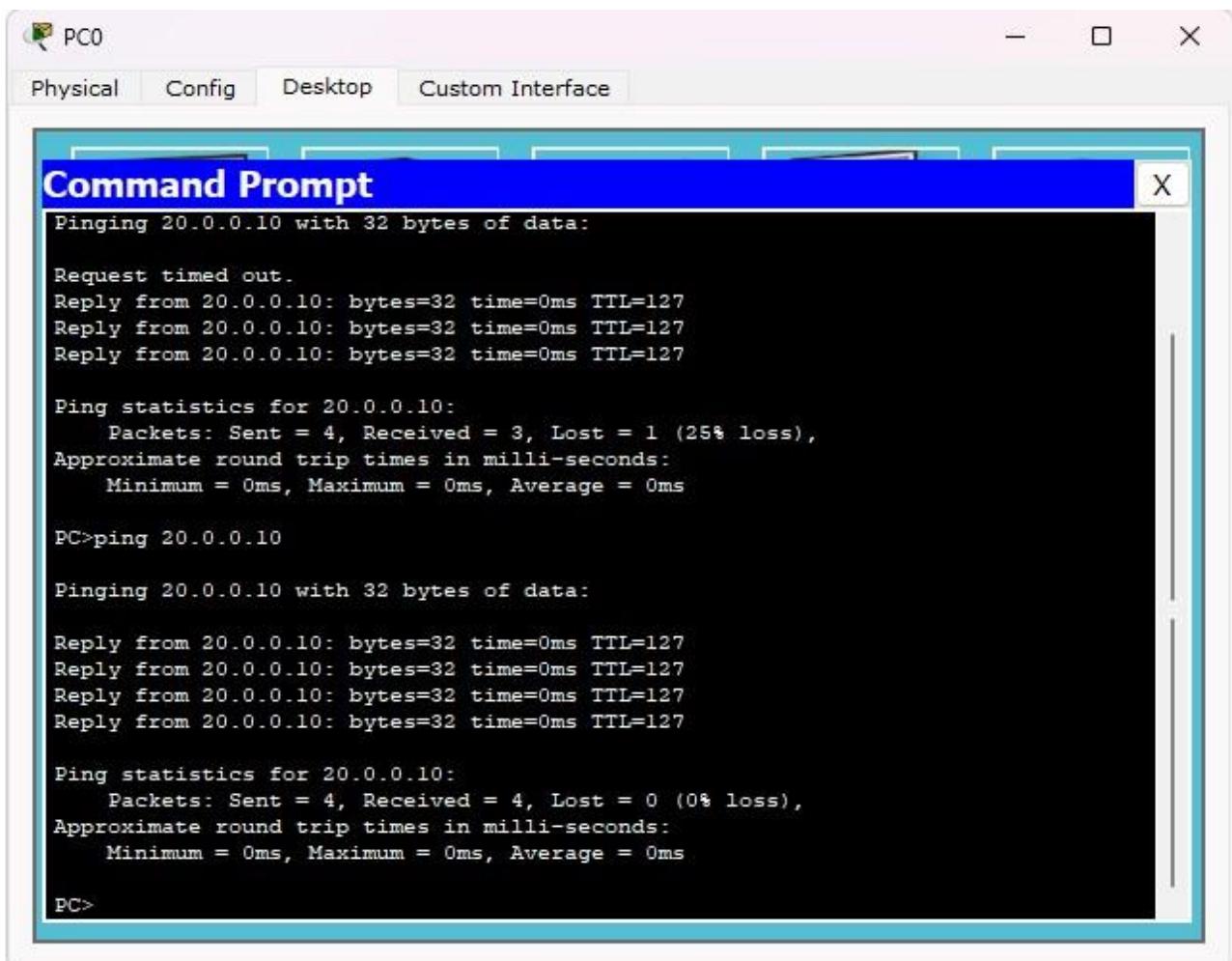
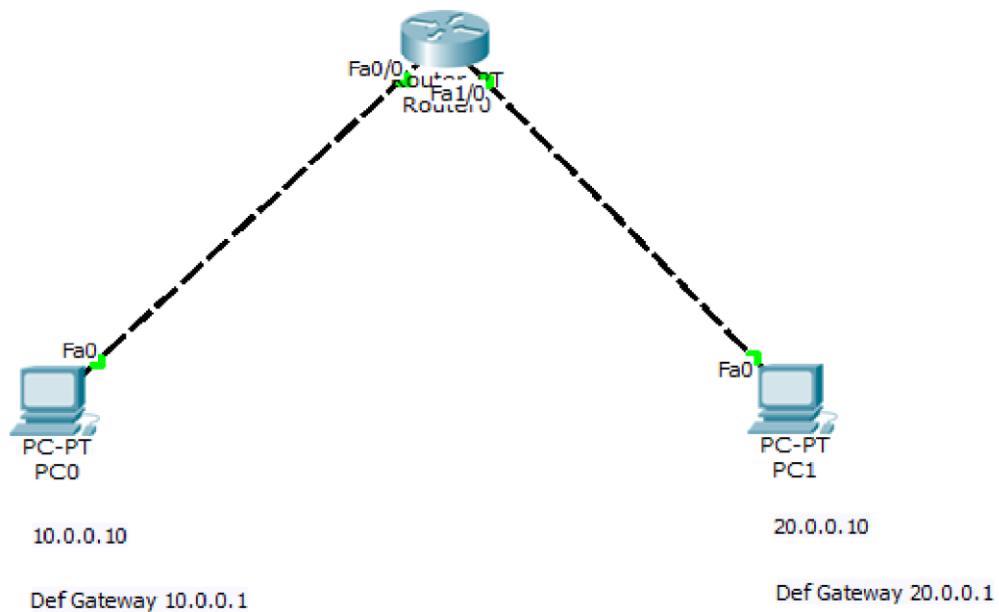
Result

Show ip route

10.0.0.0/8 is directly connected
20.0.0.0/8 is directly connected

new routes

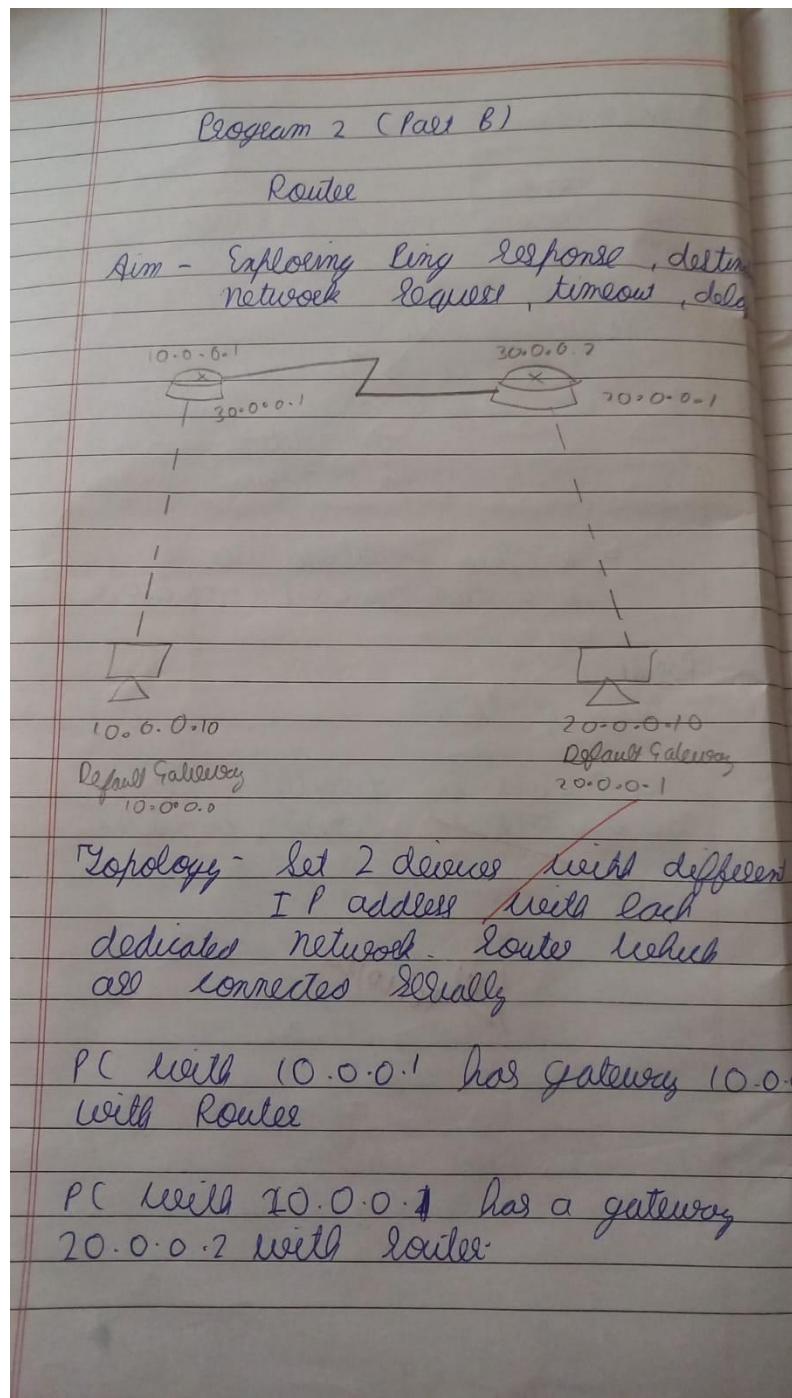
Screen Shots:



Program 3

Aim: Configure default route, static route to the Router(Part 1).

Topology , Procedure and Observation:



Router 1 is equally connected with Router 2 and having IP address 30.0.0.1 & 30.0.0.2

Procedure

- ① Take two Routers and connect the two system to two Router respectively
- ② In First Router Open CLI and type below command

enable
config terminal
interface fastethernet 0/0
ip address 10.0.0.1 255.0.0.0
no shutdown

end

interface serial 2/0
ip address 30.0.0.1 255.0.0.0
no shutdown

exit

Configuration of Second Router

enable

config terminal

interface fastethernet 1/0

ip address 20.0.0.1 255.0.0.0

no shutdown
exit
interface serial 2/0
ip address 30.0.0.2 255.0.0.0
no shutdown
exit

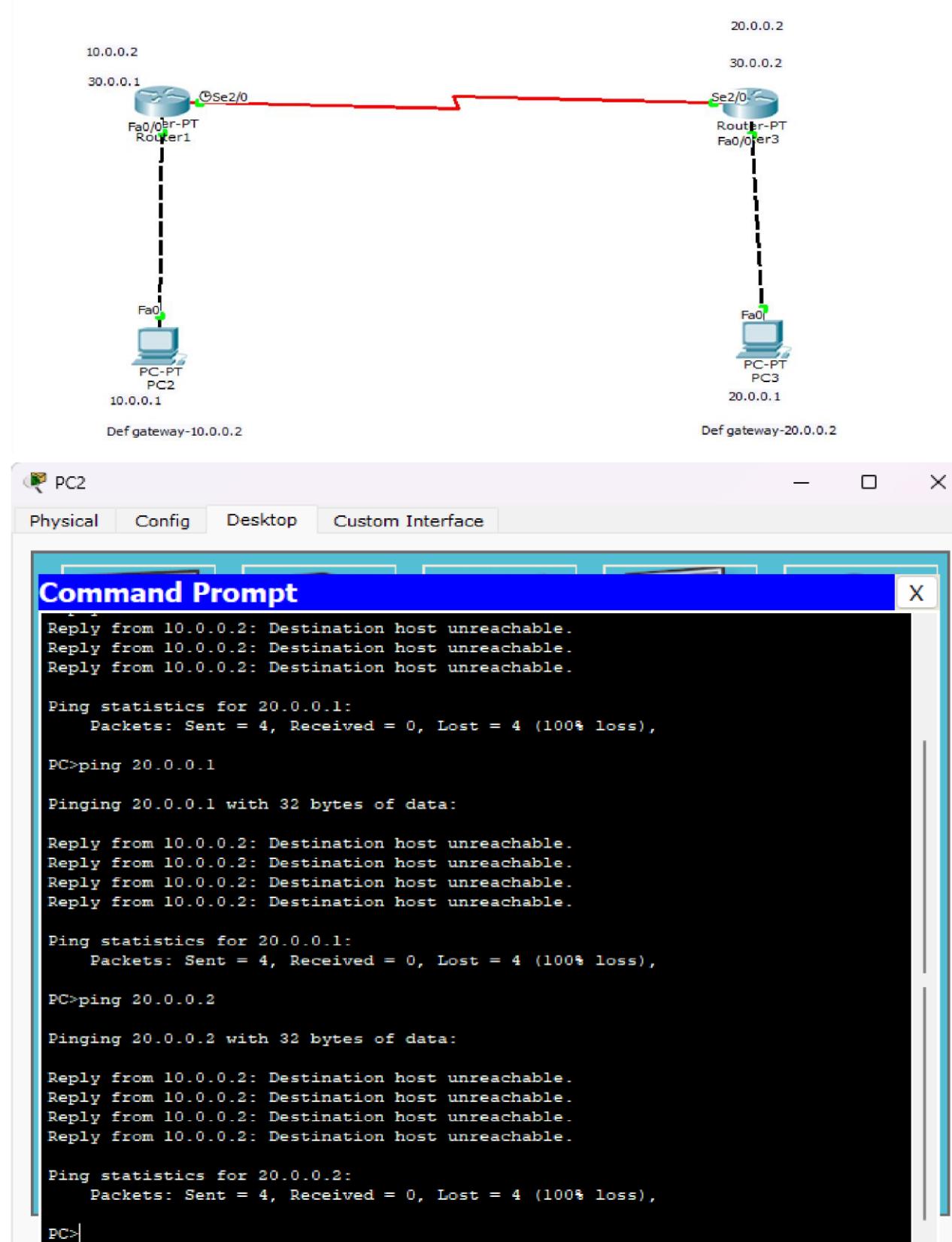
ping
command prompt
ping 20.0.0.1
Reply from 30.0.0.10 : Destination Unreachable

configuring Static Route

ip route 10.0.0.0 255.0.0.0 30.0.0.1

ping
command prompt
ping 20.0.0.1
Reply from 20.0.0.10 bytes=32 time=6ms

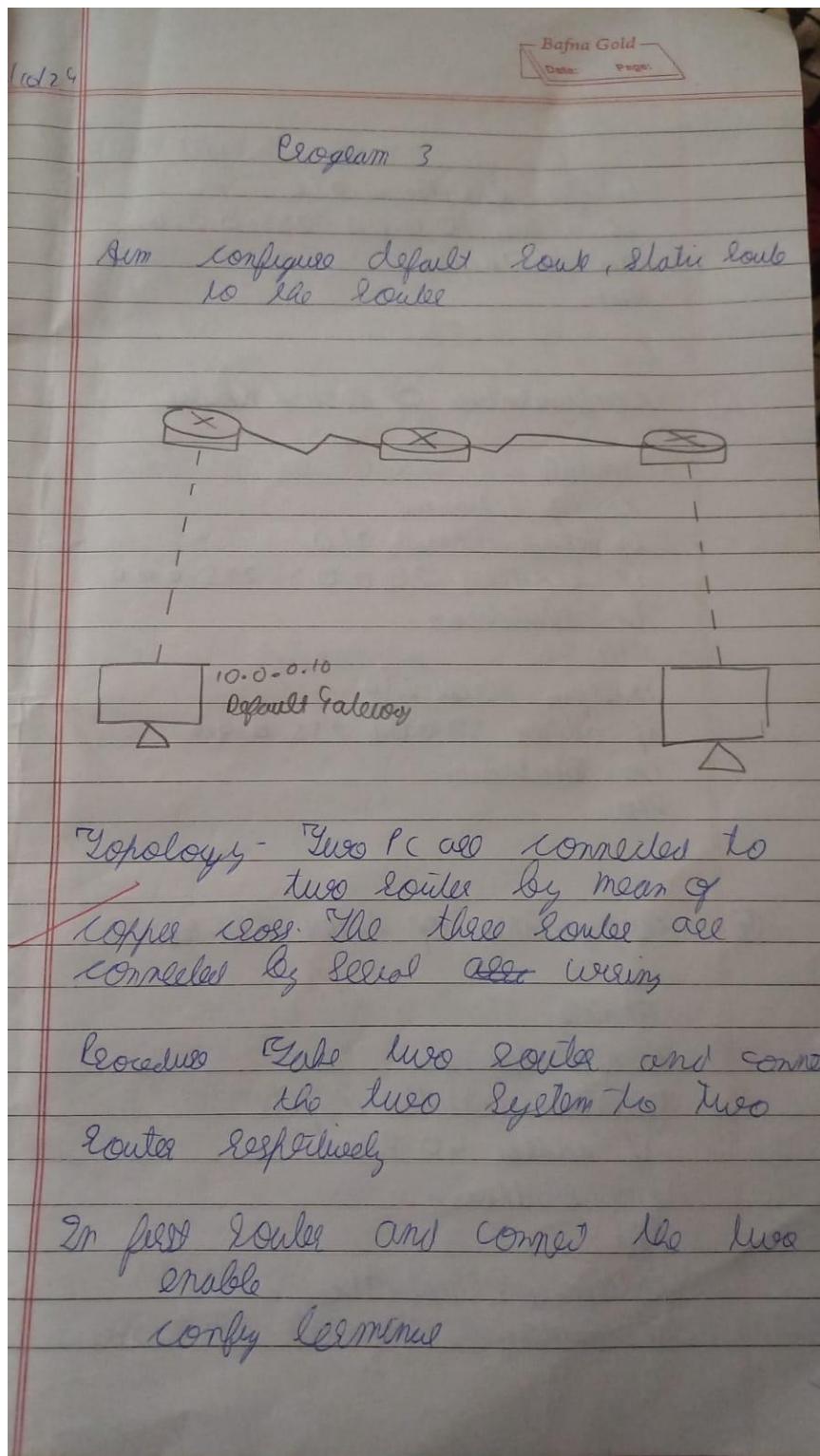
Screen Shots:



Program 4

Aim: Configure default route, static route to the Router(Part 2).

Topology , Procedure and Observation:



interface fastethernet 0/0
ip address 10.0.0.1 255.0.0.0
no shutdown
exit

③ Configuration of second Router

enable
config terminal
interface serial 3/0
ip address 20.0.0.2 255.0.0.0
no shutdown
exit
interface serial 2/0
ip address 30.0.0.1 255.0.0.0
no shutdown
exit

④ Configuration of third Router

enable
config terminal
interface fastethernet 1/0
ip address 40.0.0.1
no shutdown
exit
interface serial 3/0
ip address 30.0.0.2 255.0.0.0

no shutdown
loop

5) static configuration second Router

if route 10.0.0.0 255.0.0.0 20.0.0.1
if route 40.0.0.0 255.0.0.0 30.0.0.2

5') static configuration first Router

if route 0.0.0.0 0.0.0.0 20.0.0.1

7) static configuration third Router

if route 0.0.0.0 0.0.0.0 30.0.0.1

~~Observation~~

Router 1

Show if route

C 10.0.0.0/8 is directly connected

C 20.0.0.0/8 is directly connected Serial2

S 0.0.0.0/0 [1/0] via 20.0.0.2

Router 2

S 10.0.0.0/8 [1/0] via 20.0.0.1

C 20.0.0.0/8 is directly connected Serial3

13111

C 30.0.0.0/8 is Directly Connected Serial 2/0
S 40.0.0.0/8 [1/0]

Router 3

Show if route

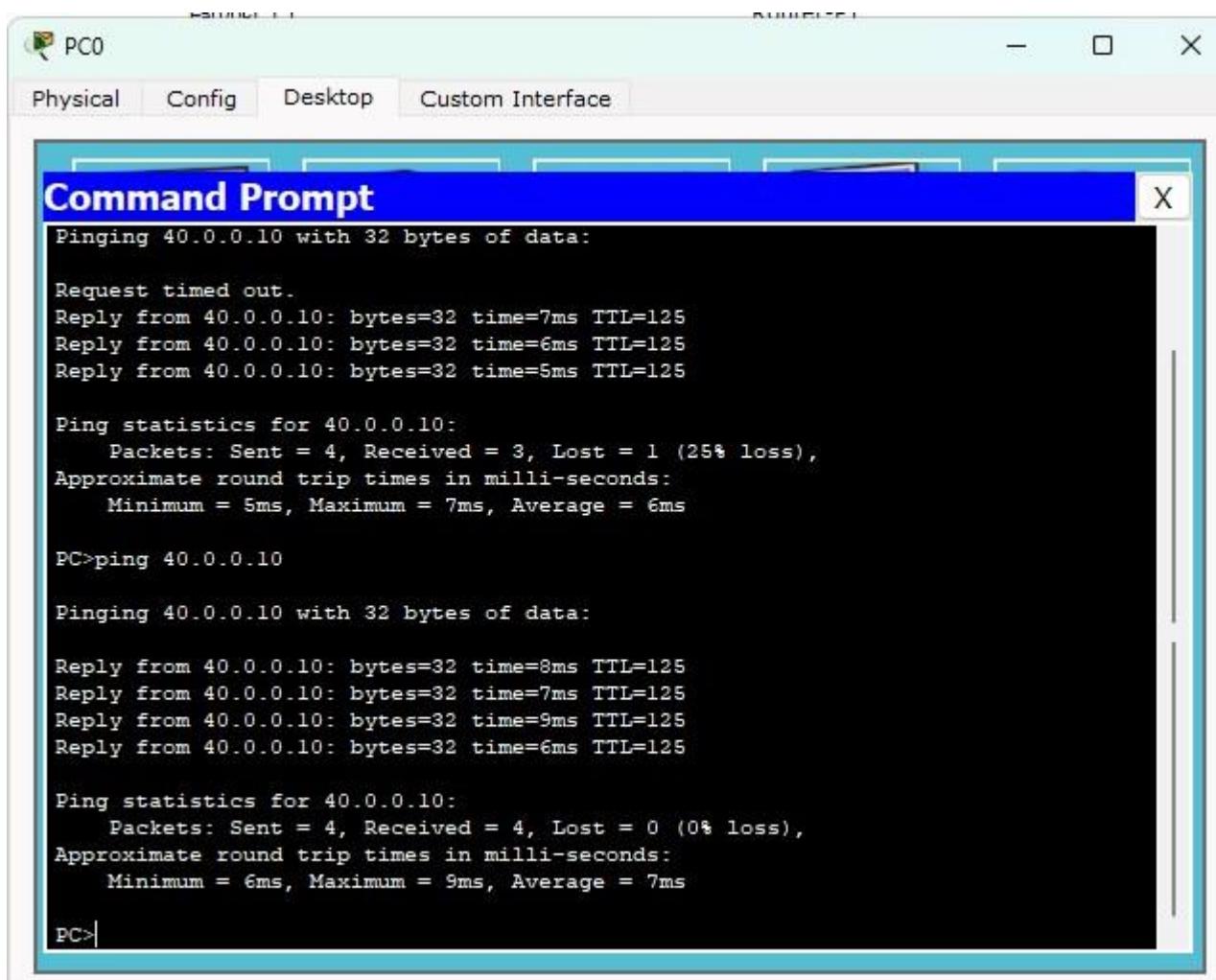
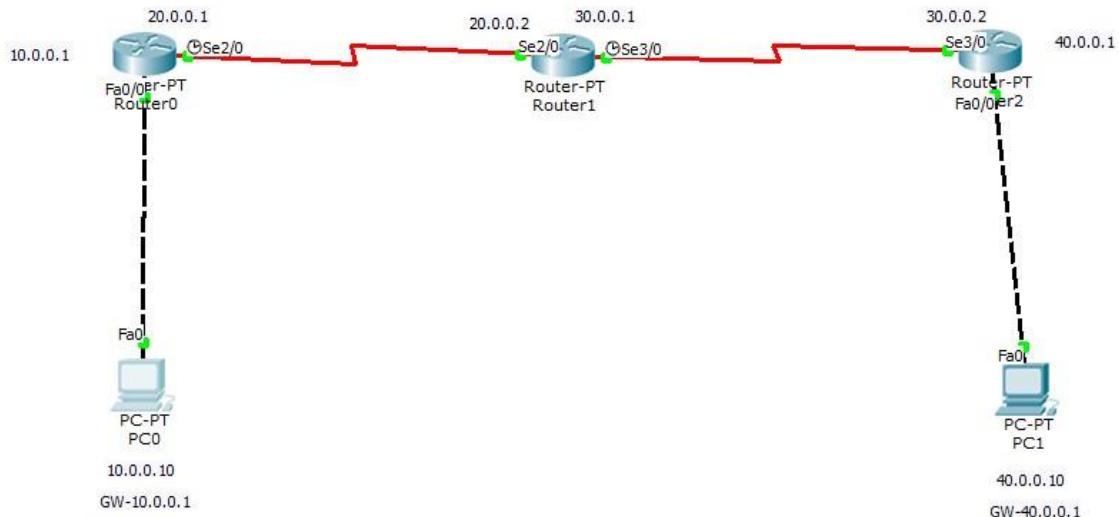
C 30.0.0.0/8 is directly Connected Serial 2/0
C 40.0.0.0/8 is Directly Connected
S* 0.0.0.0/10 [1/0] via 30.0.0.1

ping 40.0.0.10

ping statistic for 40.0.0.10

✓
23/10/20

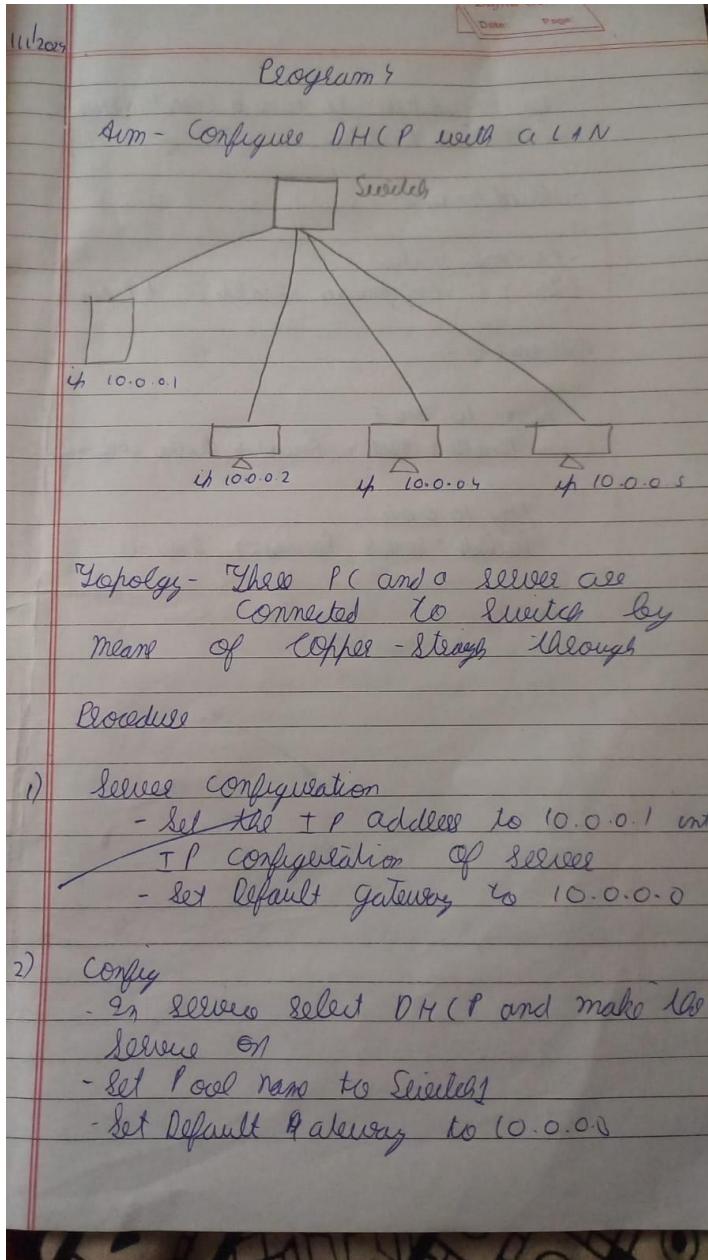
Screen Shots:



Program 5

Aim: Configure DHCP within a LAN and outside LAN.

Topology , Procedure and Observation:



- set IP address to 10.0.0.3 and select first 3 IP address

- Click on add

- PC Configuration

- In IP Configuration enable DHCP for all

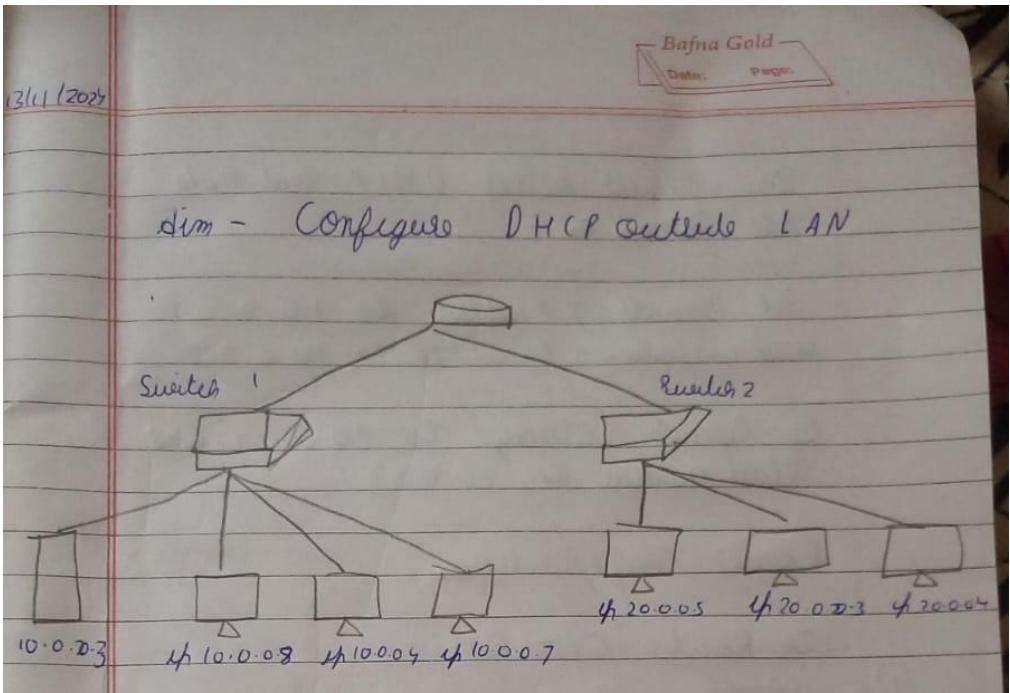
Observation -

ping 10.0.0.5

Packets : Sent: 4, Received = 4, Lost = 0 (0% of total)

ping 10.0.0.9

Packets : Sent: 4, Received = 4, Lost = 0



Topology - set 6 PCs and a server are connected to ~~respective~~ ~~respective~~ switches of a network 10.0.0.0 and 20.0.0.0 by means of Copper String Wires

Procedure

1) Server Configuration

- Set the IP address of a IP configuration server
- Set default gateway to 10.0.0.0 for switch 1 and 20.0.0.1 for switch 2

2) config

20/11

In server select DHCP and make
the server on

Set Default IP address to 10.0.0.1
and static IP address to 10.0.0.3

By default gateway to 20.0.0.1 and
subnet address to 20.0.0.3

Router Configuration

1) In Router (CLI)
enable

config-terminal
interface fastethernet 4/0
ip address 10.0.0.1 255.0.0.0
ip helper-address 10.0.0.3
no shutdown

2) interface fastethernet 0/0
ip address 20.0.0.1 255.0.0.0
ip helper-address 10.0.0.3
no shutdown
end

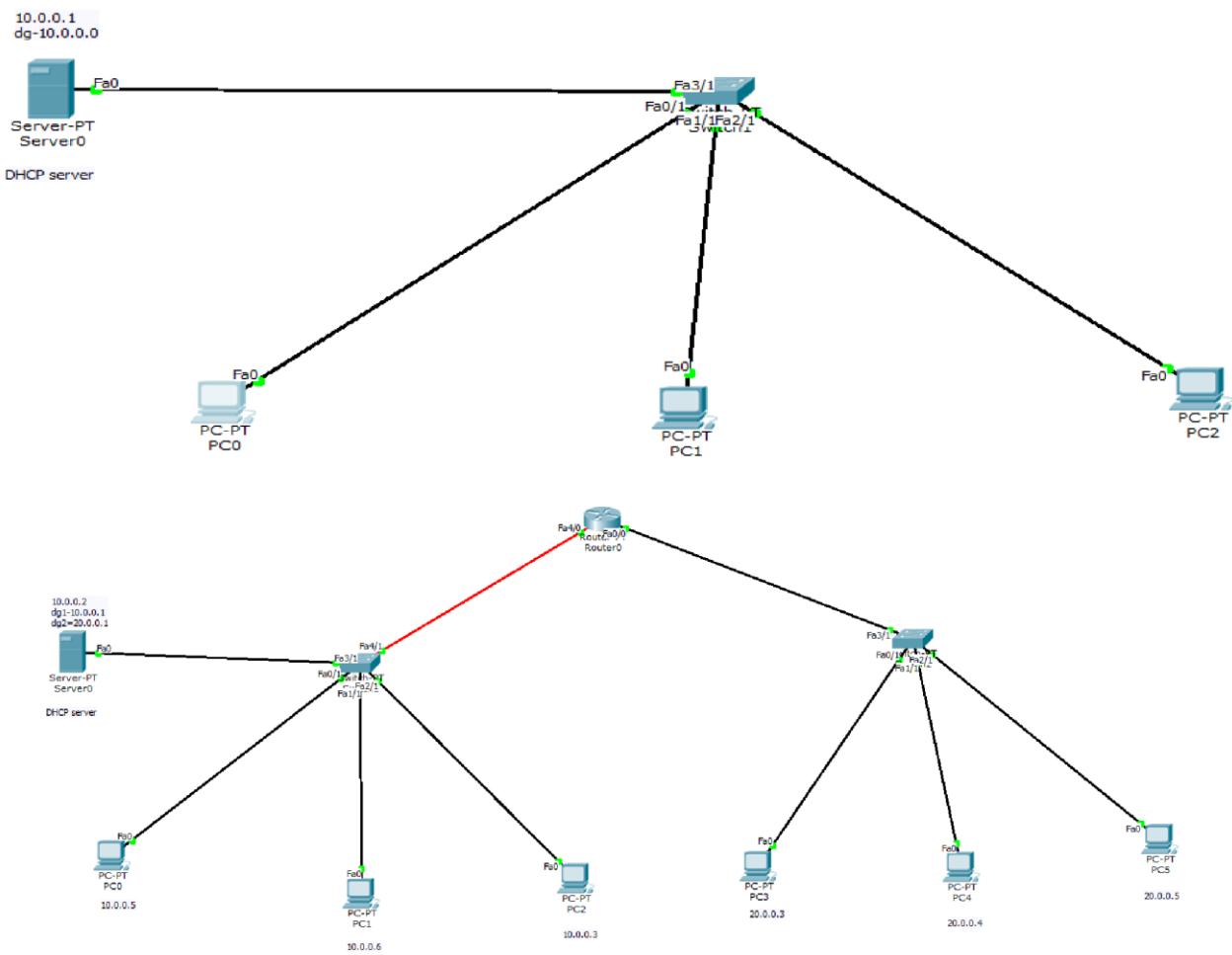
all

Observation

Show ip route

- ↳ 10.0.0.1 is directly fastethernet 0/0
- ↳ 20.0.0.1 is directly connected fastethernet 0/0

Screen Shots:



The screenshot shows a window titled "Command Prompt" from the Cisco Packet Tracer software. The window has a blue header bar with the title and a close button (X). Below the header is a toolbar with four tabs: "Physical", "Config", "Desktop", and "Custom Interface". The "Custom Interface" tab is currently selected. The main area of the window is a black terminal-like interface displaying the following text:

```
Packet Tracer PC Command Line 1.0
PC>ping 10.0.0.4

Pinging 10.0.0.4 with 32 bytes of data:

Reply from 10.0.0.4: bytes=32 time=1ms TTL=128
Reply from 10.0.0.4: bytes=32 time=0ms TTL=128
Reply from 10.0.0.4: bytes=32 time=0ms TTL=128
Reply from 10.0.0.4: bytes=32 time=0ms TTL=128

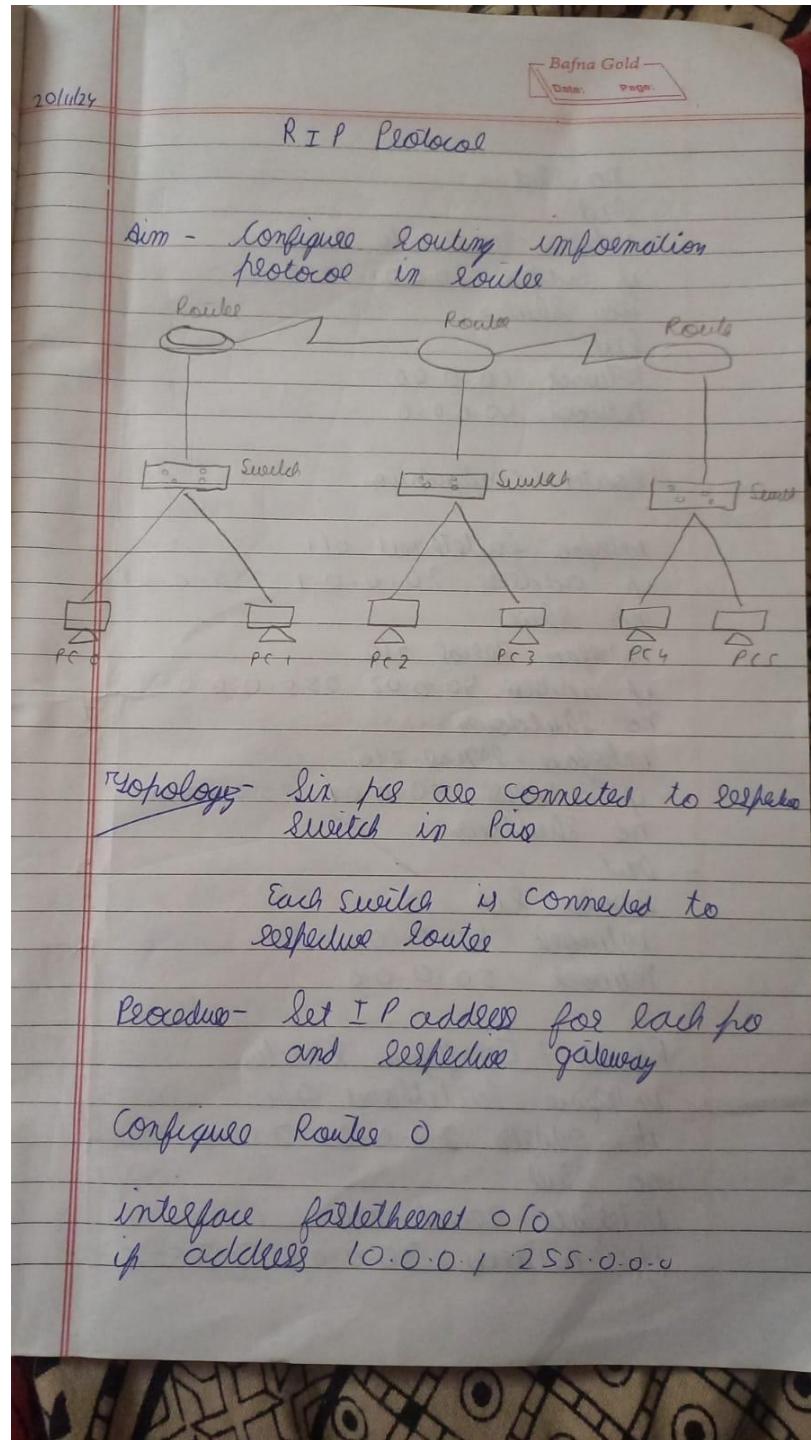
Ping statistics for 10.0.0.4:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 1ms, Average = 0ms

PC>
```

Program 6

Aim: Configure RIP routing Protocol in Routers .

Topology , Procedure and Observation:



no shut
exit
interface serial 2/0
ip address 10.0.0.1 255.0.0.0
no shut
exit
network 10.0.0.0
network 40.0.0.0

Router Configuration

interface fastethernet 0/1
ip address 20.0.0.1 255.0.0.0
no shut
interface serial 2/0
ip address 40.0.0.2 255.0.0.0
no shutdown
interface serial 3/0
ip address 50.0.0.1 255.0.0.0
no shutdown
- exit
router rip
network 40.0.0.0
network 50.0.0.0

Router - Configure Router

interface fastethernet 0/1
ip address 26.0.0.1 255.0.0.0
no shut
interface serial 2/0
ip address 30.0.0.1 255.0.0.0

no shut
exit

route rip
network 30.0.0.0
network 50.0.0.0

Observation

C 10.0.0.0/8 is directly connected
R 20.0.0.0/8 [120/1] via 40.0.0.2, 00:00:20
R 30.0.0.0/8 [120/2] via 40.0.0.2
C 40.0.0.0/8 is directly connected
R 50.0.0.0/8 via 40.0.0.2

ping

by ping 30.0.0.3
Reply from 30.0.0.3
Reply from 30.0.0.3
Reply from 30.0.0.3
Reply from 30.0.0.3

packets: send=4 Received=4 Lost=0

Select simulation

Select simple ADO

Select source to destination

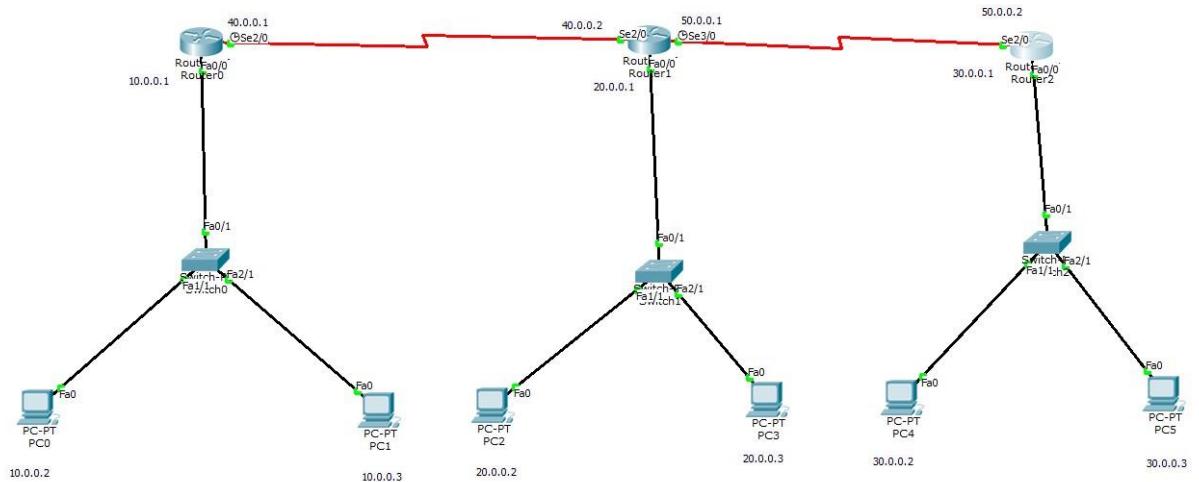
press autocapture

when the packets come to router press

click on the packet

ADO information

Screen Shots:



PC0

Physical Config Desktop Custom Interface

Command Prompt

```
Pinging 30.0.0.2 with 32 bytes of data:

Request timed out.
Reply from 30.0.0.2: bytes=32 time=7ms TTL=125
Reply from 30.0.0.2: bytes=32 time=6ms TTL=125
Reply from 30.0.0.2: bytes=32 time=7ms TTL=125

Ping statistics for 30.0.0.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 6ms, Maximum = 7ms, Average = 6ms

PC>ping 30.0.0.2

Pinging 30.0.0.2 with 32 bytes of data:

Reply from 30.0.0.2: bytes=32 time=4ms TTL=125
Reply from 30.0.0.2: bytes=32 time=7ms TTL=125
Reply from 30.0.0.2: bytes=32 time=7ms TTL=125
Reply from 30.0.0.2: bytes=32 time=7ms TTL=125

Ping statistics for 30.0.0.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 4ms, Maximum = 7ms, Average = 6ms

PC>
```

Program 7

Aim: Demonstrate the TTL/ Life of a Packet .

Topology , Procedure and Observation:

Exp-7.

Demonstrate TTL / life of a packet

TTL \rightarrow time to leave ,

Observation

Procedure:

• send a simple PDU from PC1 to PCU.

• Auto capture the event list then observe the
TTL of each router w/ PDU information.

Observation

when the packet passes Router 0

inbound PDU details :

TTL = 255 ms

outbound PDU details

R1 : TTL = 254 ms

inbound R1

inbound = 254 ms

outbound = 253 ms

R2

inbound = 253 ms

outbound = 252 ms

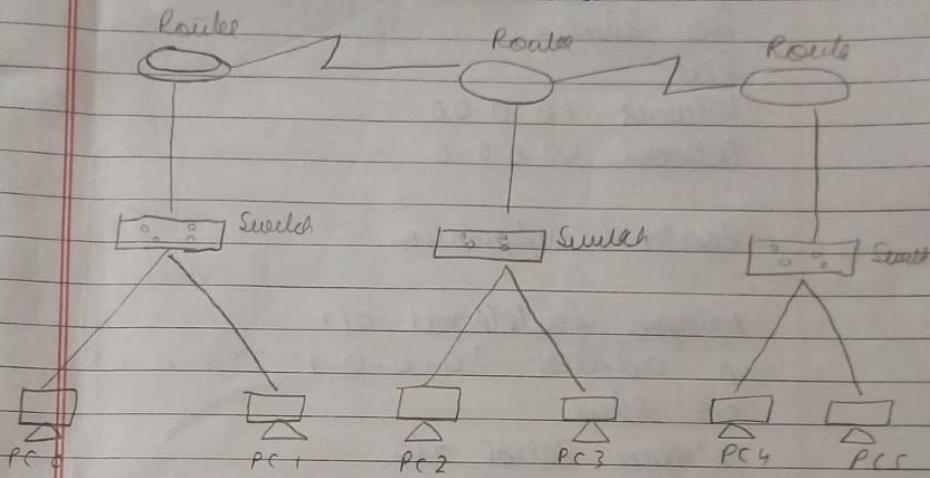
Hence, we can observe TTL decreases as it passes through
each of the router.

20/11/24

Bafna Gold
Date: _____
Page: _____

RIP Protocol

Aim - Configure routing information protocol in router



Topology - Six PCs are connected to respective switch in Pairs

Each switch is connected to respective router

Procedure - Set IP address for each PC and respective gateway

Configure Router 0

interface fastethernet 0/0
ip address 10.0.0.1 255.0.0.0

Screen Shots:

PDU Information at Device: Router0

OSI Model Inbound PDU Details Outbound PDU Details

At Device: Router0
Source: PC0
Destination: PC3

In Layers

Layer7
Layer6
Layer5
Layer4
Layer 3: IP Header Src. IP: 10.0.0.2, Dest. IP: 20.0.0.3 ICMP Message Type: 8
Layer 2: Ethernet II Header 000A.41E3.E33A >> 0010.11A0.4697
Layer 1: Port FastEthernet0/0

Out Layers

Layer7
Layer6
Layer5
Layer4
Layer 3: IP Header Src. IP: 10.0.0.2, Dest. IP: 20.0.0.3 ICMP Message Type: 8
Layer 2: HDLC Frame HDLC
Layer 1: Port(s): Serial2/0

1. FastEthernet0/0 receives the frame.

Challenge Me << Previous Layer Next Layer >>

PDU Information at Device: Router0

OSI Model Inbound PDU Details Outbound PDU Details

PDU Formats

Ethernet II

0	4	8	14	19	Bytes
PREAMBLE: 101010...1011		DEST MAC: 0010.11A0.4697		SRC MAC: 000A.41E3.E33A	
TYPE: 0x800		DATA (VARIABLE LENGTH)		FCS: 0x0	

IP

0	4	8	16	19	31 Bits
4		IHL	DSCP: 0x0	TL: 28	
		ID: 0xa	0x0	0x0	
TTL: 255		PRO: 0x1	CHKSUM		
		SRC IP: 10.0.0.2			
		DST IP: 20.0.0.3			
OPT: 0x0		0x0			
DATA (VARIABLE LENGTH)					

ICMP

0	8	16	31 Bits
TYPE: 0x8	CODE: 0x0	CHECKSUM	

PDU Information at Device: Router0

OSI Model Inbound PDU Details Outbound PDU Details

PDU Formats

HDLC

0	8	16	32	32+x	48+x	56+x
FLG: 0111 1110	ADR: 0x8f	CONTROL: 0x0	DATA: (VARIABLE LENGTH)	FCS: 0x0	FLG: 0111 1110	

IP

0	4	8	16	19	31 Bits
4	IHL	DSCP: 0x0		TL: 28	
		ID: 0xa	0x0	0x0	
TTL: 254		PRO: 0x1		CHKSUM	
		SRC IP: 10.0.0.2			
		DST IP: 20.0.0.3			
		OPT: 0x0		0x0	
		DATA (VARIABLE LENGTH)			

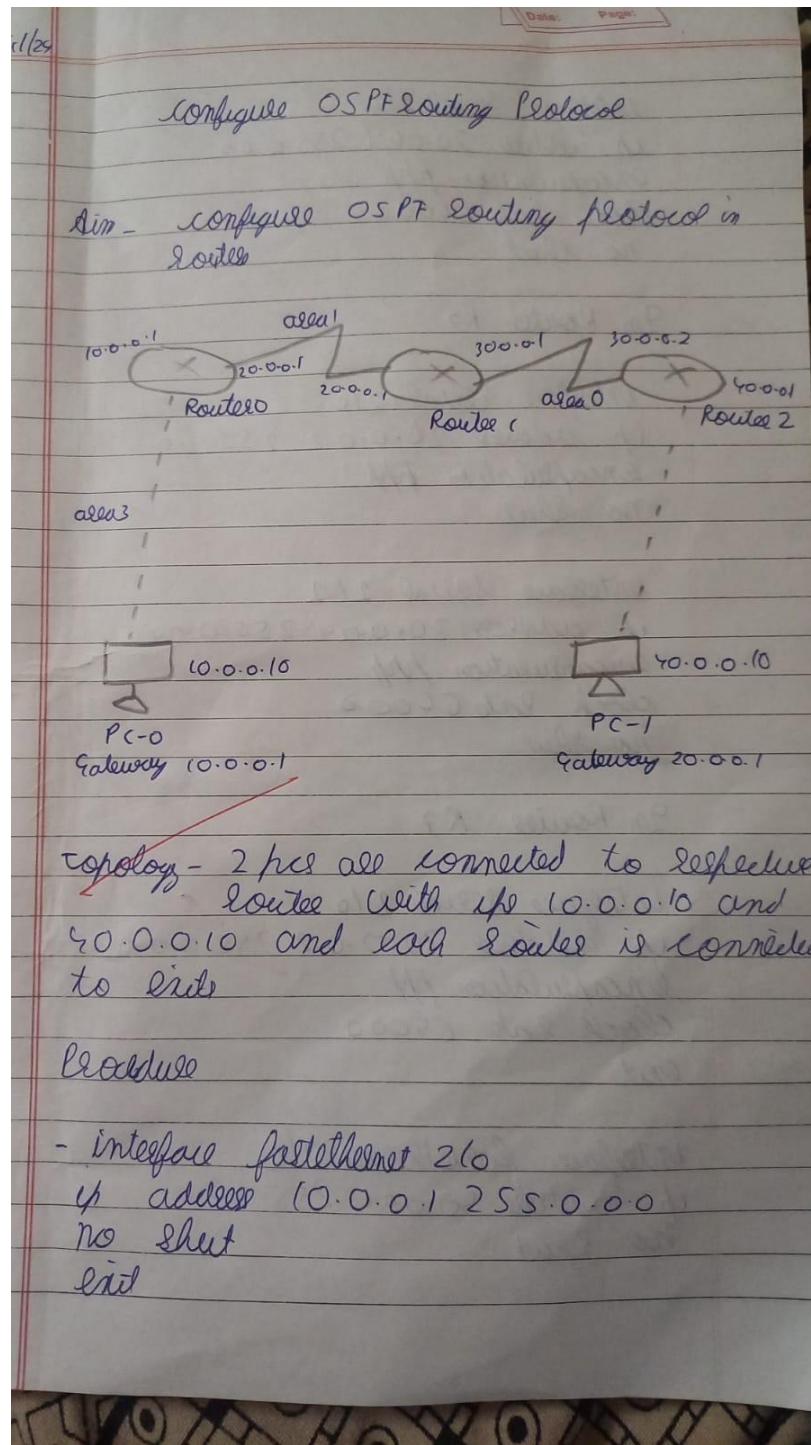
ICMP

0	8	16	31 Bits
TYPE: 0x8	CODE: 0x0	CHECKSUM	
ID: 0x5		SEQ NUMBER: 10	

Program 8

Aim: Configure OSPF routing protocol .

Topology , Procedure and Observation:



interface serial 2/0
ip address 20.0.0.1 255.0.0.0
encapsulation ~~ppp~~
clock rate 64000
no shutdown

In Router R2

interface serial 2/0
ip address 20.0.0.2 255.0.0.0
encapsulation ~~ppp~~
no shutdown

interface serial 1/1
ip address 30.0.0.1 255.0.0.0
encapsulation ~~ppp~~
clock rate 64000
no shutdown

In Router R3

interface serial 2/0
ip address 30.0.0.2 255.0.0.0
encapsulation ~~ppp~~
clock rate 64000
exit

interface fastethernet 2/0
ip address 40.0.0.1 255.0.0.0
no shutdown

In Router R1

Router OSPF 1

Router-id 1.1.1.1

Network 10.0.0.0 255.255.0 area 3

Network 20.0.0.6 255.255.255 area 1

In Router R2

Router OSPF 1

Router-id 2.2.2.2

Network 20.0.0.0 255.255.255 area 1

Network 30.0.0.0 255.255.255 area 0

In Router R3

Router OSPF 1

Router-id 3.3.3.3

Network 30.0.0.0 255.255.255 area 0

Network 40.0.0.0 255.255.255 area 2

DM

Output

show ip route

C 10.0.0.0/8

C 20.0.0.0/8

O 40.0.0.0/8

O 30.0.0.0/8

1

Router 1
interface loopback 0
ip add 172.16.1.252 255.255.0.0
no shut

In Router 2

interface loopback 0
ip add 172.16.1.253 255.255.0.0
no shut

In Router 3

interface loopback 0
ip add 172.16.1.254 255.255.0.0
no shut

Observation

- O 20.0.0.0/8
- C 40.0.0.0/8
- C 30.0.0.0/8

Router 1

router ospf 1
area 1 virtual link 2.2.2.2
loading zone

In Router 2

virtual link not found

loading Done

R₂ ≠ 1 aces (virtual link 1-1-1)
exit

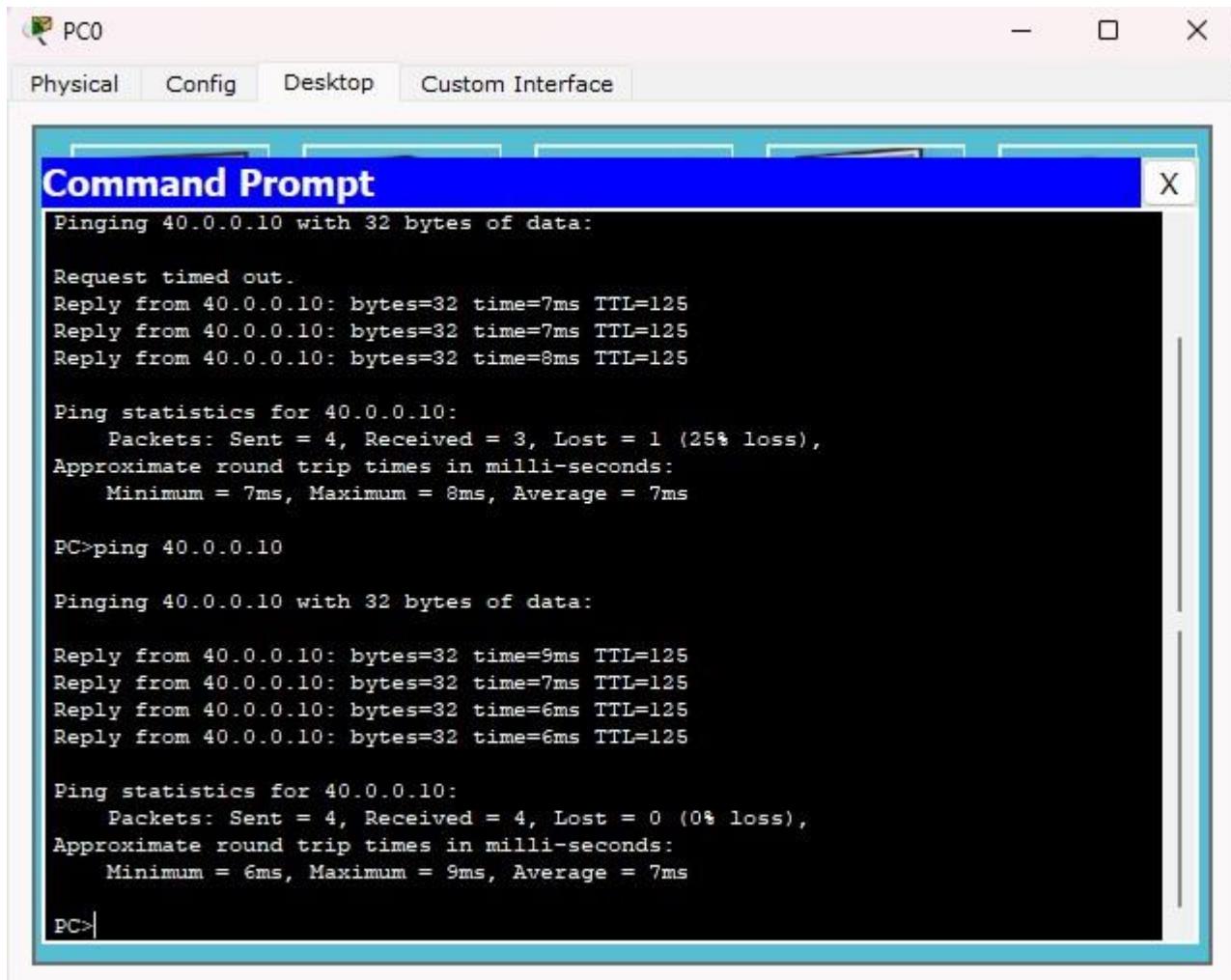
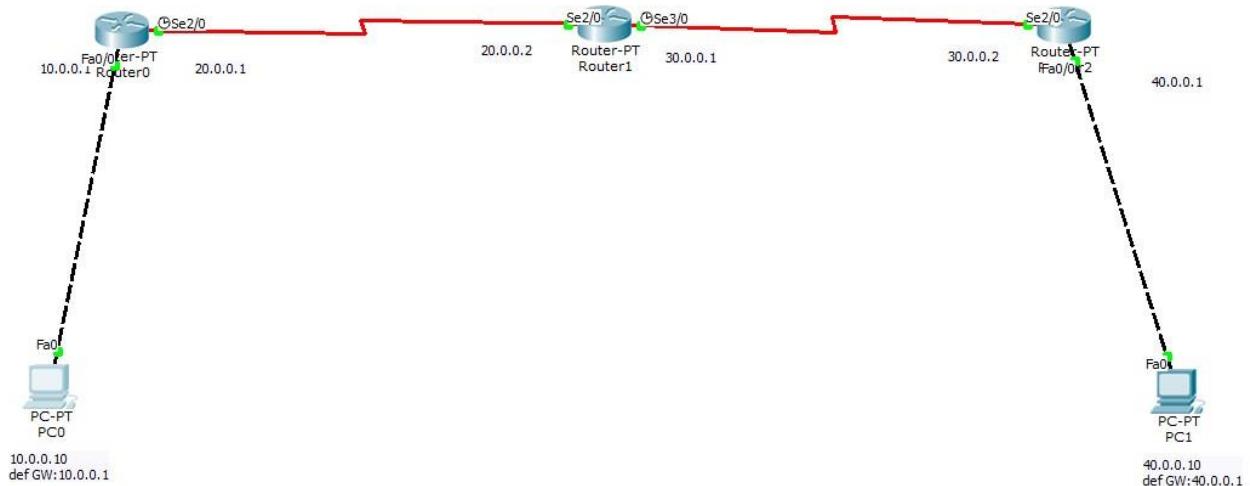
Observation

ping 40.0.0.10

packets sent = 5 Received = 5 Host = 0

Ree 27/11/14

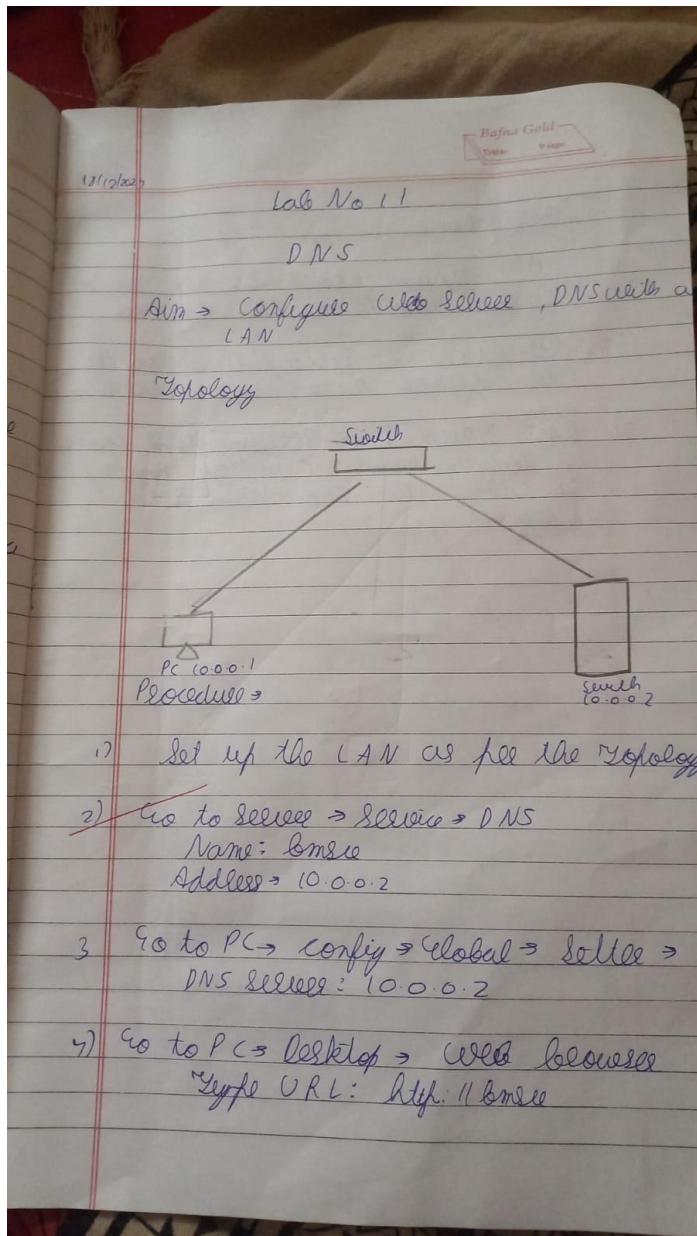
Screen Shots:



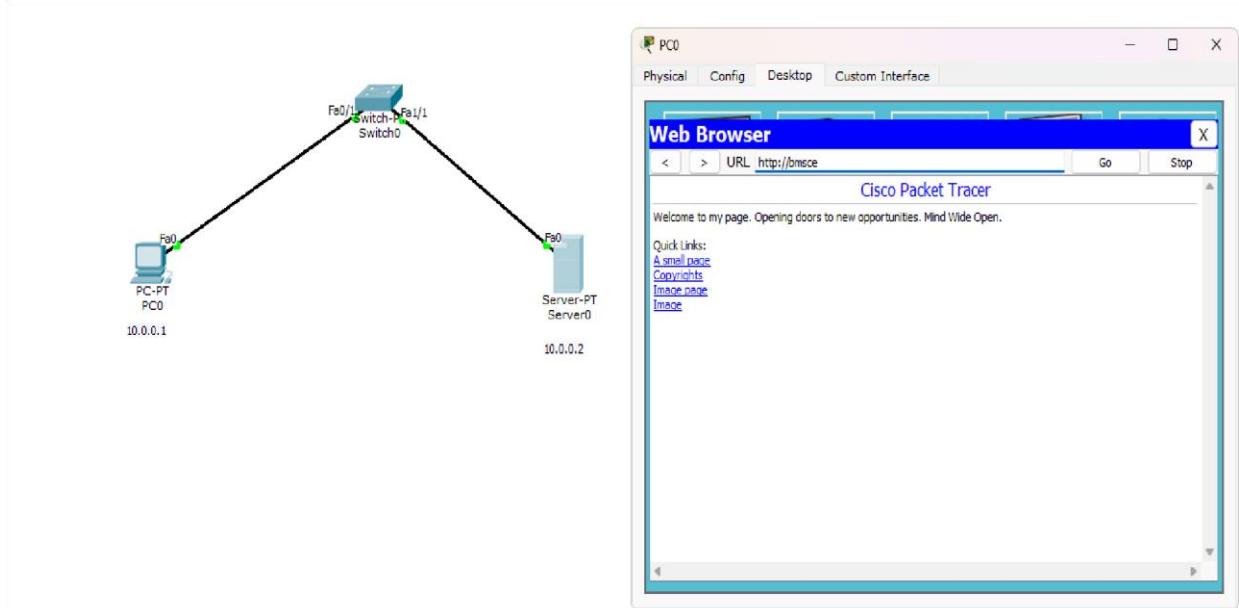
Program 9

Aim: Configure Web Server, DNS within a LAN.

Topology , Procedure and Observation:



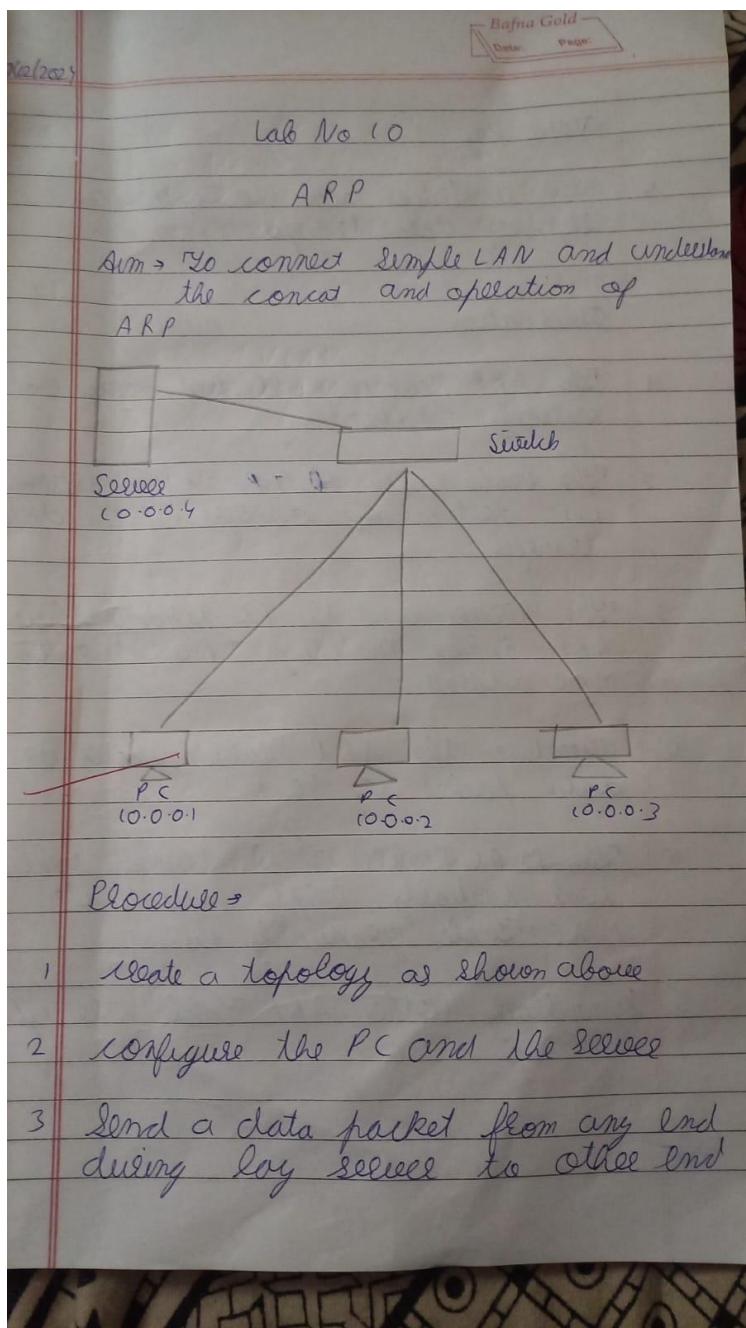
Screen Shots:



Program 10

Aim: To construct simple LAN and understand the concept and operation of Address Resolution Protocol (ARP)

Topology , Procedure and Observation:



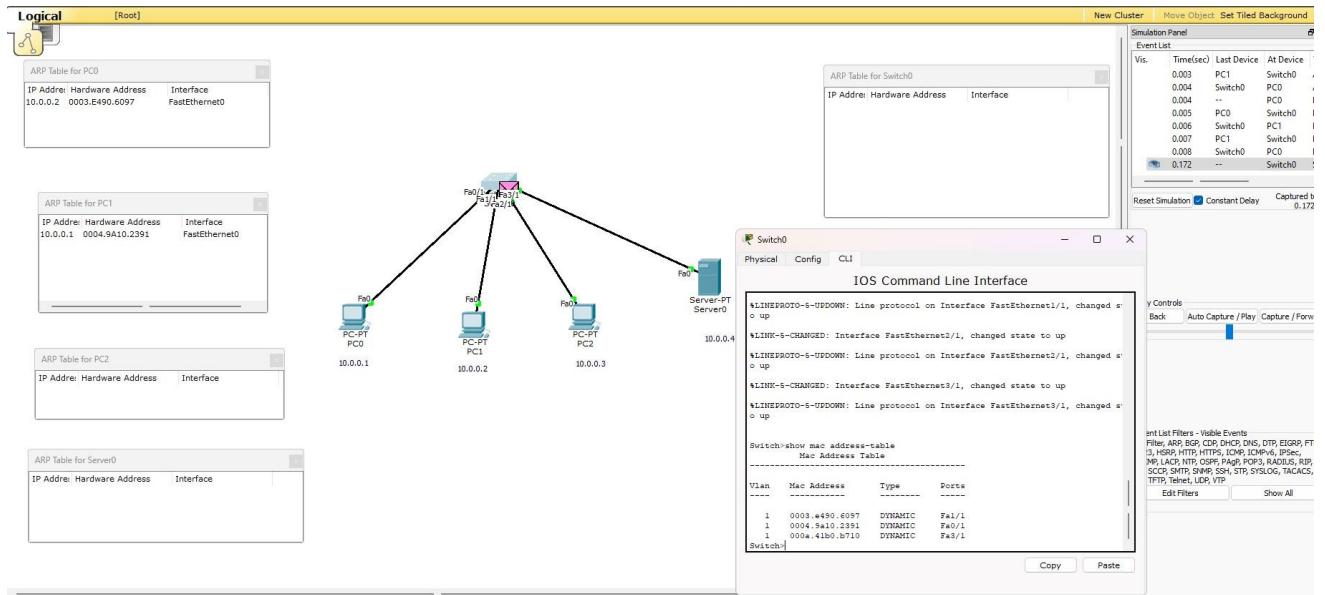
device lay

- open simulation mode to capture each step of data transfer

Observation

- The ARP Table of all end devices are initially empty
- The device with the IP address present in destination address of the data transfer
- The switch and the PC update their ARP tables matching IP address to MAC address
- Over time the ARP Table grows as data packets are sent
- The MAC Table of the switch which was initially empty update its MAC Table gradually to
- Similarly other ARP Table are updated

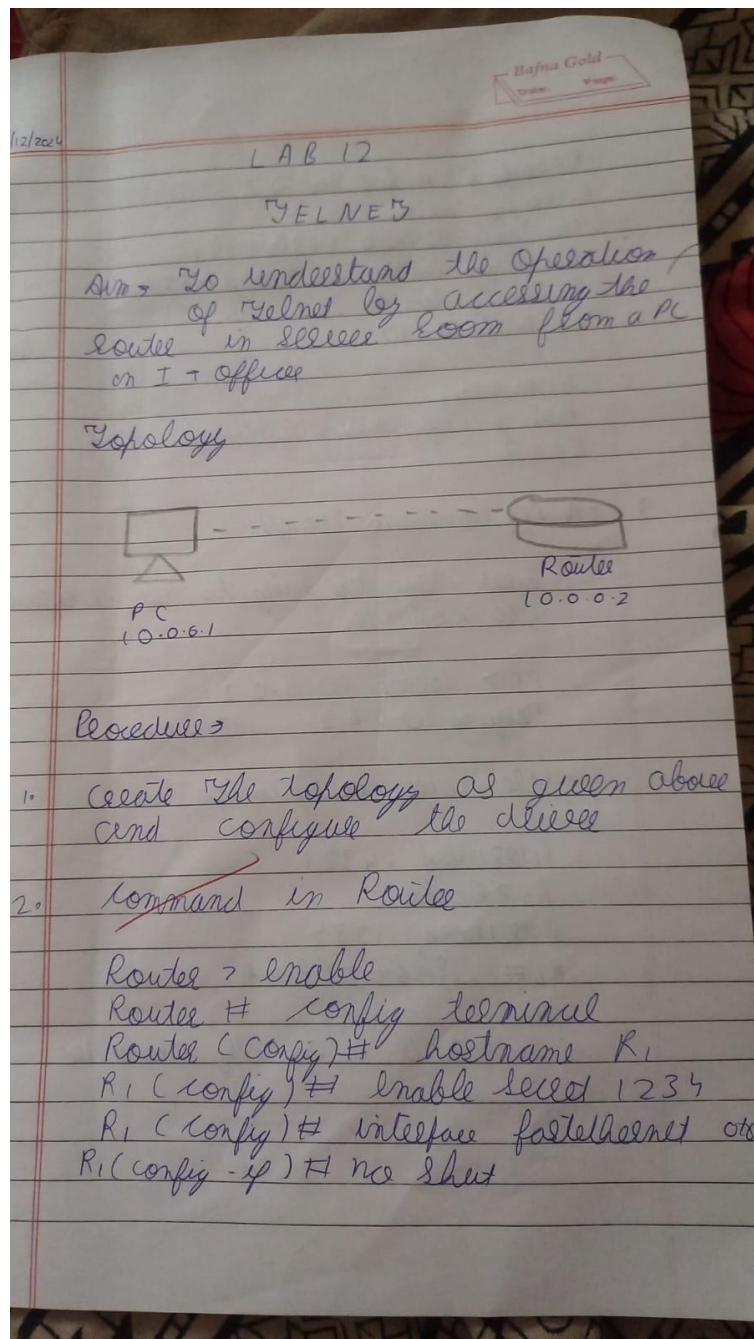
Screen Shots:



Program 11

Aim: To understand the operation of TELNET by accessing the router in the server room from a PC in the IT office.

Topology , Procedure and Observation:



R1(config)# line vty 0 3
R1(config-line)# login
%o login desirable on line 194
R1(config-line)# password 4321
R1(config-line)# exit
R1(config)# exit

R1# use
Building configuration--
[OK]

3 IN PC: command Prompt

First try to ping to see if device are connected

PC> telnet 10.0.0.2
Trying 10.0.0.2 - open

Use access ~~recognition~~

Password : 4321

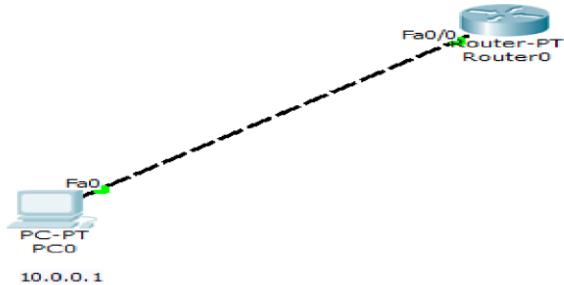
R1> enable

Password : 1234

R1# show IP config

(10.0.0.0/8 is directly connected)

Screen Shots:



Command Prompt

```
Packet Tracer PC Command Line 1.0
PC>ping 10.0.0.2

Pinging 10.0.0.2 with 32 bytes of data:

Reply from 10.0.0.2: bytes=32 time=0ms TTL=255

Ping statistics for 10.0.0.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

PC>telnet 10.0.0.2
Trying 10.0.0.2 ...Open

User Access Verification

Password:
R1>enable
Password:
R1#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
      * - candidate default, U - per-user static route, o - ODR
      P - periodic downloaded static route

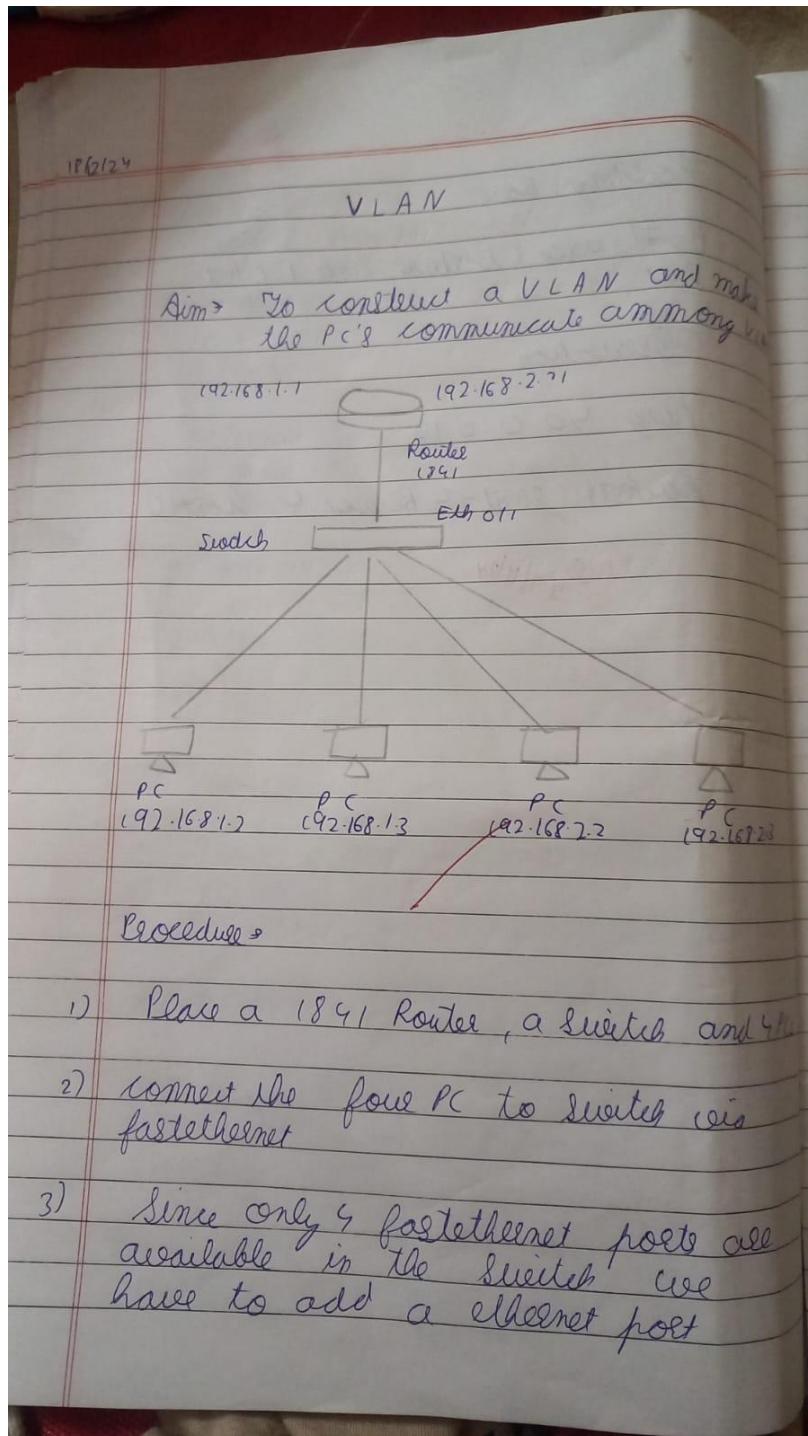
Gateway of last resort is not set

C    10.0.0.0/8 is directly connected, FastEthernet0/0
R1#|
```

Program 12

Aim: To construct a VLAN and make the PC's communicate among a VLAN .

Topology , Procedure and Observation:



- Bafna Gold
Date: _____ Page: _____
- 4) switch off power button of switch
 - Add the Ethernet Port to switch
 - 5) Add the switch on Power button
 - connect Router to switch
 - 5) In the switch go to config tab
 - Select VLAN Database
 - Give VLAN number say 2
 - Add it to Database
 - 6) Select the switch
 - Go to Config
 - Go to Ethernet 6/1 is connected to Router
 - 7) Configure the PC as shown in Topology
 - 8) ~~Select switch~~
 - Go to Config
 - Go to fastethernet 2/1
 - Set VLAN number as 2
 - 9) Configure the Router
- Router (config)# interface fastethernet
Router (config-if)# no shutdown
Router (config-if)# exit

Now to Configure the Router VLAN interface
Router (config)# interface fastethernet 0/0
Router (config-subif)# encapsulation dot1q
Router (config-subif)# no shutdown
Router (config-subif)# shutdown

10/12/2020

10

Observation

1) When device are pinged within same LAN

→ Pinging 192.168.1.3 from 192.168.1.2

⇒ The data packet doesn't go to Router

2) When device ping a device of another LAN

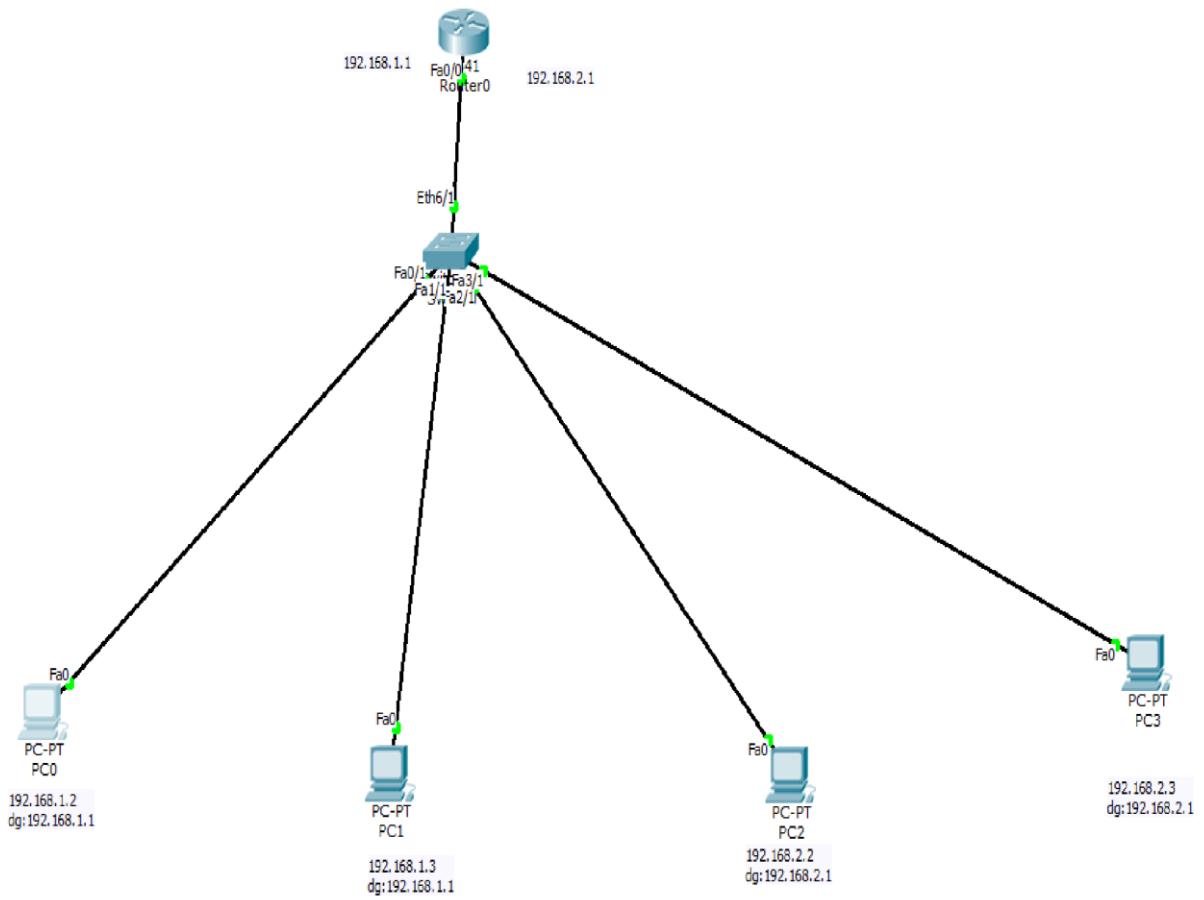
→ Pinging 192.168.2.3 from 192.168.1.2

The data packet journey is as follows

192.168.1.2 → Switch → Router

192.168.1.3 ← Switch ←

Screen Shots:



Command Prompt

```
Packet Tracer PC Command Line 1.0
PC>ping 192.168.2.2

Pinging 192.168.2.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.2.2: bytes=32 time=0ms TTL=127
Reply from 192.168.2.2: bytes=32 time=0ms TTL=127
Reply from 192.168.2.2: bytes=32 time=4ms TTL=127

Ping statistics for 192.168.2.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 4ms, Average = 1ms

PC>ping 192.168.2.2

Pinging 192.168.2.2 with 32 bytes of data:

Reply from 192.168.2.2: bytes=32 time=0ms TTL=127
Reply from 192.168.2.2: bytes=32 time=0ms TTL=127
Reply from 192.168.2.2: bytes=32 time=2ms TTL=127
Reply from 192.168.2.2: bytes=32 time=0ms TTL=127

Ping statistics for 192.168.2.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 2ms, Average = 0ms

PC>ping 192.168.2.3

Pinging 192.168.2.3 with 32 bytes of data:

Request timed out.
Reply from 192.168.2.3: bytes=32 time=3ms TTL=127
Reply from 192.168.2.3: bytes=32 time=2ms TTL=127
Reply from 192.168.2.3: bytes=32 time=1ms TTL=127

Ping statistics for 192.168.2.3:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 3ms, Average = 2ms

PC>ping 192.168.2.3

Pinging 192.168.2.3 with 32 bytes of data:

Reply from 192.168.2.3: bytes=32 time=0ms TTL=127
Reply from 192.168.2.3: bytes=32 time=0ms TTL=127
Reply from 192.168.2.3: bytes=32 time=2ms TTL=127
Reply from 192.168.2.3: bytes=32 time=0ms TTL=127

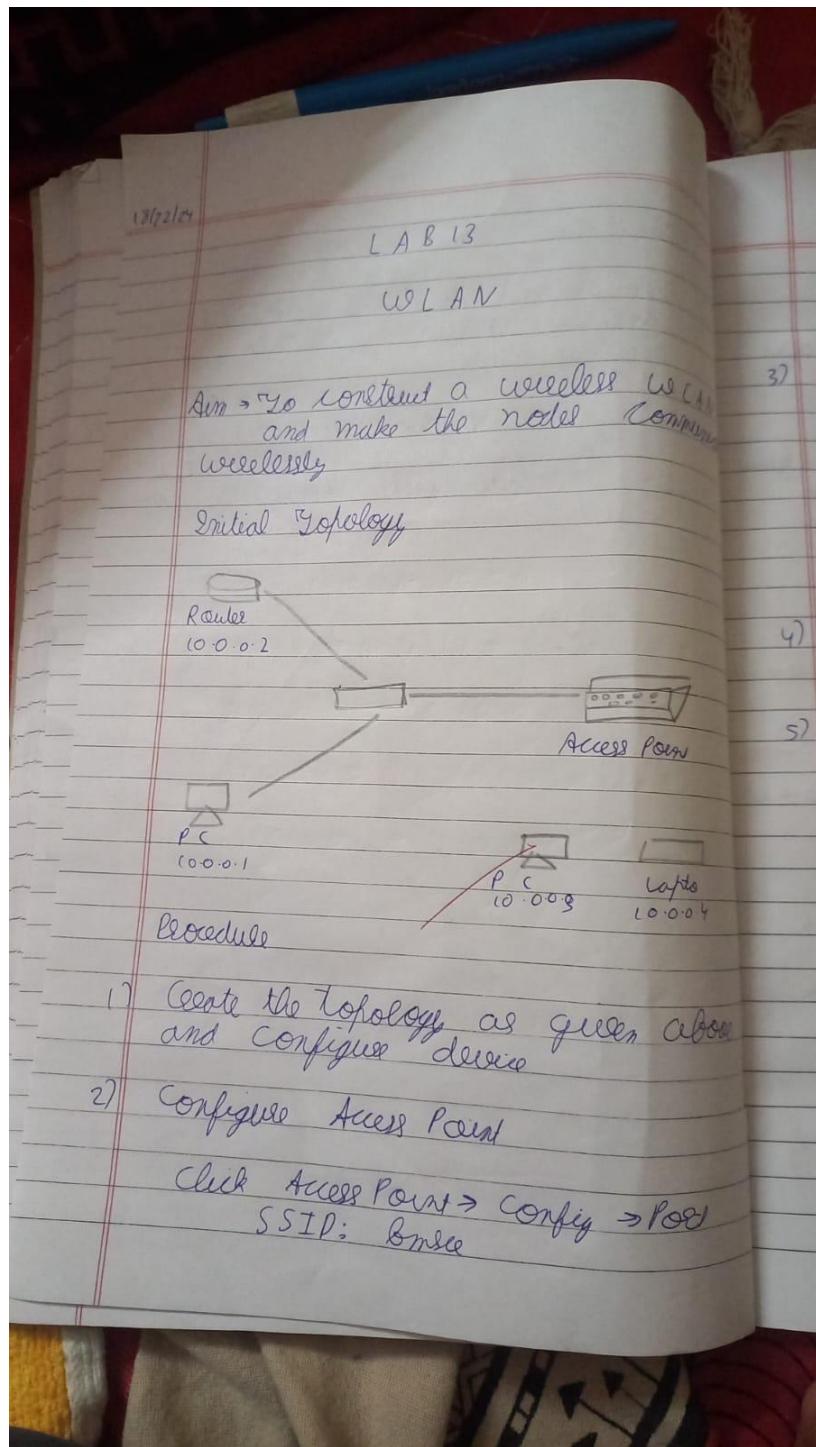
Ping statistics for 192.168.2.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 2ms, Average = 0ms

PC>
```

Program 13

Aim: To construct a WLAN and make the nodes communicate wirelessly.

Topology , Procedure and Observation:



Select : WEP

Set Key : 1234567890

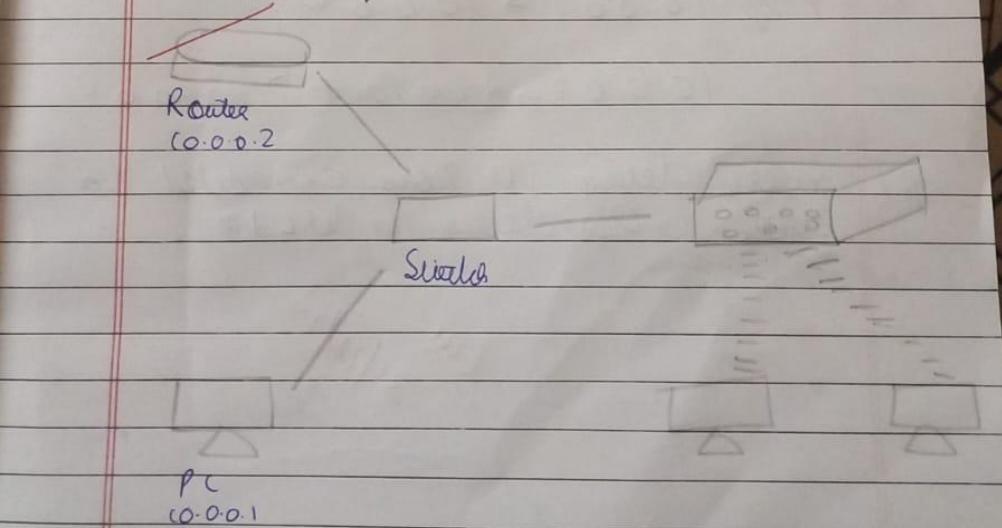
3) Config PC and Laptop with Wireless Standards

- Select off device
- Drag WMP 300 N wireless interface to the empty port
- Select off the device

4) In the config tab, a new wireless interface was added

5) Config the device by entering SSID
WEP key, IP address & Gateway

Topology after wireless configuration



Observation

1. We were able to ping from every device to every other device

2. WMP 300 N wireless interface:
→ Wireless network adapt that enables device to communicate with access point using wireless signals

3. Pinging 10.0.0.1 → 10.0.0.3

10.0.0.1 → Switch → Access Point → 10.0.0.3

4) Pinging 10.0.0.3 to 10.0.0.1

10.0.0.3 → Access Point → Switch → 10.0.0.1

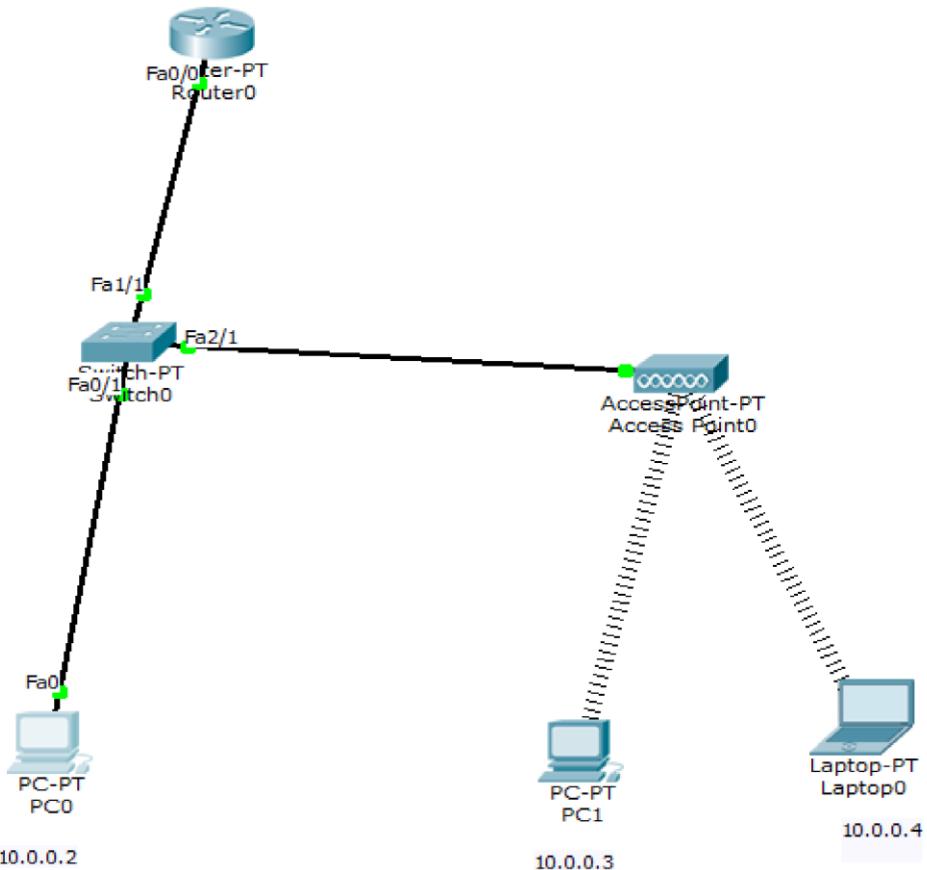
5) Ping 10.0.0.3 → 10.0.0.4

10.0.0.3 → Access Point → 10.0.0.4

6) Every device is now connected to every other device in WLAN

*Aut
31/12/2011*

Screen Shots:



PC0

Physical Config Desktop Custom Interface

Command Prompt

```

Packet Tracer PC Command Line 1.0
PC>ping 10.0.0.3

Pinging 10.0.0.3 with 32 bytes of data:

Reply from 10.0.0.3: bytes=32 time=22ms TTL=128
Reply from 10.0.0.3: bytes=32 time=6ms TTL=128
Reply from 10.0.0.3: bytes=32 time=3ms TTL=128
Reply from 10.0.0.3: bytes=32 time=7ms TTL=128

Ping statistics for 10.0.0.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 3ms, Maximum = 22ms, Average = 9ms

PC>ping 10.0.0.4

Pinging 10.0.0.4 with 32 bytes of data:

Reply from 10.0.0.4: bytes=32 time=19ms TTL=128
Reply from 10.0.0.4: bytes=32 time=5ms TTL=128
Reply from 10.0.0.4: bytes=32 time=6ms TTL=128
Reply from 10.0.0.4: bytes=32 time=7ms TTL=128

Ping statistics for 10.0.0.4:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 5ms, Maximum = 19ms, Average = 9ms

PC>

```

PART-B

Program 14

Write a program for error detecting code using CRC-CCITT (16-bits).

Code :

```
Bafna Gold  
Gullon  
Date: _____  
Page: _____  
  
Program 9  
CRC-CCITT (16-Bit)  
  
def cc.crc_ccitt(data: str, polynomial:  
    int = 0x1021, initial_val:  
    int = 0xFFFF) ->  
  
    int:  
        data_bytes = data.encode()  
        crc = initial_value  
  
        for byte in data_bytes:  
            if byte >= 8:  
                for i in range(8):  
                    if crc & 0x8000:  
                        crc = (crc << 1) ^ polynomial  
                    else:  
                        crc <<= 1  
                    crc &= 0xFFFF  
            else:  
                crc <<= 1  
  
def main():  
  
    message = input("Enter the message: ")  
    cc = cc.crc_ccitt(message)  
    print(f"RCCIT (16 Bit) value for  
    message '{message}' is {hex(cc)}")  
  
    if name == "-main":  
        main()
```

Output

```
Enter data: 1100110
Enter generator polynomial: 1101
CRC: 100
Transmitted Data: 1100110100
Enter received data: 1100110100
No Error

==== Code Execution Successful ===
```

Program 15

Write a program for congestion control using Leaky bucket algorithm.

Code :

Cycle - 2

Program 2

L Bucket.c

```
int main()
{
    int packet_size[NOF_PACKETS], B_dly;
    float p_dly, p_sq_m=0, p_sq_dly;
    for(i=0; i<NOF_PACKETS; i++)
        packet_size[i] = Random(1)%100;
    for(i=0; i<NOF_PACKETS; i++)
        p_sq_dly = Unpacker["od"] * dly_Bigle[i];
    cout ("Enter the Output rate : ");
    cin ("%d", &p_dly);
    cout ("Bucket Capacity : ");
    cin ("%d", &B_dly);

    for(i=0; i<NOF_PACKETS; i++)
        if (packet_size[i] + p_sq_m) > B_dly)
            if (packet_size[i] > B_dly)
                cout ("In Bucket Capacity Exceeded");
        else
            p_sq_m += packet_size[i];
            cout ("Bytes Remaining to Transmit : ");
            cout (p_sq_m);
    while (p_sq_m != 0)
        sleep(1);
    if (p_sq_m) {
}
```

```
if (h_sz_m == 0_sz) {  
    or = h_sz_m, h_sz_m = 0;  
} else {  
    or = 0_sz, h_sz_m = 0_sz;  
}  
>    penalty ("No packet of size " + sz + " transmitted");  
else {  
    penalty ("No packet to transmit");  
}  
y  
z  
zz
```

Output

Clear

Generated packets: [80, 63, 57, 12, 69]

Enter bucket size: 60

Enter output rate: 30

Packet of size 80 bytes exceeds bucket capacity (60 bytes) - REJECTED

Packet of size 63 bytes exceeds bucket capacity (60 bytes) - REJECTED

Packet of size 57 bytes added to bucket

Bytes in bucket: 57

Transmitting 30 bytes

Bytes remaining in bucket: 27

Transmitting 27 bytes

Bytes remaining in bucket: 0

Packet of size 12 bytes added to bucket

Bytes in bucket: 12

Transmitting 12 bytes

Bytes remaining in bucket: 0

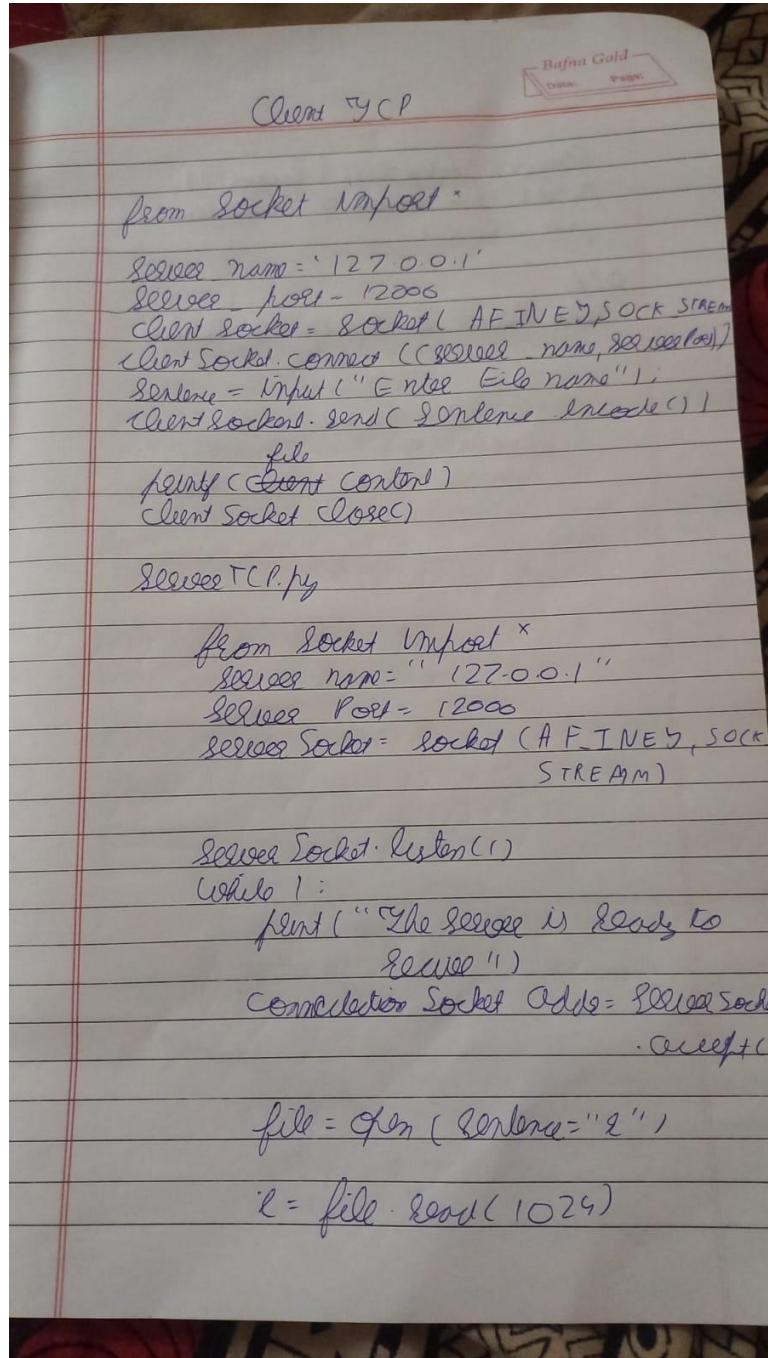
Packet of size 69 bytes exceeds bucket capacity (60 bytes) - REJECTED

==== Code Execution Successful ===

Program 16

Using TCP/IP sockets, write a client-server program to make the client send the file name and the server to send back the contents of the requested file if present.

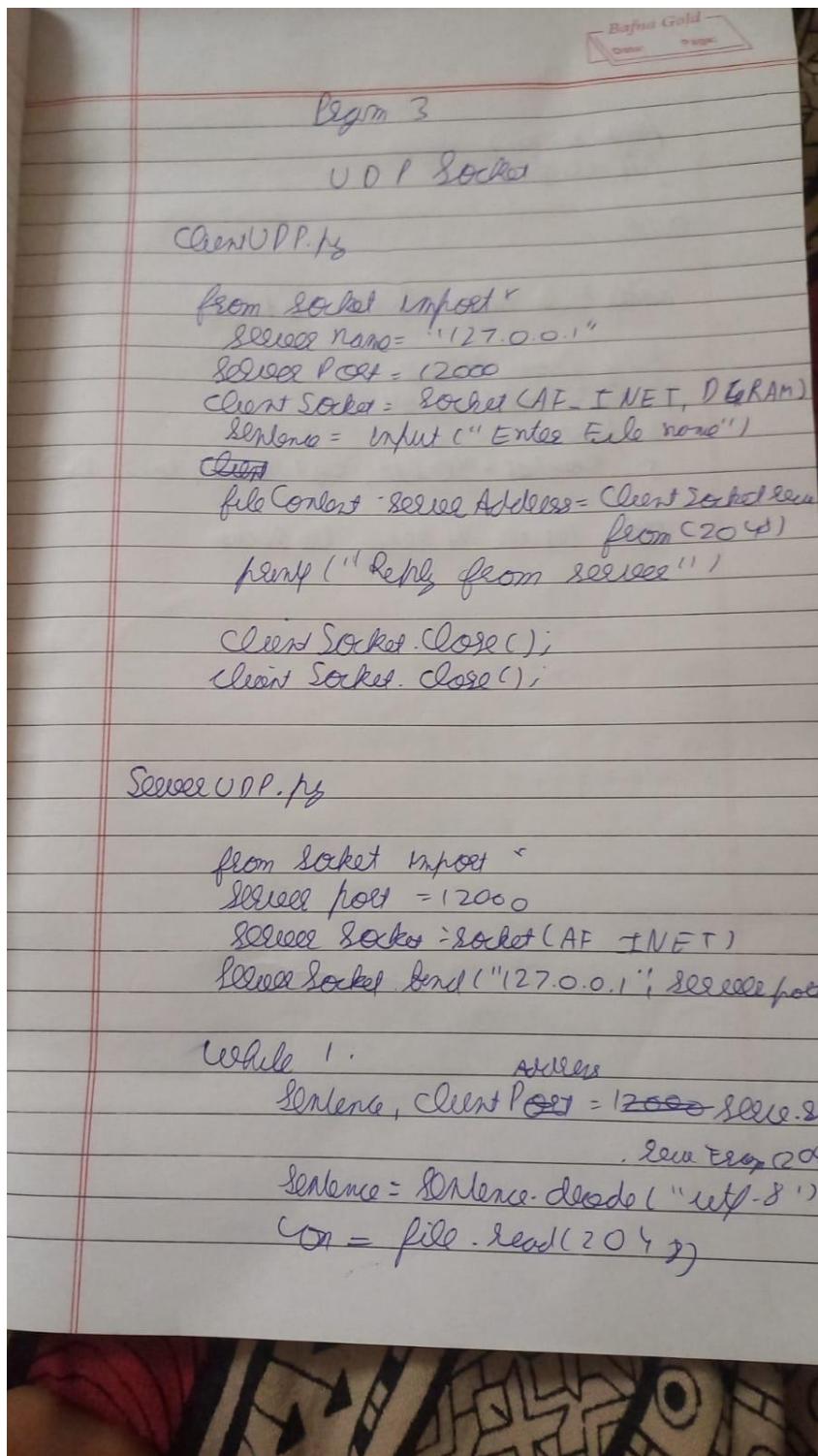
Code and Output:



Program 17

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

Code and Output:



from (socket)
file.close()

OIP

Enter File name: server UDP.py

Reply from server:
from socket import *

s_port = 12000

s_socket = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)

The server is ready to receive