VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"JnanaSangama", Belgaum -590014, Karnataka.

LAB REPORT
on

COMPUTER NETWORK LAB

Submitted by

ANAGHA M S (1BM21CS022)

in partial fulfillment for the award of the degree of
BACHELOR OF ENGINEERING
in
COMPUTER SCIENCE AND ENGINEERING

B.M.S. COLLEGE OF ENGINEERING
(Autonomous Institution under VTU)
BENGALURU-560019
JUN-2023 to SEP-2023

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 "LAB COURSE **TITLE**" carried out by **ANAGHA M S (1BM21CS022),** who is bonafide student of **B. M. S. College of Engineering.** It is in partial fulfillment for the award of **Bachelor of Engineering in Computer Science and Engineering** of the Visvesvaraya Technological University, Belgaum during the year 2023. The Lab report has been approved as it satisfies the academic requirements in respect of a **Course** a **Computer Networks - (22CS4PCCON)** work prescribed for the said degree.

Dr Shyamala G

Assistant Professor
Department of CSE
BMSCE, Bengaluru

Dr. Jyothi Nayak

Professor anHead
Department of CSE
BMSCE, Bengaluru

`

Index

		communicate among a VLAN	
10)	10/8/23	Demonstrate the TTL/ Life of a Packet	95-101
11)	10/8/23	To construct a WLAN and make the nodes communicate wirelessly	102-110
12)	10/8/23	To understand the operation of TELNET by accessing the router in server room from a PC in IT office	111-115
13)	17/8/23	Write a program for error detecting code using CRCCCITT (16-bits)	116-121
14)	17/8/23	Write a program for congestion control using Leaky bucket algorithm	122-126
15)	24/08/23	Using TCP/IP 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	127-131
16)	24/08/23	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	132-135
17)	31/08/23	Tool Exploration -Wireshark	136-138

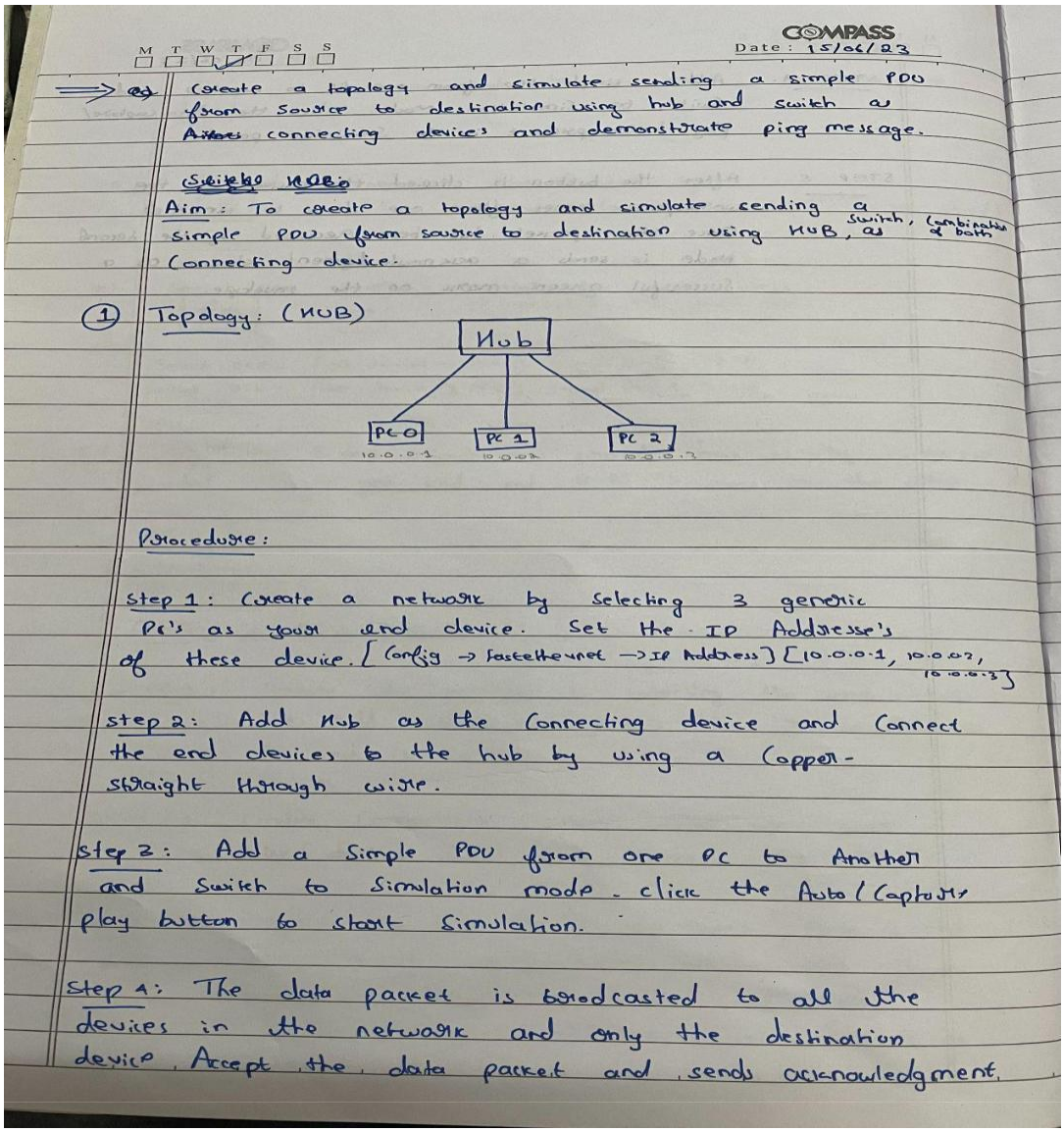
LAB PROGRAM 1

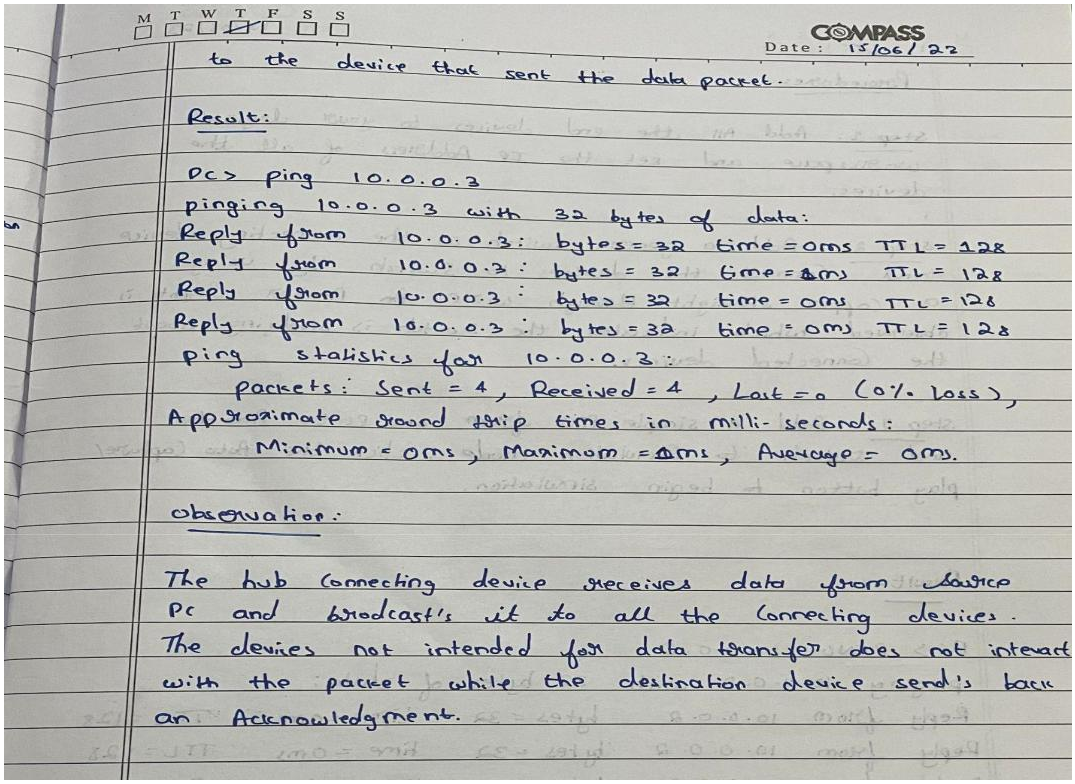
Aim:

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

1) Hub

Procedure :





to the device that sent the data packet.

Result:

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 = 0ms TTL = 128

Reply from 10.0.0.3: bytes = 32 time = 0ms TTL = 128

Reply from 10.0.0.3: bytes = 32 time = 0ms TTL = 128

Reply from 10.0.0.3: bytes = 32 time = 0ms TTL = 128

ping statistics for 10.0.0.3:

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

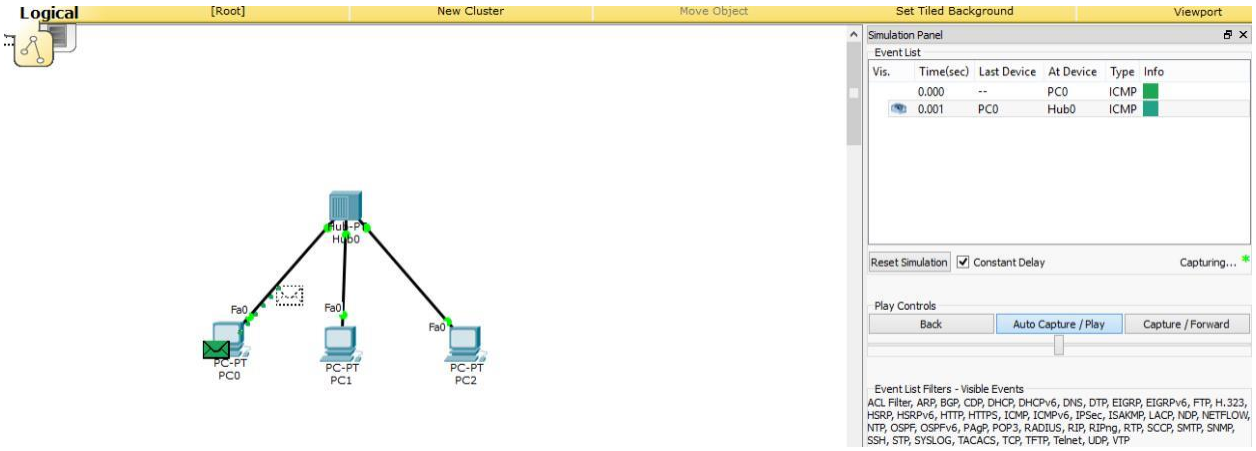
Approximate round trip times in milli-seconds:

Minimum = 0ms, Maximum = 0ms, Average = 0ms.

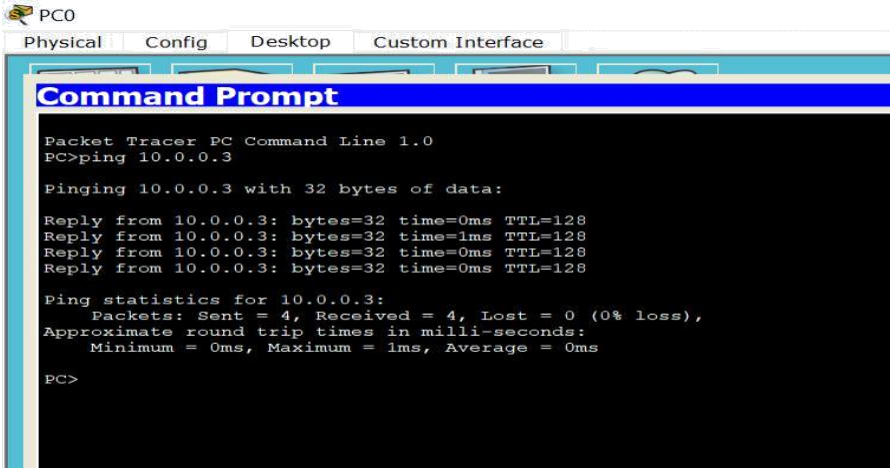
observation:

The hub connecting device receives data from source Pc and brodcast's it to all the connecting devices. The devices not intended for data transfer does not interact with the packet while the destination device send's back an Acknowledgment.

Topology :



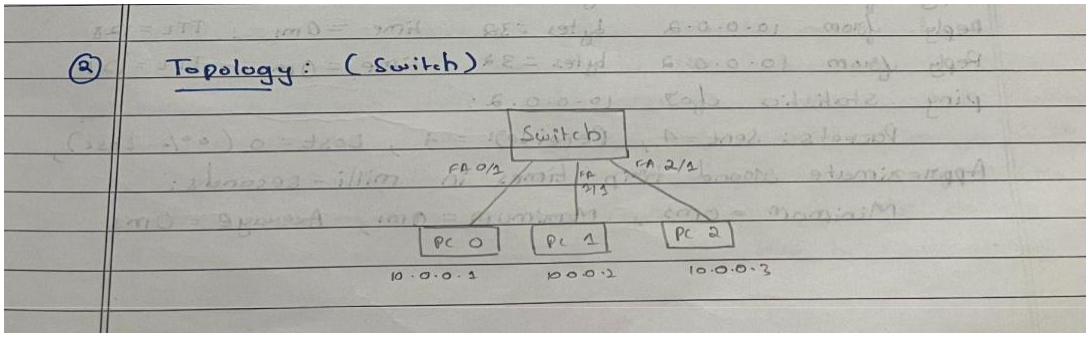
Ping Result :

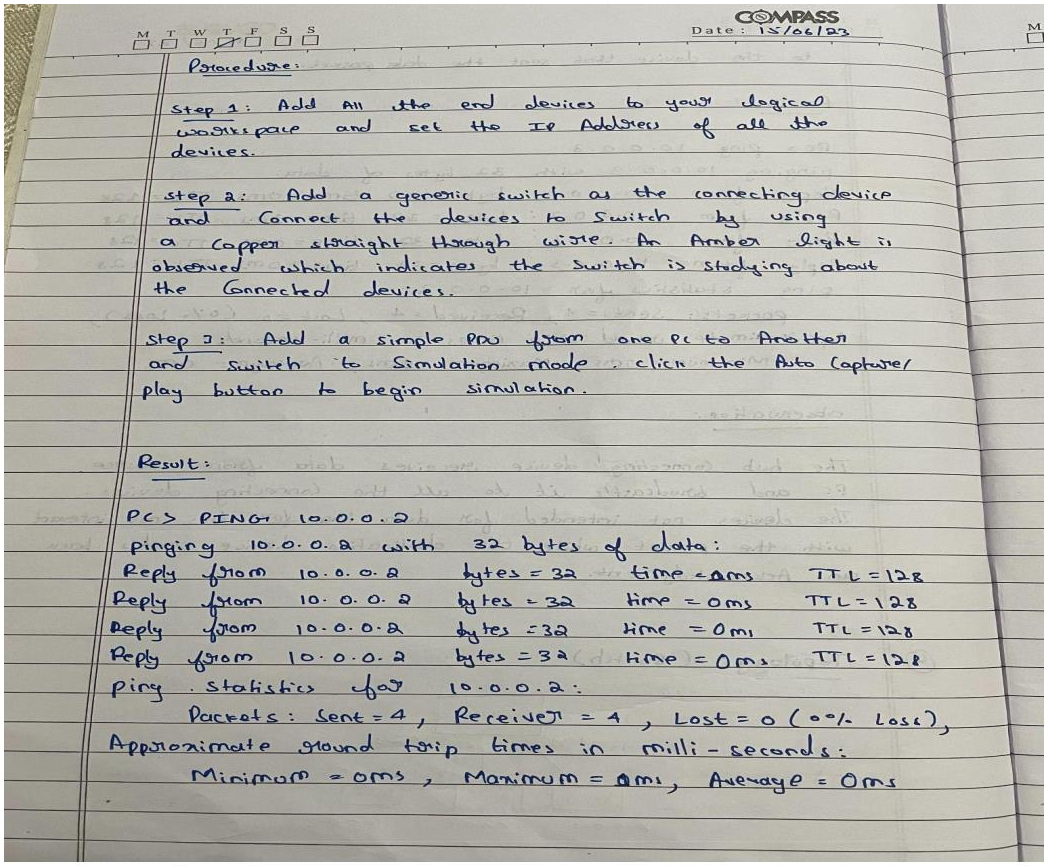


---

2) Switch

Procedure :





Procedure:

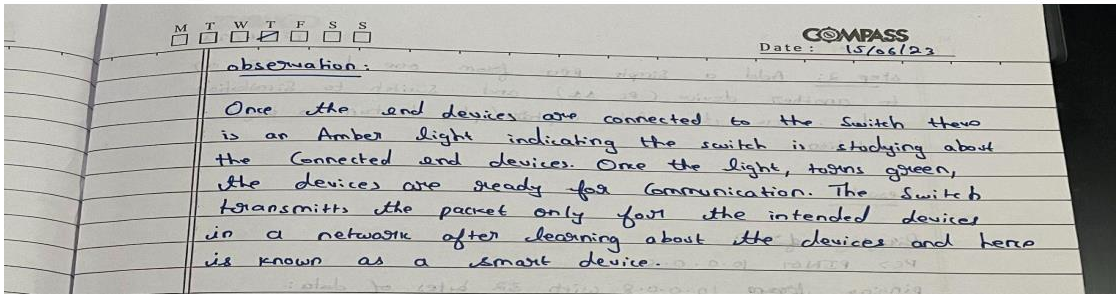
Step 1: Add All the end devices to your logical workspace and set the IP Addresses of all the devices.

step 2: Add a generic switch as the connecting device and Connect the devices to Switch by using a Copper straight Through wire. An Amber light is observed which indicates the Switch is studying about the Connected devices.

step 3: Add a simple PDU from one PC to Another and Switch to Simulation mode. click the Auto Capture/ play button to begin simulation.

Result:

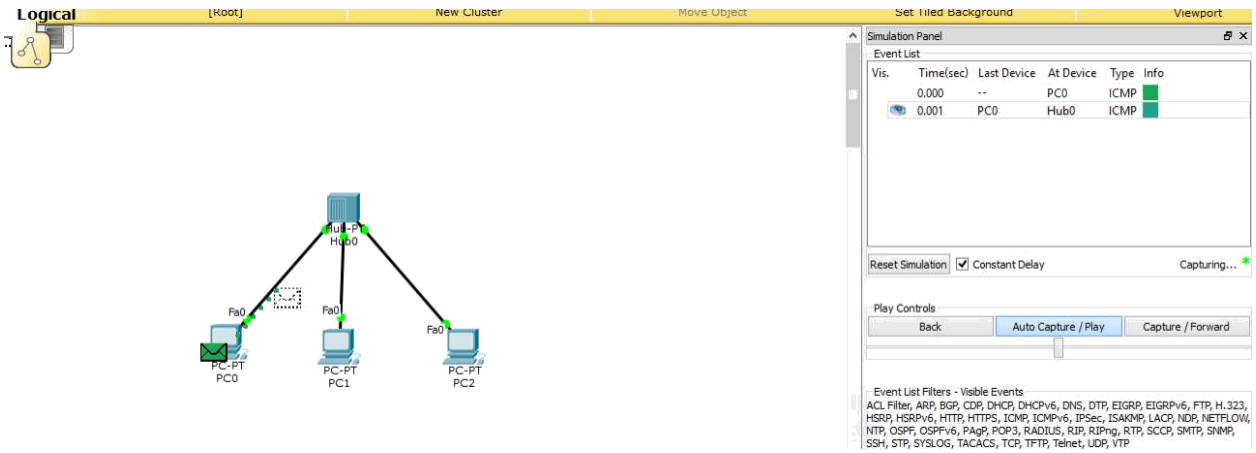
PC> PING 10.0.0.2
pinging 10.0.0.2 with 32 bytes of data:
Reply from 10.0.0.2 bytes = 32 time = 0ms TTL = 128
Reply from 10.0.0.2 bytes = 32 time = 0ms TTL = 128
Reply from 10.0.0.2 bytes = 32 time = 0ms TTL = 128
Reply from 10.0.0.2 bytes = 32 time = 0ms TTL = 128
ping Statistics for 10.0.0.2:
Packets: Sent = 4, Receiver = 4, Lost = 0 (0% Loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 0ms, Average = 0ms



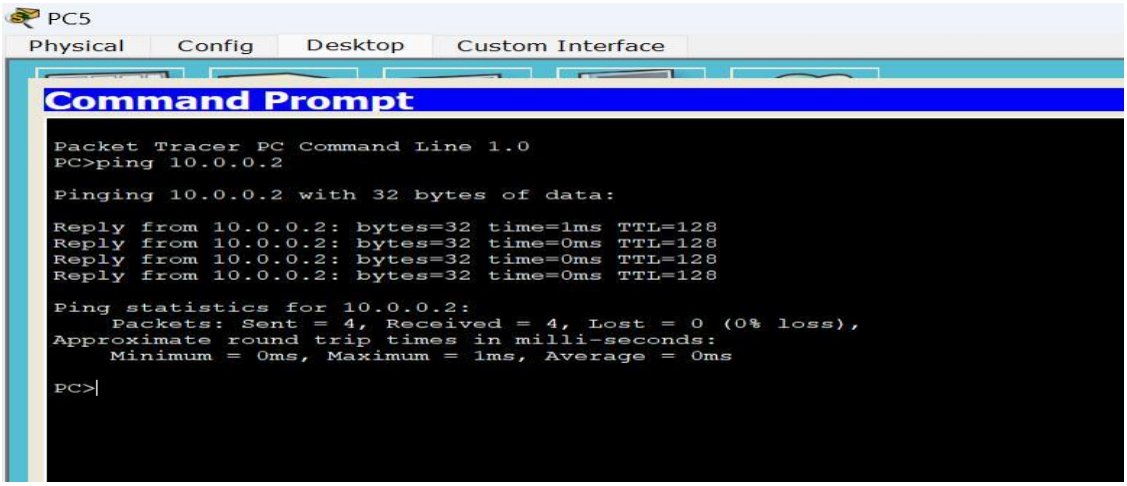
observation:

Once the end devices are connected to the Switch there is an Amber light indicating the switch is studying about the Connected end devices. Once the light, turns green, the devices are ready for Communication. The Switch transmitts the packet only for the intended devices in a network after learning about the devices and hence is known as a smart device.

Topology :



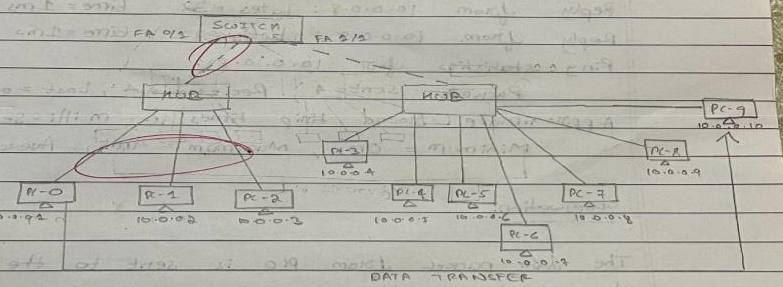
Ping Results :



---

3) Hybrid(Hub and Switch) Procedure :

(3) Topology: (Hybrid)



Procedure:

Step 1: Add About 4 generic PC's to your logical workspace and Connect it to the Hub using a Copper straight through wire. Add Another 7 devices and Connect it to Another hub using a Copper straight through wire. Connect the 2 Hub's to a Switch using a Copper Cross-over wire and set the IP Addresses of all the devices. (More ports can be added to the Hub by switching off the device)

COMPASS
Date: 15/06/22

M T W T F S S

Step 2: Add a simple PDU from one device (PC 0) to another device (PC 11) and Switch to Simulation mode. click the Auto Capture/play button to begin simulation.

Result:

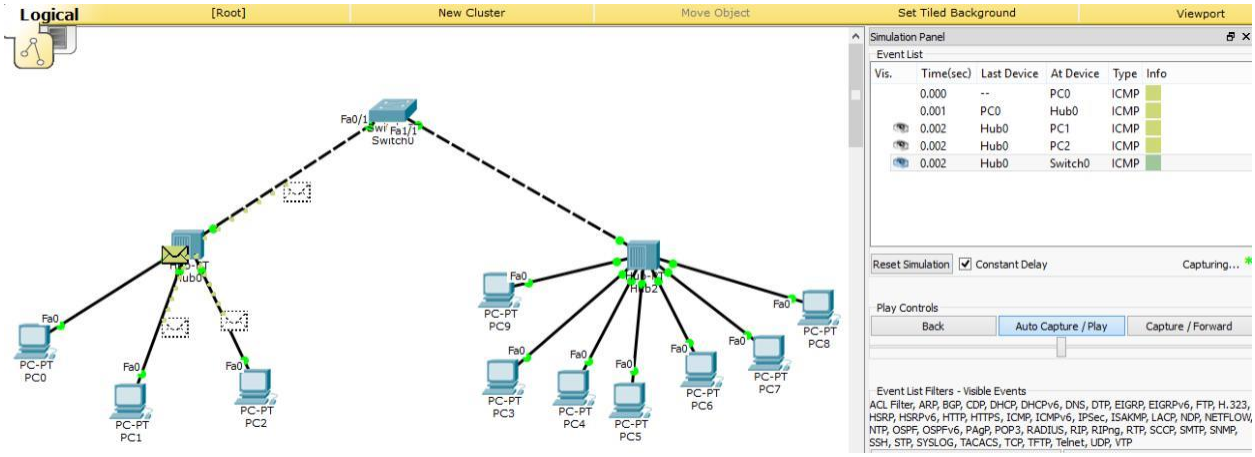
Pinging from
PC> PING 10.0.0.8
Pinging 10.0.0.8 with 32 bytes of data:
Reply from 10.0.0.8: bytes = 32 time = 1ms TTL = 127
Reply from 10.0.0.8: bytes = 32 time = 1ms TTL = 128
Reply from 10.0.0.8: bytes = 32 time = 1ms TTL = 128
Reply from 10.0.0.8: bytes = 32 time = 1ms TTL = 128
Ping statistics for 10.0.0.8:
Packets: Sent = 4, Recieved = 4, Lost = 0 (0% Loss),
Approximate round trip times in milli-Seconds:
Minimum = 0ms, Maximum = 4ms, Average = 1ms

Observation:

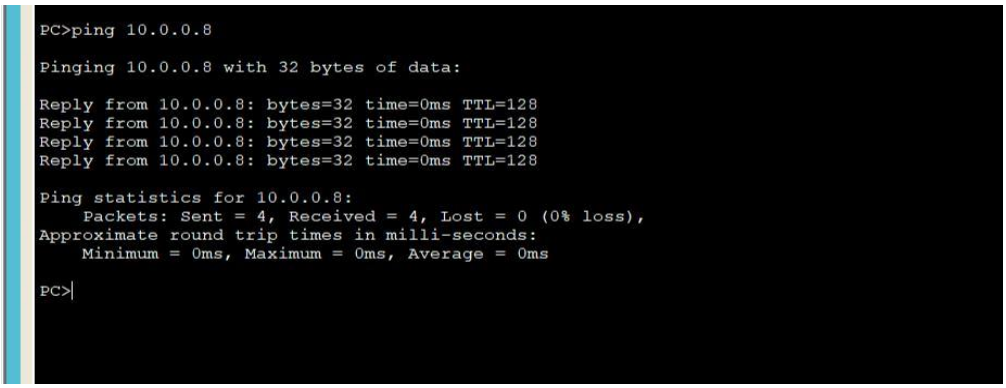
The data packet from PC0 is sent to the Hub which then broadcasts the packet to all the devices in that network and the Switch. The Switch then transmits the data packet to the second hub which in-turn broadcasts the packet to all the devices the hub. The destination device Accepts the data packet while the remaning the discard the data packet. Finally, the destination device sends an Acknowledgment to the Source device that data communication is Complete.

22/6/22

Topology :



Ping Result :

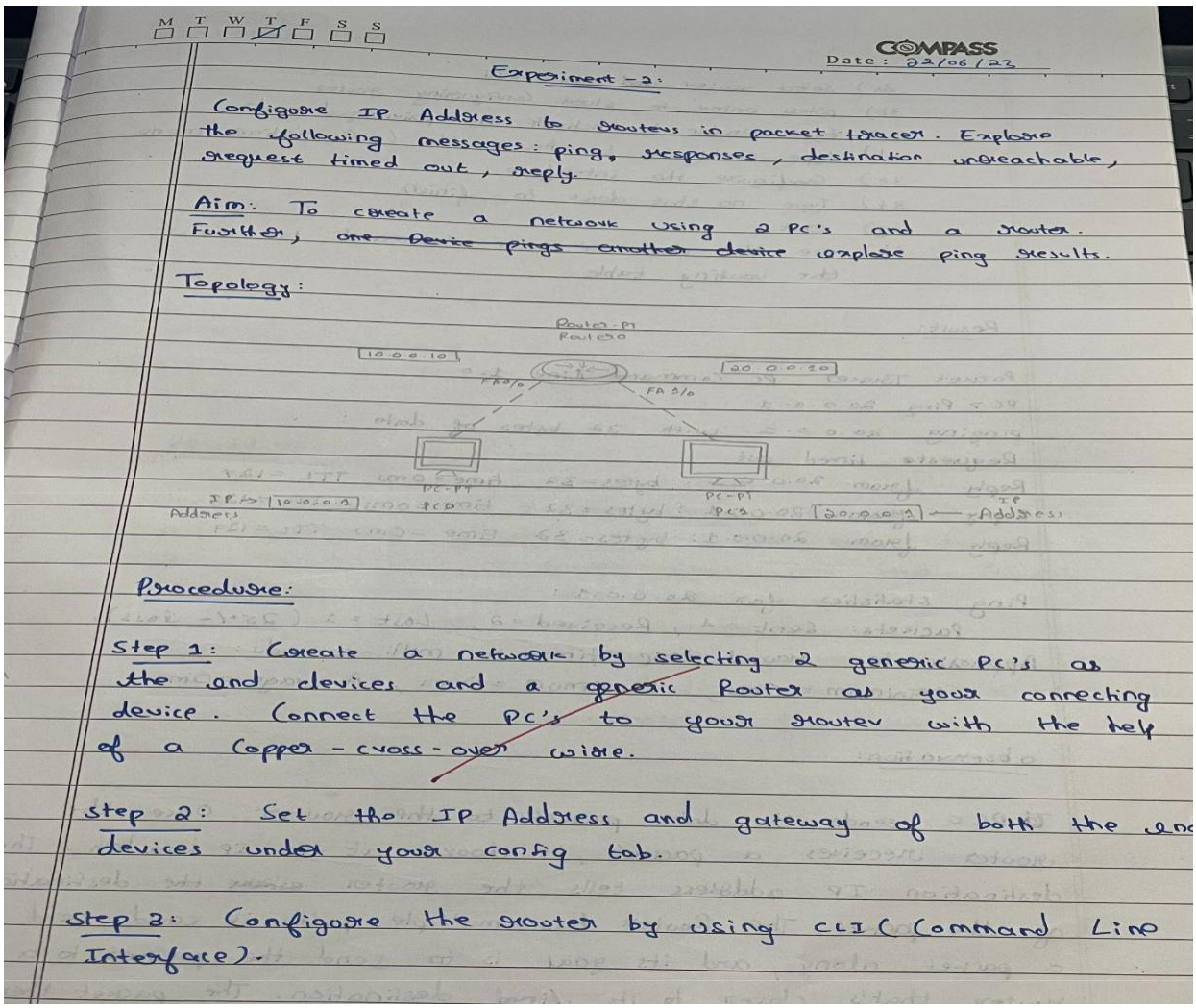


LAB PROGRAM -2

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

1) Single Router

Procedure :



COMPASS
Date: 22/06/23

M T W T F S S

3a) Select router and open CLI
3b) press enter to start Configuring router
3c) type enable to Activate the privileged mode
3d) type config t to Access the Configuration menu
3e) Configure the interface
3f) Type no shut down to finish
3g) exit
3h) use the show IP route Command to view the routing table

Result:

Packet Tracer PC Command Line 1.0
PC > Ping 20.0.0.1

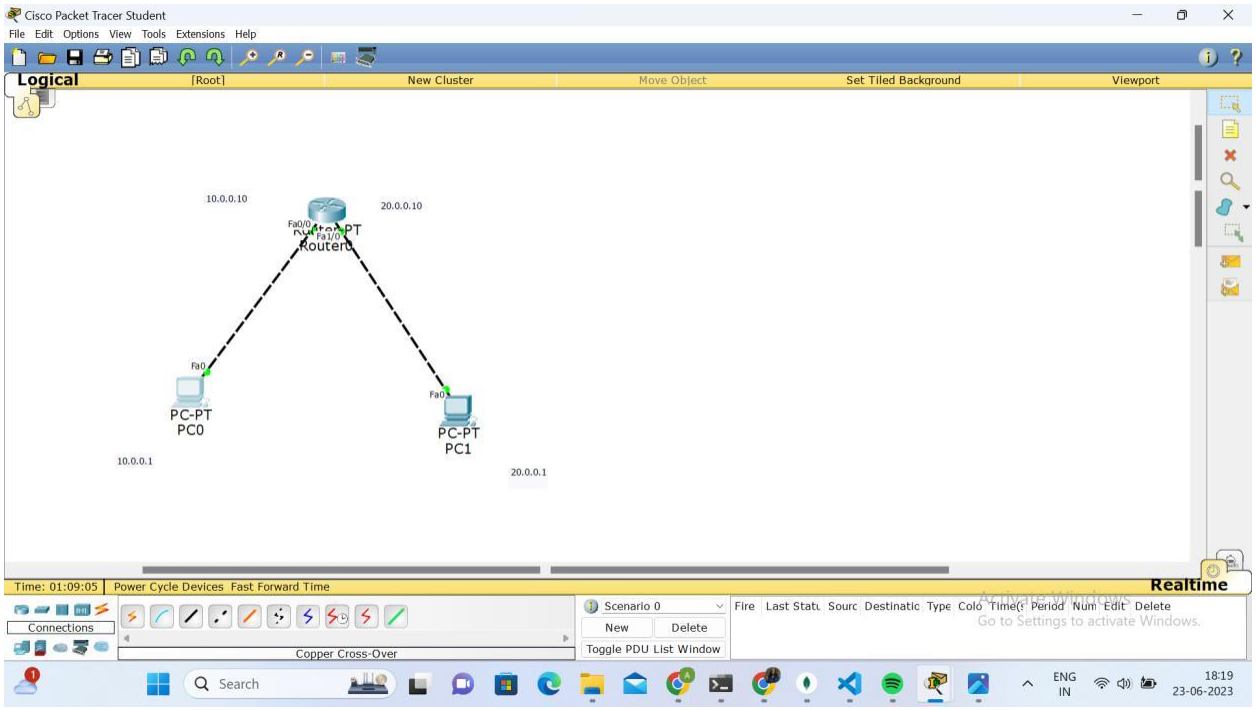
pinging 20.0.0.1 with 32 bytes of data :
Request timed out
Reply from 20.0.0.1 : bytes = 32 time = 0ms TTL = 127
Reply from 20.0.0.1 : bytes = 32 time = 0ms TTL = 127
Reply from 20.0.0.1 : bytes = 32 time = 0ms TTL = 127

Ping statistics for 20.0.0.1 :
Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 0ms, Average = 0ms

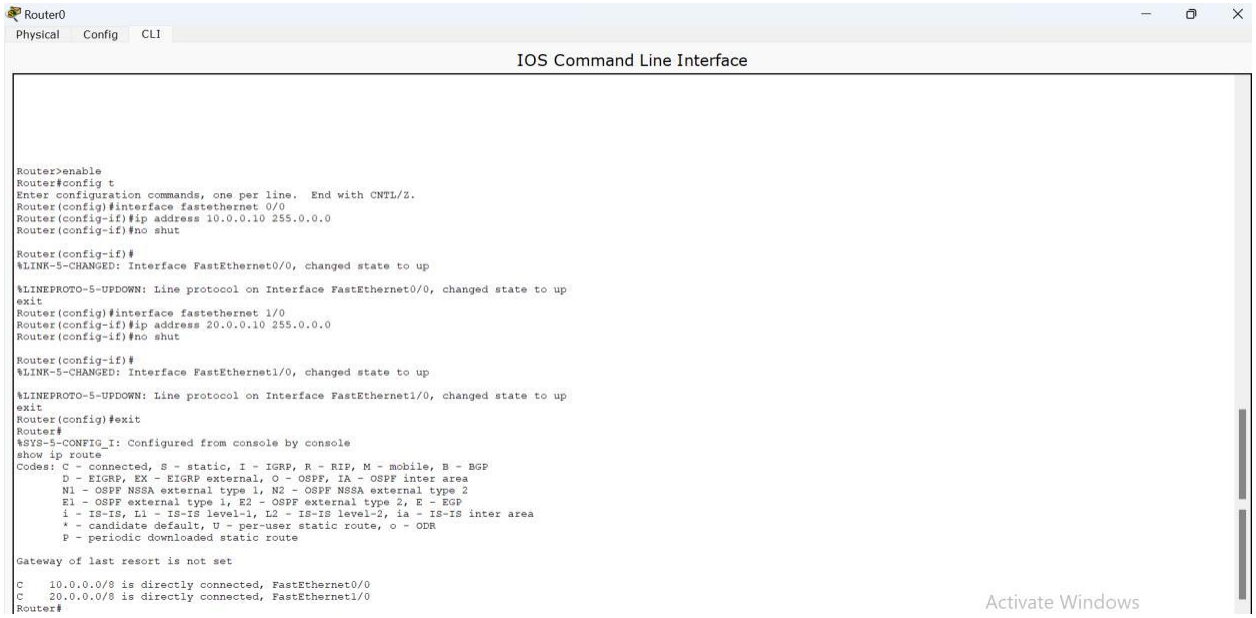
observation:

PC0 sends the data packet to the router. Once the router receives a packet, it looks at its IP header. The destination IP address tells the router ~~where~~ the destination of the packet. The Router has multiple paths it could send a packet along, and its goal is to send the packet to a router that's closer to its final destination. The packet then arrives at the destination, i.e, PC1.

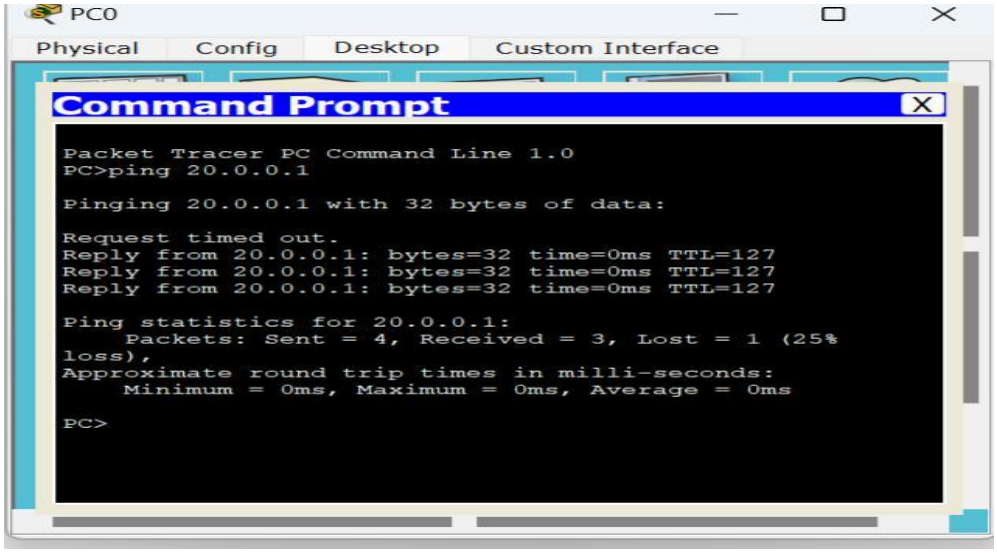
Topology :



Router Configuration :

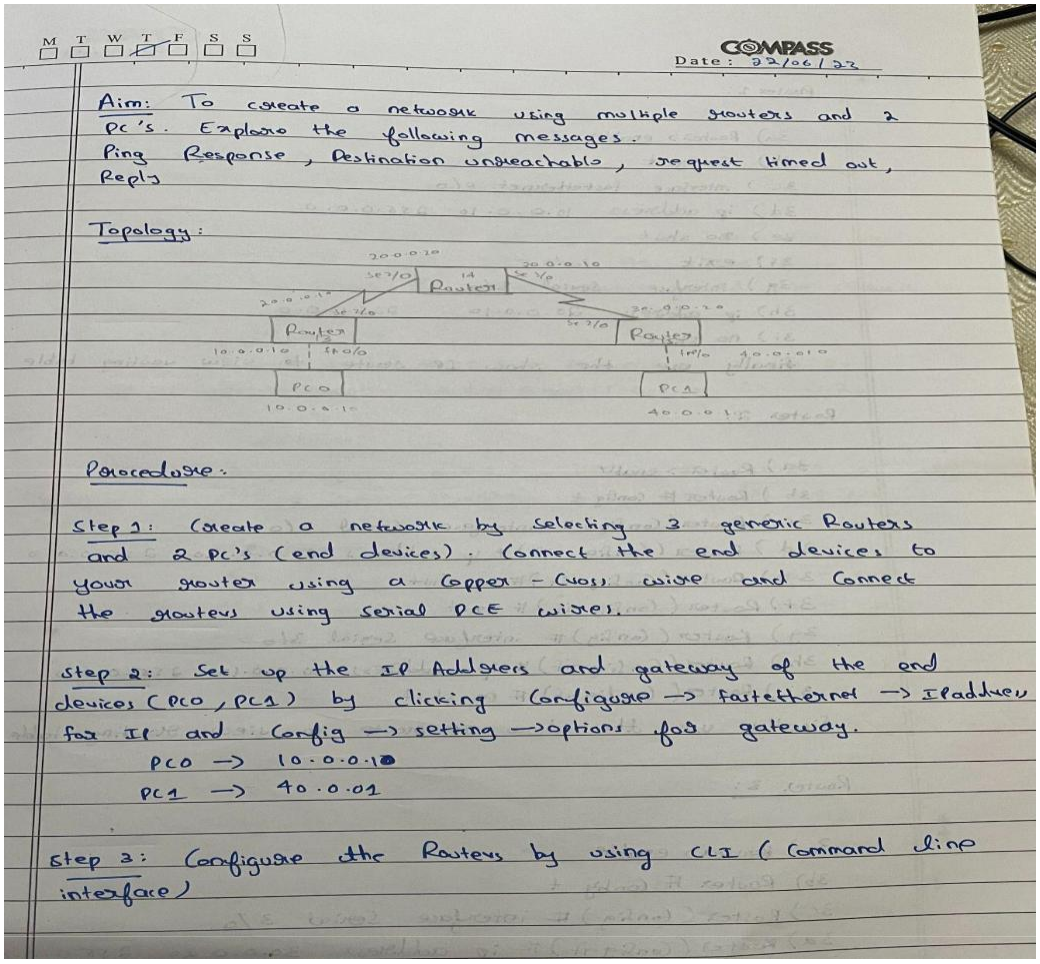


Ping Results :



2) Multi Router

Procedure :



M T W T F S S

COMPASS

Date : 22/06/22

Aim: To create a network using multiple routers and 2 PC's. Explore the following messages.
Ping Response, Destination unreachable, Request timed out, Reply

Topology:

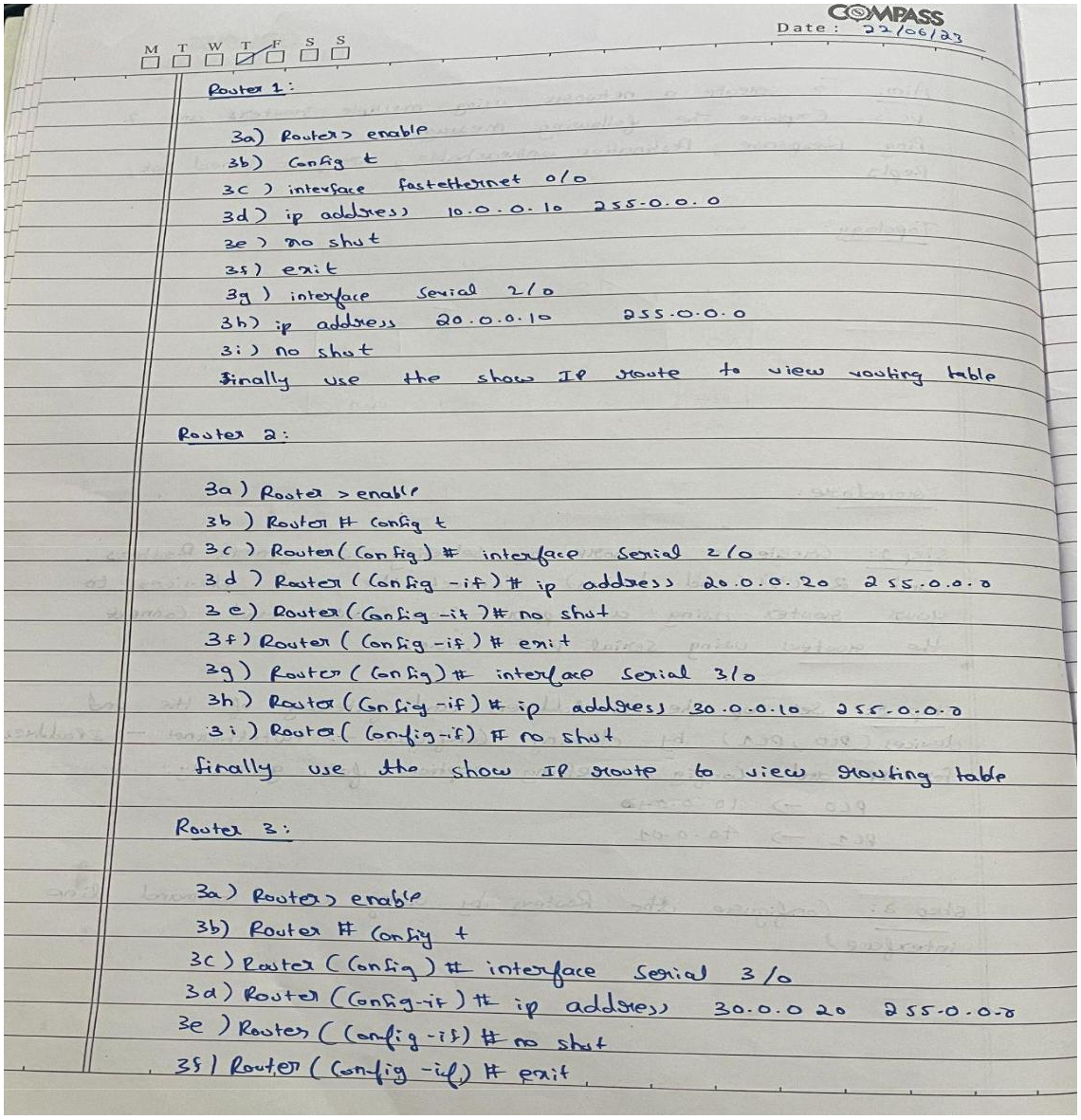
Procedure:

Step 1: Create a network by selecting 3 generic Routers and 2 PC's (end devices). Connect the end devices to your router using a Copper - Cross wire and Connect the routers using serial DCE wires.

Step 2: Set up the IP Addresses and gateway of the end devices (PC0, PC1) by clicking Configure → fastethernet → IPaddress for IP and Config → setting → options for gateway.

PC0 → 10.0.0.10
PC1 → 40.0.0.01

Step 3: Configure the Routers by using CLI (Command line interface)



M T W T F S S

COMPASS
Date: 22/06/23

Router 1:

3a) Router> enable
3b) Config t
3c) interface fastethernet 0/0
3d) ip address 10.0.0.10 255.0.0.0
3e) no shut
3f) exit
3g) interface serial 2/0
3h) ip address 20.0.0.10 255.0.0.0
3i) no shut
finally use the show IP route to view routing table

Router 2:

3a) Router > enable
3b) Router # Config t
3c) Router(Config) # interface serial 2/0
3d) Router(Config-if) # ip address 20.0.0.20 255.0.0.0
3e) Router(Config-if) # no shut
3f) Router(Config-if) # exit
3g) Router(Config) # interface serial 3/0
3h) Router(Config-if) # ip address 30.0.0.10 255.0.0.0
3i) Router(Config-if) # no shut
finally use the show IP route to view routing table

Router 3:

3a) Router> enable
3b) Router # Config t
3c) Router(Config) # interface serial 3/0
3d) Router(Config-if) # ip address 30.0.0.20 255.0.0.0
3e) Router((Config-if) # no shut
3f) Router(Config-if) # exit

M T W T F S S

COMPASS
Date : 22/06/23

3g) Router (config)# interface fastethernet 0/0
3b) Router (config-if)# ip address 10.0.0.10 255.0.0.0
3i) Router (config-if)# exit
finally use the show ip route to view routing table

Step 4: ping the different gateway and observe the output.
Now configure the Router such that the Routers have data/knowledge about the other network, add the network in CLI

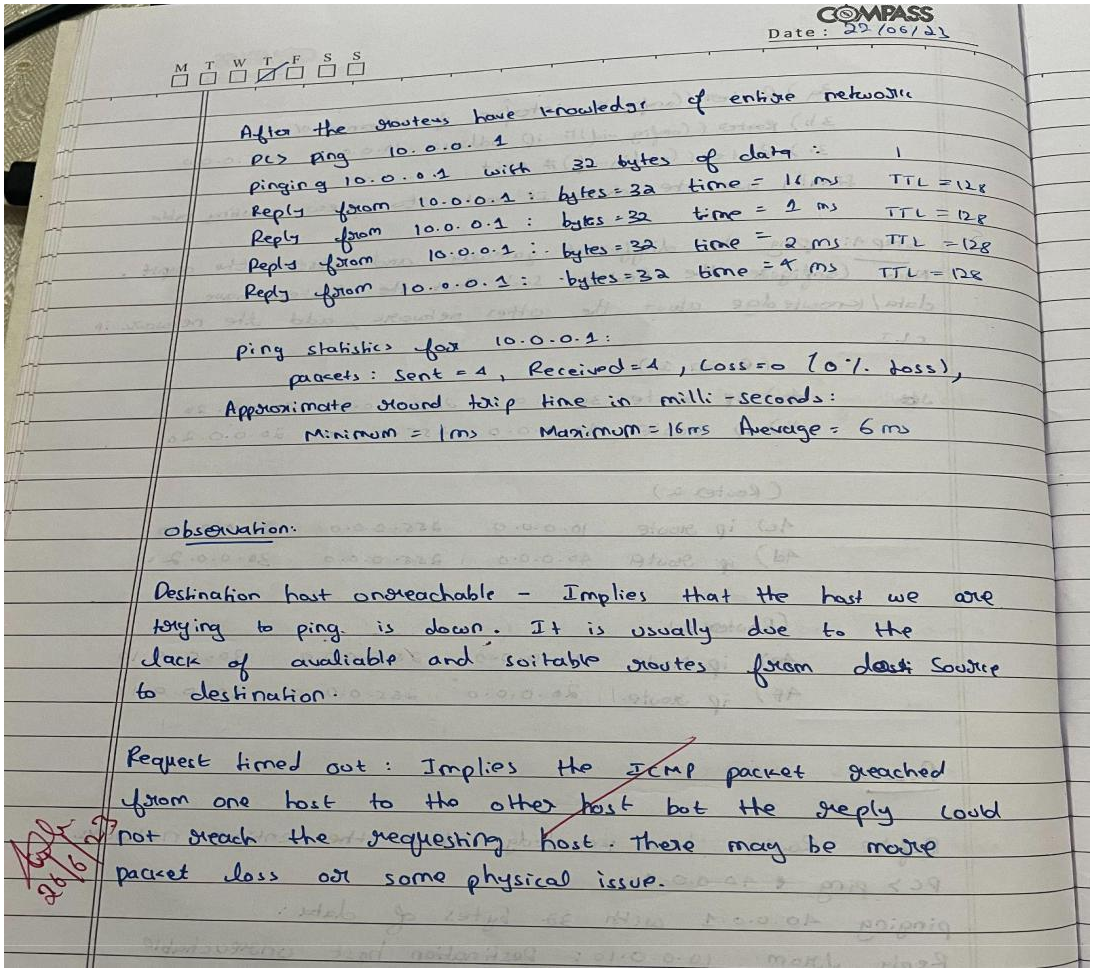
(Router 1)
4a) ip route 30.0.0.0 255.0.0.0 20.0.0.20
4b) ip route 40.0.0.0 255.0.0.0 20.0.0.20

(Router 2)
4c) ip route 10.0.0.0 255.0.0.0 20.0.0.10
4d) ip route 40.0.0.0 255.0.0.0 30.0.0.20

(Router 3)
4e) ip route 10.0.0.0 255.0.0.0 30.0.0.10
4f) ip route 20.0.0.0 255.0.0.0 30.0.0.10

Result:

Before Routers have knowledge about the entire network
PC> ping 40.0.0.1
pinging 40.0.0.1 with 32 bytes of data:
Reply from 10.0.0.10: Destination host unreachable
Reply from 10.0.0.10: Destination host unreachable
Reply from 10.0.0.10: Destination host unreachable
Reply from 10.0.0.10: Destination host unreachable
Ping statistics for 40.0.0.1:
Packets: Sent = 4, Received = 0, Lost = 4 (100% Loss)



After the routers have knowledge of entire network

```
PC> ping 10.0.0.1

Pinging 10.0.0.1 with 32 bytes of data:

Reply from 10.0.0.1: bytes=32 time=16ms TTL=128
Reply from 10.0.0.1: bytes=32 time=1ms TTL=128
Reply from 10.0.0.1: bytes=32 time=2ms TTL=128
Reply from 10.0.0.1: bytes=32 time=4ms TTL=128

Ping statistics for 10.0.0.1:
    Packets: Sent = 4, Received = 4, Loss = 0 (0% loss),
Approximate round trip time in milli-seconds:
    Minimum = 1ms, Maximum = 16ms, Average = 6ms
```

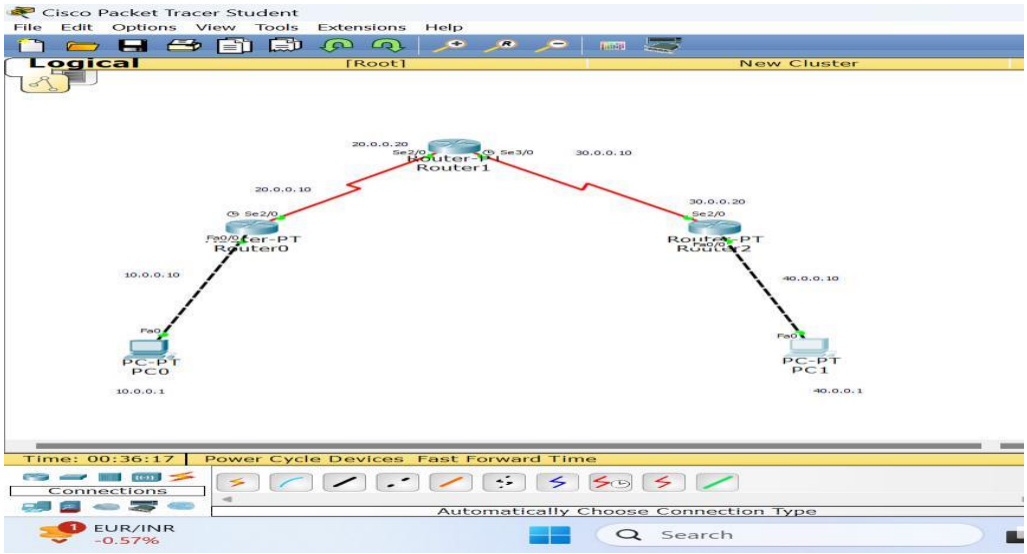
Observation:

Destination host unreachable - Implies that the host we are trying to ping is down. It is usually due to the lack of available and suitable routes from source to destination.

Request timed out : Implies the ICMP packet reached from one host to the other host but the reply could not reach the requesting host. There may be major packet loss or some physical issue.

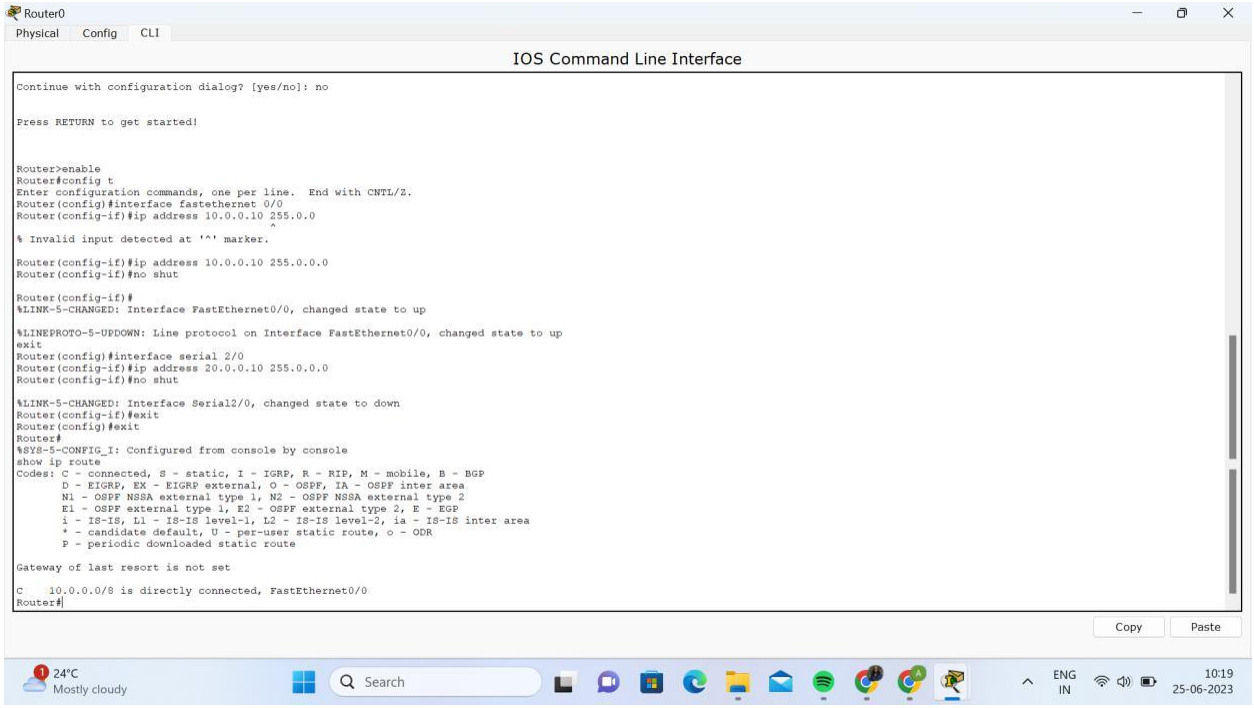
26/6/23

Topology :

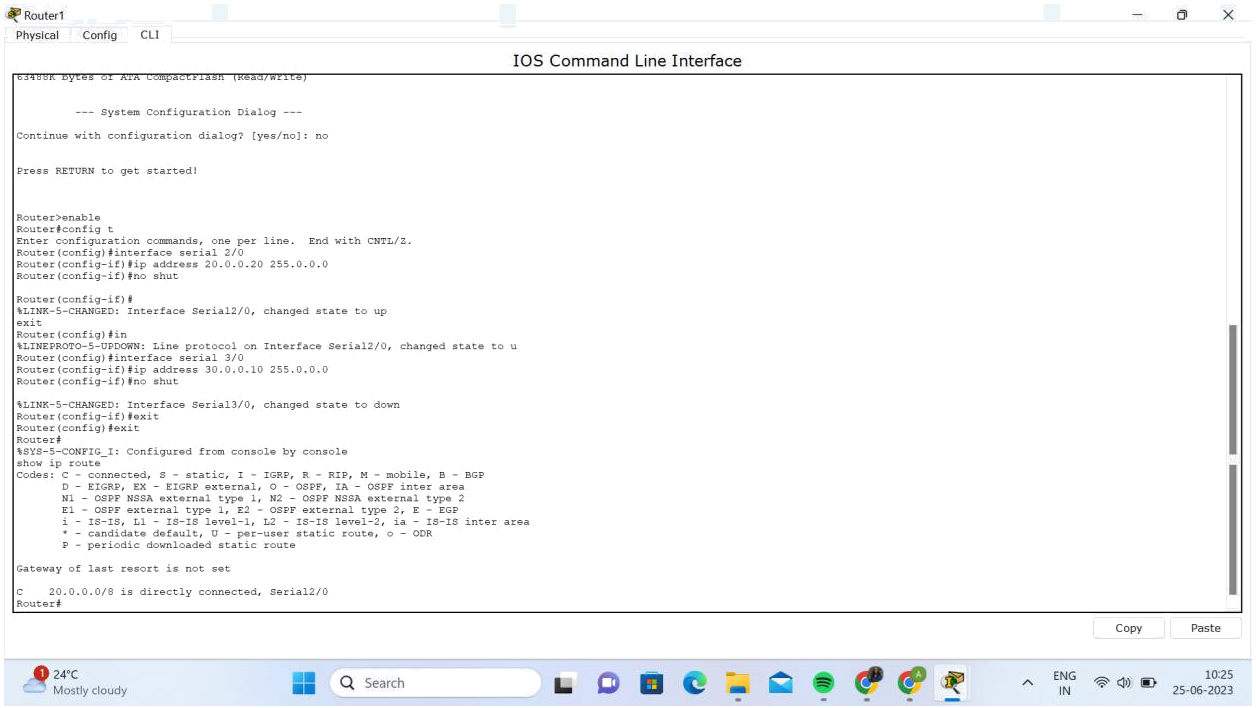


Router Configuration :

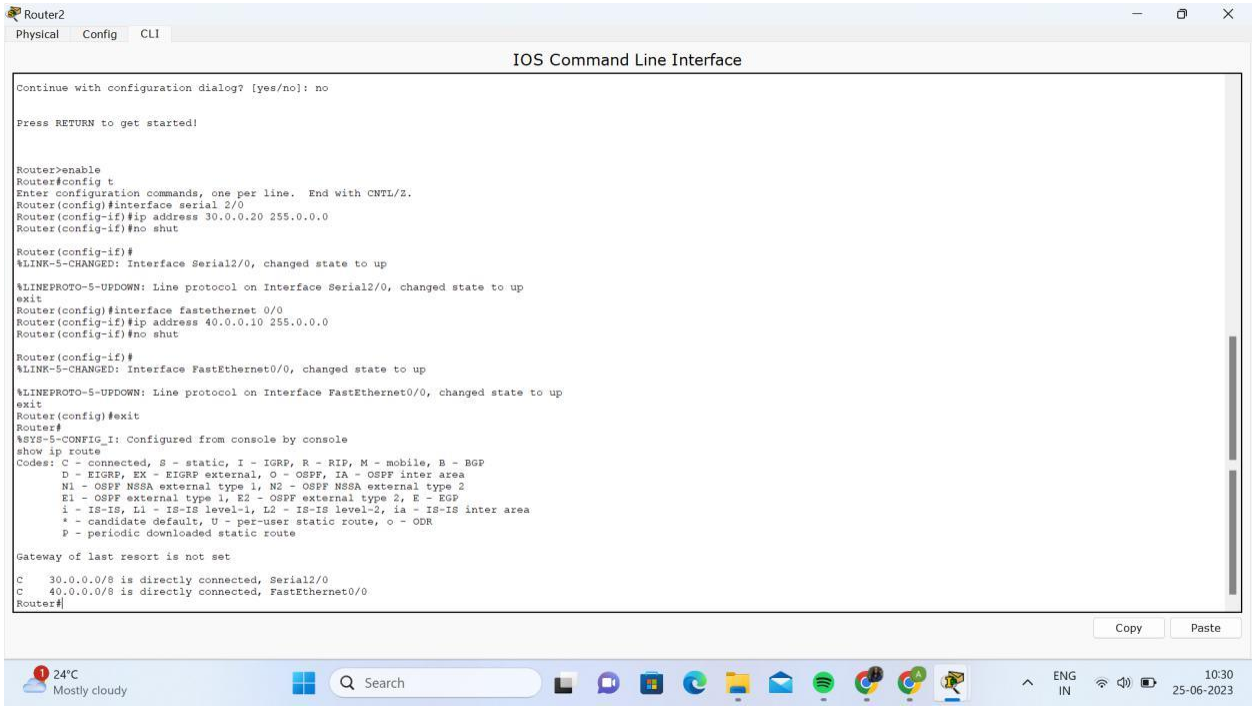
R0 :



R1 :

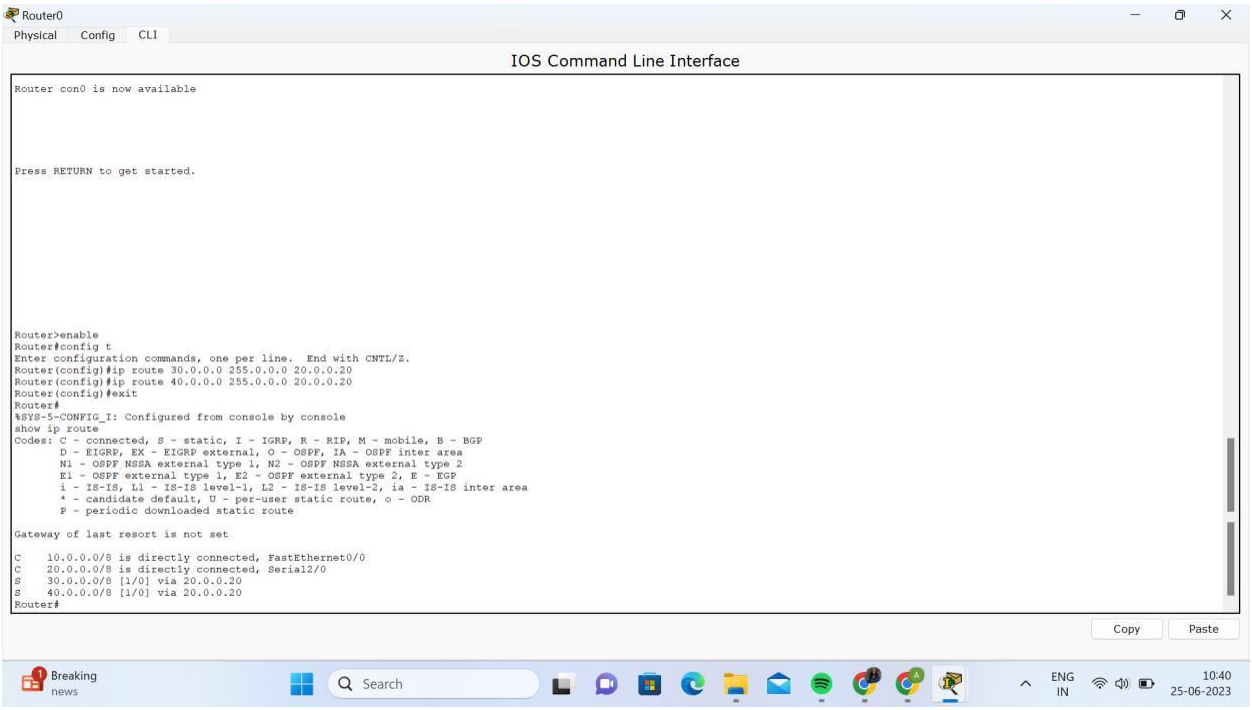

Router1
Physical
Config
CLI
IOS Command Line Interface
63488K bytes of ATA CompactFlash (Read/Write)
--- System Configuration Dialog ---
Continue with configuration dialog? [yes/no]: no
Press RETURN to get started!
Router>enable
Router#config t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface serial 2/0
Router(config-if)#ip address 20.0.0.20 255.0.0.0
Router(config-if)#no shut
Router(config-if)#
%LINK-5-CHANGED: Interface Serial2/0, changed state to up
exit
Router(config)#in
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to u
Router(config)#interface serial 3/0
Router(config-if)#ip address 30.0.0.10 255.0.0.0
Router(config-if)#no shut
%LINK-5-CHANGED: Interface Serial3/0, changed state to down
Router(config-if)#exit
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console
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 20.0.0.0/8 is directly connected, Serial2/0
Router#
Copy
Paste
24°C
Mostly cloudy
Search
ENG
IN
10:25
25-06-2023

R2 :

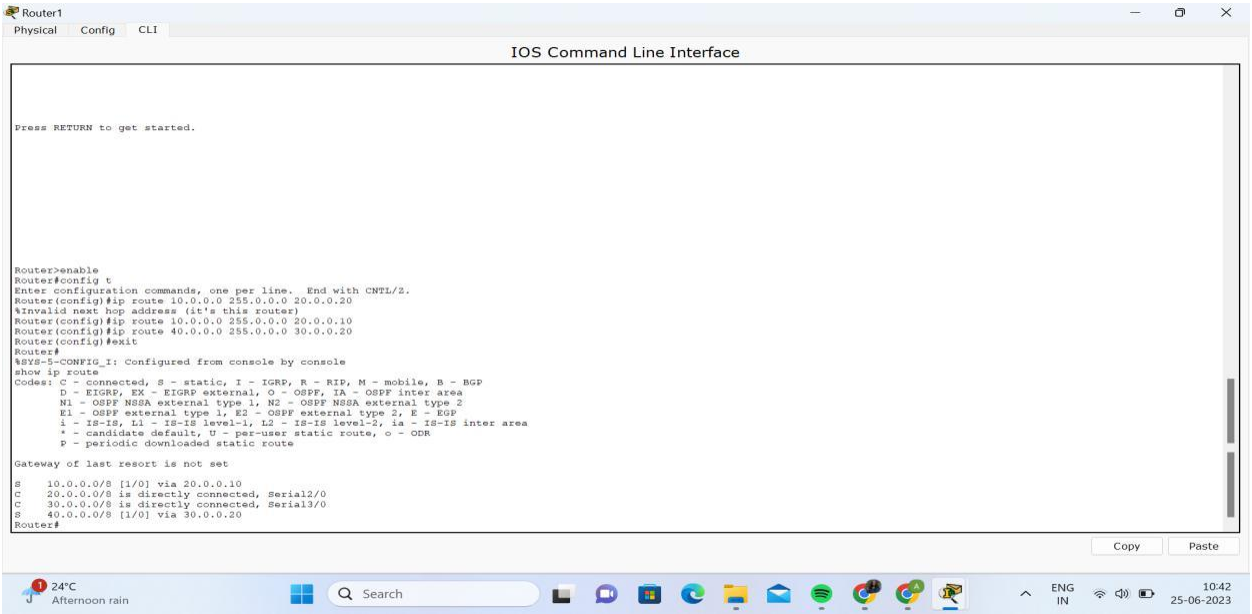

Router2
Physical
Config
CLI
IOS Command Line Interface
Continue with configuration dialog? [yes/no]: no
Press RETURN to get started!
Router>enable
Router#config t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface serial 2/0
Router(config-if)#ip address 30.0.0.20 255.0.0.0
Router(config-if)#no shut
Router(config-if)#
%LINK-5-CHANGED: Interface Serial2/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to up
exit
Router(config)#interface fastethernet 0/0
Router(config-if)#ip address 40.0.0.10 255.0.0.0
Router(config-if)#no shut
Router(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up
exit
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console
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 30.0.0.0/8 is directly connected, Serial2/0
C 40.0.0.0/8 is directly connected, FastEthernet0/0
Router#
Copy
Paste
24°C
Mostly cloudy
Search
ENG
IN
10:30
25-06-2023

Router Configuration (To gain knowledge about routers of diff Networks) :

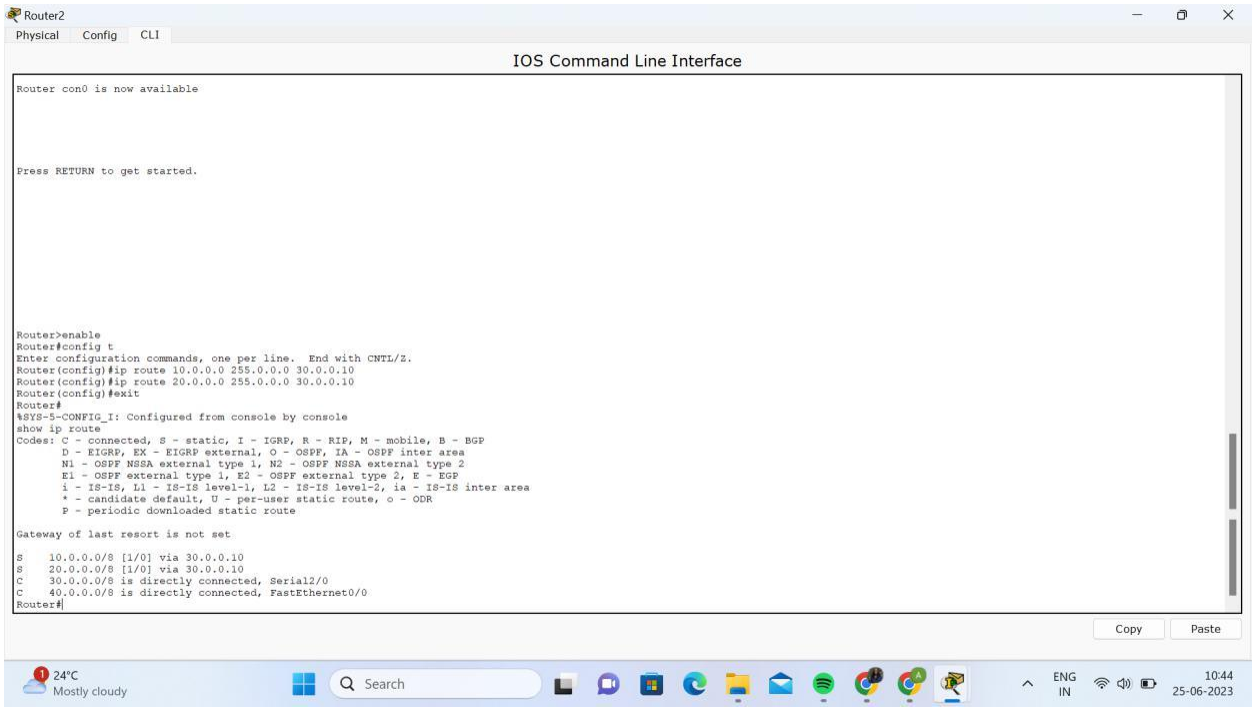
R0 :



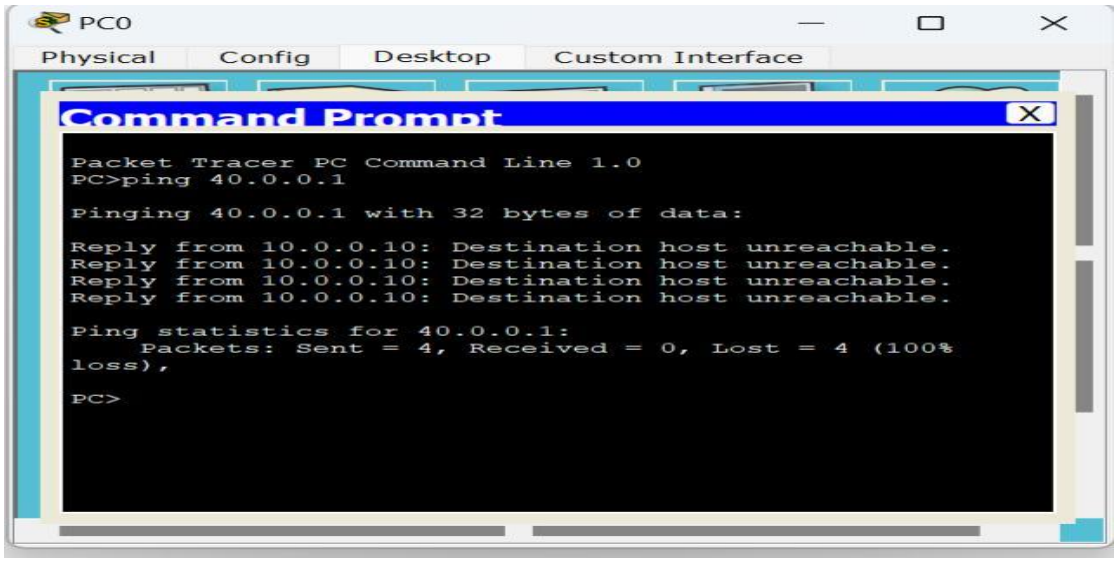
R1 :



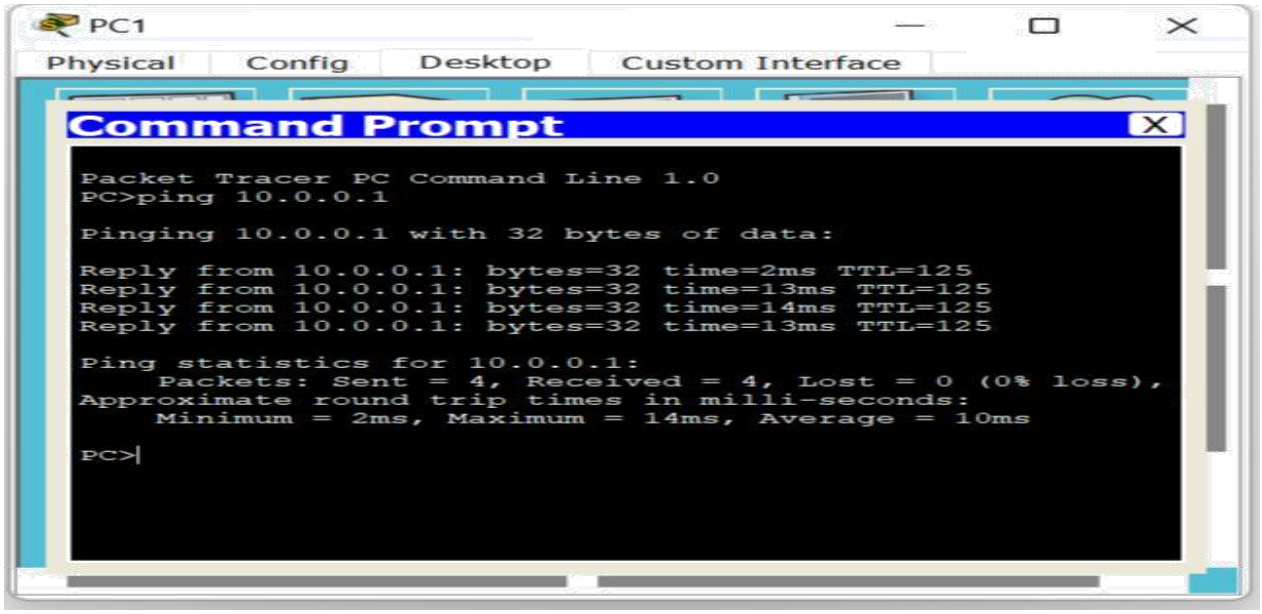
R2 :



Ping Results(Before Routers Had idea about other Routers) :



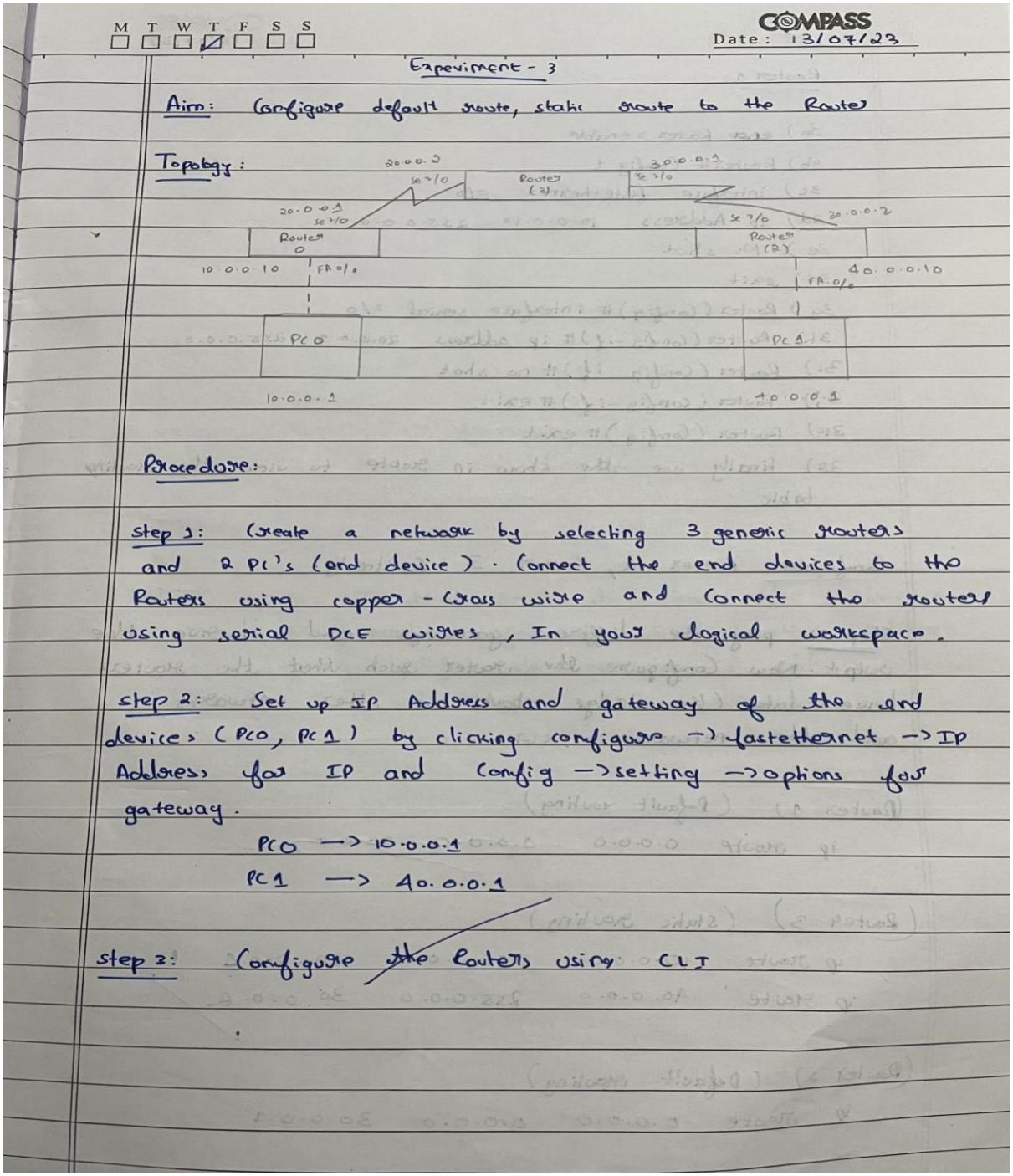
Ping Results(After Routers have idea about Routers of other Networks) :


PC1
Physical
Config
Desktop
Custom Interface
Command Prompt
Packet Tracer PC Command Line 1.0
PC>ping 10.0.0.1
Pinging 10.0.0.1 with 32 bytes of data:
Reply from 10.0.0.1: bytes=32 time=2ms TTL=125
Reply from 10.0.0.1: bytes=32 time=13ms TTL=125
Reply from 10.0.0.1: bytes=32 time=14ms TTL=125
Reply from 10.0.0.1: bytes=32 time=13ms TTL=125
Ping statistics for 10.0.0.1:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 2ms, Maximum = 14ms, Average = 10ms
PC>

LAB PROGRAM – 3

Aim :
Configure default route, static route to the Router

Procedure :



COMPASS
Date : 13/07/23

M T W T F S S

Router 1 :

3a) enable Router > enable
3b) Router# Config t
3c) interface fastethernet 0/0
3d) ip Address 10.0.0.10 255.0.0.0
3e) No shut
3f) exit
3g) Router (Config)# interface serial 2/0
3h) Router (Config-if)# ip address 20.0.0.1 255.0.0.0
3i) Router (Config-if)# no shut
3j) Router (Config-if)# exit
3k) Router (Config)# exit
3l) finally use the show ip route to view the routing table

Configure Router 2, Router 3 similarly

Step 4: ping the different gateways and observe the output. Now Configure the router such that the routers have data (knowledge about the other networks, add the network in CLI

(Router 1) (Default routing)
ip route 0.0.0.0 0.0.0.0 20.0.0.2

(Router 3) (Static routing)
ip route 10.0.0.0 255.0.0.0 20.0.0.1
ip route 40.0.0.0 255.0.0.0 30.0.0.2

(Router 2) (Default routing)
ip route 0.0.0.0 0.0.0.0 30.0.0.1

COMPASS
Date: 12/07/23

Result:

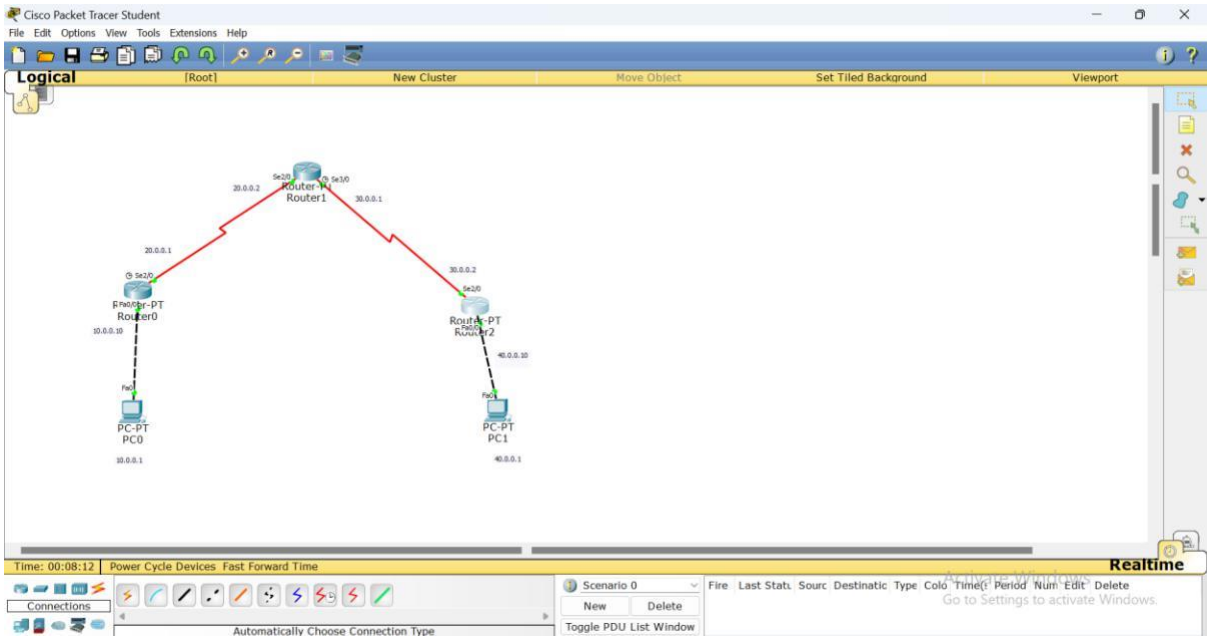
```
PC> Ping 10.0.0.1

Pinging 10.0.0.1 with 32 bytes of data
Reply from 10.0.0.1 : bytes = 32 time = 20ms TTL = 125
Reply from 10.0.0.1 : bytes = 32 time = 20ms TTL = 125
Reply from 10.0.0.1 : bytes = 32 time = 20ms TTL = 125
Reply from 10.0.0.1 : bytes = 32 time = 20ms TTL = 125
Ping statistics for 10.0.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 7ms, Maximum = 20ms, Average = 13ms
```

Observation:

Default route: Is that route that take effect when no other route is available for an IP destination address. A default route identifies the gateway IP address to which the router sends all IP packets that it does not have a learned route for. It establishes a forwarding rule for packets when no specific address of a next-hop host is available from the routing table or other routing mechanisms.

Static route: Process in which we have to manually add routes to the routing table. It is a pre-determined pathway that a packet must travel to reach a specific host or Network. It performs routing decisions with preconfigured routes in the routing table. It is implemented when route selections are limited or only a single default route is available. static Routing is handled by internet protocol.

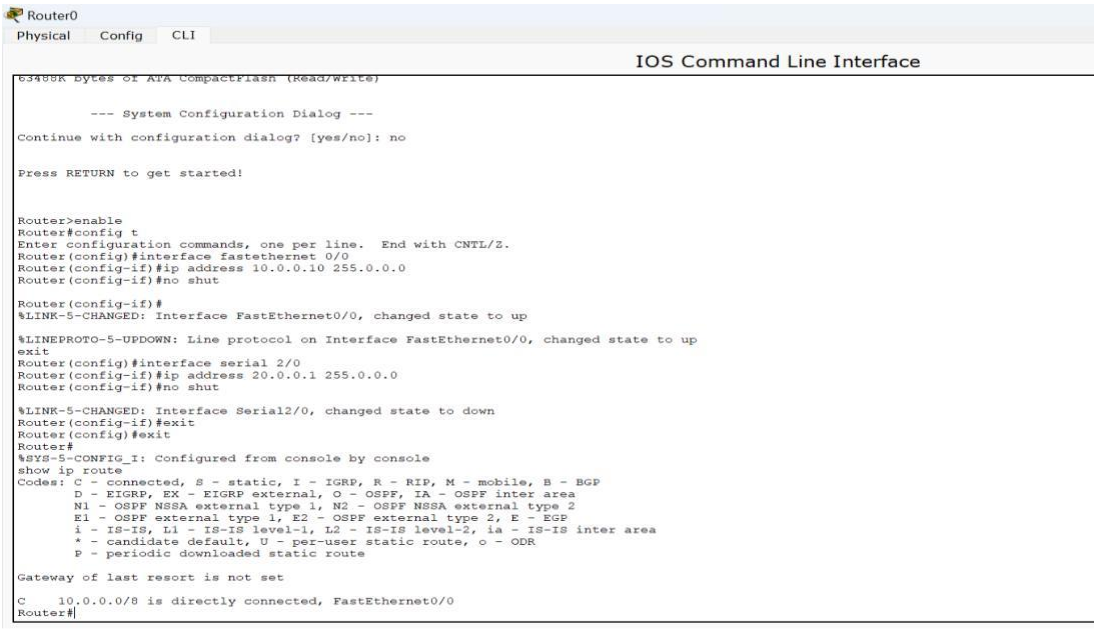
Topology :



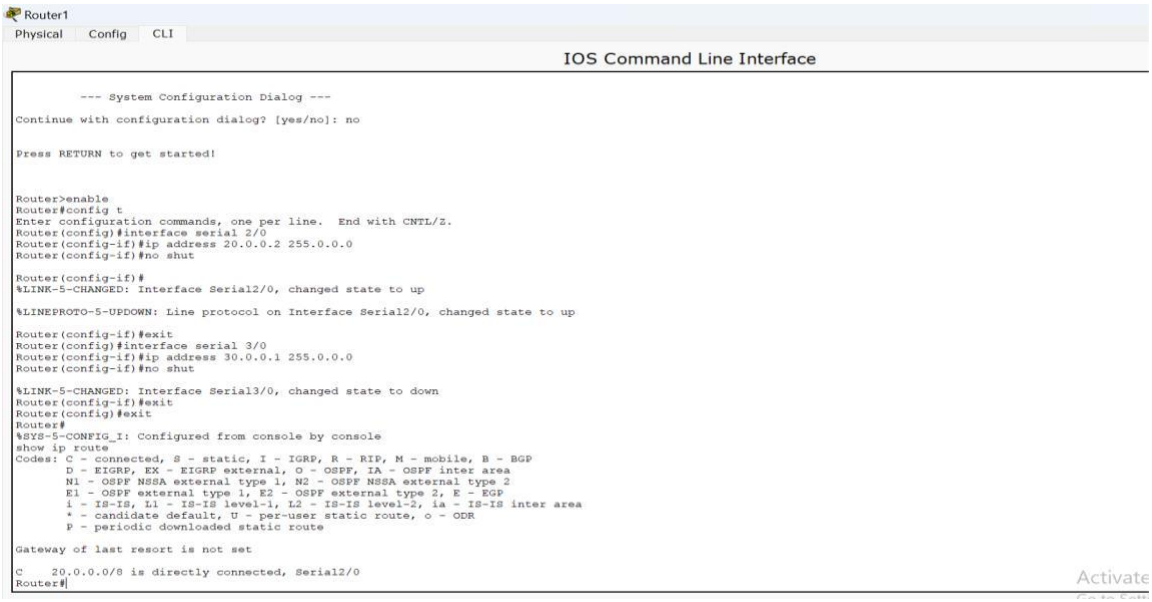
Router Configuration :

- Configuring Individual Routers

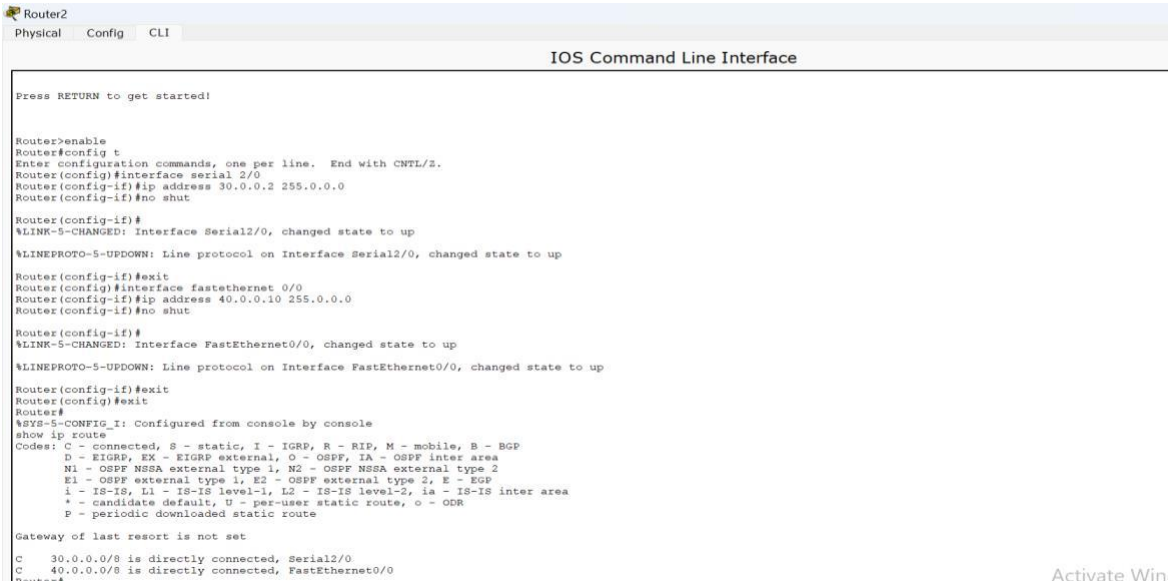
R0 :



R1 :

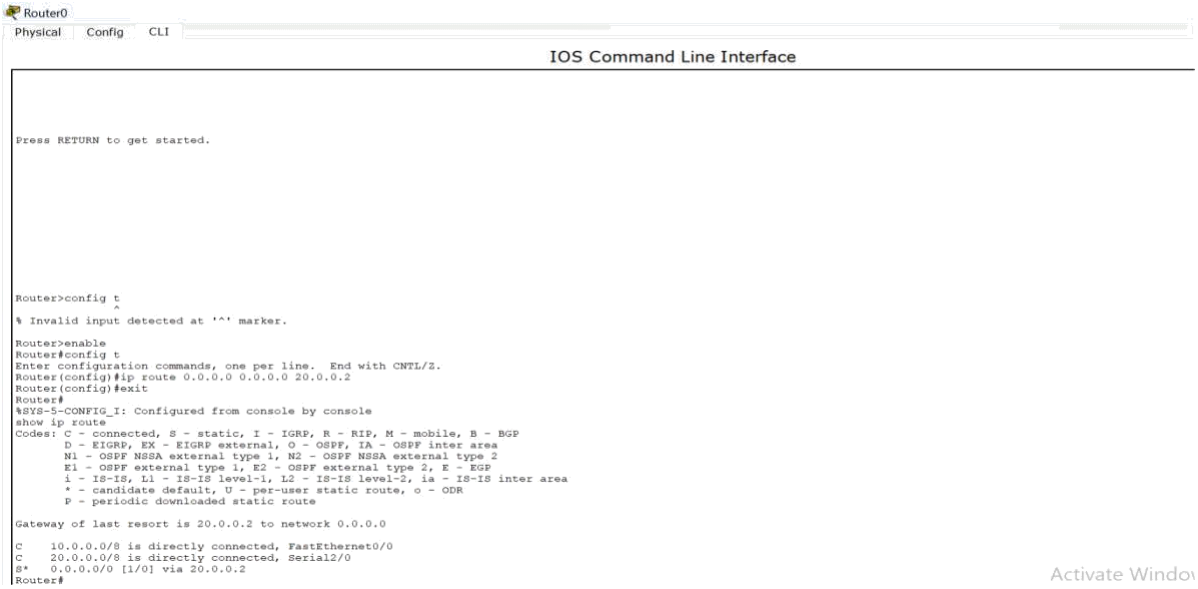


R2 :

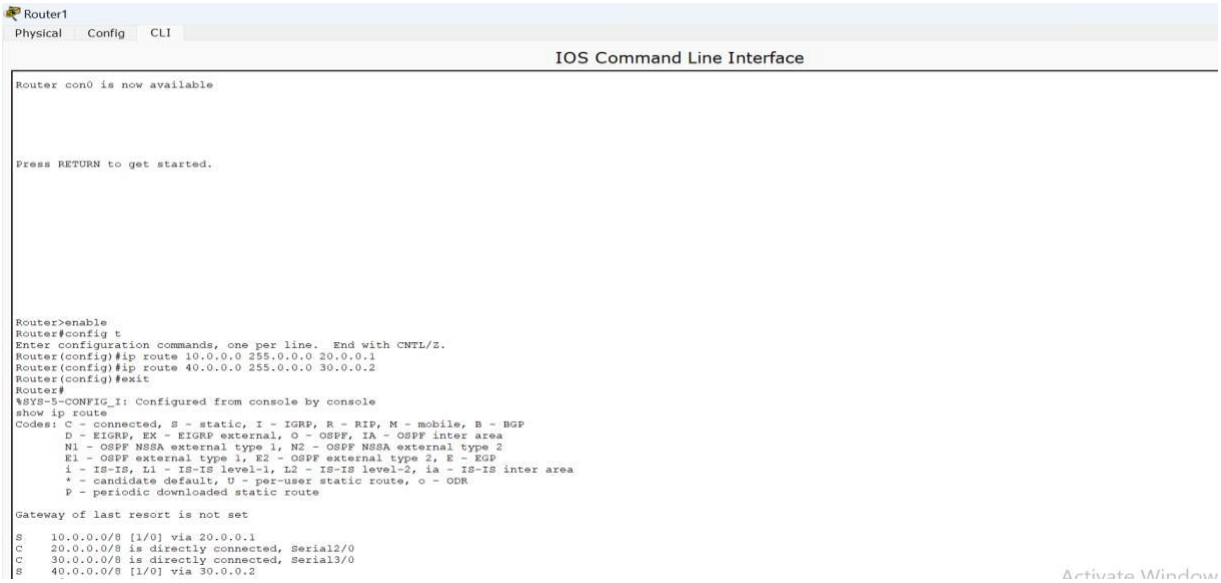


- Configure the routers gain knowledge about

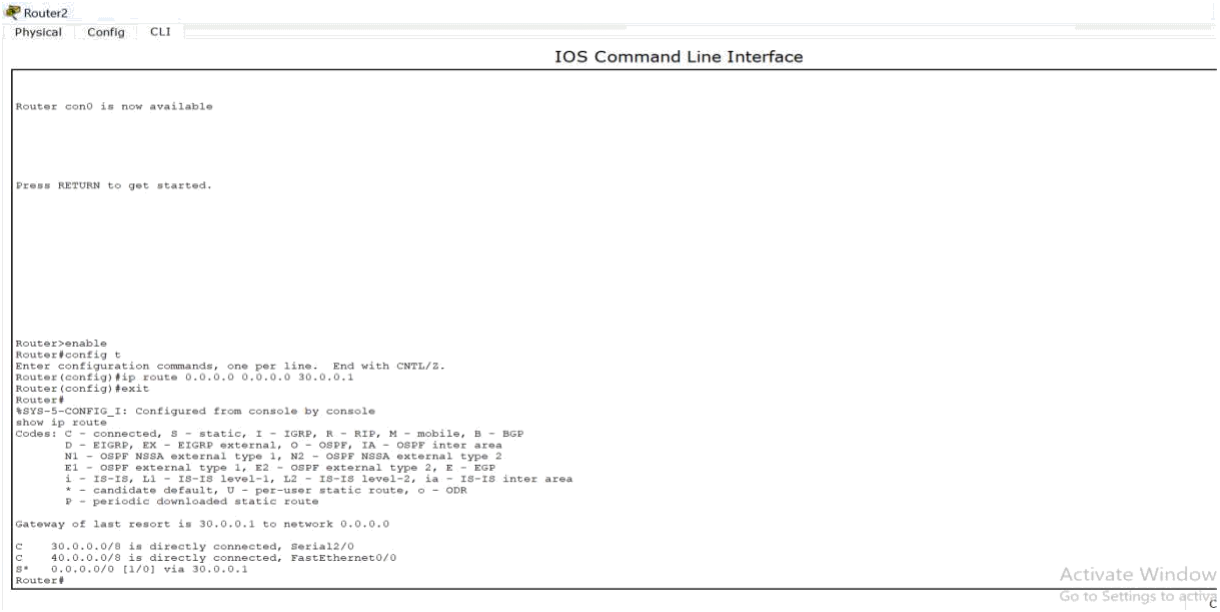
R0 : (Default Routing)



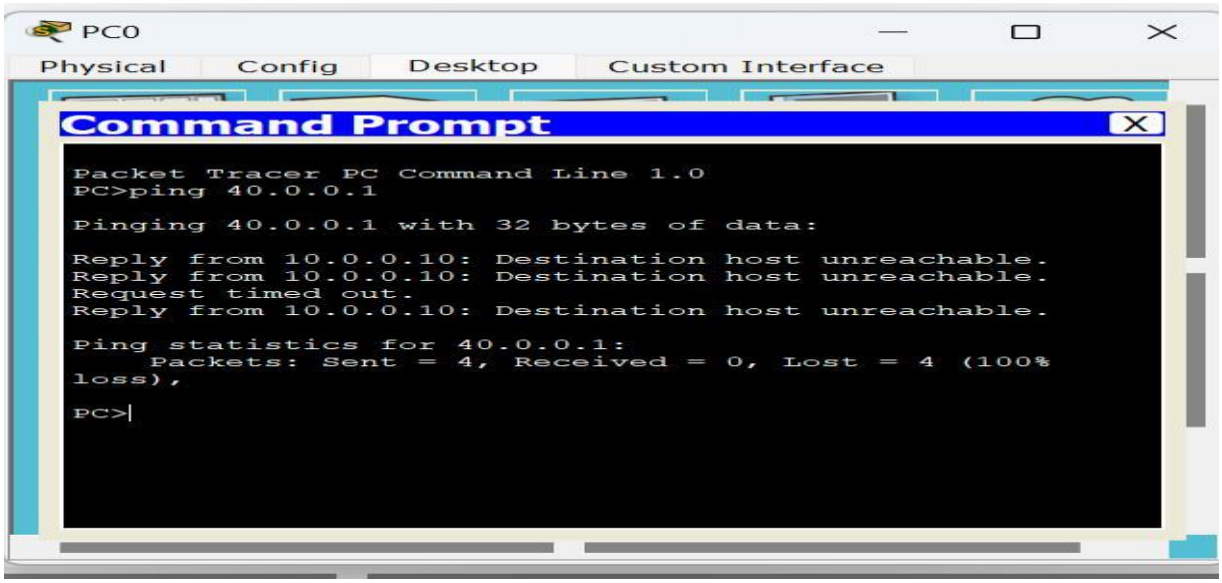
R1 : (Static Routing)



R2 : (Default Routing)

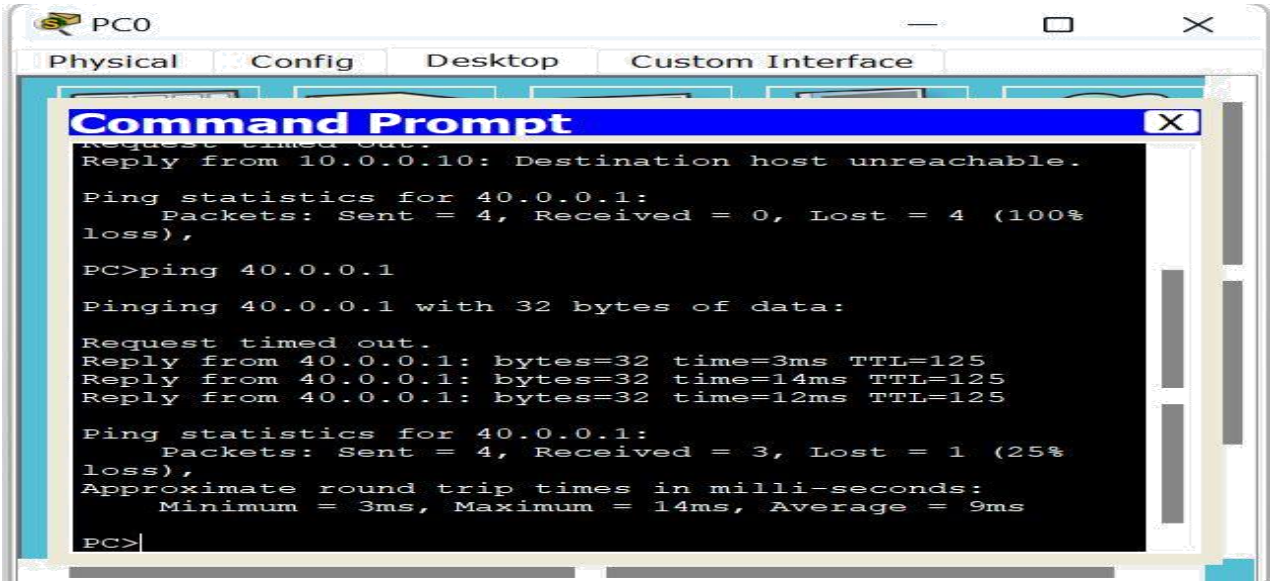


Ping Results(Before Routers gained Knowledge about other networks)

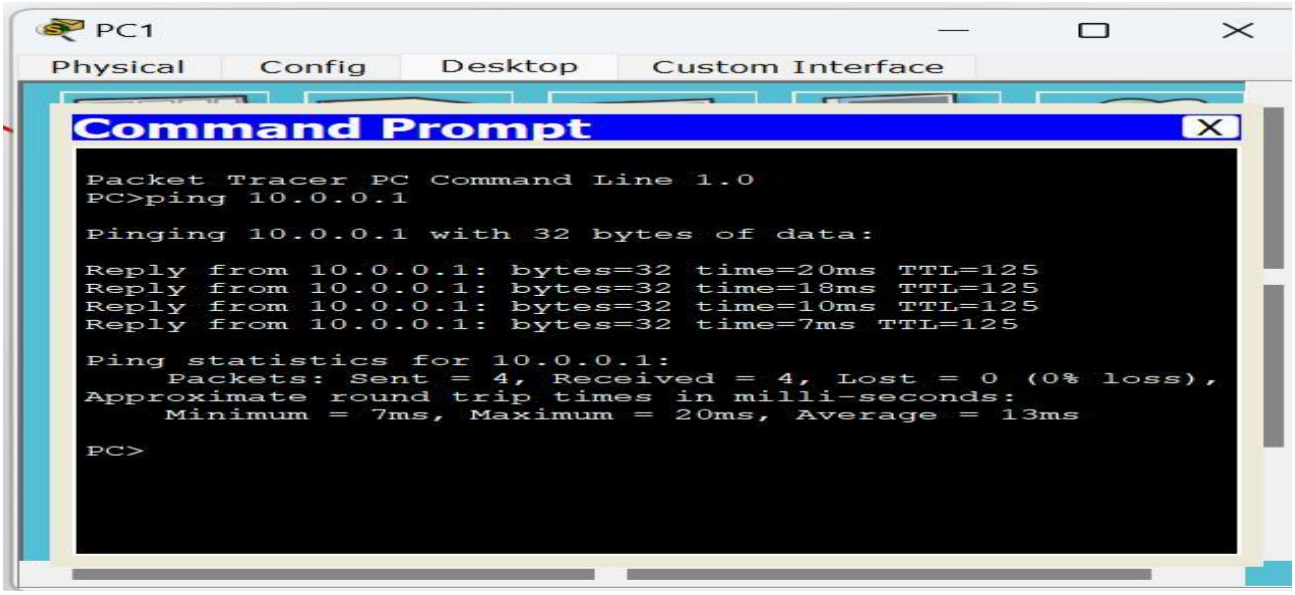


Ping Results(After Routers Have Knowledge About Other Networks)

- From PC-0 to PC-1



- From PC-1 to PC-0

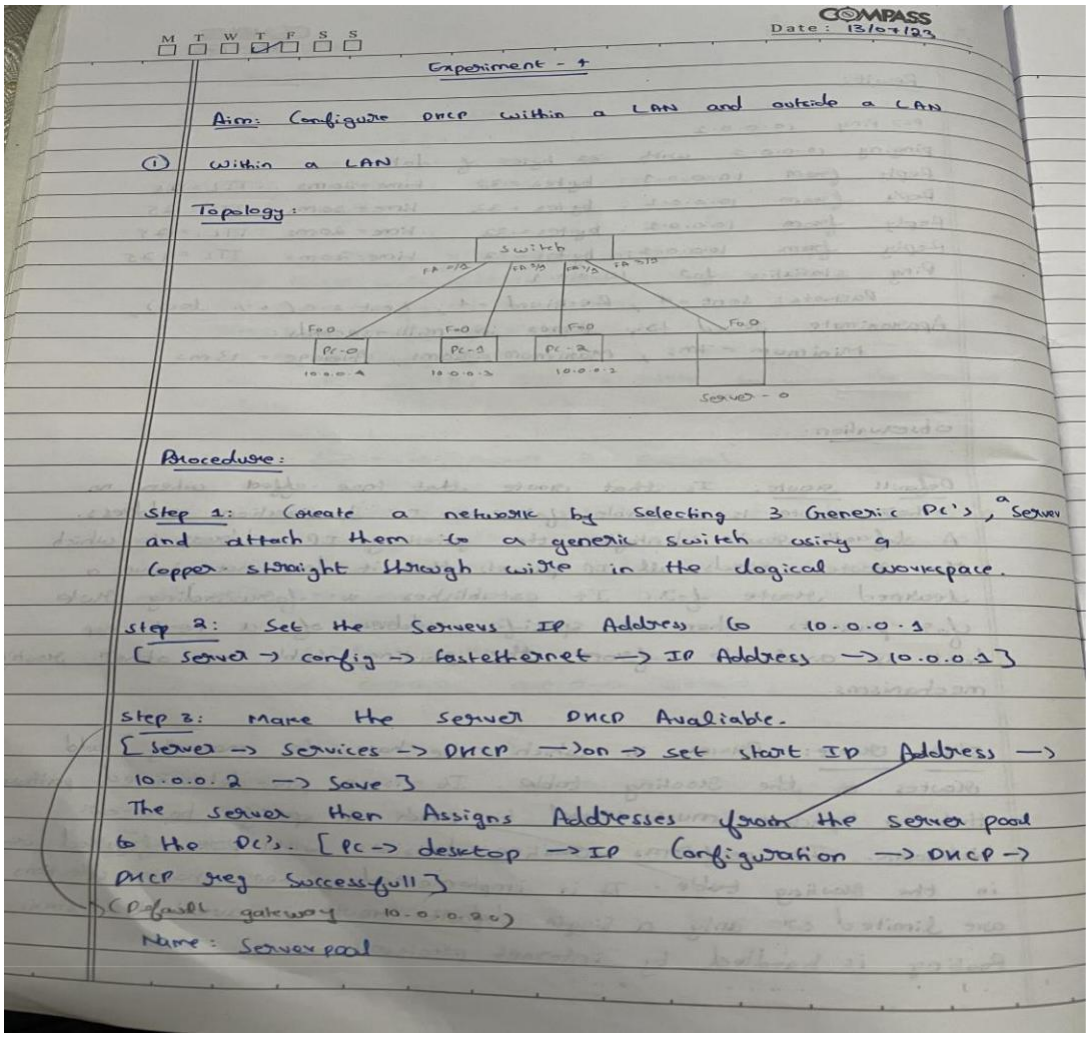


LAB PROGRAM – 4

Aim :
Configure DHCP within a LAN and outside LAN.

1) Within a LAN :

Procedure :



M T W T F S S

COMPASS

Date : 13/07/23

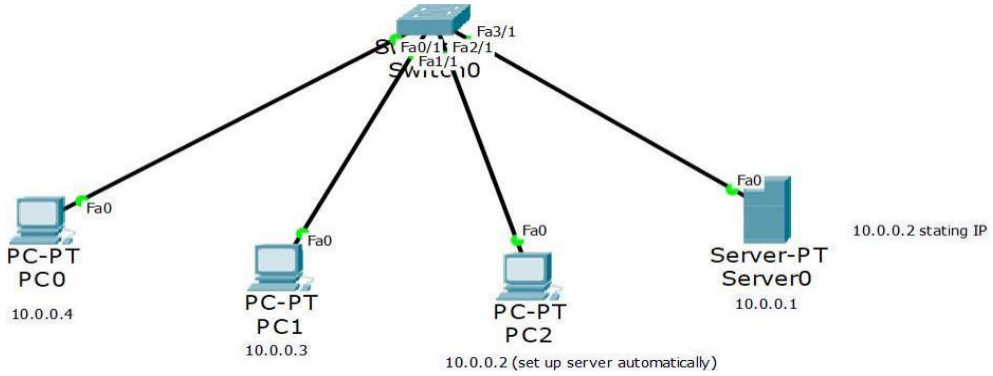
Result:

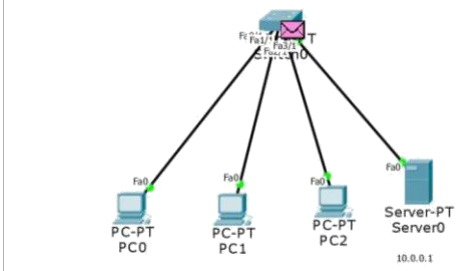
```
PC> Ping 10.0.0.4

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
```

observation:

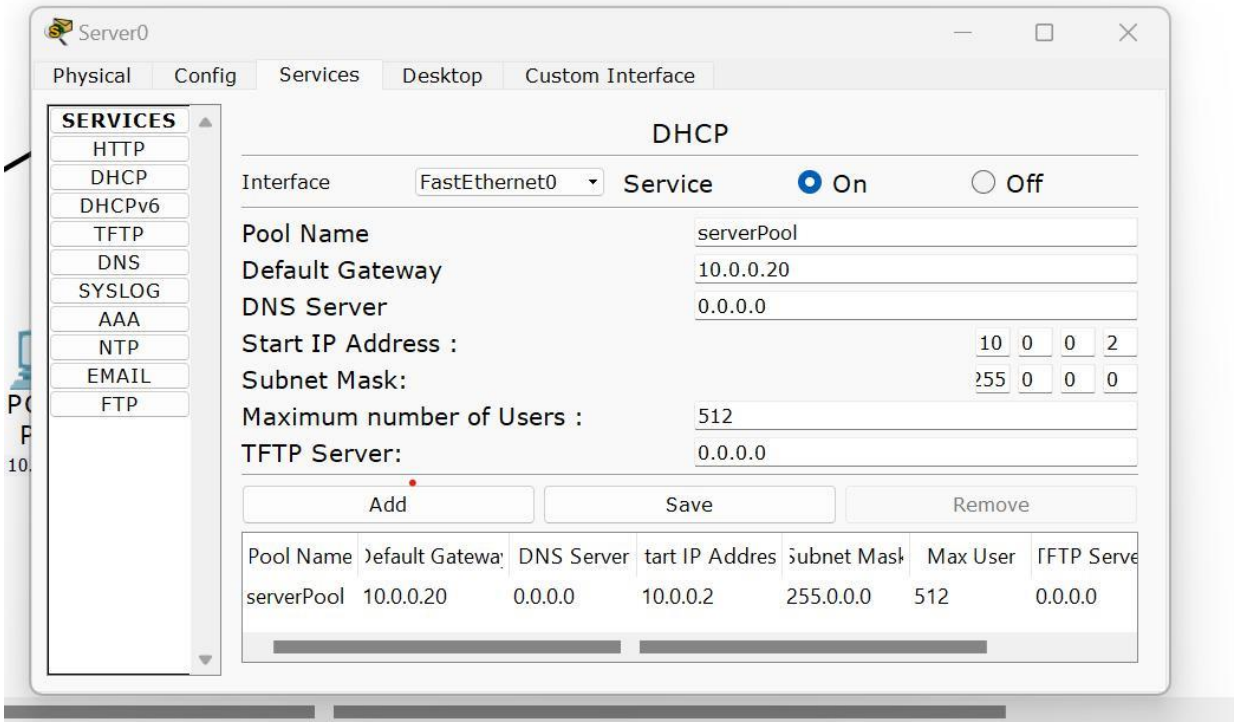
DHCP -> Dynamic host Configuration protocol. This protocol helps manage Allocation of IP Address to end-users. When a device wants Access to a network that's using DHCP, it sends a request for an IP address that is picked up by a DHCP server. The server responds by delivering an IP Address to the device from a pool of IP Addresses, then monitors the use of the Address and takes it back after a specified time/ when system shuts down.

Topology :

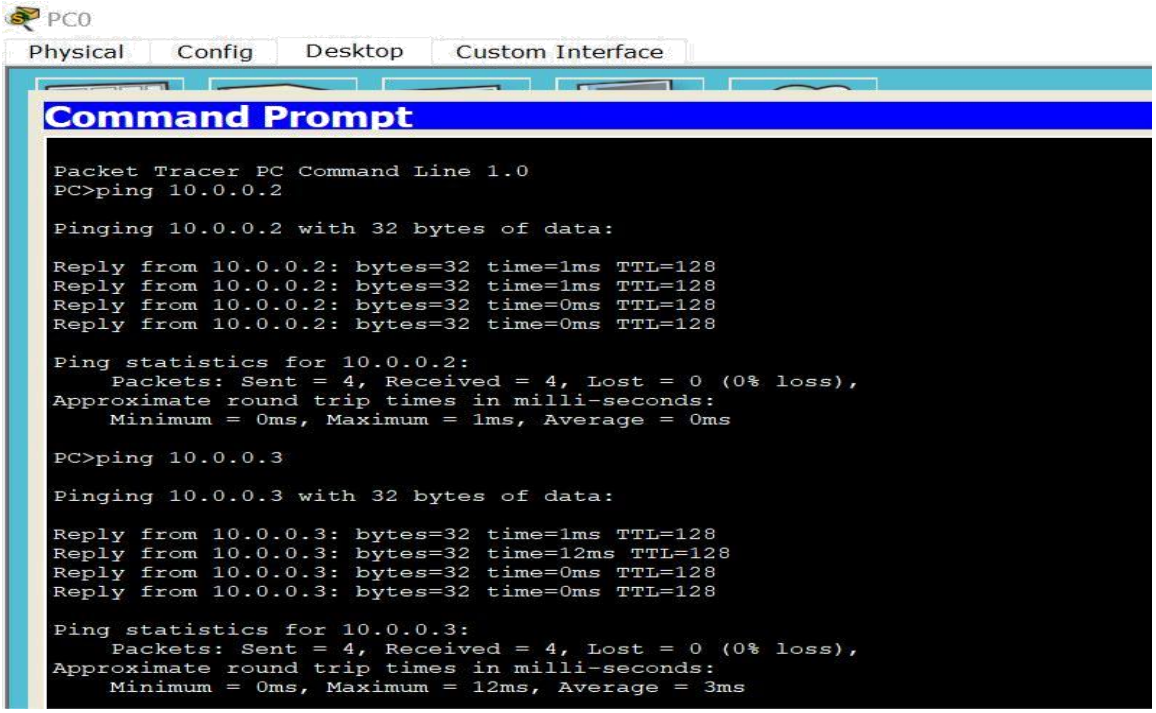




Server 0 :

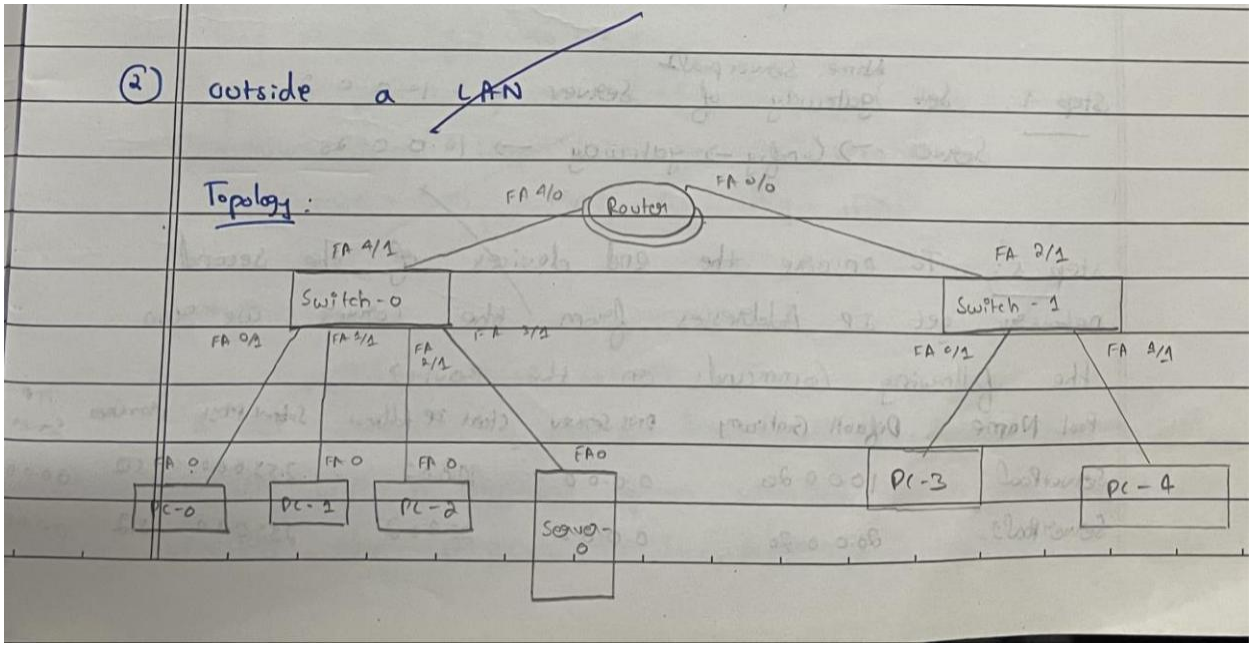


Ping Results :



2) Outside a LAN :

Procedure :



M T W T F S S

COMPASS
Date: 13/03/23

Procedure:

step 1: follow the steps mentioned under the experiment Configuring DHCP within a LAN.

step 2: Create another network with 2 PC's connected to a server using a copper straigh through wire. Connect the switch from the second network to the switch in the first network using a Router.

step 3: Configure the router [static route]
[Router → CLI]

step 3a) enable
3b) Config t
3c) interface fastethernet 1/0
3d) ip Address 10.0.0.20 255.00.0
3e) no shut
3f) exit
3g) interface fastethernet 0/0
3h) ip Address 20.0.0.20 255.0.0.0
3i) no shut
3j) exit
3k) exit
3l) show ip route

step 4: Set gateway of Server to 10.0.0.20
Name: ServerPool1
Server → Config → gateway → 10.0.0.20

step 5: To ensure the end devices of the second network get IP Addresses from the server, we run the following commands on the Router.

Pool Name	Default Gateway	DNS Server	Start IP Address	Subnet Mask	Max User	TFTP Server
ServerPool	10.0.0.20	0.0.0.0	10.0.0.2	255.0.0.0	512	0.0.0
ServerPool1	20.0.0.20	0.0.0.0	20.0.0.2	255.0.0.0	512	0.0.0

M T W T F S S

COMPASS
Date: 13/07/23

5a) enable
5b) Config t
5c) interface fastethernet 0/0
5d) ip helper-Address 10.0.0.1
5e) no shut
5f) exit

step 6: Server → Services → DHCP → serverpool 1 → start IP Address → 20.0.0.2 → Add. (Making DHCP Available to the second network)
The server then Assigns Addresses from the server pool to the PC's of second network.
(PC → desktop → IP Configuration → DHCP → DHCP req successful)
> (Default gateway - 20.0.0.20)

Result:

Pc> ping 40.0.0.1
pinging 40.0.0.1 with 32 bytes of data
Request timed out
Reply from 40.0.0.1 : bytes = 32 time = 20ms TTL = 125
Reply from 40.0.0.1 : bytes = 32 time = 9ms TTL = 125
Reply from 40.0.0.1 : bytes = 32 time = 21ms TTL = 125
ping statistics for 40.0.0.1
packets sent = 4, recieved = 3, host = 1 (Lost = 25%)
Approx round trip times in milli seconds
min = 9ms, Max = 21, Avg = 16ms

observation:

DHCP → used in TCP/IP network
A DHCP server is a way to Automatically configure the system in the LAN. A requestor from an end device sends out a message that requests an IP Address. The server responds

COMPASS
Date: 13/07/23

M T W T F S S

with an IP lease i.e an IP Address from the Address pool and some additional configuration. DHCP automates and Centrally manages these configurations rather than requiring network administrators to manually Assign IP Address to all network devices.

→ when you connect a new device, it still does not have an IP Address. It will call over the network for a DHCP server. This request arrives to all devices and the Server Also.

→ The DHCP hears the call and answers with an IP Address to the newly Connected device.

→ The IP Address will Arrive to the device and the device will Accept it and will send a request to use it.

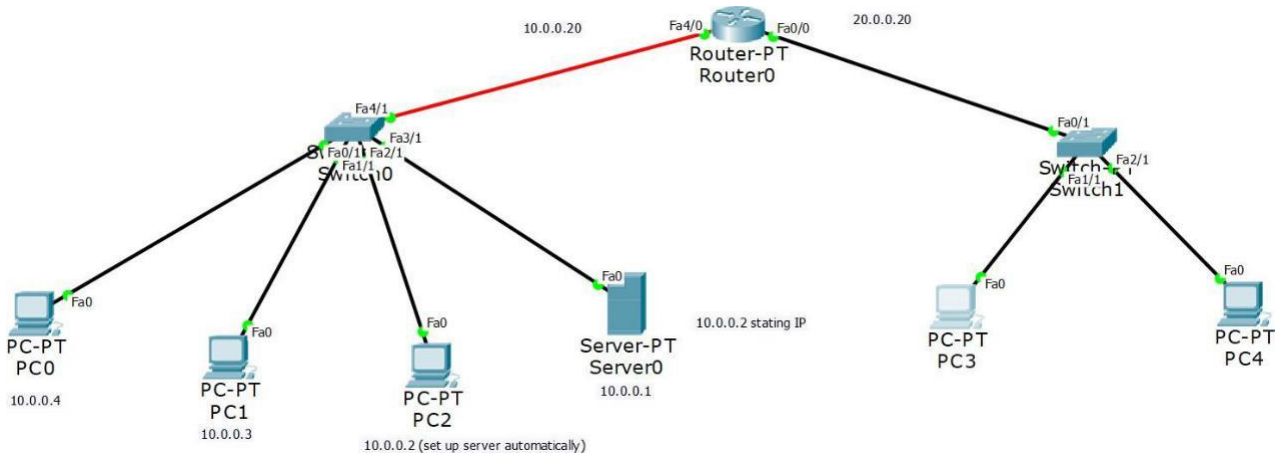
→ The server gets accepting message from the device. It will provide IP address to the device along with Subnet mask and DNS server.

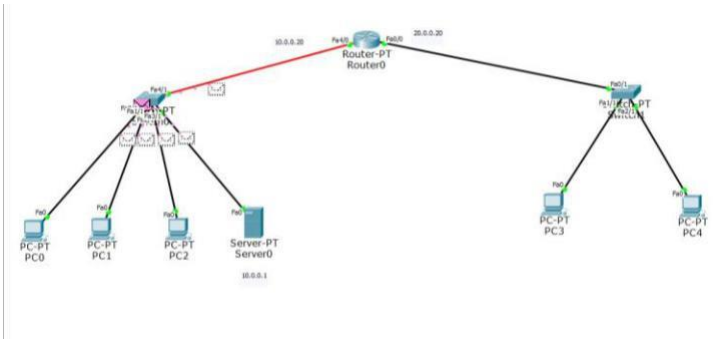
The DHCP leases the IP Address for a limited time. After the time passes, the IP Address will go back to the IP pool of available IP addresses.

18/12/23

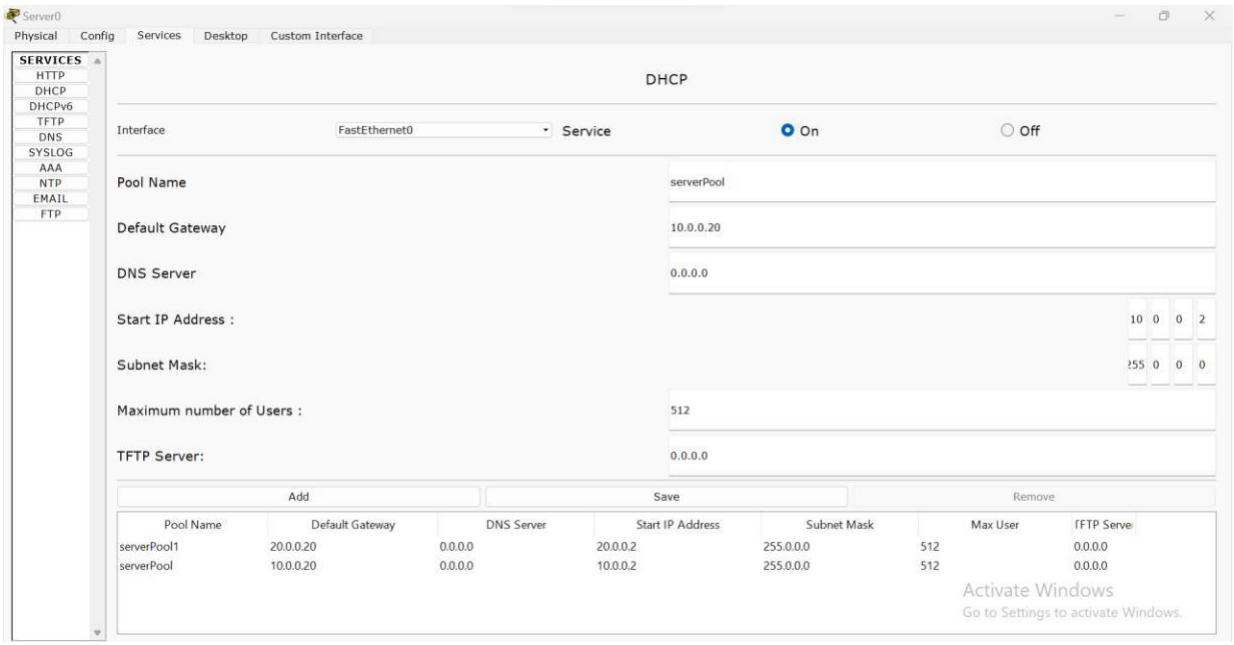
(10)

Topology :


10.0.0.20
Fa4/0
Fa0/0
20.0.0.20
Router-PT
Router0
Fa4/1
Fa3/1
Fa0/1
Fa2/1
Fa1/1
Switch1
Fa0
PC-PT
PC0
10.0.0.4
PC-PT
PC1
10.0.0.3
PC-PT
PC2
10.0.0.2 (set up server automatically)
Server-PT
Server0
10.0.0.1
10.0.0.2 stating IP
PC-PT
PC3
PC-PT
PC4



Server 0


Server0
Physical
Config
Services
Desktop
Custom Interface
SERVICES
HTTP
DHCP
DHCPv6
TFTP
DNS
SYSLOG
AAA
NTP
EMAIL
FTP
DHCP
Interface
FastEthernet0
Service
On
Off
Pool Name
serverPool
Default Gateway
10.0.0.20
DNS Server
0.0.0.0
Start IP Address :
10 0 0 2
Subnet Mask:
255 0 0 0
Maximum number of Users :
512
TFTP Server:
0.0.0.0
Add
Save
Remove
Pool Name
Default Gateway
DNS Server
Start IP Address
Subnet Mask
Max User
TFTP Serve
serverPool1
20.0.0.20
0.0.0.0
20.0.0.2
255.0.0.0
512
0.0.0.0
serverPool
10.0.0.20
0.0.0.0
10.0.0.2
255.0.0.0
512
0.0.0.0
Activate Windows
Go to Settings to activate Windows.

Router Configuration : (Router 0)

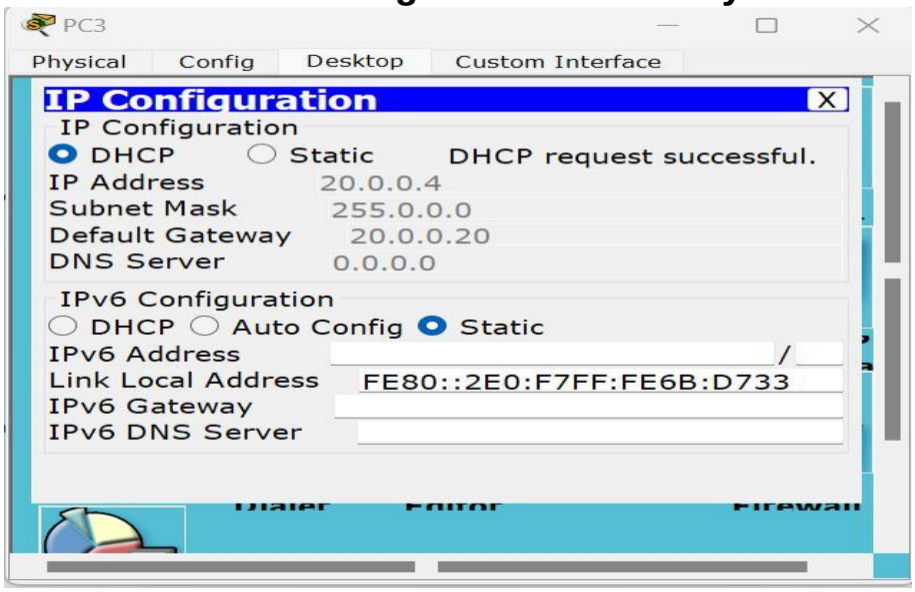
Router0

Physical | Config | CLI

IOS Command

```
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet4/0, changed state to up
exit
Router(config)#interface fastethernet0/0
Router(config-if)#ip address 20.0.0.20 255.0.0.0
Router(config-if)#no shut

Router(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up
exit
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console
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, FastEthernet4/0
C    20.0.0.0/8 is directly connected, FastEthernet0/0
Router#config t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#interface fastethernet0/0
Router(config-if)#ip helper-address 10.0.0.1
Router(config-if)#no shut
Router(config-if)#exit
Router(config)#exit
```

Automation IP is assigned in the PCs by Server 0 via DHCP:



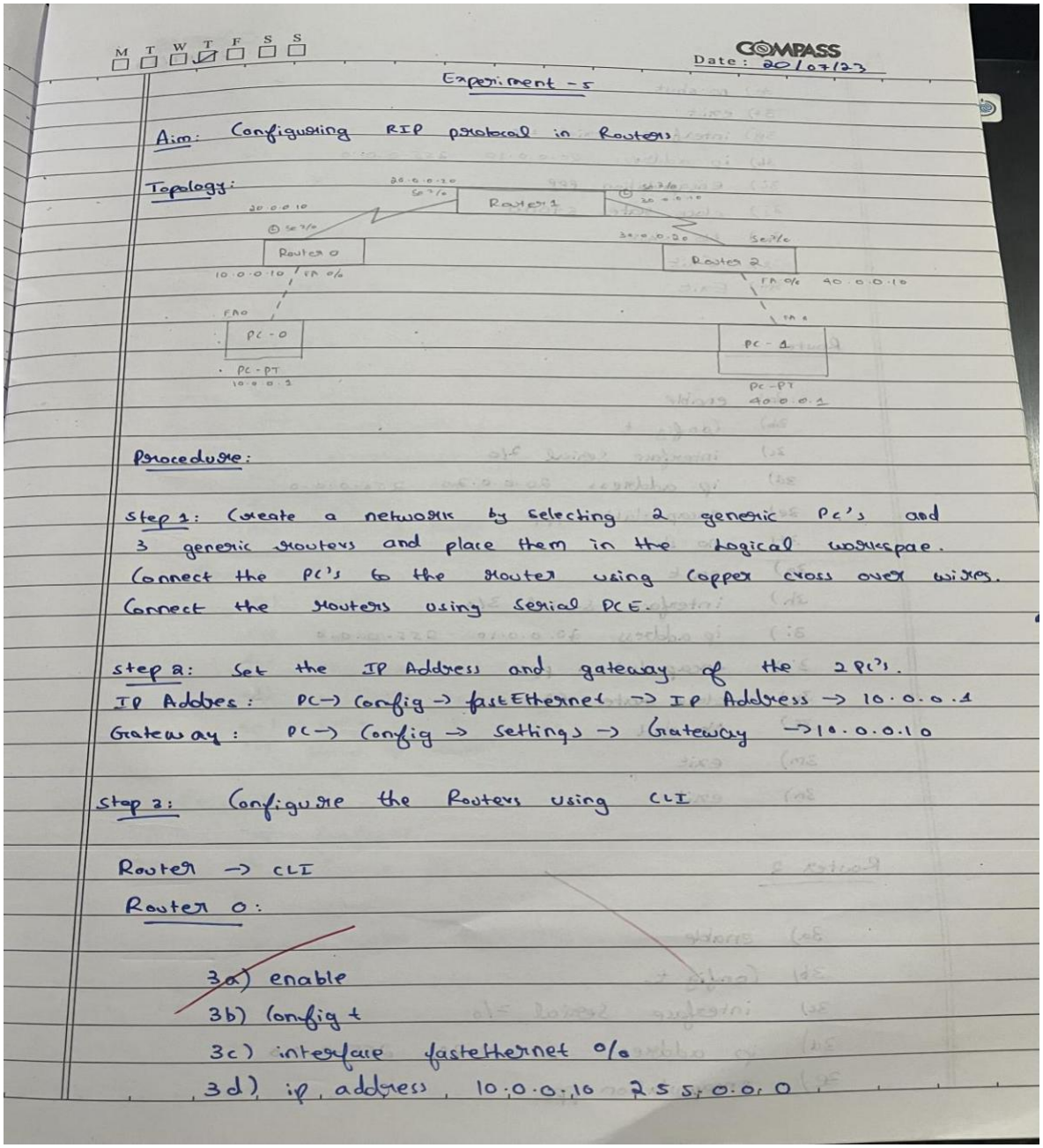
Ping Results :

```
PC>ping 20.0.0.2

Pinging 20.0.0.2 with 32 bytes of data:

Reply from 20.0.0.2: bytes=32 time=1ms TTL=127
Reply from 20.0.0.2: bytes=32 time=0ms TTL=127
Reply from 20.0.0.2: bytes=32 time=0ms TTL=127
Reply from 20.0.0.2: bytes=32 time=0ms TTL=127

Ping statistics for 20.0.0.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 1ms, Average = 0ms
```

LAB PROGRAM -5

Aim :
Configure RIP routing Protocol In Routers.

Procedure :



M T W T F S S

COMPASS
Date : 20/07/23

Experiment - 5

Aim: Configuring RIP protocol in Routers

Topology:

Procedure:

Step 1: Create a network by selecting 2 generic Pc's and 3 generic routers and place them in the logical workspace. Connect the Pc's to the router using copper cross over wires. Connect the routers using serial DCE.

step 2: Set the IP Address and gateway of the 2 Pc's.
IP Address: PC→ Config → fastEthernet → IP Address → 10.0.0.1
Gateway: PC→ Config → Settings → Gateway →10.0.0.10

Step 3: Configure the Routers using CLI

Router → CLI
Router 0:

3a) enable
3b) config t
3c) interface fastethernet 0/0
3d) ip address 10.0.0.10 255.0.0.0

COMPASS
Date: 20/07/23

M T W T F S S

3e) no shut
3f) exit
3g) interface serial 2/0
3h) ip address 20.0.0.10 255.0.0.0
3i) encapsulation ppp
3j) clock rate 64000
3k) No shut
3l) Exit
3m) Exit

Router 1

3a) enable
3b) Config t
3c) interface Serial 2/0
3d) ip address 20.0.0.20 255.0.0.0
3e) encapsulation ppp
3f) no shut
3g) exit
3h) interface Serial 3/0
3i) ip address 30.0.0.10 255.0.0.0
3j) encapsulation ppp
3k) clock rate 64000
3l) No shut
3m) exit
3n) exit

Router 2

3a) enable
3b) Config t
3c) interface Serial 2/0
3d) ip address 30.0.0.20 255.0.0.0
3e) Encapsulation ppp

M T W T F S S

COMPASS
Date: 20/07/23

3f) no shut
3g) exit
3h) interface fastethernet 0/0
3i) ip address 40.0.0.10 255.0.0.0
3j) no shut
3k) exit
3l) Exit

Step 4: Router -> CLI

Router 0:

4a) Config t
4b) Router RIP
4c) network 10.0.0.0
4d) network 20.0.0.0
4e) exit
4f) show ip route

Router 1:

4a) Config t
4b) Router RIP
4c) network 20.0.0.0
4d) network 30.0.0.0
4e) exit
4f) show ip route

Router 2:

4a) Config t
4b) Router RIP
4c) network 30.0.0.0
4d) network 40.0.0.0
4e) exit
4f) show ip route

COMPASS
Date: 20/07/23

M T W T F S S

The show ip route command will show ip Address Associated with the path as 'c' and other IP Addresses are labelled 'R'.

Result:

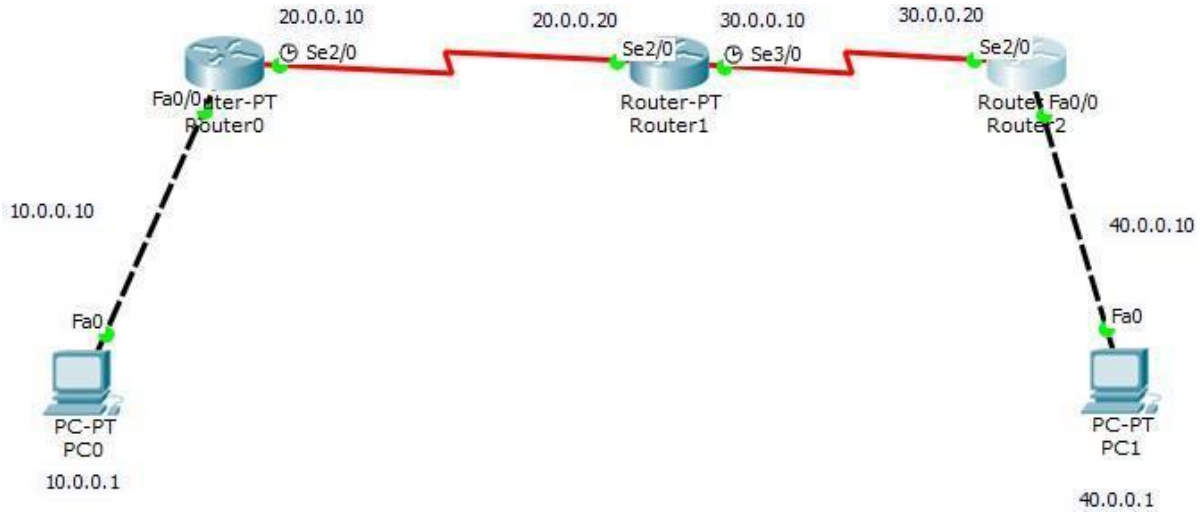
Pc > ping 40.0.0.1
pinging 40.0.0.1 with 32 bytes of data:
Request timed out
Reply from 40.0.0.1 : bytes = 32 time = 11ms TTL = 125
Reply from 40.0.0.1 : bytes = 32 time = 10ms TTL = 125
Reply from 40.0.0.1 : bytes = 32 time = 2ms TTL = 125
ping statistics for 40.0.0.1:
packets : Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
Minimum = 2ms, Maximum = 11ms, Average = 7ms

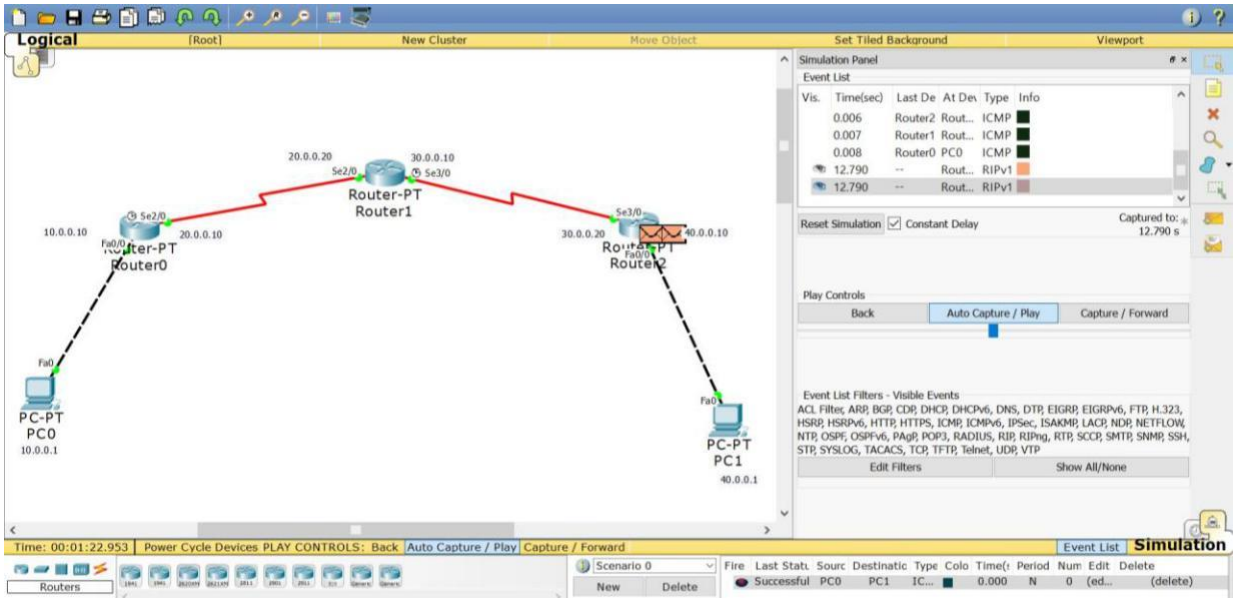
observation:

Routing information protocol (RIP) is a distance vector protocol that uses hop count as its primary metric. RIP defines how routers should share information when moving traffic among an interconnected group of local area networks.

RIP uses a distance vector Algo to decide which path to put a packet on to get to its destination. Each RIP router maintains a routing table which is a list of all the destinations the router knows how to reach. Each router broadcasts its entire routing table to it's closest neighbors every 30 seconds. The neighbors (routers) pass the information to nearest neighbors until all the RIP hosts within the network have the same knowledge of Routing paths. (∴ RIP uses the shortest no of hops to determine the best path to a remote network.)

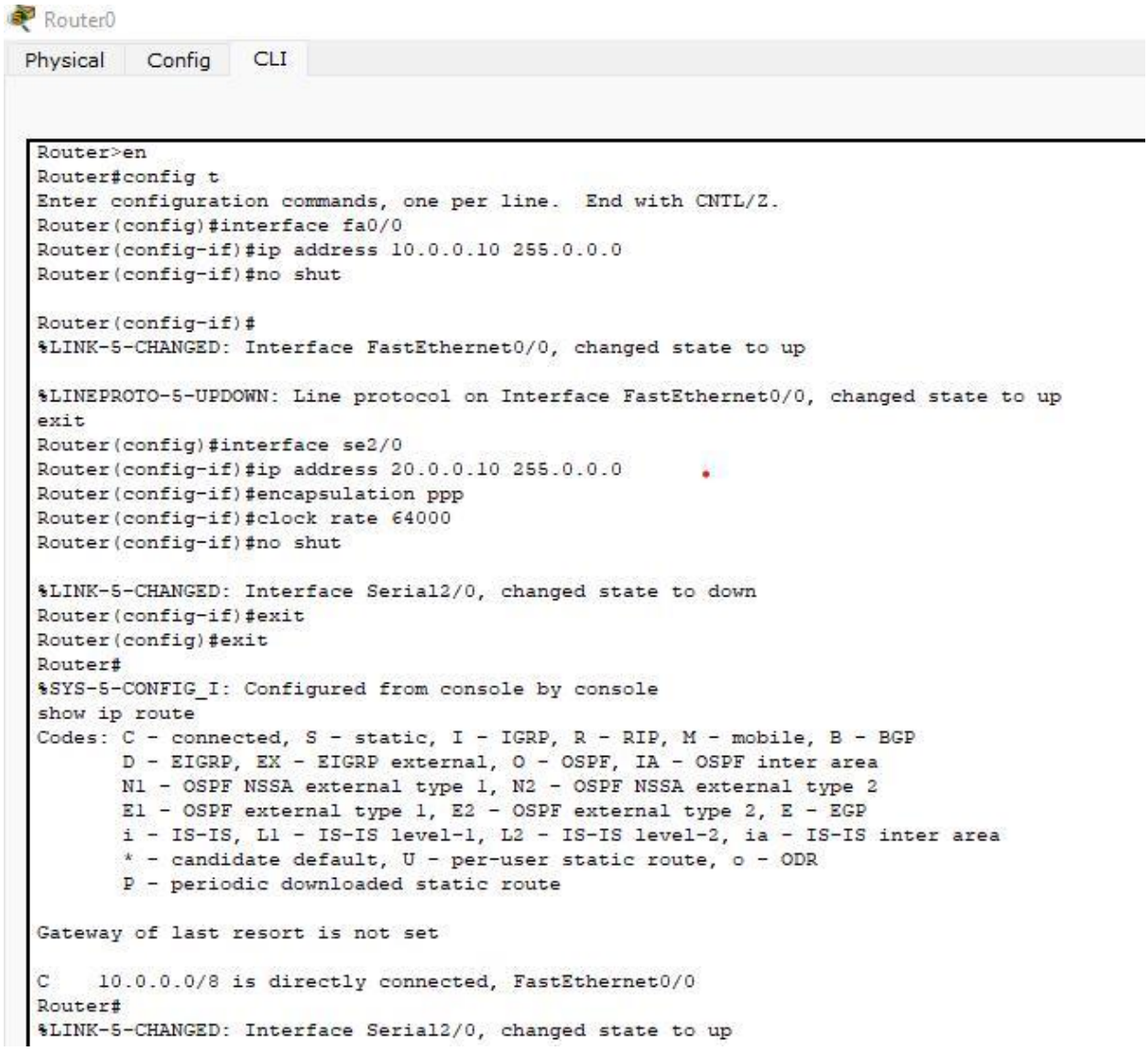
Topology :


20.0.0.10
Se2/0
20.0.0.20
Se2/0
30.0.0.10
Se3/0
30.0.0.20
Se2/0
Fa0/0
Router-PT
Router0
Router-PT
Router1
Router-PT
Router2
Fa0/0
10.0.0.10
40.0.0.10
Fa0
Fa0
PC-PT
PC0
10.0.0.1
PC-PT
PC1
40.0.0.1

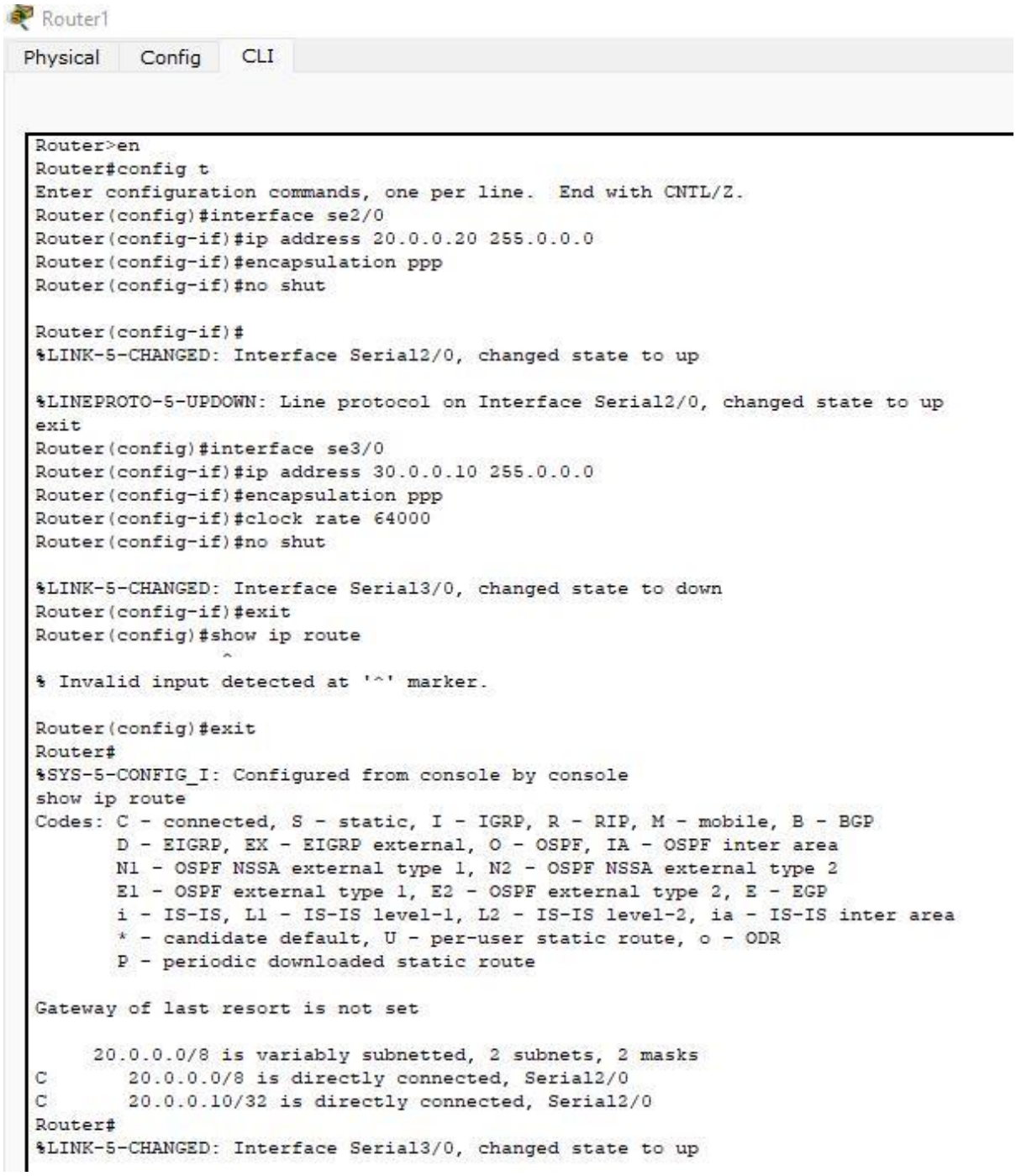

Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
Simulation Panel
Event List
Vis.
Time(sec)
Last De
At Dev
Type
Info
0.006
Router2
Rout...
ICMP
0.007
Router1
Rout...
ICMP
0.008
Router0
PC0
ICMP
12.790
Rout...
RIPv1
12.790
Rout...
RIPv1
Reset Simulation
Constant Delay
Captured to:
12.790 s
Play Controls
Back
Auto Capture / Play
Capture / Forward
Event List Filters - Visible Events
ACL Filter, ARP, BGP, CDP, DHCP, DHCPv6, DNS, DTP, EIGRP, EIGRPv6, FTP, H.323, HSRP, HSRPv6, HTTP, HTTPS, ICMP, ICMPv6, IPSec, ISAKMP, LACP, NDP, NETFLOW, NTP, OSPF, OSPFv6, PAgP, POP3, RADIUS, RIP, RIPng, RTP, SCCP, SMTP, SNMP, SSH, STP, SYSLOG, TACACS, TCP, TFTP, Telnet, UDP, VTP
Edit Filters
Show All/None
Time: 00:01:22.953
Power Cycle Devices
PLAY CONTROLS:
Back
Auto Capture / Play
Capture / Forward
Event List
Simulation
Routers
Scenario 0
New
Delete
Fire
Last Stat
Sourc
Destinatic
Type
Colo
Time(
Period
Num
Edit
Delete
Successful
PC0
PC1
0.000
N
0
(edit)
(delete)

Router Configuration:

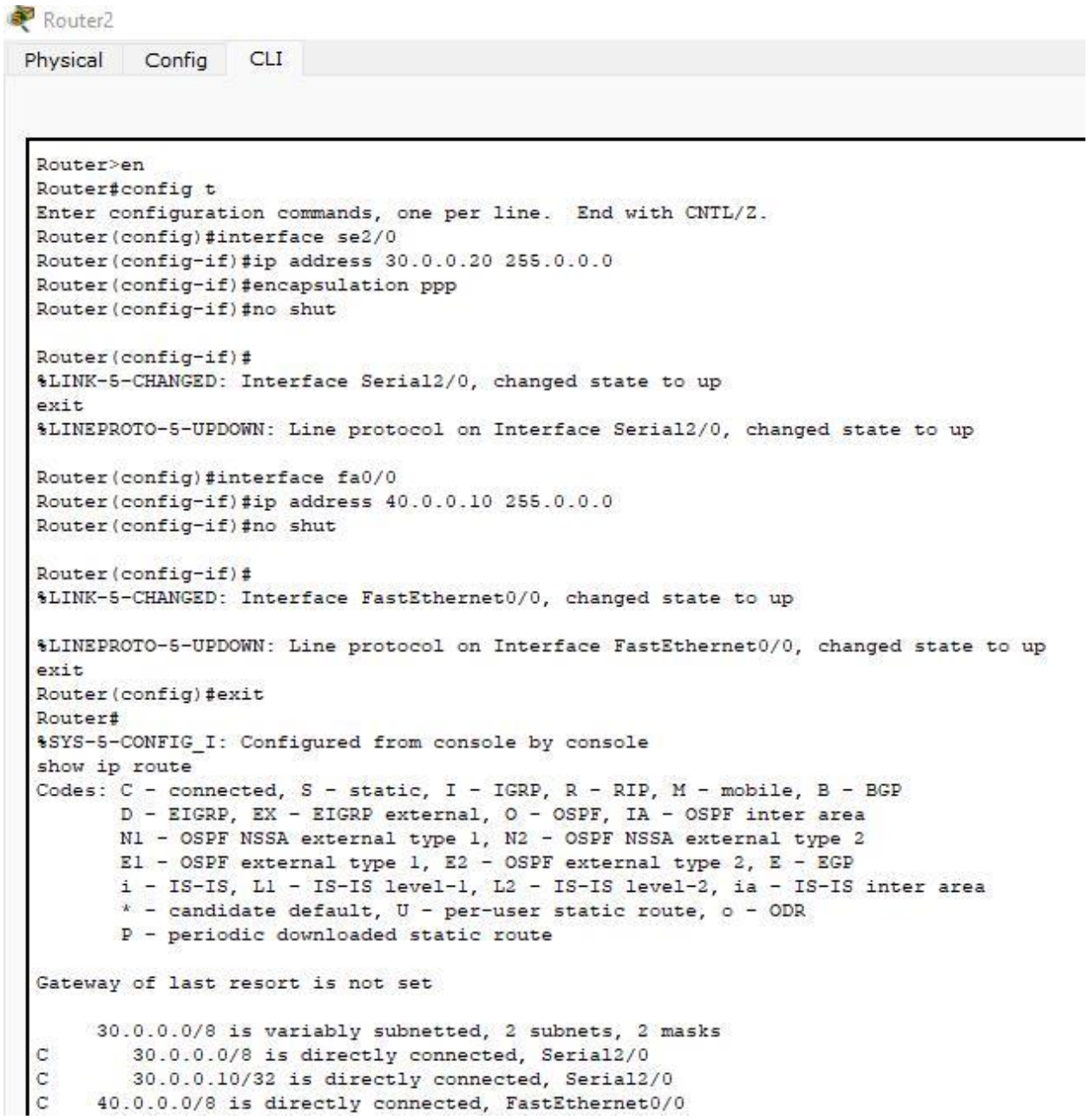
Router 0 :



Router 1 :


Router1
Physical
Config
CLI
Router>en
Router#config t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface se2/0
Router(config-if)#ip address 20.0.0.20 255.0.0.0
Router(config-if)#encapsulation ppp
Router(config-if)#no shut
Router(config-if)#
%LINK-5-CHANGED: Interface Serial2/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to up
exit
Router(config)#interface se3/0
Router(config-if)#ip address 30.0.0.10 255.0.0.0
Router(config-if)#encapsulation ppp
Router(config-if)#clock rate 64000
Router(config-if)#no shut
%LINK-5-CHANGED: Interface Serial3/0, changed state to down
Router(config-if)#exit
Router(config)#show ip route
% Invalid input detected at '^' marker.
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console
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
20.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C 20.0.0.0/8 is directly connected, Serial2/0
C 20.0.0.10/32 is directly connected, Serial2/0
Router#
%LINK-5-CHANGED: Interface Serial3/0, changed state to up

Router 2 :


Router2
Physical
Config
CLI
Router>en
Router#config t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface se2/0
Router(config-if)#ip address 30.0.0.20 255.0.0.0
Router(config-if)#encapsulation ppp
Router(config-if)#no shut
Router(config-if)#
%LINK-5-CHANGED: Interface Serial2/0, changed state to up
exit
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to up
Router(config)#interface fa0/0
Router(config-if)#ip address 40.0.0.10 255.0.0.0
Router(config-if)#no shut
Router(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up
exit
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console
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
30.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C 30.0.0.0/8 is directly connected, Serial2/0
C 30.0.0.10/32 is directly connected, Serial2/0
C 40.0.0.0/8 is directly connected, FastEthernet0/0

RIP routing:

Router 0:

```
Router#
%LINK-5-CHANGED: Interface Serial2/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to up
config t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#router rip
Router(config-router)#network 10.0.0.0
Router(config-router)#network 20.0.0.0
Router(config-router)#exit
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console
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
     20.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C       20.0.0.0/8 is directly connected, Serial2/0
C       20.0.0.20/32 is directly connected, Serial2/0
R    30.0.0.0/8 [120/1] via 20.0.0.20, 00:00:18, Serial2/0
R    40.0.0.0/8 [120/2] via 20.0.0.20, 00:00:18, Serial2/0
Router#
```

Router 1:

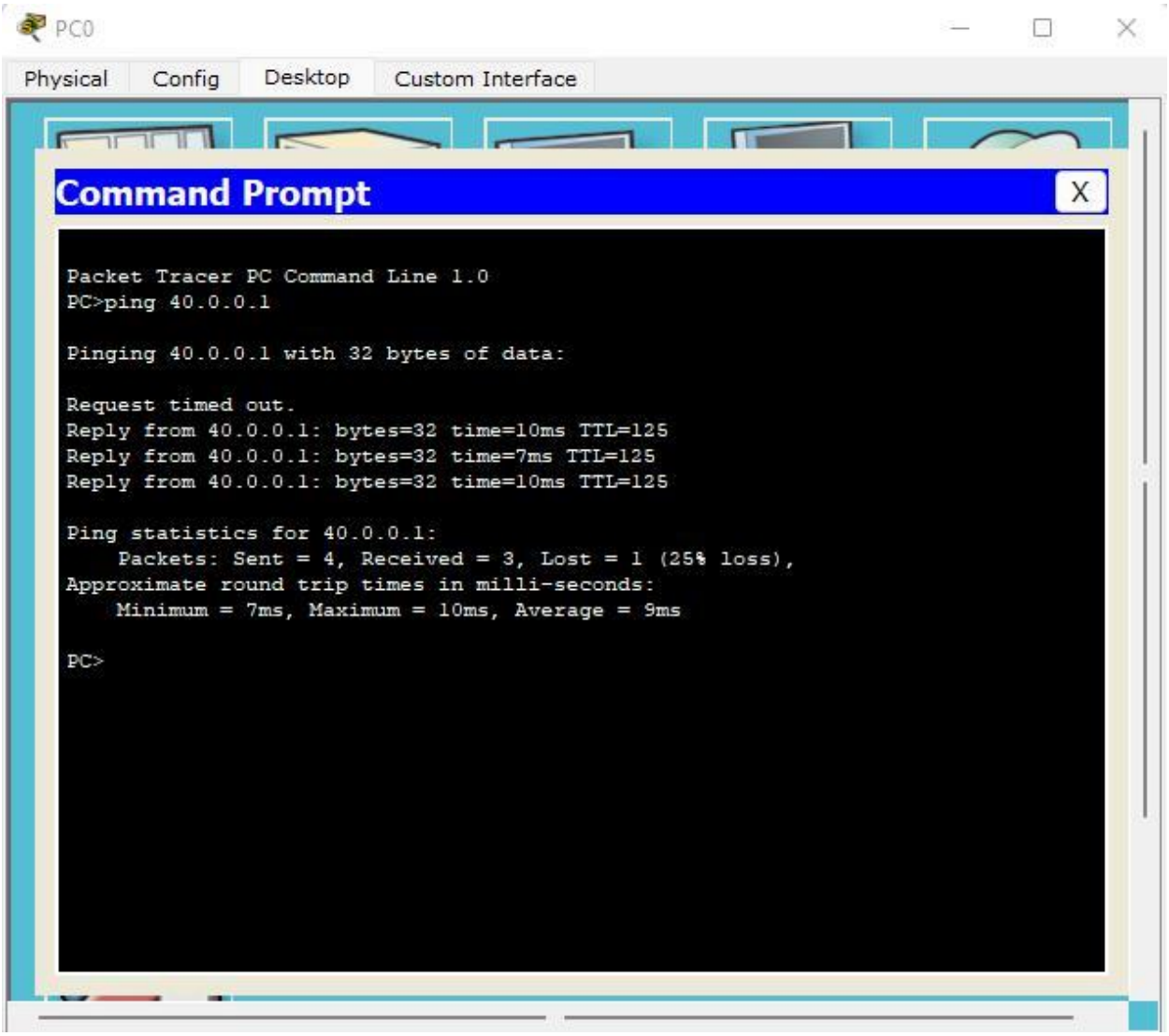
```
Router#
%LINK-5-CHANGED: Interface Serial3/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial3/0, changed state to up
config t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#router rip
Router(config-router)#network 20.0.0.0
Router(config-router)#network 30.0.0.0
Router(config-router)#exit
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console
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

R    10.0.0.0/8 [120/1] via 20.0.0.10, 00:00:20, Serial2/0
     20.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C       20.0.0.0/8 is directly connected, Serial2/0
C       20.0.0.10/32 is directly connected, Serial2/0
     30.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C       30.0.0.0/8 is directly connected, Serial3/0
C       30.0.0.20/32 is directly connected, Serial3/0
R    40.0.0.0/8 [120/1] via 30.0.0.20, 00:00:19, Serial3/0
Router#
```

Router 2:

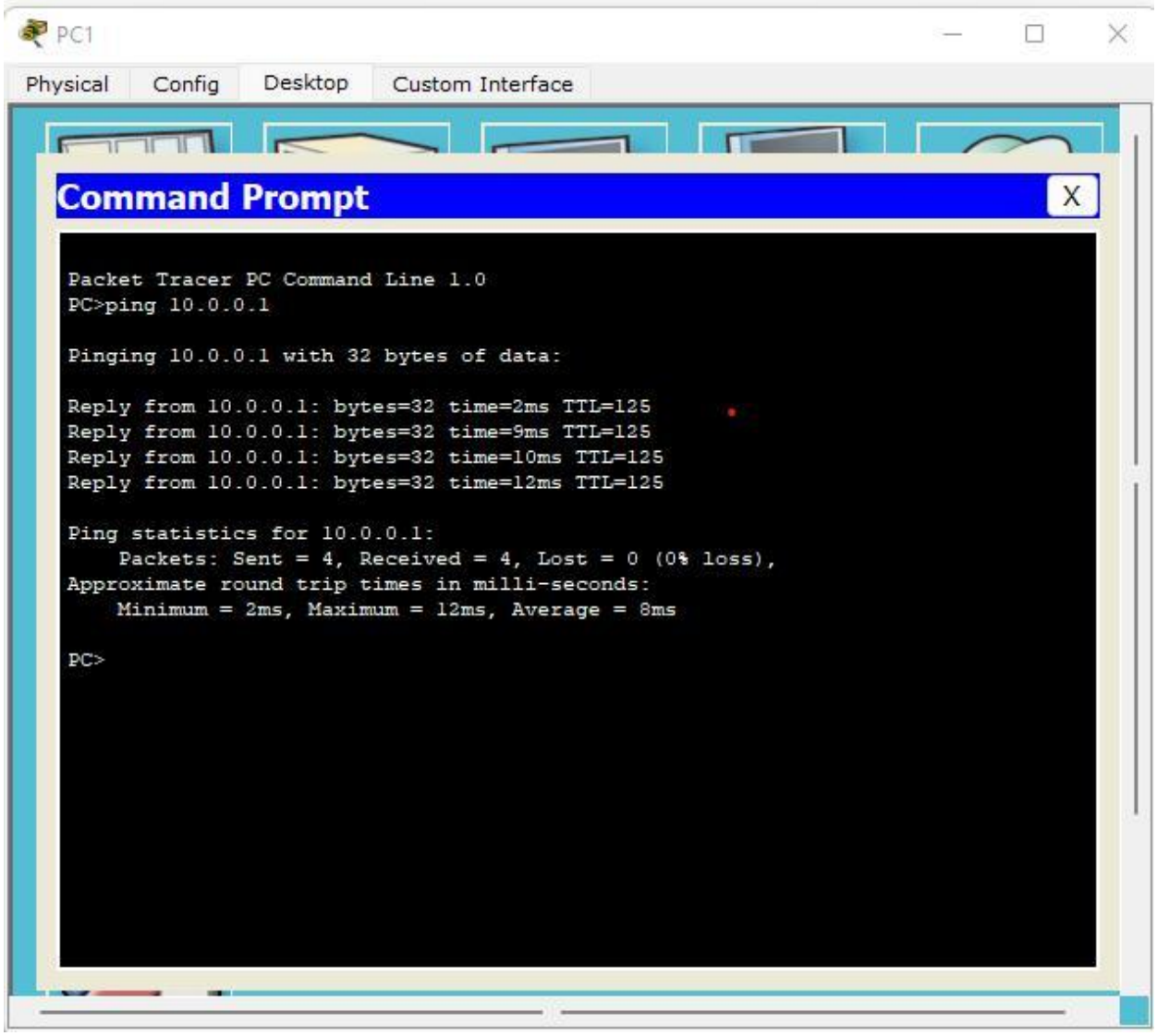
```
Router#config t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#network 30.0.0.0
                ^
% Invalid input detected at '^' marker.

Router(config)#router rip
Router(config-router)#network 30.0.0.0
Router(config-router)#network 40.0.0.0
Router(config-router)#exit
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console
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

R    10.0.0.0/8 [120/2] via 30.0.0.10, 00:00:14, Serial2/0
R    20.0.0.0/8 [120/1] via 30.0.0.10, 00:00:14, Serial2/0
     30.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C       30.0.0.0/8 is directly connected, Serial2/0
C       30.0.0.10/32 is directly connected, Serial2/0
C    40.0.0.0/8 is directly connected, FastEthernet0/0
Router#
```

Ping Result :

P0:



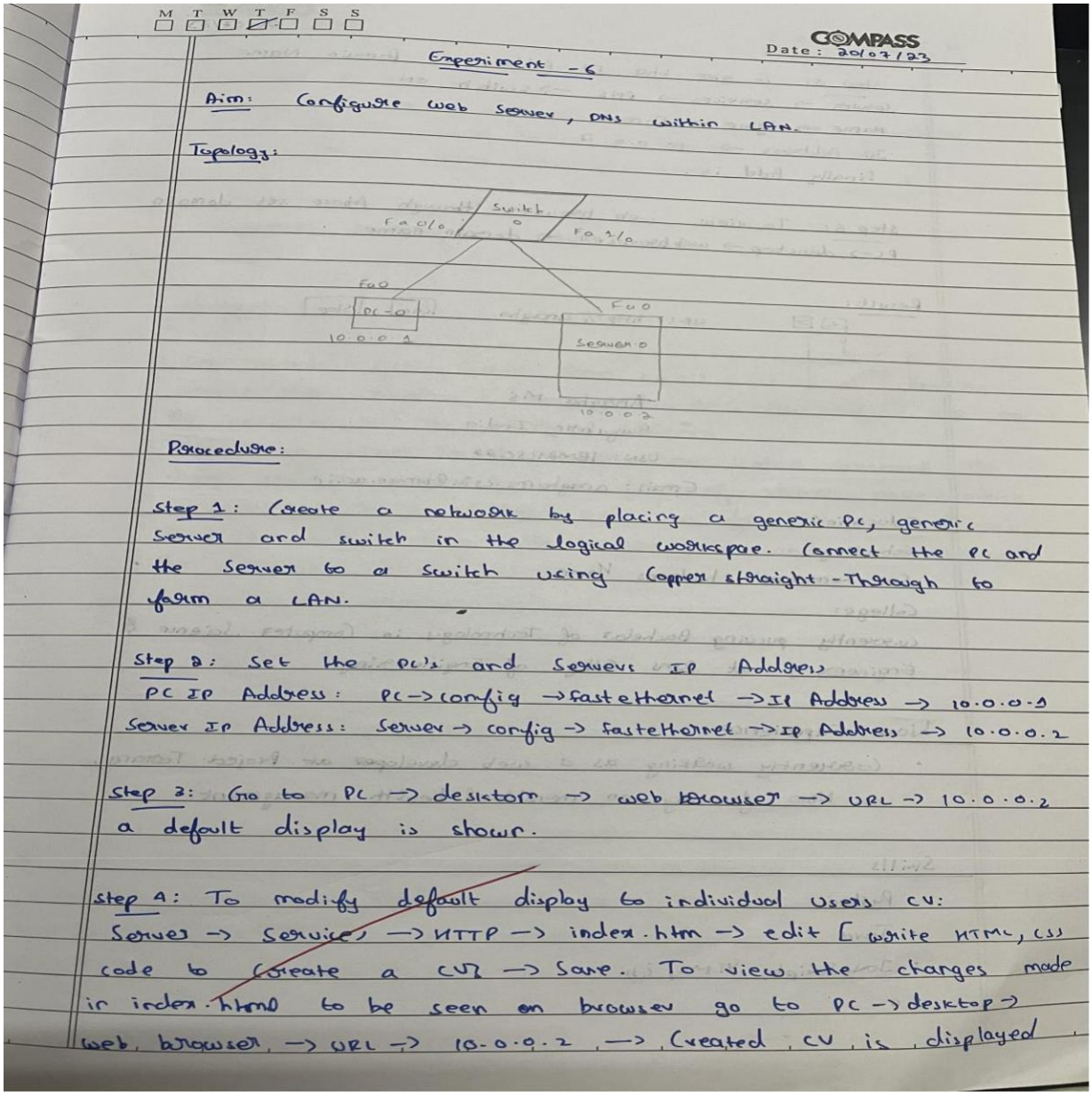
P1:


PC1
Physical
Config
Desktop
Custom Interface
Command Prompt
X
Packet Tracer PC Command Line 1.0
PC>ping 10.0.0.1
Pinging 10.0.0.1 with 32 bytes of data:
Reply from 10.0.0.1: bytes=32 time=2ms TTL=125
Reply from 10.0.0.1: bytes=32 time=9ms TTL=125
Reply from 10.0.0.1: bytes=32 time=10ms TTL=125
Reply from 10.0.0.1: bytes=32 time=12ms TTL=125
Ping statistics for 10.0.0.1:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 2ms, Maximum = 12ms, Average = 8ms
PC>

LAB PRGRAM -6

Aim :
Configure Web Server, DNS within LAN.

Procedure :



Experiment - 6

Aim: Configure web server, DNS within LAN.

Topology:

Procedure:

Step 1: Create a network by placing a generic PC, generic Server and switch in the logical workspace. Connect the PC and the Server to a switch using Copper straight-Through to form a LAN.

Step 2: Set the PC's and Servers IP Address
PC IP Address: PC -> config -> fastethernet -> IP Address -> 10.0.0.1
Server IP Address: Server -> config -> fastethernet -> IP Address -> 10.0.0.2

Step 3: Go to PC -> desktop -> web browser -> URL -> 10.0.0.2
a default display is shown.

Step 4: To modify default display to individual users CV:
Server -> Services -> HTTP -> index.htm -> edit [write HTML, CSS code to create a CV] -> Save. To view the changes made in index.html to be seen on browser go to PC -> desktop -> web browser -> URL -> 10.0.0.2 -> Created CV is displayed.

COMPASS
Date: 20/07/23

M T W T F S S

Step 5: To give the IP Address a Domain Name
Server → services → DNS → Switch ON
Name → give a Domain name
IP Address → 10.0.0.2
finally Add it.

Step 6: To view web browser through Above set domain
PC → desktop → web browser → domain name.

Result:

URL: http://anagha Go Stop

Image to be placed here

Anagha MS
Bangalore, India
USN: 1BM21CS022
Email: anaghams.cs21@bmsce.ac.in

Education

Schooling:
Sri Kumarans Childrens Home

College:
Currently pursuing Bachelor of Technology in Computer Science & Engineering at BMS College of Engineering.

Work Experience

- Currently working as a web developer at Project Tarang, an NGO focusing on menstrual health management

Skills

- Python
- Java
- JavaScript, HTML, CSS
- C

M T W T F S S

COMPASS

Date: 20/07/23

Projects

College Activity Page

- Was part of a 4-membered team to design a website
- Provides students Ability to Register for events.
- Designed using Javascript, HTML.

References

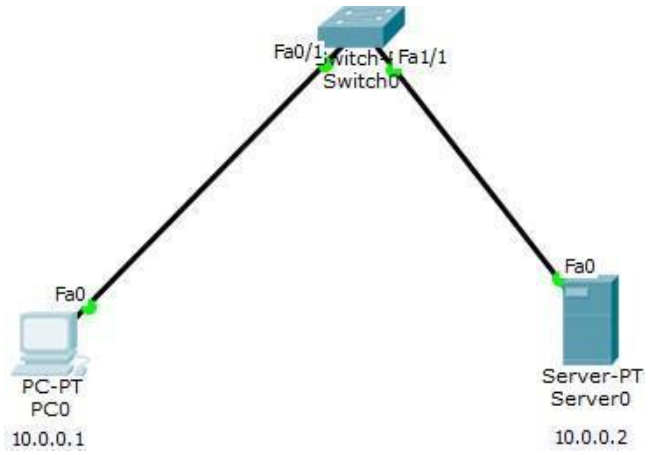
References Available upon request.

observation:

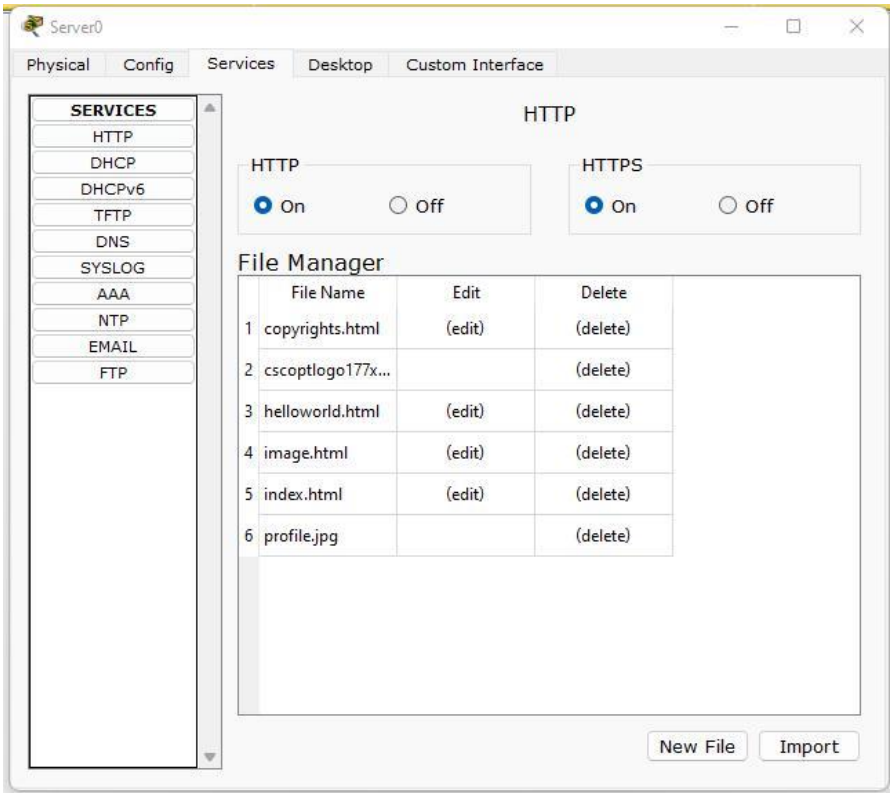
DNS stands for Domain Name System. It is a directory service that provides a mapping b/w the name of a host on the network and its numerical Address.

DNS is a client/server network Communication protocol. The client sends a request (in the form of domain name) which is converted to IP Address knows as forward DNS lookup. The Computer is then able to retrieve and interact with the webpage.

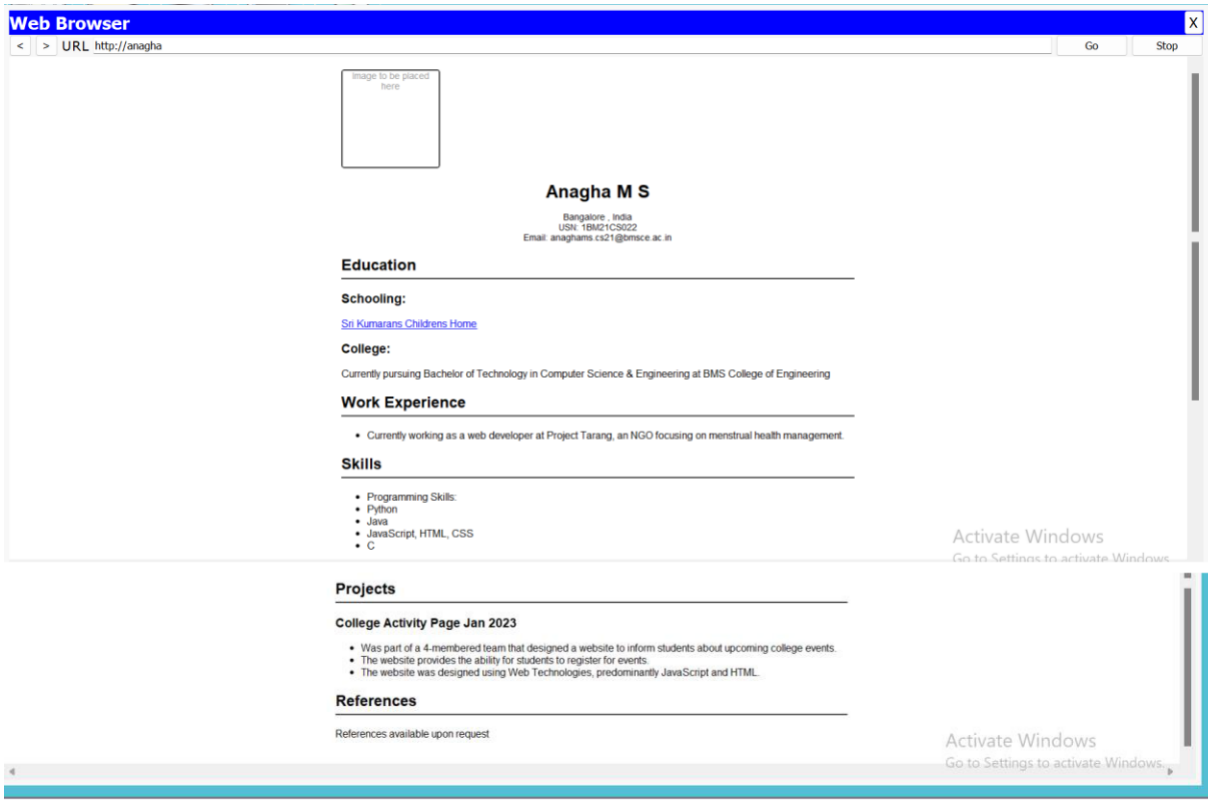
Topology :



Server 0 :



Website:



Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Your CV</title>
<style>
body {
font-family: Arial, sans-serif;
margin: 0;
padding: 0;
}

.container {
display: flex;
flex-direction: column-reverse; /* Reversed column order */
align-items: flex-end; /* Right alignment */
max-width: 800px;
margin: 0 auto;
padding: 20px;
}
```

```
.left-column {
flex-basis: 30%;
text-align: center;
}

.right-column {
flex-basis: 70%;
}

.profile-picture {
width: 150px;
height: 150px;
border: 2px solid #333; /* Border style for the image box */
border-radius: 5px;
position: relative; /* Positioning for the placeholder */
overflow: hidden;
margin-bottom: 20px;
}

.profile-picture img {
width: 100%;
height: 100%;
object-fit: cover;
}

.placeholder-text {
position: absolute;
top: 0;
right: 0;
bottom: 0;
left: 0;
display: flex;
justify-content: center;
align-items: center;
font-size: 14px;
color: #999;
background-color: rgba(255, 255, 255, 0.7);
}

.name {
font-size: 28px;
margin-bottom: 10px;
}

.contact-info {
font-size: 14px;
}

h2 {
margin-top: 20px;
```

```
border-bottom: 2px solid #333;
padding-bottom: 5px;
}

/* Additional styling */
.work-experience,
.skills,
.projects,
.references {
margin-bottom: 20px;
}

.work-experience ul,
.skills ul,
.projects ul {
list-style-type: disc;
padding-left: 20px;
}

.projects h3 {
margin-top: 10px;
}
</style>
</head>
<body>
<div class="container">
<div class="left-column">
<div class="profile-picture">
<!-- Placeholder for the image -->
<div class="placeholder-text">Image to be placed
here</div> </div>
<h1 class="name">Anagha M S</h1>
<p class="contact-info">
Bangalore , India<br>
USN: 1BM21CS022<br>
Email: anaghams.cs21@bmsce.ac.in
</p>
</div>
<div class="right-column">
<h2>Education</h2>
<h3>Schooling:</h3>
<!-- Link to school website -->
<p><a href="https://kumarans.org/" target="_blank">Sri Kumarans Childrens
Home</a></p>
<h3>College:</h3>
<p>Currently pursuing Bachelor of Technology in Computer Science & Engineering at BMS
College of Engineering</p>

<h2 class="work-experience">Work
Experience</h2> <ul>
```

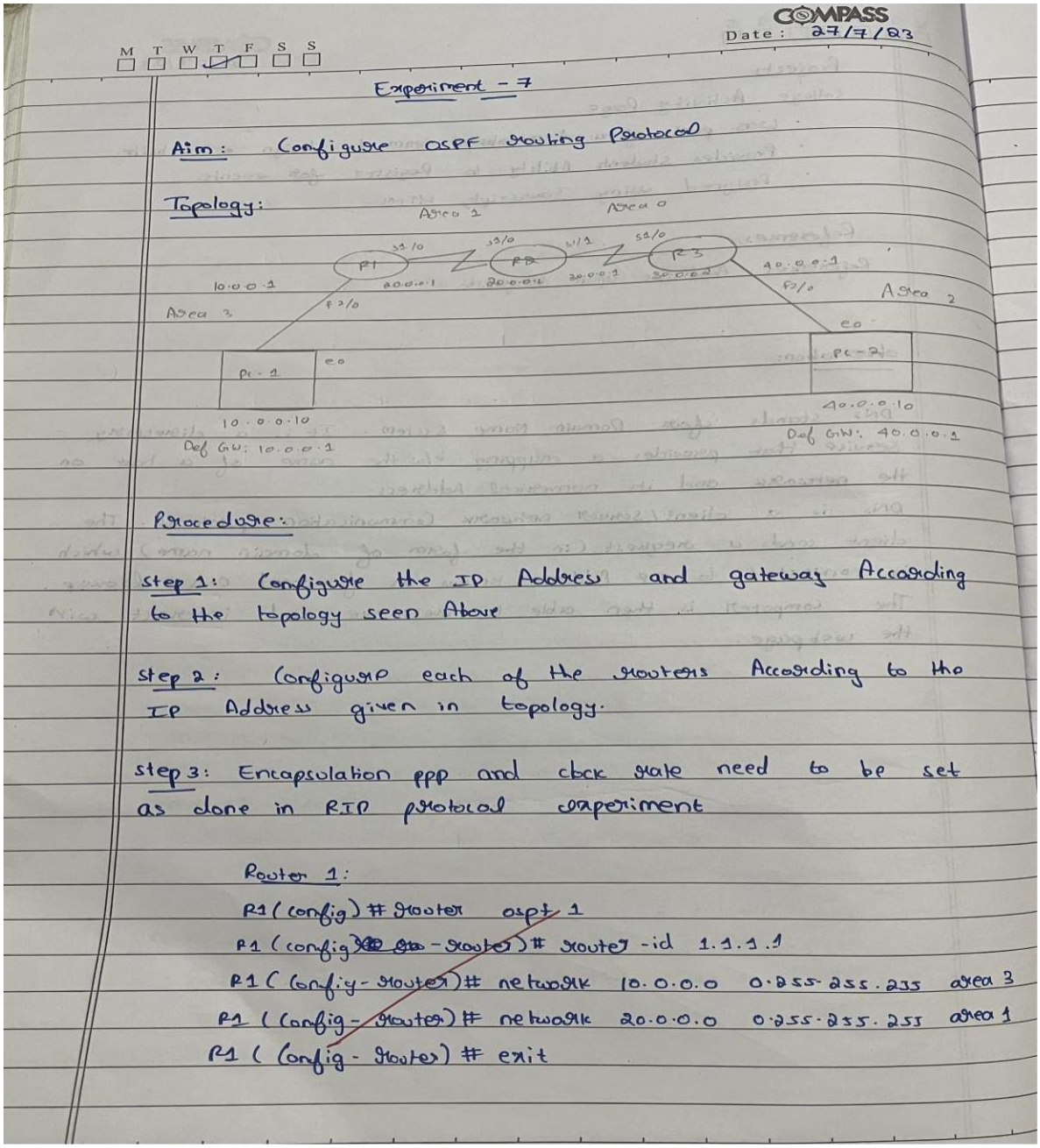
```
<li>Currently working as a web developer at Project Tarang, an NGO focusing on menstrual health management.</li>
</ul>

<h2 class="skills">Skills</h2>
<ul>
<li>Programming Skills:</li>
<li>Python</li>
<li>Java</li>
<li>JavaScript, HTML, CSS</li>
<li>C</li>
</ul>

<h2 class="projects">Projects</h2>
<h3>College Activity Page Jan 2023</h3>
<ul>
<li>Was part of a 4-membered team that designed a website to inform students about upcoming college events.</li>
<li>The website provides the ability for students to register for events.</li>
<li>The website was designed using Web Technologies, predominantly JavaScript and HTML.</li>
</ul>

<h2 class="references">References</h2>
<p>References available upon request</p>
</div>
</div>
</body>
</html>
```

LAB PROGRAM -7

Aim :

Configure OSPF Routing Protocol

Procedure :



COMPASS
Date : 27/7/23

M T W T F S S

Experiment - 7

Aim : Configure OSPF routing Protocol

Topology:

Procedure:

step 1: Configure the IP Address and gateway According to the topology seen Above

step 2: Configure each of the routers According to the IP Address given in topology.

step 3: Encapsulation ppp and clock rate need to be set as done in RIP protocol experiment

Router 1:

R1(config)# router ospf 1
R1(config-router)# router-id 1.1.1.1
R1(config-router)# network 10.0.0.0 0.255.255.255 area 3
R1(config-router)# network 20.0.0.0 0.255.255.255 area 1
R1(config-router)# exit

M T W T F S S

COMPASS
Date: 27/7/23

Router 2:

R2(config)# router ospf 1
R2(config-router)# router-id 2.2.2.2
R2(config-router)# network 20.0.0.0 0.255.255.255 area 1
R2(config-router)# network 30.0.0.0 0.255.255.255 area 0
R2(config-router)# exit

Router 3:

R3(config)# router ospf 1
R3(config-router)# router-id 3.3.3.3
R3(config-router)# network 30.0.0.0 0.255.255.255 area 0
R3(config-router)# network 40.0.0.0 0.255.255.255 area 2
R3(config-router)# exit

step 1: To keep the routers Active we have to configure interface loopback

Router 1:

R1(config-if)# interface loopback 0
R1(config-if)# ip address 172.16.1.252 255.255.0.0
R1(config-if)# no shutdown

Router 2:

R2(config-if)# interface loopback 0
R2(config-if)# ip address 172.16.1.253 255.255.0.0
R2(config-if)# no shutdown

Router 3:

R3(config-if)# interface loopback 0
R3(config-if)# ip address 172.16.1.254 255.255.0.0
R3(config-if)# no shutdown

M T W T F S S

COMPASS

Date : 27/7/23

Step 5: Create virtual link b/w R1, R2 by this we can create a virtual link to Connect to Area 0

Router 1:

```
R1(config)# router ospf 1
R1(config-router)# area 1 virtual-link 2.2.2.2
```

Router 2:

```
R2(config)# router ospf 1
R2(config-router)# area 1 virtual-link 1.1.1.1
R2(config-router)# exit
```

Finally, After creating virtual link, show ip route for all routers.

Result:

```
Pc> ping 40.0.0.10
pinging 40.0.0.10 with 32 bytes of data:
Request timed out
Reply from 40.0.0.10: bytes=32 time=10ms TTL=125
Reply from 40.0.0.10: bytes=32 time=2ms TTL=125
Reply from 40.0.0.10: bytes=32 time=9ms TTL=125
ping statistics for 40.0.0.10:
    Packets: Sent = 4, Recieved = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 2ms, Maximum = 10ms, Average = 7ms
```

M T W T F S S

COMPASS

Date: 27/7/23

Router 1:

```
show ip route
O IA  10.0.0.0/8 [110/65] via 20.0.0.1 00:00:11, Serial2/0
      20.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        20.0.0.0/8 is directly connected, Serial2/0
C        20.0.0.2/32 is directly connected, Serial2/0
      30.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        30.0.0.0/8 is directly connected, Serial 3/0
C        30.0.0.2/32 is directly connected Serial 3/0
O IA  40.0.0.0/8 [110/65] via 30.0.0.2, 00:04:44, Serial 3/0
C     172.16.0.0/16 is directly connected, Loopback0
```

observation:

The OSPF protocol is a link-state routing protocol, which means that the routers exchange topology information with their nearest neighbors.

It is an intradomain protocol, which means that it is used within an area. Each router contains the information of every domain, and based on this information, it determines shortest path. The goal of routing is to learn routes. The OSPF achieves by learning about every router and subnet within the entire network. The way the router learns this information by sending LSA. These LSAs contain information about every router, subnet, and other networking information. [To keep routers Active we use interface loop back.].

We use virtual link to connect to the backbone through a non-backbone area.

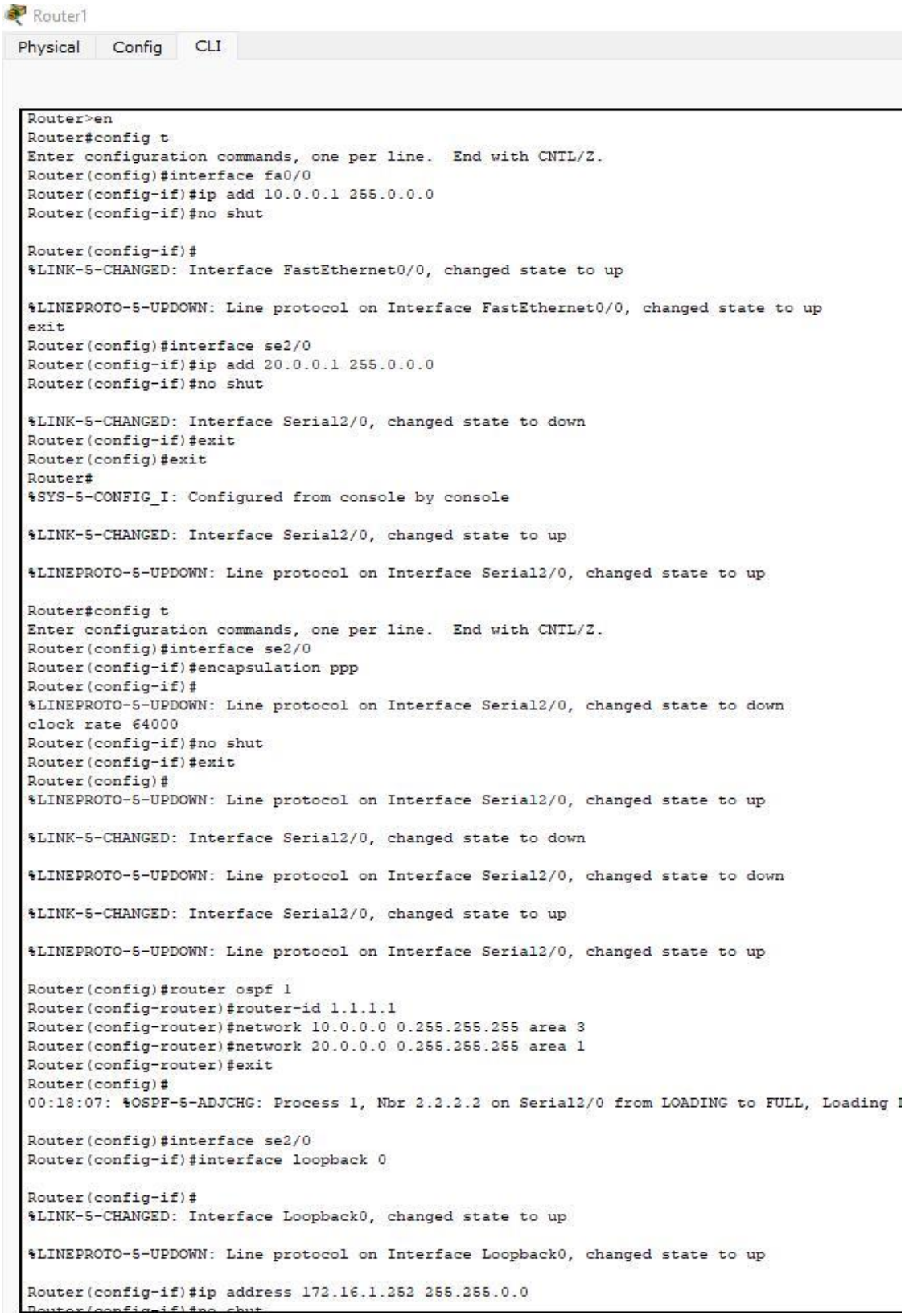
27/7/23

Topology :


Area 1
Area 0
20.0.0.1
20.0.0.2
30.0.0.1
30.0.0.2
Se2/0
Se2/0
Se3/0
Se2/0
Fa0/0
Fa0/0
Router-PT
Router1
Router-PT
Router2
Router-PT
Router3
10.0.0.1
40.0.0.1
Area 3
Area 2
Fa0
Fa0
PC-PT
PC0
PC-PT
PC1
10.0.0.10
40.0.0.10

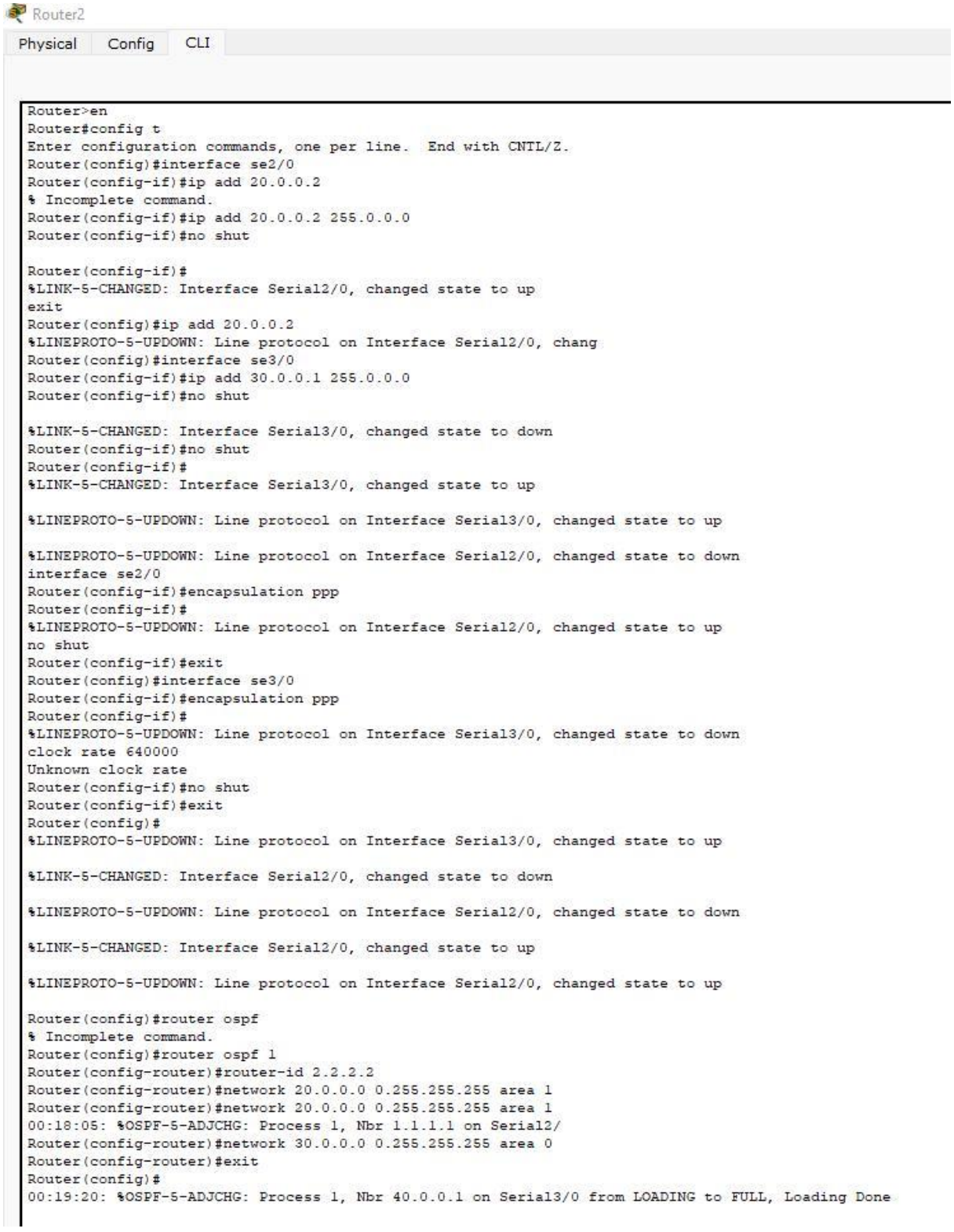
Router Configuration :

Router 1 :



```
Router(config)#interface se2/0
Router(config-if)#interface loopback 0

Router(config-if)#
%LINK-5-CHANGED: Interface Loopback0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, changed state to up

Router(config-if)#ip address 172.16.1.252 255.255.0.0
Router(config-if)#no shut
Router(config-if)#exit
Router(config)#router ospf 1
Router(config-router)#area 1 virtual-link 2.2.2.2
Router(config-router)#exit
Router(config)#
00:24:20: %OSPF-5-ADJCHG: Process 1, Nbr 2.2.2.2 on OSPF_VL0 from LOADING to FULL, Loading Done
exit
Router#
%SYS-5-CONFIG_I: Configured from console by console
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
     20.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C       20.0.0.0/8 is directly connected, Serial2/0
C       20.0.0.2/32 is directly connected, Serial2/0
O    30.0.0.0/8 [110/128] via 20.0.0.2, 00:00:54, Serial2/0
O IA 40.0.0.0/8 [110/129] via 20.0.0.2, 00:00:54, Serial2/0
C    172.16.0.0/16 is directly connected, Loopback0
Router#
```

Router 2 :


Router2
Physical
Config
CLI
Router>en
Router#config t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface se2/0
Router(config-if)#ip add 20.0.0.2
% Incomplete command.
Router(config-if)#ip add 20.0.0.2 255.0.0.0
Router(config-if)#no shut
Router(config-if)#
%LINK-5-CHANGED: Interface Serial2/0, changed state to up
exit
Router(config)#ip add 20.0.0.2
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, chang
Router(config)#interface se3/0
Router(config-if)#ip add 30.0.0.1 255.0.0.0
Router(config-if)#no shut
%LINK-5-CHANGED: Interface Serial3/0, changed state to down
Router(config-if)#no shut
Router(config-if)#
%LINK-5-CHANGED: Interface Serial3/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial3/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to down
interface se2/0
Router(config-if)#encapsulation ppp
Router(config-if)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to up
no shut
Router(config-if)#exit
Router(config)#interface se3/0
Router(config-if)#encapsulation ppp
Router(config-if)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial3/0, changed state to down
clock rate 640000
Unknown clock rate
Router(config-if)#no shut
Router(config-if)#exit
Router(config)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial3/0, changed state to up
%LINK-5-CHANGED: Interface Serial2/0, changed state to down
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to down
%LINK-5-CHANGED: Interface Serial2/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to up
Router(config)#router ospf
% Incomplete command.
Router(config)#router ospf 1
Router(config-router)#router-id 2.2.2.2
Router(config-router)#network 20.0.0.0 0.255.255.255 area 1
Router(config-router)#network 20.0.0.0 0.255.255.255 area 1
00:18:05: %OSPF-5-ADJCHG: Process 1, Nbr 1.1.1.1 on Serial2/
Router(config-router)#network 30.0.0.0 0.255.255.255 area 0
Router(config-router)#exit
Router(config)#
00:19:20: %OSPF-5-ADJCHG: Process 1, Nbr 40.0.0.1 on Serial3/0 from LOADING to FULL, Loading Done


Router2
Physical
Config
CLI
IOS Command Line Interface
Router(config)#interface se3/0
Router(config-if)#interface loopback 0
Router(config-if)#
%LINK-5-CHANGED: Interface Loopback0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, changed state to up
ip add 172.16.1.253 255.255.0.0
Router(config-if)#no shut
Router(config-if)#
%LINK-5-CHANGED: Interface Serial3/0, changed state to down
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial3/0, changed state to down
00:23:22: %OSPF-5-ADJCHG: Process 1, Nbr 40.0.0.1 on Serial3/0 from FULL to DOWN, Neighbor Down: Interface down or detached
%LINK-5-CHANGED: Interface Serial3/0, changed state to up
Router(config-if)#exi
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial3/0, changed state to up
t
00:23:33: %OSPF-4-ERRRCV: Received invalid packet: mismatch area ID, from backbone area must be virtual-link but not found from 20.0.0.2, Serial2/0
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console
00:23:41: %OSPF-5-ADJCHG: Process 1, Nbr 40.0.0.1 on Serial3/0 from LOADING to FULL, Loading Done
Router#
00:23:43: %OSPF-4-ERRRCV: Received invalid packet: mismatch area ID, from backbone area must be virtual-link but not found from 20.0.0.2, Serial2/0
config t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#router ospf 1
Router(config-router)#
00:23:53: %OSPF-4-ERRRCV: Received invalid packet: mismatch area ID, from backbone area must be virtual-link but not found from 20.0.0.2, Serial2/0
Router(config-router)#area 1 virtual-link
00:24:03: %OSPF-4-ERRRCV: Received invalid packet: mismatch area ID, from backbone area must be virtual-link but not found from 20.0.0.2, Serial2/0
1.1.1.1
Router(config-router)#
Router#
%SYS-5-CONFIG_I: Configured from console by console
Router#c
00:24:18: %OSPF-5-ADJCHG: Process 1, Nbr 1.1.1.1 on OSPF_VL0 from LOADING to FULL, Loading Done
% Ambiguous command: "c"
Router#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
20.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C 20.0.0.0/8 is directly connected, Serial2/0
C 20.0.0.1/32 is directly connected, Serial2/0
30.0.0.0/8 is variably subnetted, 2 subnets, 2 masks

```
Router#config t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#router ospf 1
Router(config-router)#area 1 virtual-link 1.1.1.1
Router(config-router)#exit
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console
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

O IA 10.0.0.0/8 [110/65] via 20.0.0.1, 00:00:34, Serial2/0
     20.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C       20.0.0.0/8 is directly connected, Serial2/0
C       20.0.0.1/32 is directly connected, Serial2/0
     30.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C       30.0.0.0/8 is directly connected, Serial3/0
C       30.0.0.2/32 is directly connected, Serial3/0
O IA 40.0.0.0/8 [110/65] via 30.0.0.2, 00:01:24, Serial3/0
C    172.16.0.0/16 is directly connected, Loopback0
Router#
```

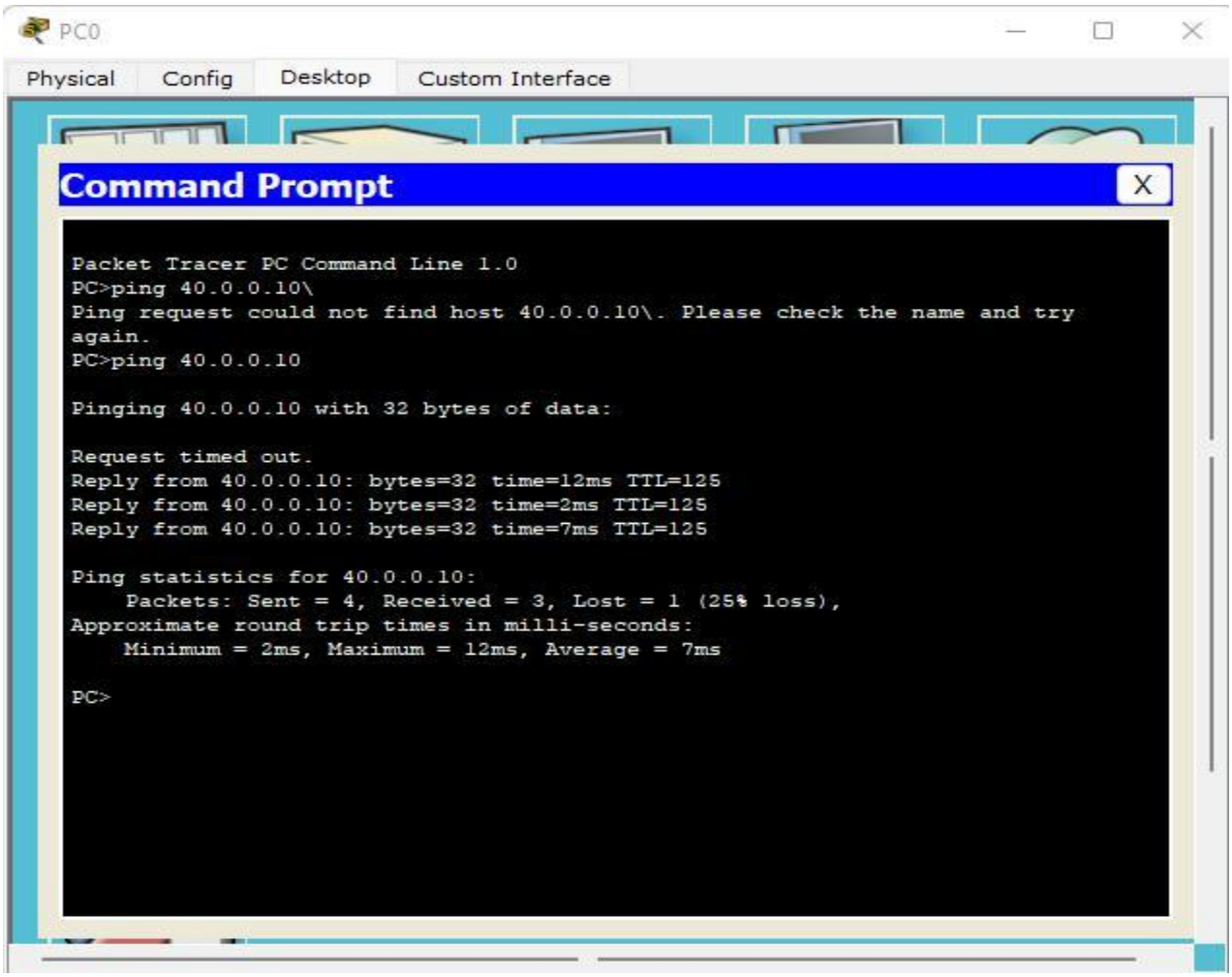
Router 3 :


Router3
Physical
Config
CLI
IOS Commar
Router>en
Router#config t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface fa0/0
Router(config-if)#ip add 40.0.0.1 255.0.0.0
Router(config-if)#no shut
Router(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up
exit
Router(config)#interface se2/0
Router(config-if)#ip add 30.0.0.2 255.0.0.0
Router(config-if)#no shut
Router(config-if)#
%LINK-5-CHANGED: Interface Serial2/0, changed state to up
exit
Router(config)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to down
Router(config)#interface se2/0
Router(config-if)#encapsulation ppp
Router(config-if)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to up
no shut
Router(config-if)#exit
Router(config)#router ospf 1
Router(config-router)#network 30.0.0.0 0.255.255.255 area 0
Router(config-router)#
00:18:56: %OSPF-5-ADJCHG: Process 1, Nbr 2.2.2.2 on Serial2/0 from LOADING to FULL, Loading Done
Router(config-router)#network 40.0.0.0 0.255.255.255 area 2
Router(config-router)#exit
Router(config)#interface se2/0
Router(config-if)#interface loopback 0
Router(config-if)#
%LINK-5-CHANGED: Interface Loopback0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, changed state to up
ip add 172.16.1.254 255.255.0.0
Router(config-if)#no shut
Router(config-if)#
%LINK-5-CHANGED: Interface Serial2/0, changed state to down
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to down
00:22:58: %OSPF-5-ADJCHG: Process 1, Nbr 2.2.2.2 on Serial2/0 from FULL to DOWN, Neighbor Down: Interface down or detached
%LINK-5-CHANGED: Interface Serial2/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to up
00:23:18: %OSPF-5-ADJCHG: Process 1, Nbr 2.2.2.2 on Serial2/0 from LOADING to FULL, Loading Done
Router(config-if)#exit
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console

```
Router#
%SYS-5-CONFIG_I: Configured from console by console
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

O IA 10.0.0.0/8 [110/129] via 30.0.0.1, 00:05:53, Serial2/0
O IA 20.0.0.0/8 [110/128] via 30.0.0.1, 00:06:30, Serial2/0
     30.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C       30.0.0.0/8 is directly connected, Serial2/0
C       30.0.0.1/32 is directly connected, Serial2/0
C    40.0.0.0/8 is directly connected, FastEthernet0/0
C    172.16.0.0/16 is directly connected, Loopback0
Router#
```

Ping Result :

P0

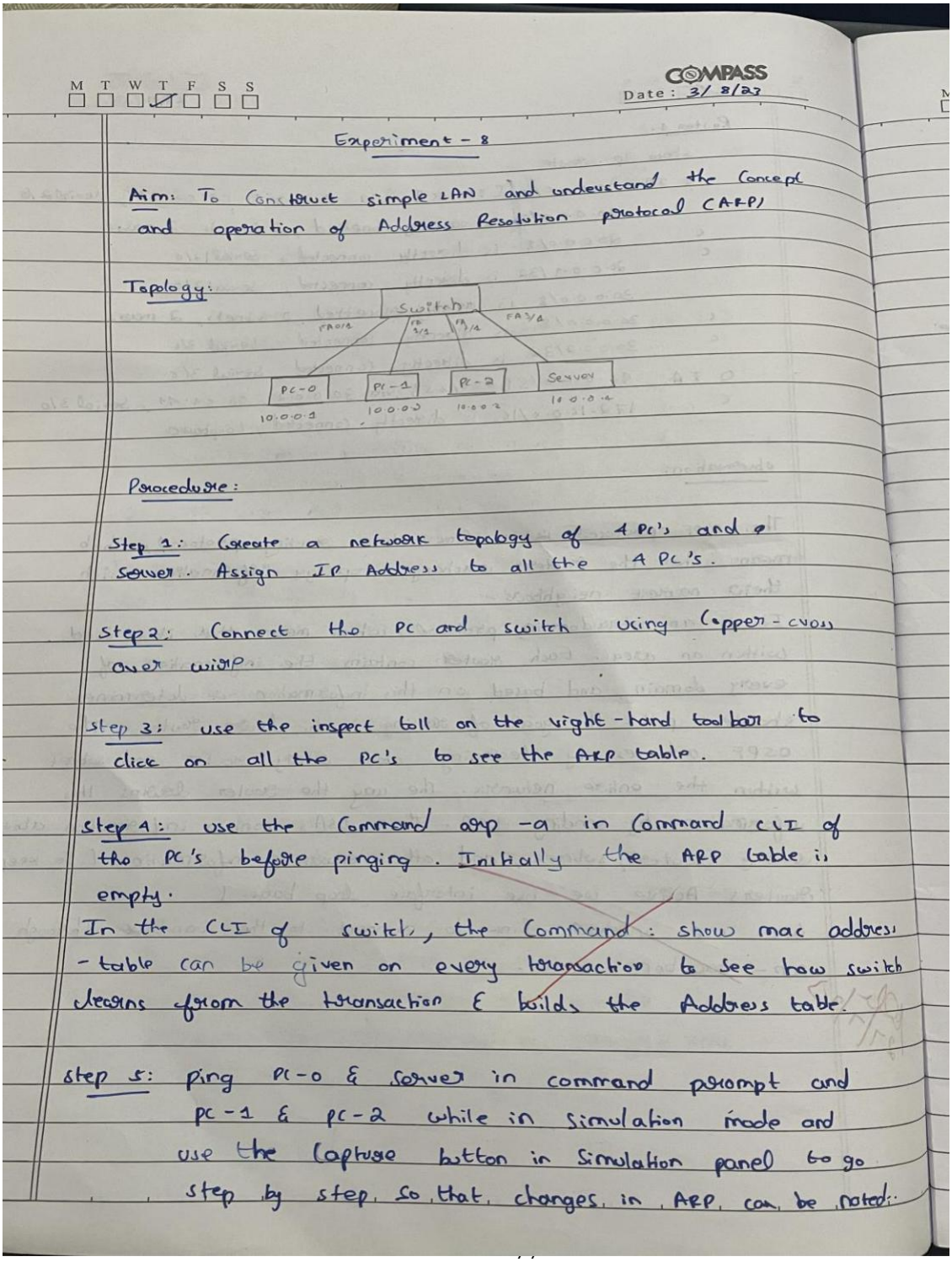


LAB PROGRAM – 8

Aim :

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

Procedure :



M T W T F S S

COMPASS

Date : 3/ 8/23

Experiment - 8

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

Topology:

Procedure:

Step 1: Create a network topology of 4 PC's and a server. Assign IP Address to all the 4 PC's.

Step 2: Connect the PC and switch using Copper - cross over wire.

Step 3: use the inspect tool on the right-hand tool bar to click on all the PC's to see the ARP table.

Step 4: use the Command arp -a in Command CLI of the PC's before pinging. Initially the ARP table is empty.

In the CLI of switch, the Command: show mac address - table can be given on every transaction to see how switch learns from the transaction & builds the Address table.

Step 5: ping PC-0 & Server in command prompt and PC-1 & PC-2 while in Simulation mode and use the Capture button in Simulation panel to go step by step so that changes in ARP can be noted.

M T W T F S S

COMPASS

Date: 3/8/23

observe the switch as well the nodes update the ARP table as and when a new communication starts.

Result:

→ Before Pinging : PC0 Command prompt

PC> arp -a

No ARP entries found

→ After Pinging : PC0 Command prompt

PC> 10.0.0.4

pinging 10.0.0.4 with 32 bytes of data

Reply from 10.0.0.4: bytes=32 time=8ms TTL=128

Reply from 10.0.0.4: bytes=32 time=4ms TTL=128

Reply from 10.0.0.4: bytes=32 time=4ms TTL=128

Reply from 10.0.0.4: bytes=32 time=4ms 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 = 4ms, Maximum = 8ms, Average = 5ms

PC > arp - a

Internet Address	Physical Address	Type
10.0.0.4	0001.42e8.32c1	dyramic

→ Switch 0 CLI

Switch > show mac address-table

MAC Address Table

Vlan	Mac Address	Type	Port
1	0001.42e8.32c1	DYNAMIC	FA1/1
1	0001.96b5.9a0	DYNAMIC	FA0/1
1	0007.ecc9.8080	DYNAMIC	FA1/1
1	000d.bd98.b636	DYNAMIC	FA2/1

M T W T F S S

COMPASS

Date: 3/1/27

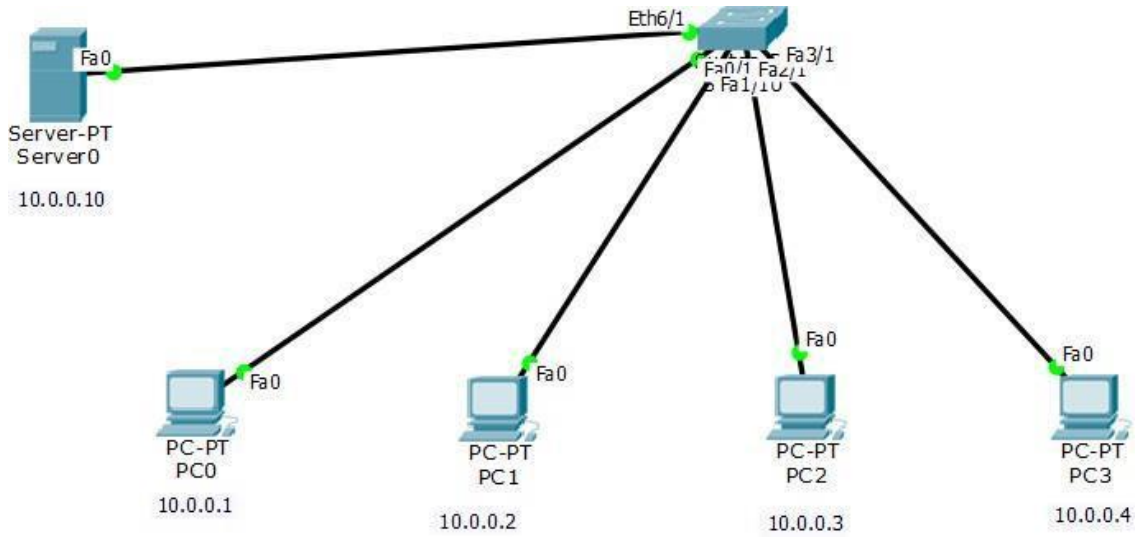
Observation:

ARP (Address Resolution protocol). It is a network layer protocol. It is responsible for finding the hardware address (MAC Address) of a host from a known IP Address.

At the network layer, when the source wants to communicate with the destination, it needs to know the mac Address of the destination. For this, it will check ARP table for the MAC Address of the destination. If the MAC Address of the destination is present, communication will take place.

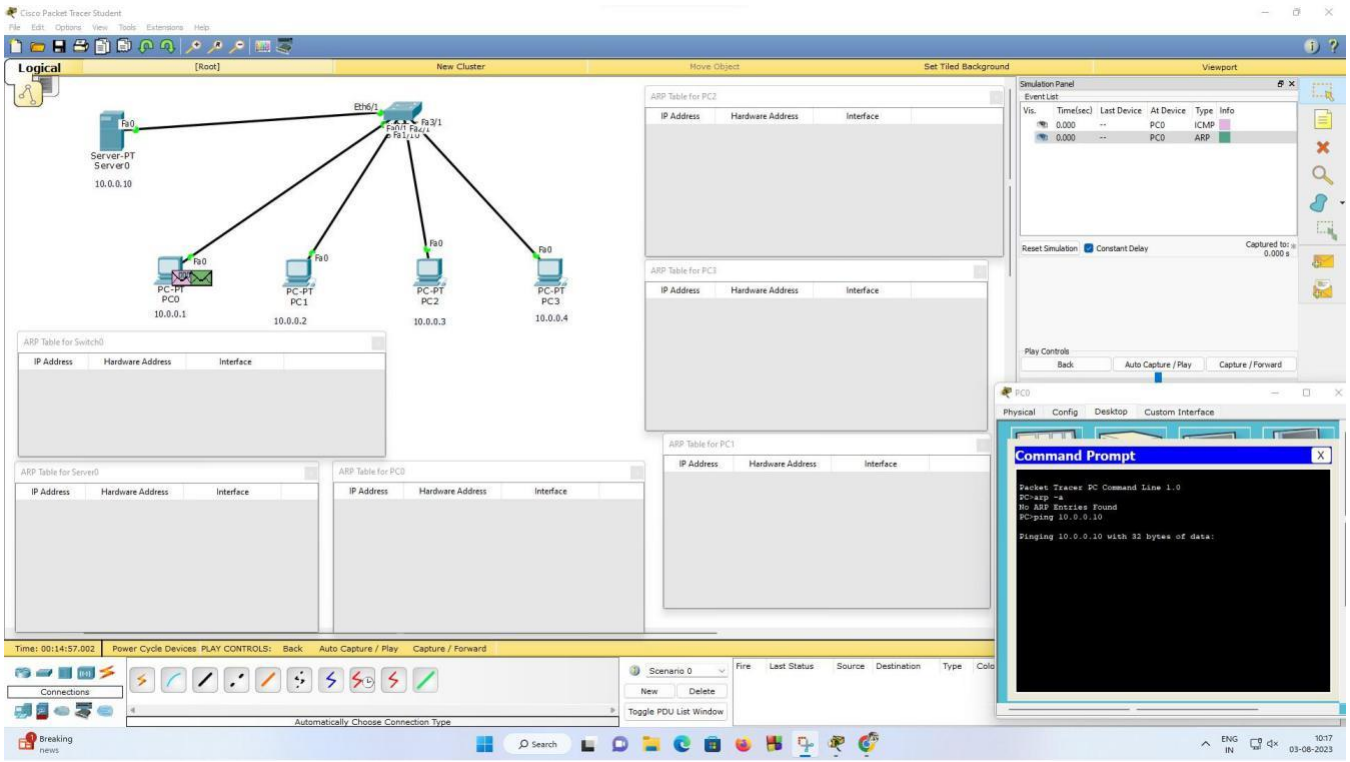
If the MAC Address is not a part of the ARP table the source sends a ARP request message (broadcast message.). All devices in the network will compare dest IP Address with it's own ip Address and if it matches, the device sends a ARP reply message (unicast). which contains the MAC Address. Now source and destination will communicate.

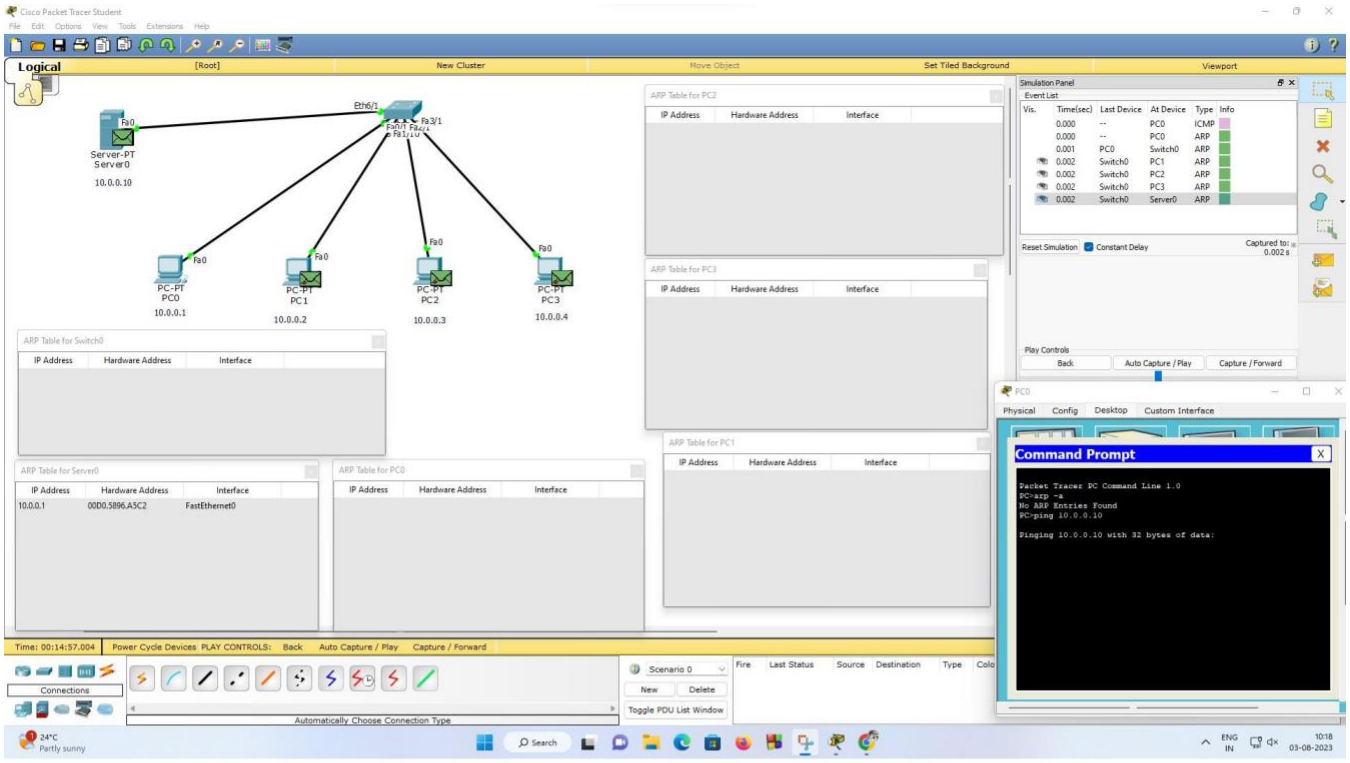
Topology:

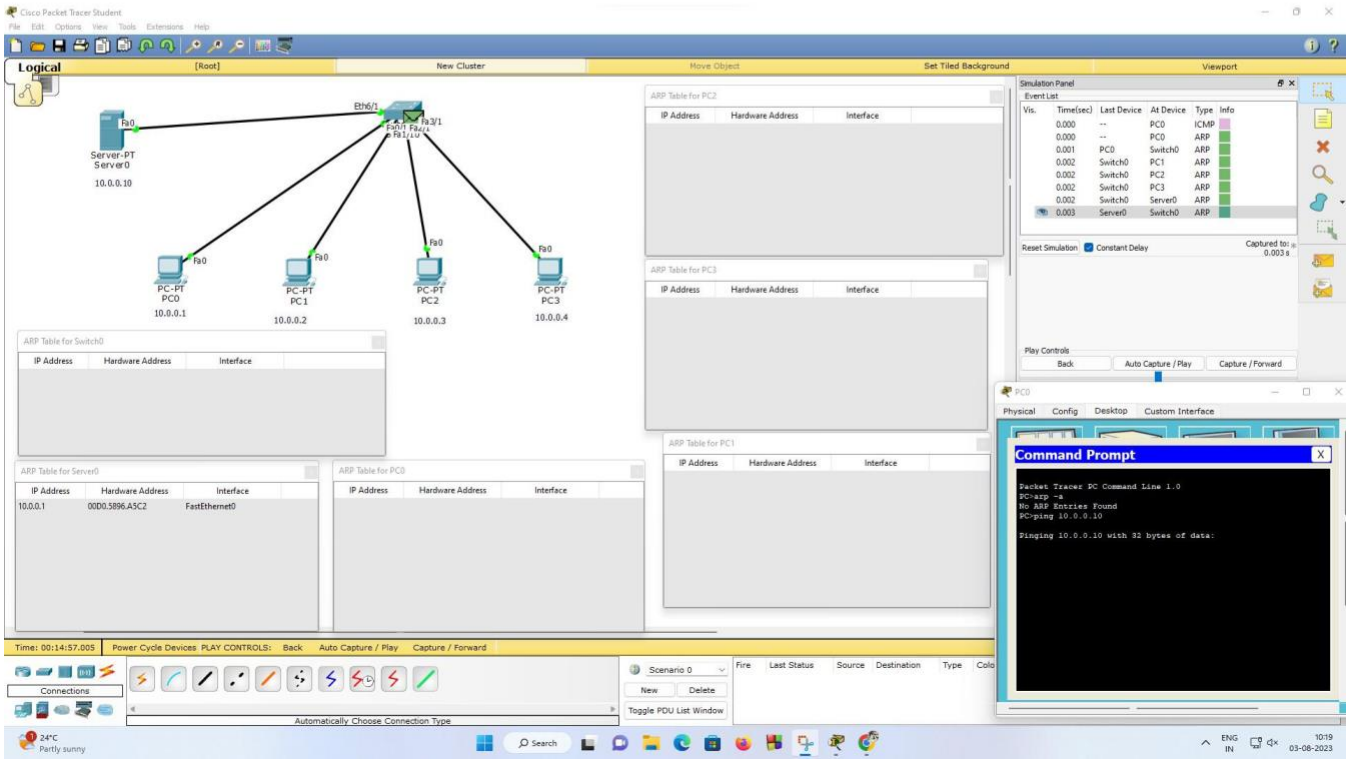


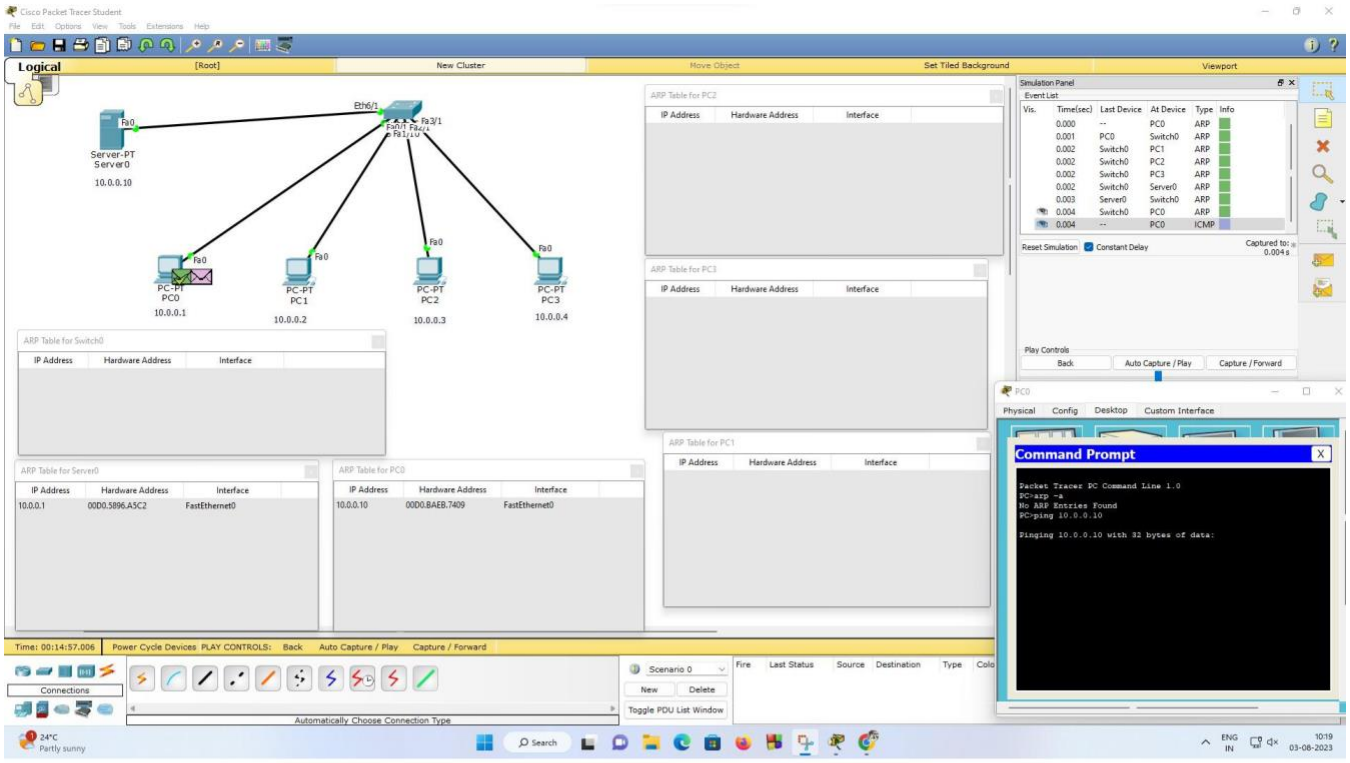
Ping Results(ARP Tables)

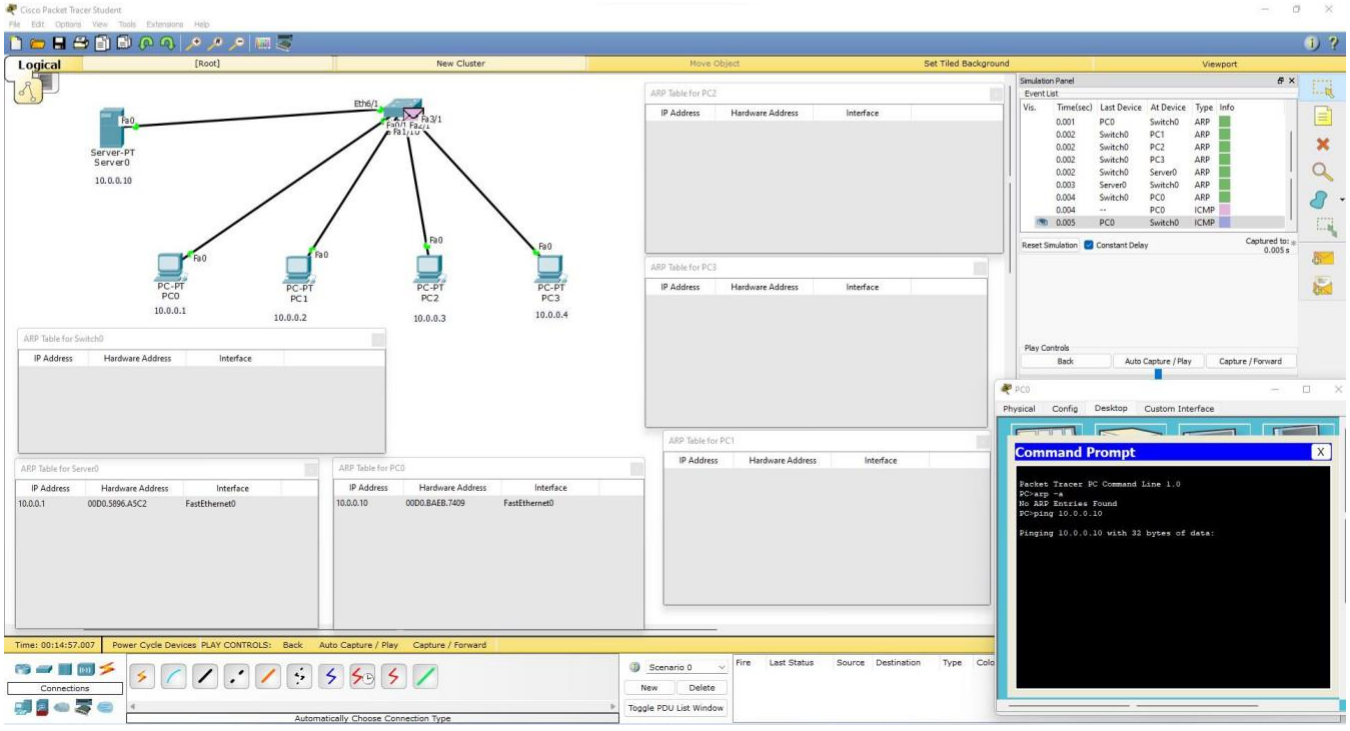
PC0 to Server0 :

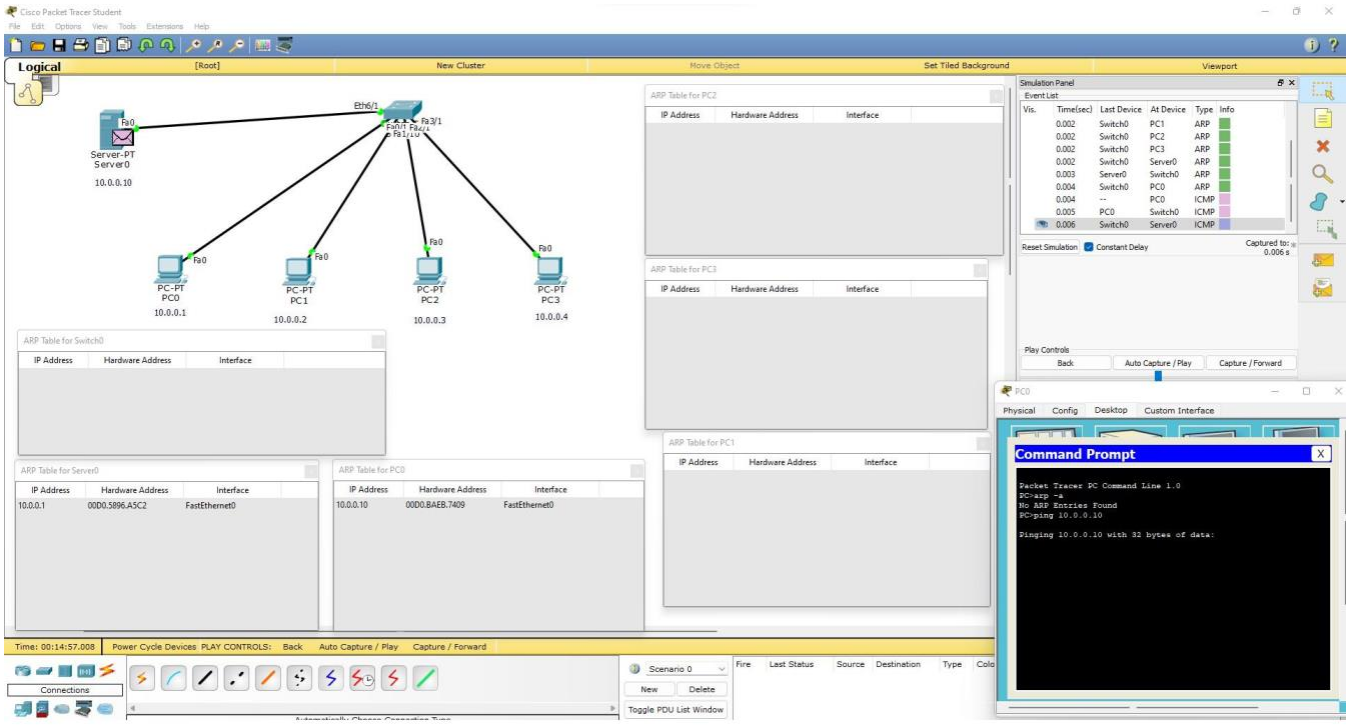


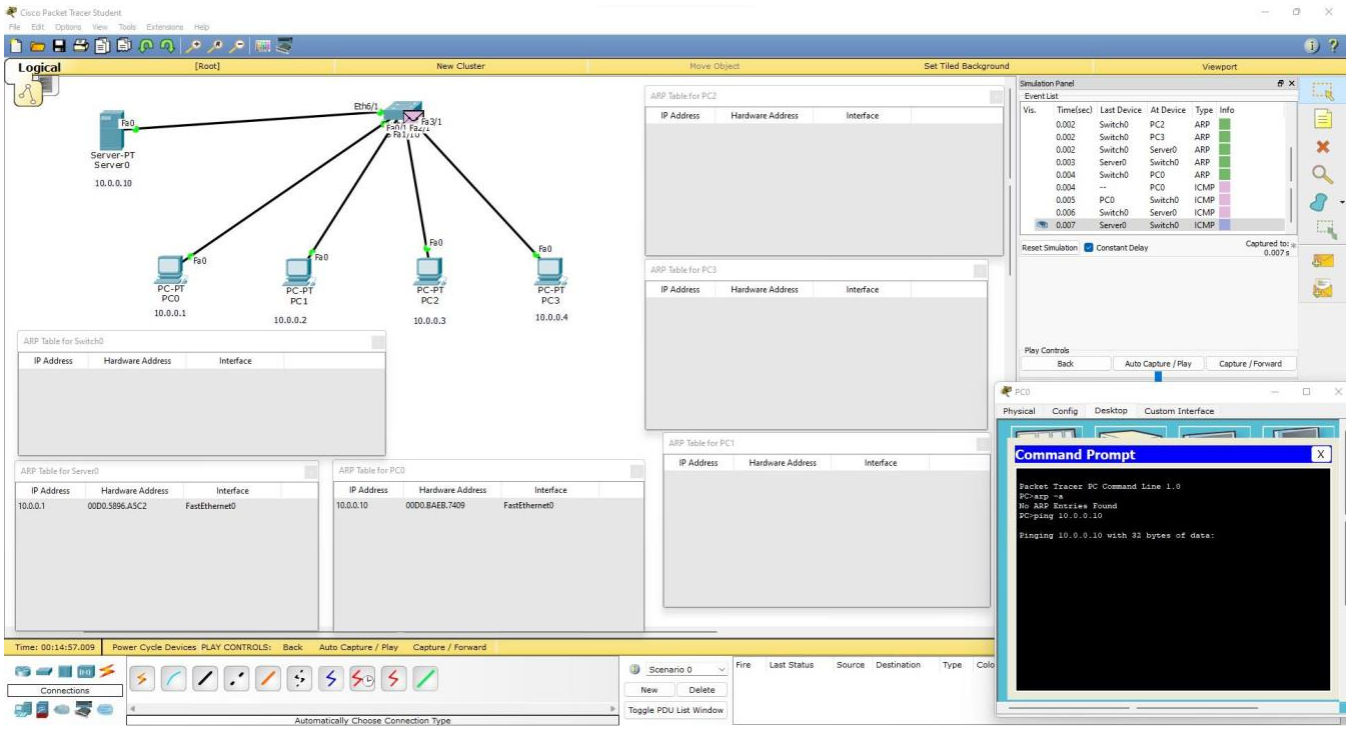

Cisco Packet Tracer Student
Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
Server-PT Server0
10.0.0.10
PC-PT PC0
10.0.0.1
PC-PT PC1
10.0.0.2
PC-PT PC2
10.0.0.3
PC-PT PC3
10.0.0.4
Simulation Panel
Event List
Command Prompt
Packet Tracer PC Command Line 1.0
PC>arp -a
No ARP Entries Found
PC>ping 10.0.0.10
Pinging 10.0.0.10 with 32 bytes of data:
Time: 00:14:57.004
Power Cycle Devices PLAY CONTROLS: Back Auto Capture / Play Capture / Forward

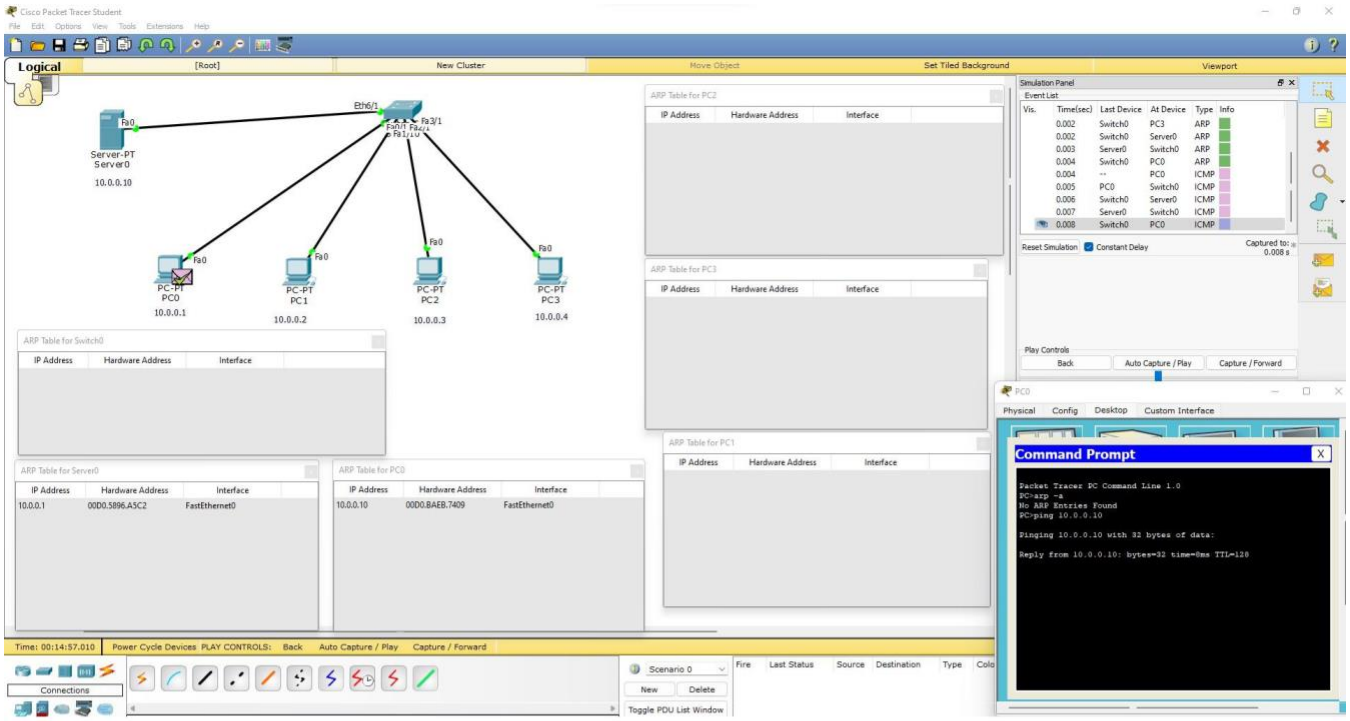

Cisco Packet Tracer Student
Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
Server-PT Server0
10.0.0.10
PC-PT PC0
10.0.0.1
PC-PT PC1
10.0.0.2
PC-PT PC2
10.0.0.3
PC-PT PC3
10.0.0.4
Simulation Panel
Event List
Command Prompt
Packet Tracer PC Command Line 1.0
PC>arp -a
No ARP Entries Found
PC>ping 10.0.0.10
Pinging 10.0.0.10 with 32 bytes of data:
Time: 00:14:57.005
Power Cycle Devices PLAY CONTROLS: Back Auto Capture / Play Capture / Forward

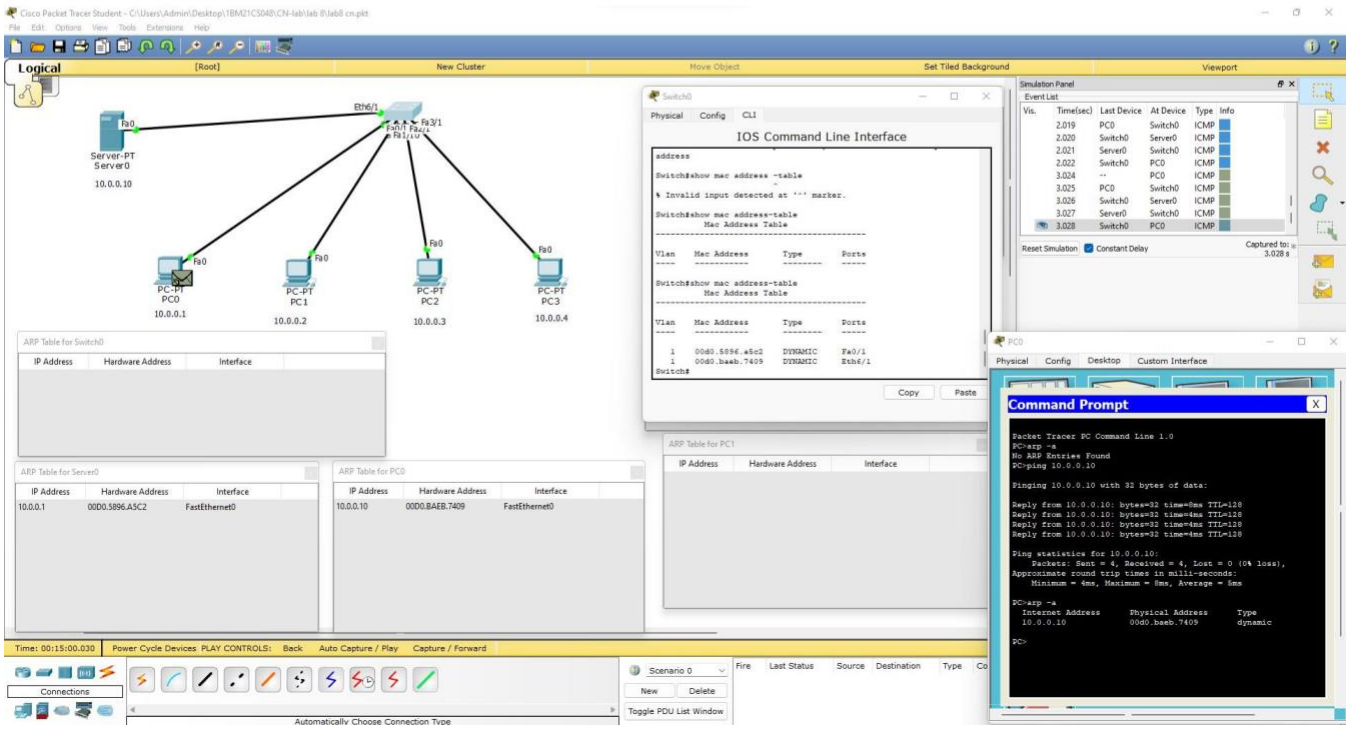

Logical
Server-PT Server0
10.0.0.10
PC-PT PC0
10.0.0.1
PC-PT PC1
10.0.0.2
PC-PT PC2
10.0.0.3
PC-PT PC3
10.0.0.4
Command Prompt
Packet Tracer PC Command Line 1.0
PC>arp -a
No ARP Entries Found
PC>ping 10.0.0.10
Pinging 10.0.0.10 with 32 bytes of data:


Logical
Server-PT Server0
10.0.0.10
PC-PT PC0
10.0.0.1
PC-PT PC1
10.0.0.2
PC-PT PC2
10.0.0.3
PC-PT PC3
10.0.0.4
Command Prompt
Packet Tracer PC Command Line 1.0
PC>arp -a
No ARP Entries Found
PC>ping 10.0.0.10
Pinging 10.0.0.10 with 32 bytes of data:

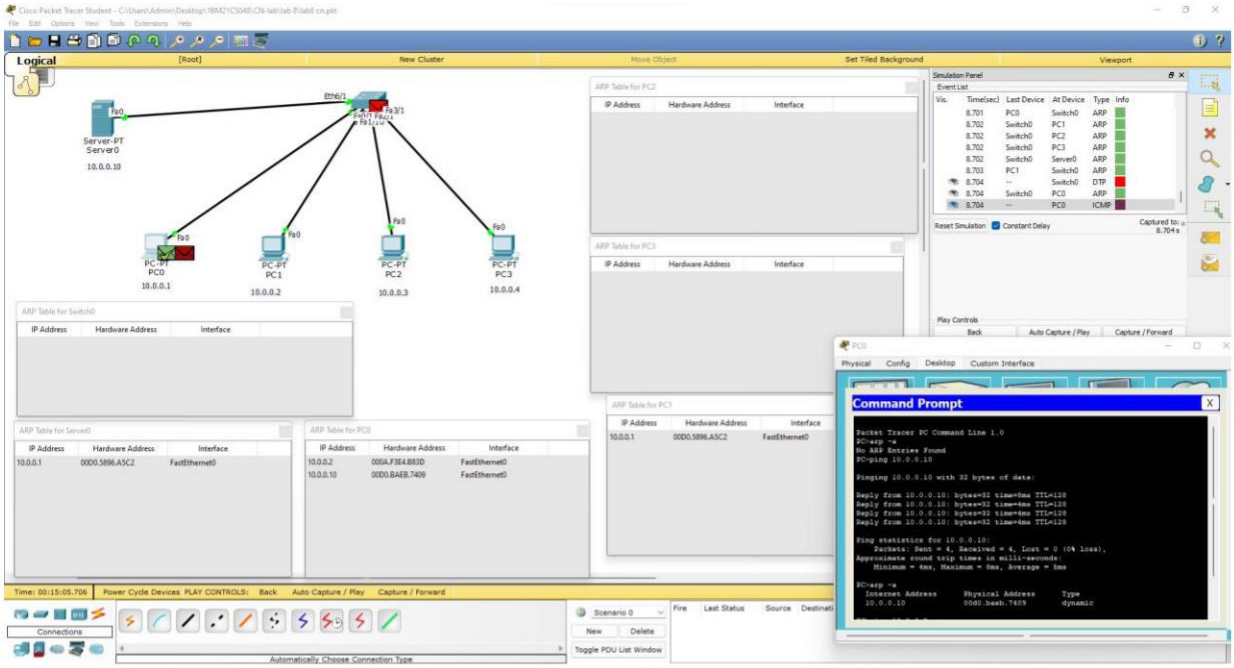

Logical
Server-PT Server0
10.0.0.10
PC-PT PC0
10.0.0.1
PC-PT PC1
10.0.0.2
PC-PT PC2
10.0.0.3
PC-PT PC3
10.0.0.4
Command Prompt
Packet Tracer PC Command Line 1.0
PC>arp -a
No ARP Entries Found
PC>ping 10.0.0.10
Pinging 10.0.0.10 with 32 bytes of data:

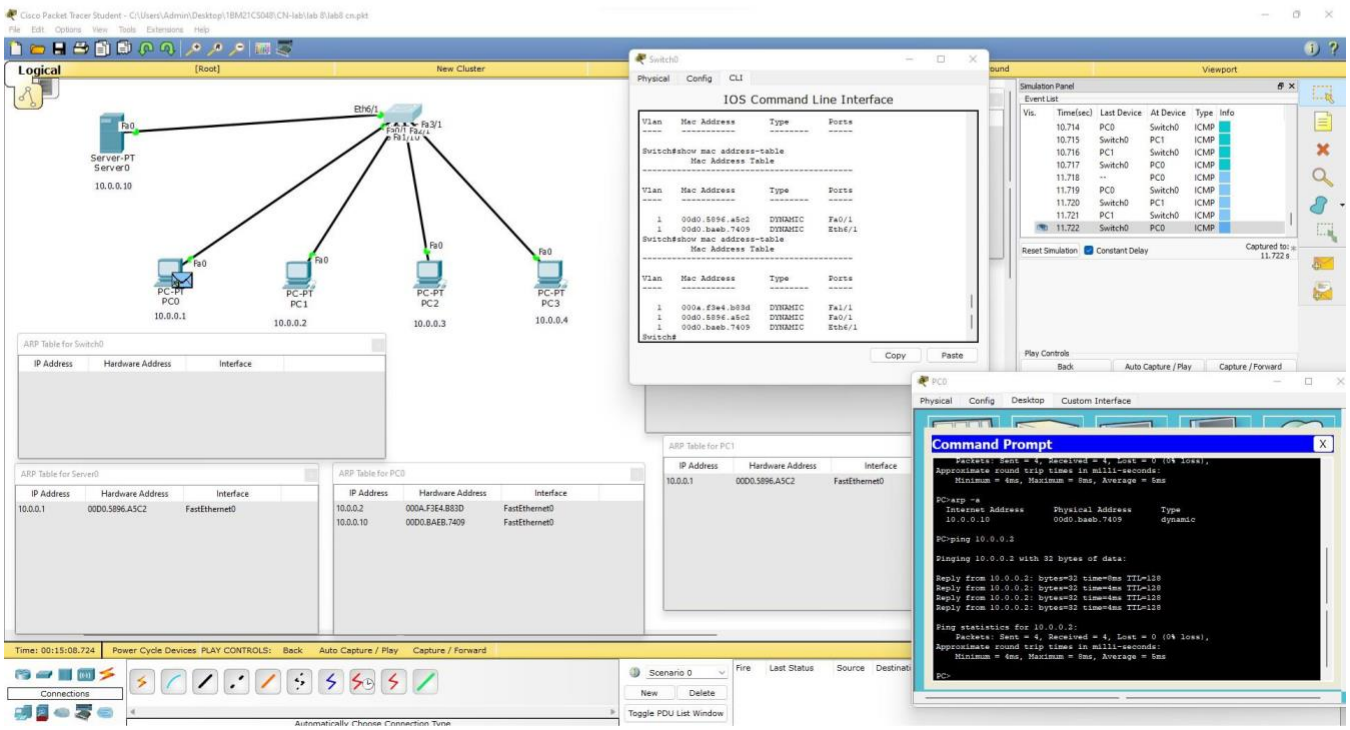

Logical
Server-PT Server0
10.0.0.10
PC-PT PC0
10.0.0.1
PC-PT PC1
10.0.0.2
PC-PT PC2
10.0.0.3
PC-PT PC3
10.0.0.4
Command Prompt
Packet Tracer PC Command Line 1.0
PC>arp -a
No ARP Entries Found
PC>ping 10.0.0.10
Pinging 10.0.0.10 with 32 bytes of data:


Logical
Server-PT Server0
10.0.0.10
PC-PT PC0
10.0.0.1
PC-PT PC1
10.0.0.2
PC-PT PC2
10.0.0.3
PC-PT PC3
10.0.0.4
Command Prompt
Packet Tracer PC Command Line 1.0
PC>arp -a
No ARP Entries Found
PC>ping 10.0.0.10
Pinging 10.0.0.10 with 32 bytes of data:
Reply from 10.0.0.10: bytes=32 time=8ms TTL=128

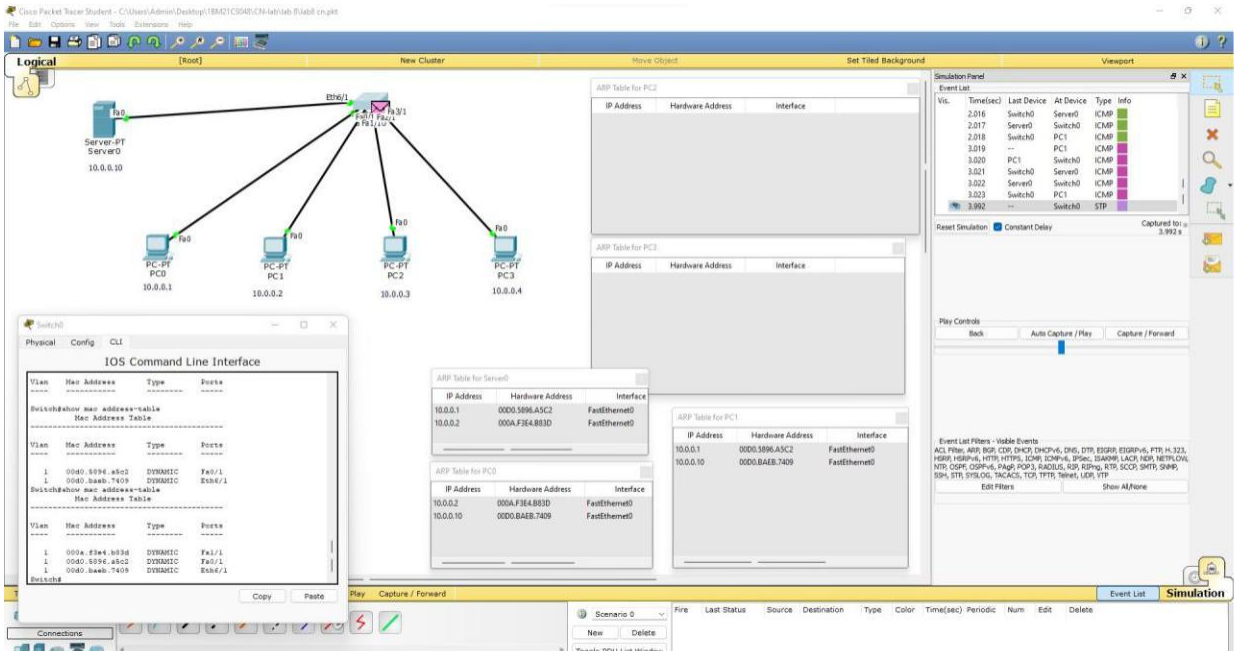

Logical
Server-PT Server0
10.0.0.10
PC-PT PC0
10.0.0.1
PC-PT PC1
10.0.0.2
PC-PT PC2
10.0.0.3
PC-PT PC3
10.0.0.4
IOS Command Line Interface
Command Prompt
Packet Tracer PC Command Line 1.0
PC>arp -a
No ARP Entries Found
PC>ping 10.0.0.10
Pinging 10.0.0.10 with 32 bytes of data:
Reply from 10.0.0.10: bytes=32 time=8ms TTL=128
Reply from 10.0.0.10: bytes=32 time=4ms TTL=128
Reply from 10.0.0.10: bytes=32 time=4ms TTL=128
Reply from 10.0.0.10: bytes=32 time=4ms TTL=128
Ping statistics for 10.0.0.10:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 4ms, Maximum = 8ms, Average = 5ms
PC>arp -a
Internet Address Physical Address Type
10.0.0.10 00d0.baeb.7409 dynamic
PC>

Ping from PC0 to PC1:





Final ARP Tables after pinging:

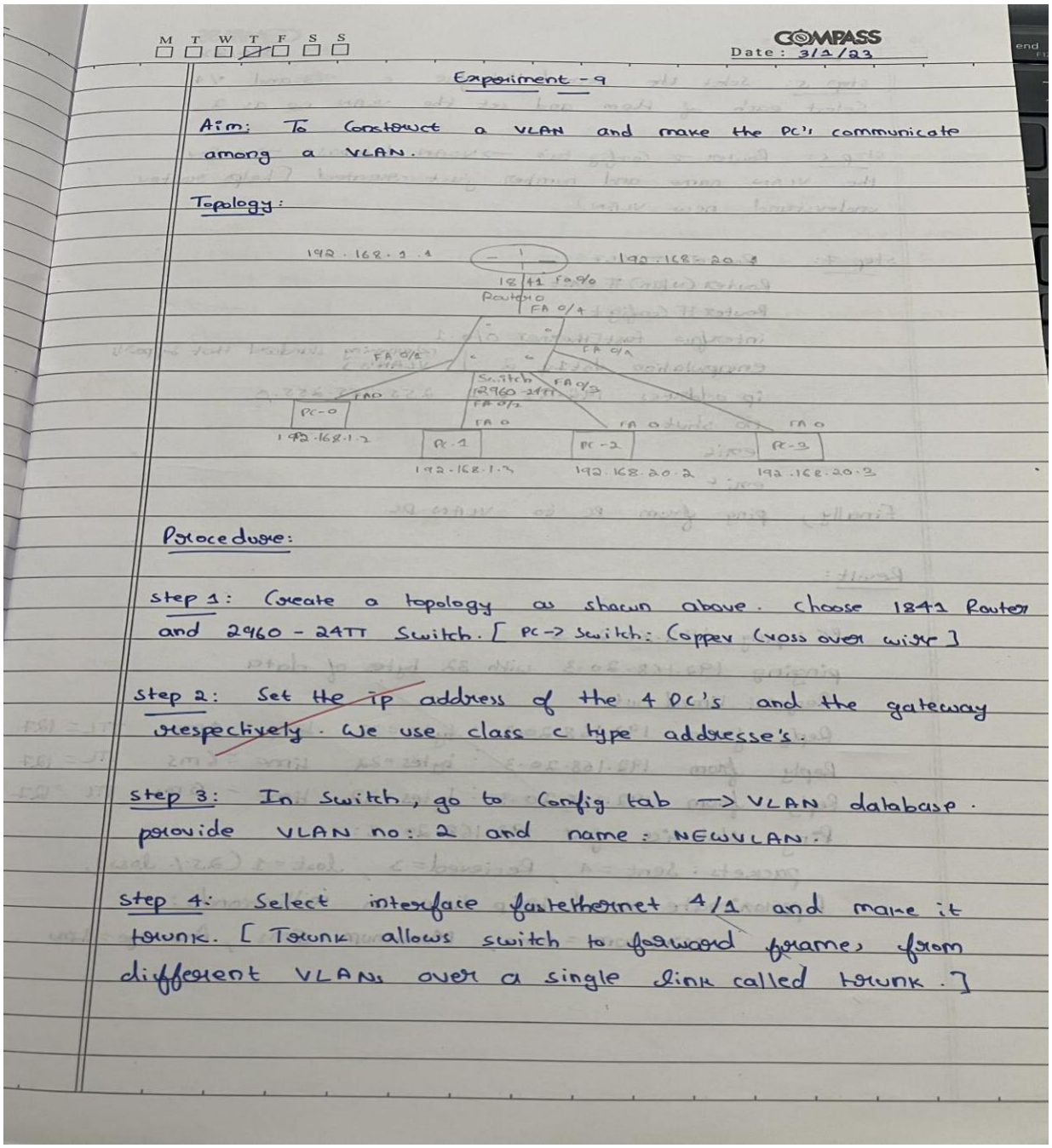


LAB PROGRAM – 9

Aim :

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

Procedure :



COMPASS

Date: 3/1/23

M T W T F S S

step 5: Select the second interface i.e 0/3 and 0/4. Select each of them and set the VLAN no as 2.

step 6: Router → Config tab → VLAN database → enter the VLAN name and number just created. [helps router understand new VLAN]

step 7:
```
Router →CLI
Router(vlan)# exit
Router# Config t
interface fastEthernet 0/0.1
encapsulation dot1q 2 [ networking standard that supports VLAN's ]
ip address 192.168.2.1 255.255.255.0
no shut
exit
exit
```

Finally, ping from PC to VLAN PC

Result:

```
PC > ping 192.168.20.3
pinging 192.168.20.3 with 32 bytes of data
Request timed out
Reply from 192.168.20.3 : bytes = 32 time = 0ms TTL = 127
Reply from 192.168.20.3 : bytes = 32 time = 6ms TTL = 127
Reply from 192.168.20.3 : bytes = 32 time = 0ms TTL = 127
Ping statistics for 192.168.20.3:
  packets: Sent = 4, Recieved = 3, lost = 1 (25% loss),
Approximate round trip times in milliseconds:
  Minimum = 0ms, Maximum = 6ms, Average = 1ms
```

M T W T F S S

COMPASS

Date : 3/1/23

observation:

VLAN - (virtual local area network) is a logical subnetwork of devices in a broadcast domain that is partioned by network switches so as to Act as it's own LAN.

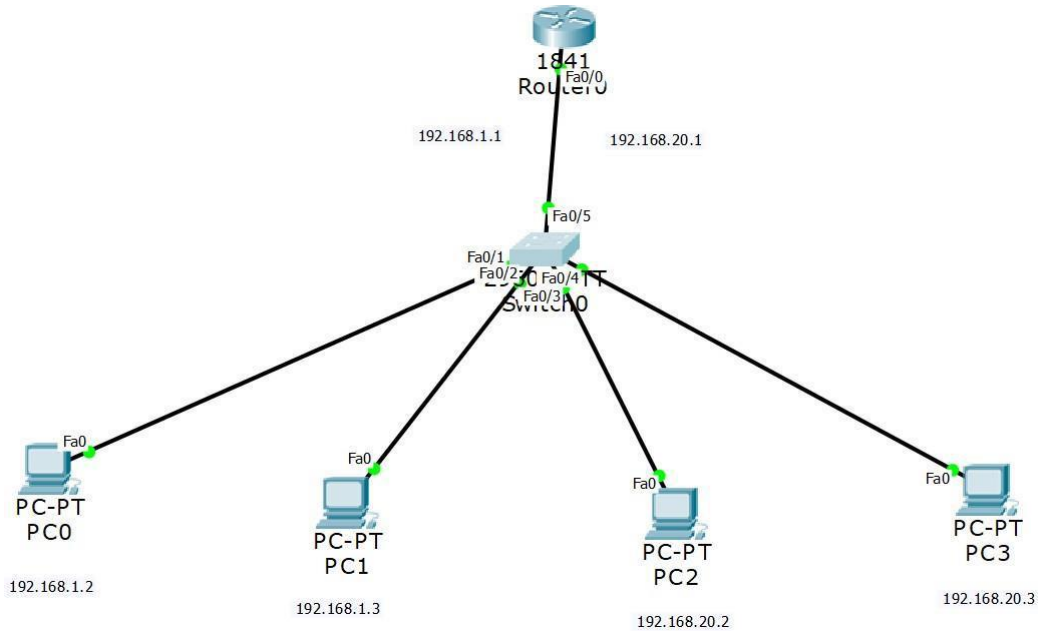
VLAN operate either at layer 2 (DLL) or layer 3 NL.

VLAN creates multiple standalone networks out of the same networking backbone. This is more secure, and it reduces the no of broadcasts individual devices receive.

VLAN's don't use IP addresses they deal with subnets or class (type Addresses.

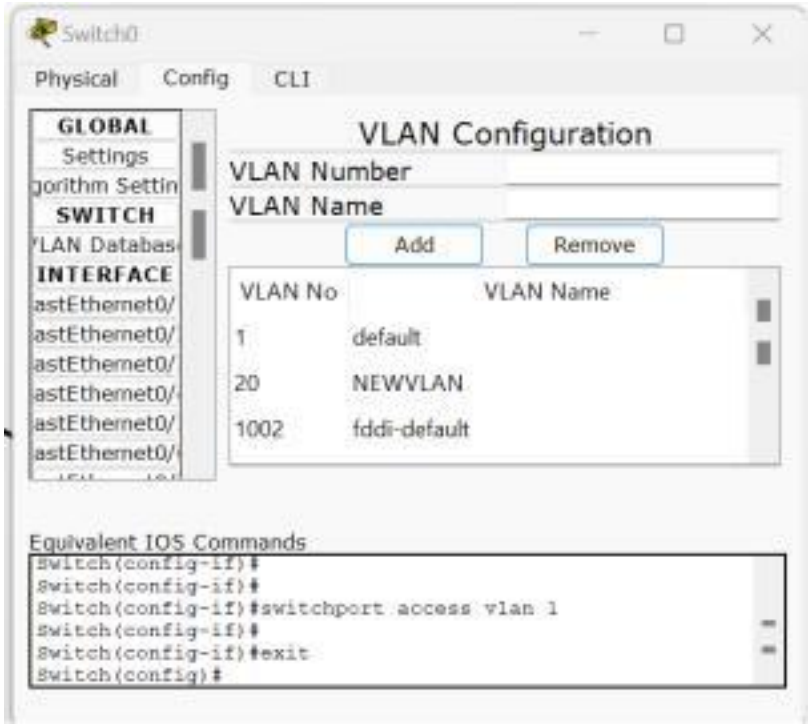
31/8/23

Topology:

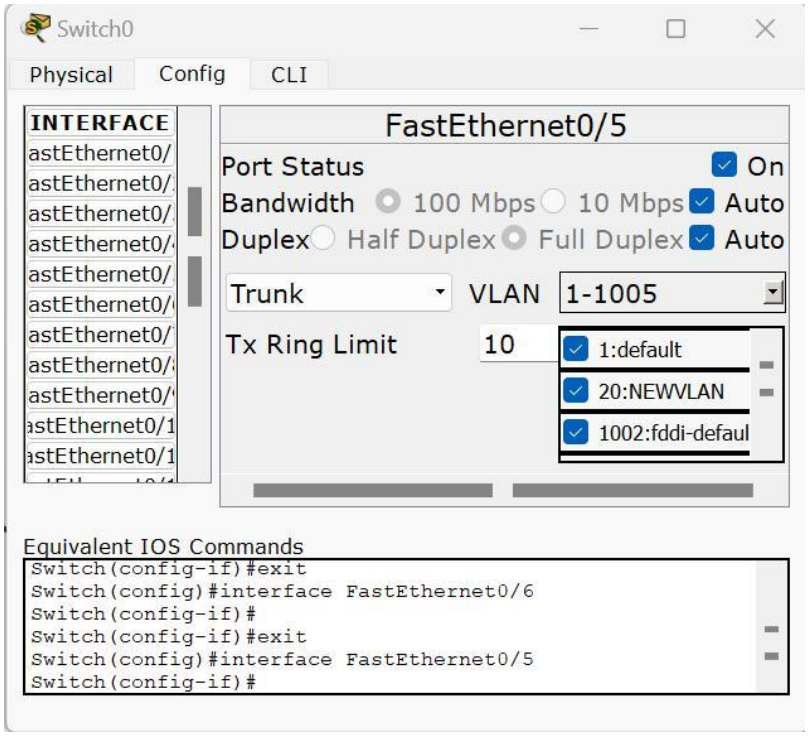


Configurations:

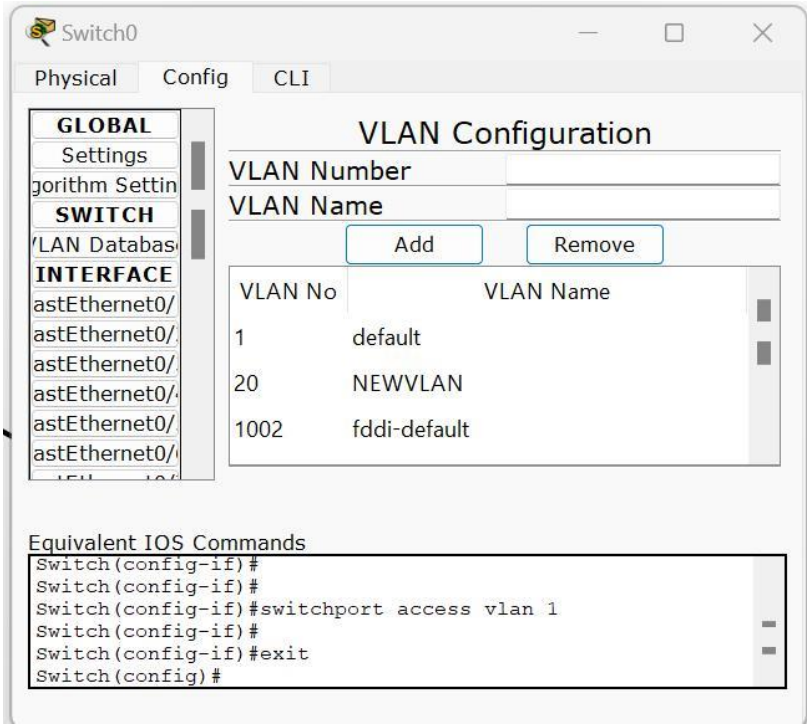
Switch VLAN Database:



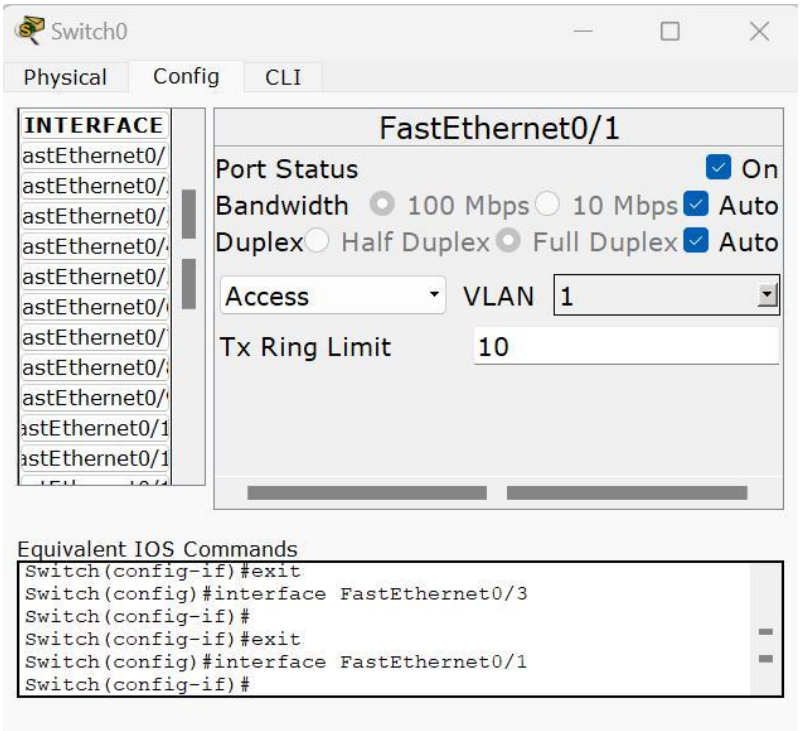
Switch FastEthernet0/5



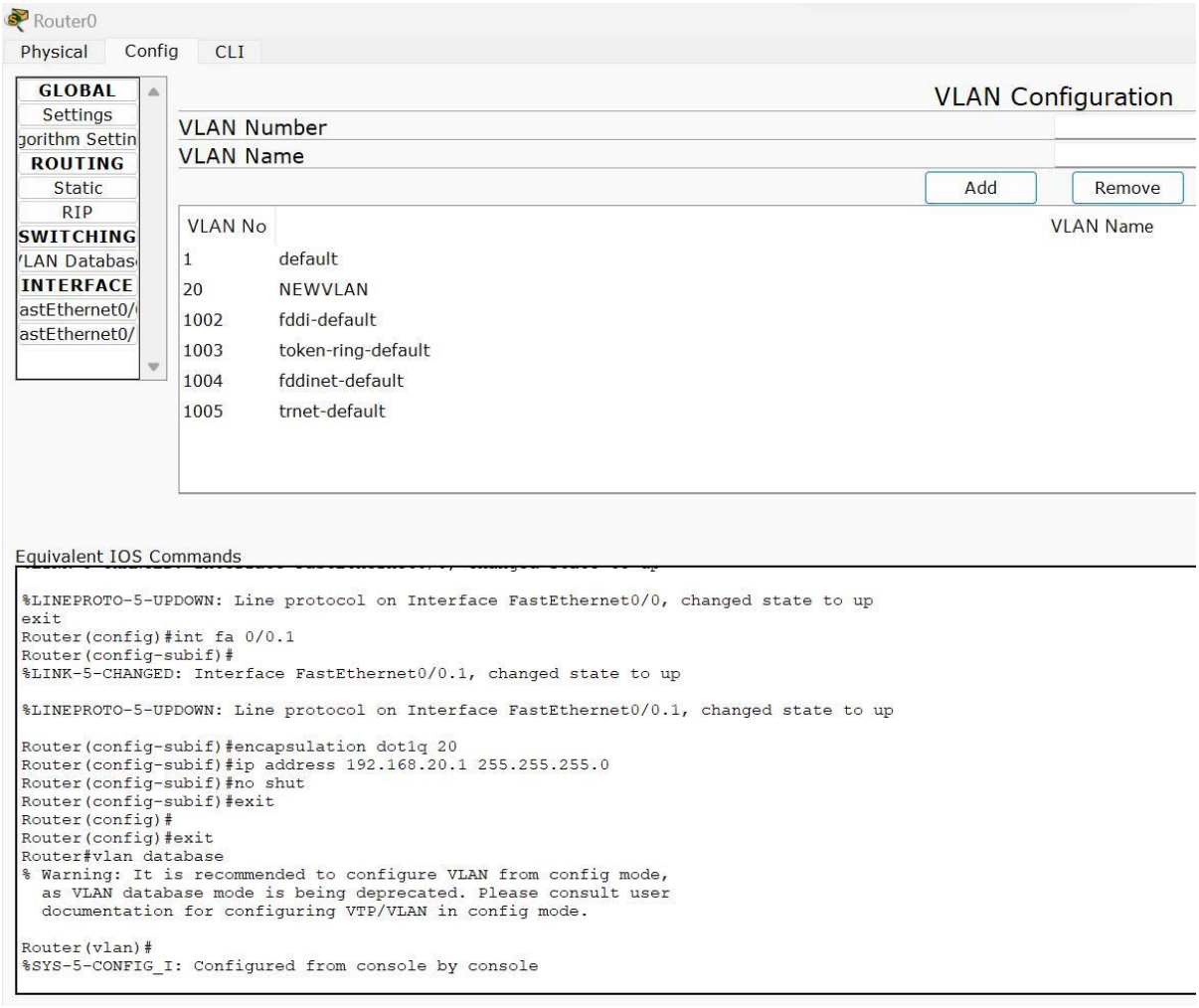
Switch FastEthernet0/3 and FastEthernet0/4



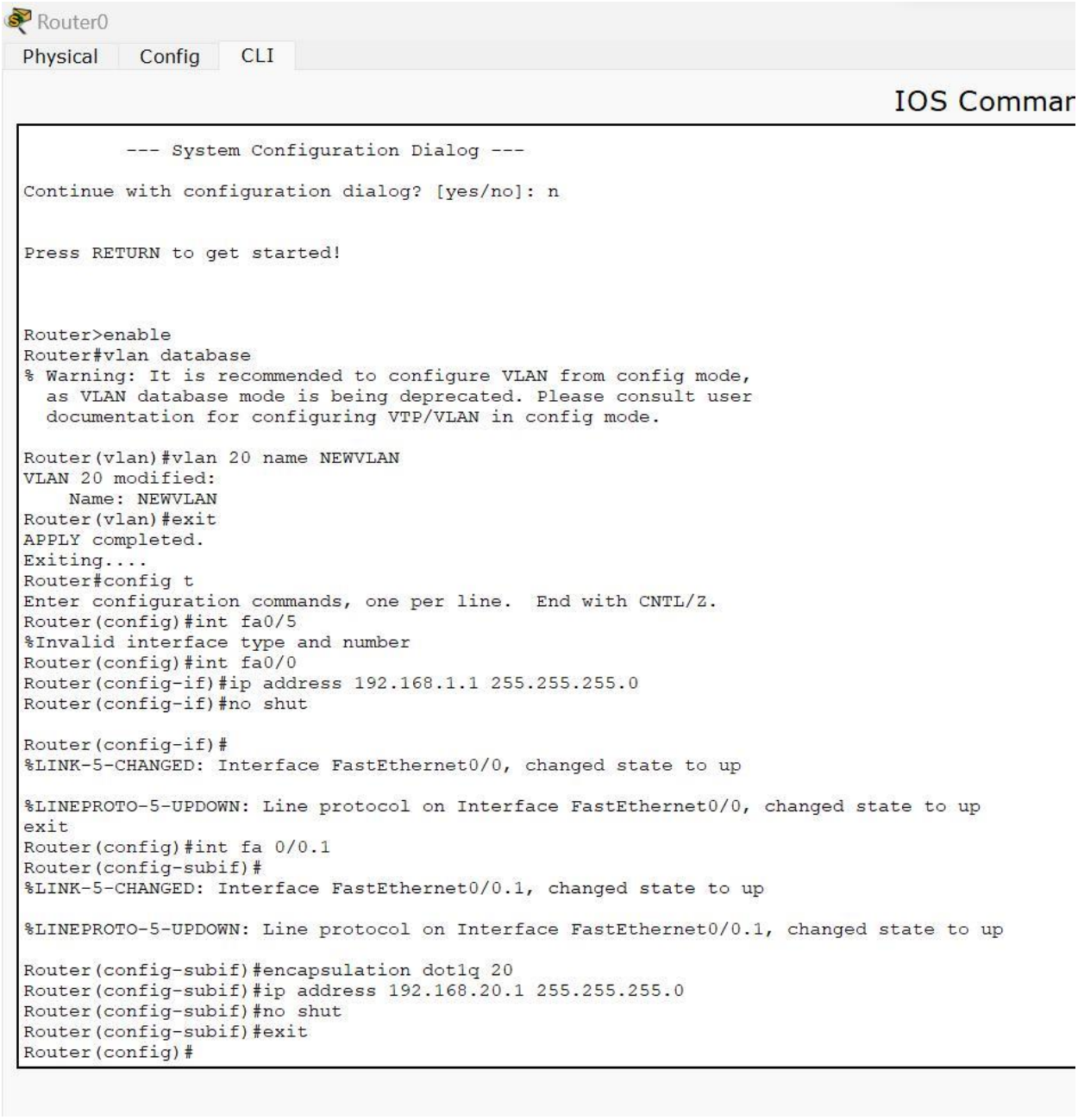
Switch FastEthernet0/1 and FastEthernet0/2



Router R0 :
VLAN DataBase:

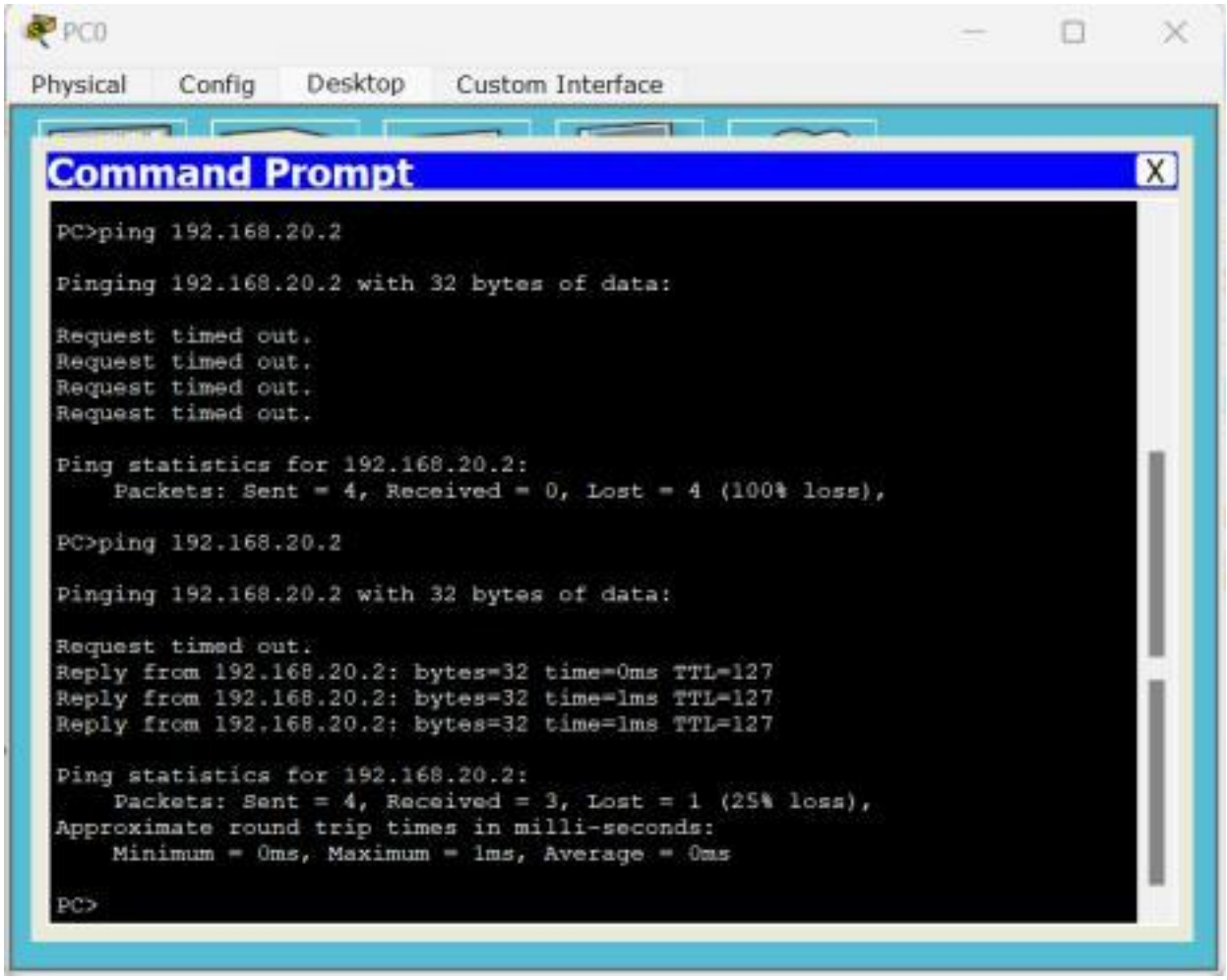


Router R0 :
CLI:



Ping Result :

P0: [Before and after VLAN configuration was successful.]

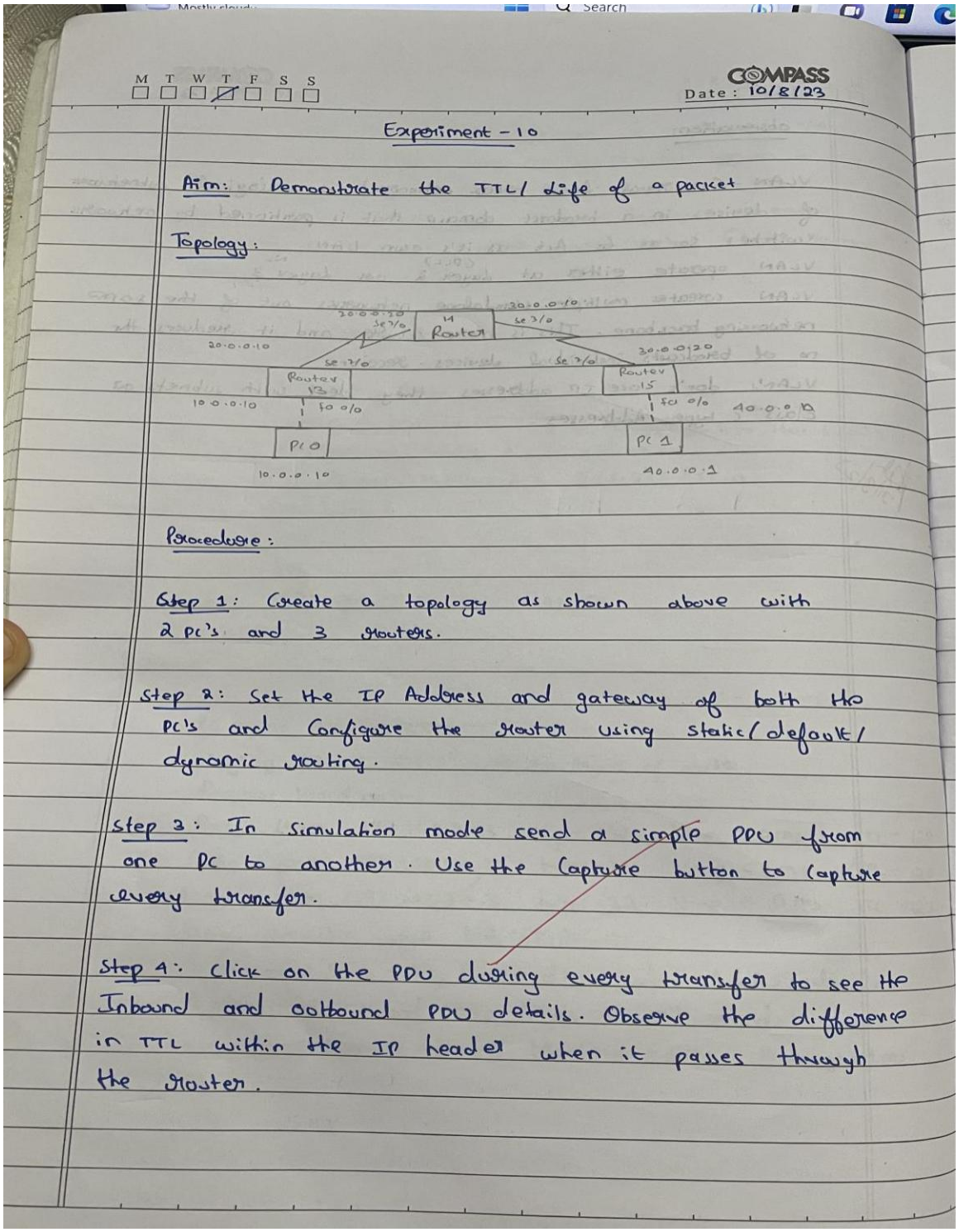


LAB PROGRAM -10

Aim:

Demonstrate the TTL/ Life of a Packet Topology

Procedure :



Experiment - 10

Aim: Demonstrate the TTL/ Life of a packet

Topology:

Procedure:

Step 1: Create a topology as shown above with 2 PC's and 3 routers.

Step 2: Set the IP Address and gateway of both the PC's and Configure the router using static/default/ dynamic routing.

Step 3: In simulation mode send a simple PDU from one PC to another. Use the Capture button to capture every transfer.

Step 4: Click on the PDU during every transfer to see the Inbound and outbound PDU details. Observe the difference in TTL within the IP header when it passes through the router.

M T W T F S S

COMPASS

Date: 10/8/23

output:

IP header: (PC 0) (Inbound)

0	4	8	16	19	31 bits
4	IHL	DSCP:	TL: 28		
ID: 0x7		0x	0x0		
TTL: (255)	PRO: 0x1	CHKSUM			
SRC IP: 10.0.0.1					
DST IP: 40.0.0.1					
OPT: 0x0		0x0			
DATA (variable length)					

IP header: (Router 0) (outbound)

↳ Inbound IP header for router 0 will look the same as above

0	4	8	16	19	31 bits
4	IHL	DSCP:	TL: 28		
ID: 0x6		0x	0x0		
TTL: (254)	PRO: 0x1	CHKSUM			
SRC IP: 10.0.0.1					
DST IP: 40.0.0.1					
OPT: 0x0		0x0			
DATA (variable length)					

→ Similar results are seen for the other 2 routers and PC.

observation:

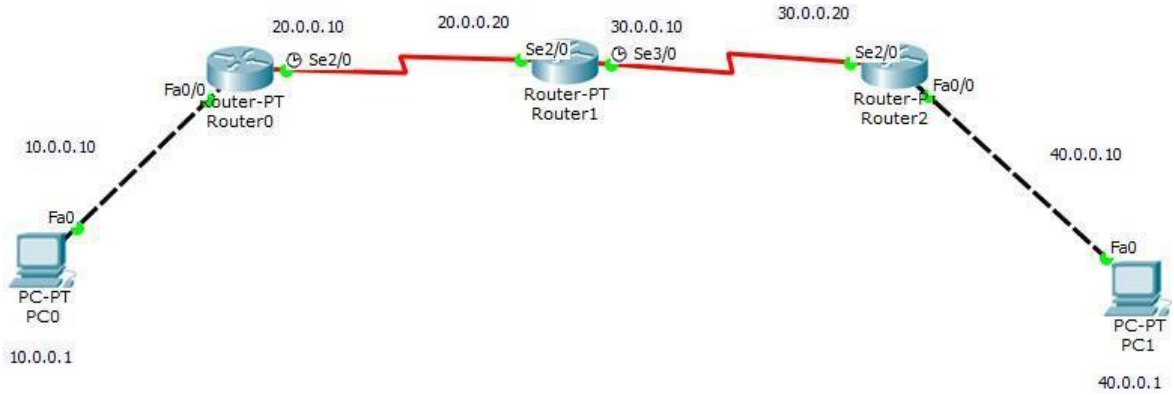
→ TTL: time to live indicates the no of hops the packet can travel before it is discarded by the router to prevent a packet from travelling aimlessly within the network. This field is set by the sender & reduced by 1 by the router it passes through. When the TTL = 0, the router discards the packet and sends ICMP message to source.

Topology :

Configurations:

Configure the devices as per static / default / dynamic routing.

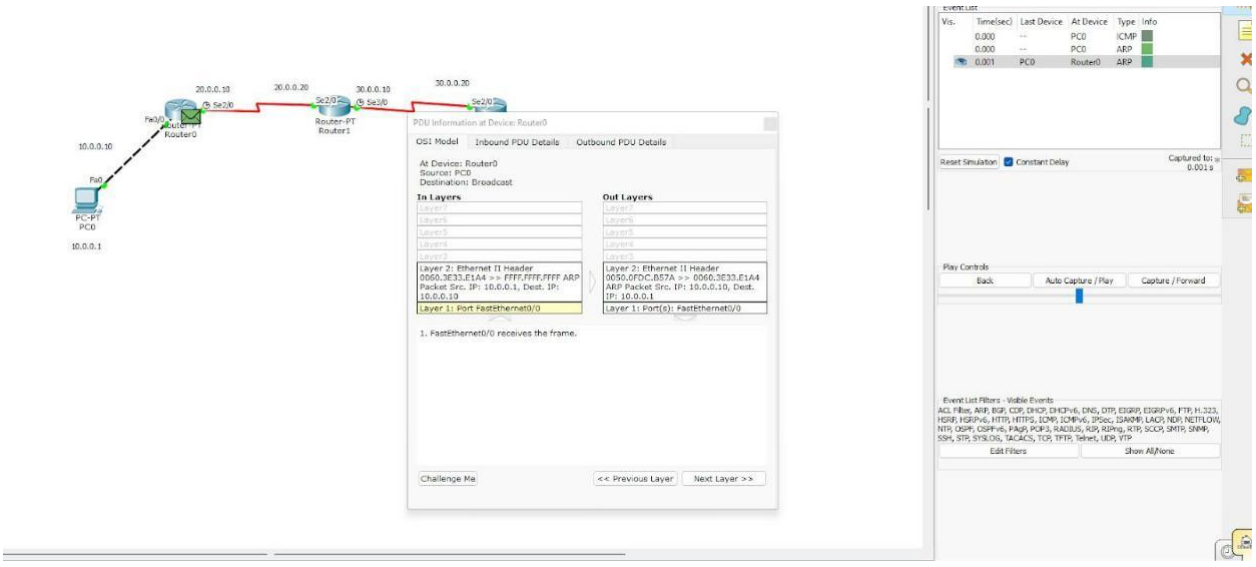
(Below has been done using static routing)

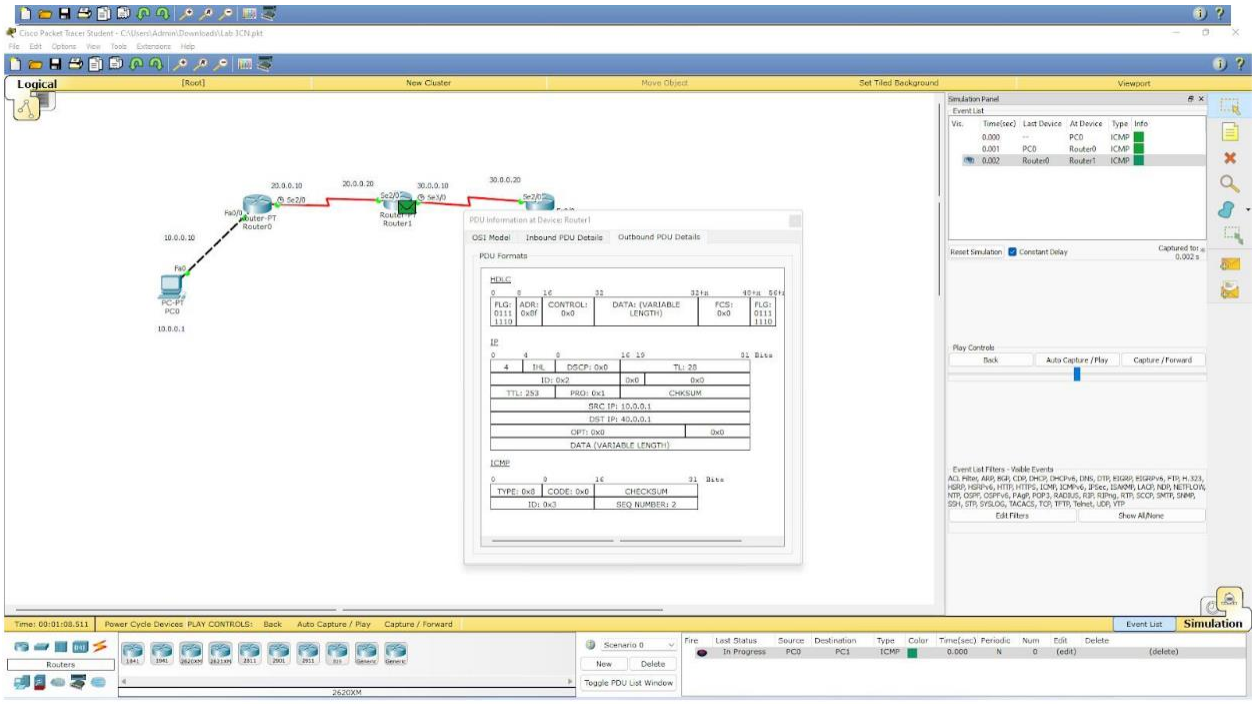


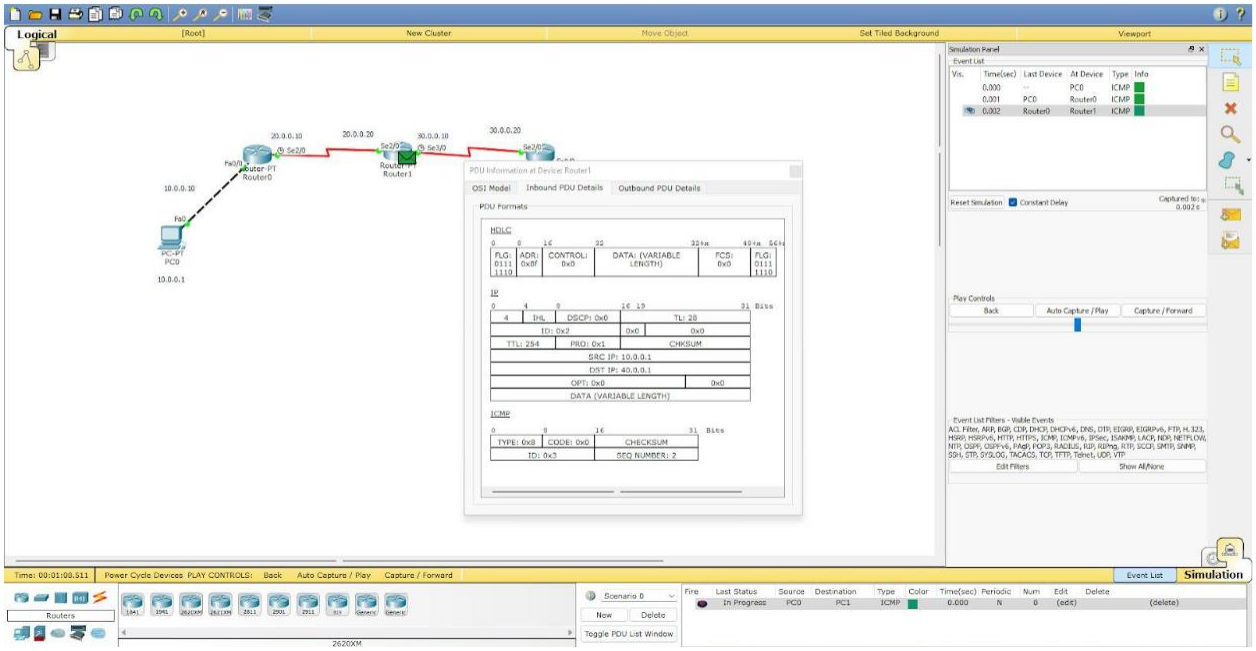
Output :

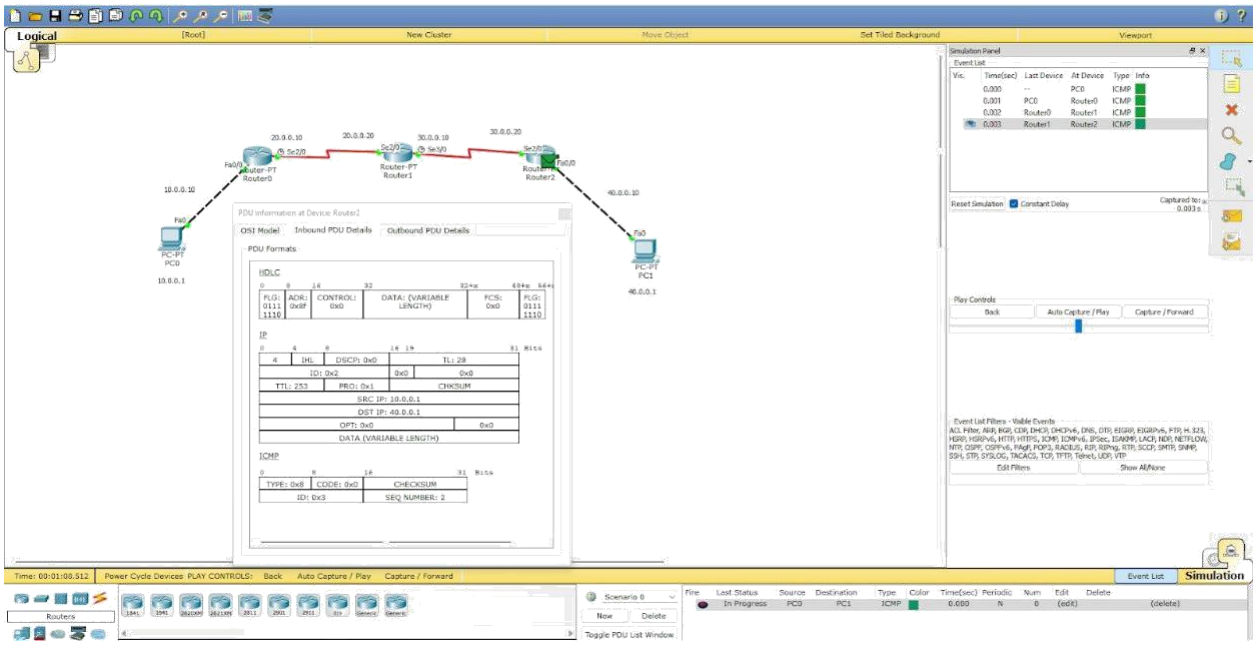
PDU Details :

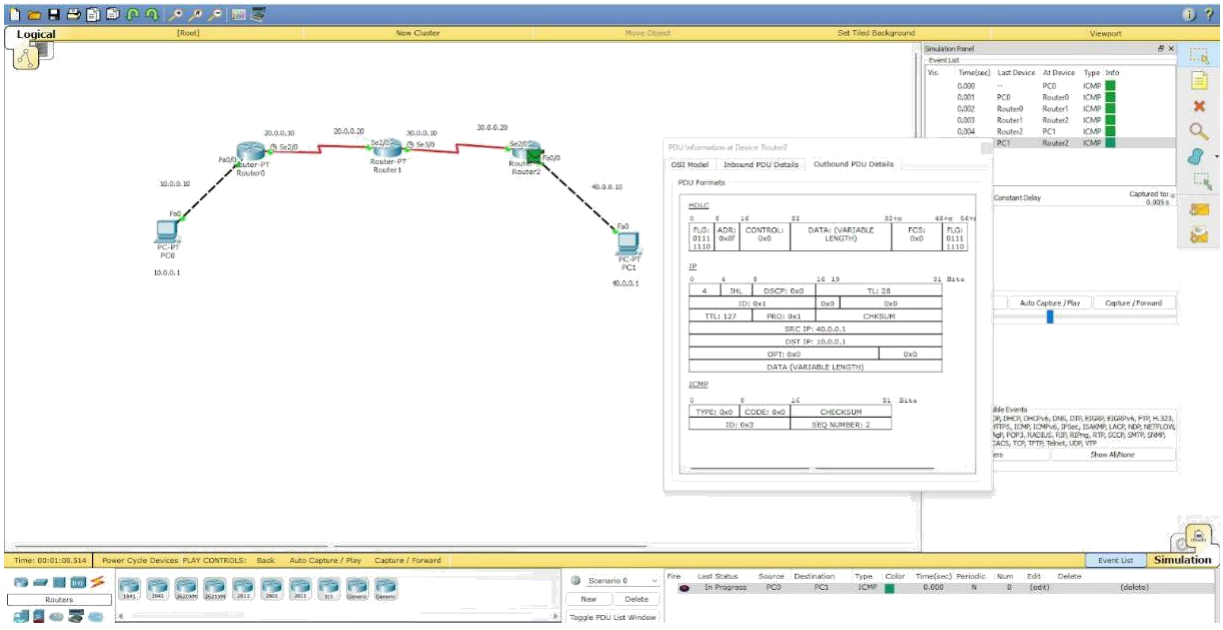
Simple PDU sent from PC0 to PC1 in simulation mode.

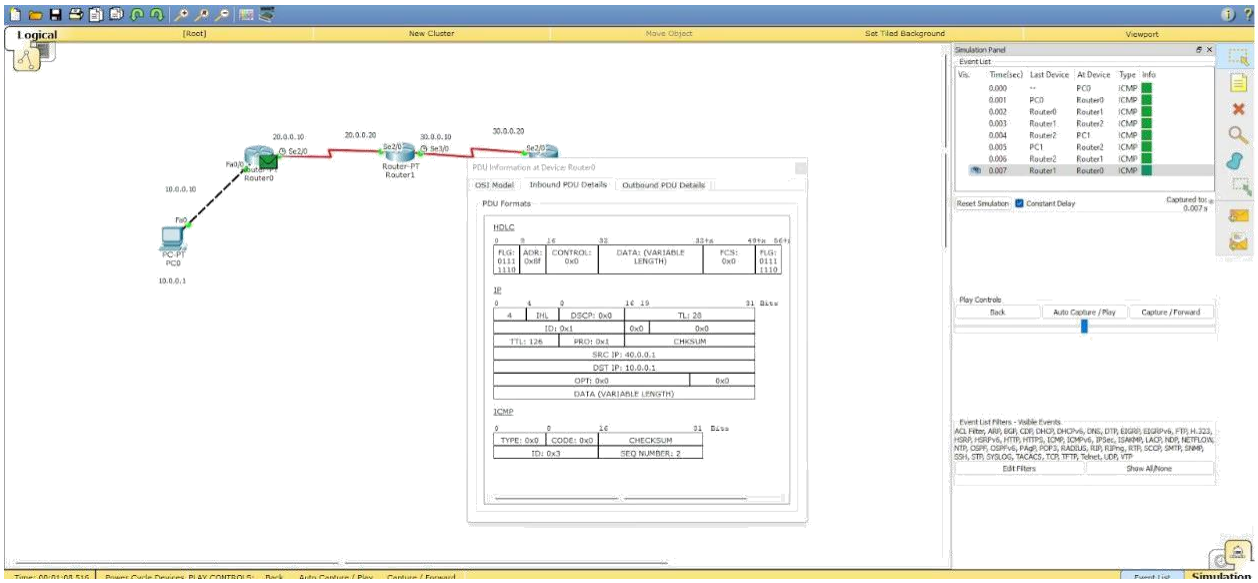


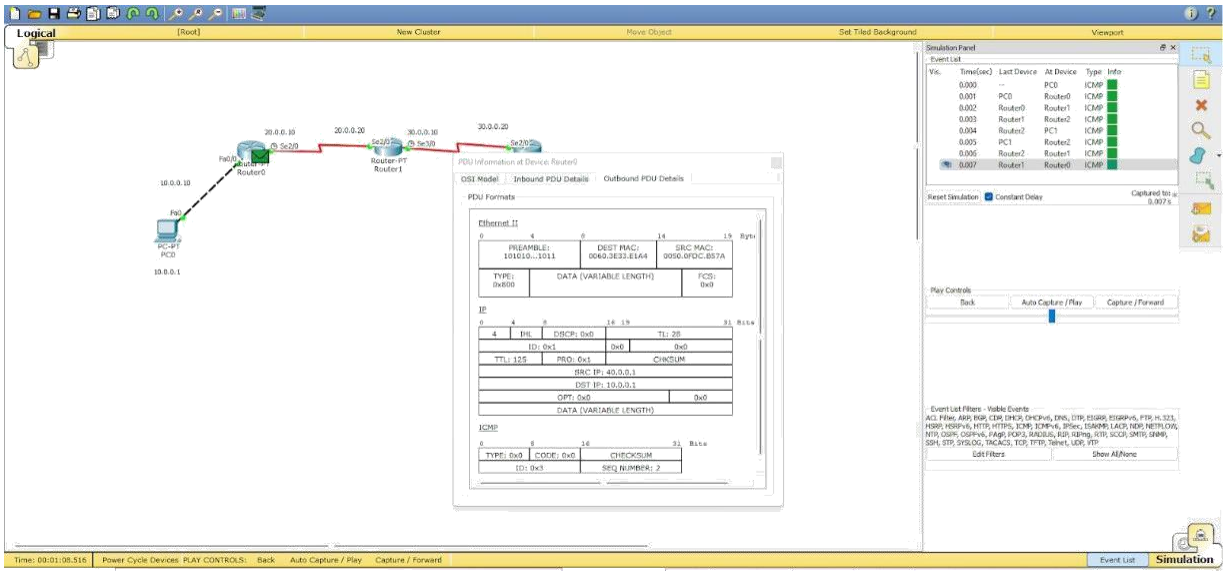

Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
Simulation Panel
Event List
PDU Information at Device: Router1
OSI Model
Inbound PDU Details
Outbound PDU Details
PDU Formats
HDLC
IP
ICMP
Play Controls
Back
Auto Capture / Play
Capture / Forward
Event List Filters - Visible Events
Edit Filters
Show All/None
Time: 00:01:08.511
Power Cycle Devices
PLAY CONTROLS:
Back
Auto Capture / Play
Capture / Forward
Event List
Simulation
Routers
Scenario 0
New
Delete
Toggle PDU List Window
2620XM

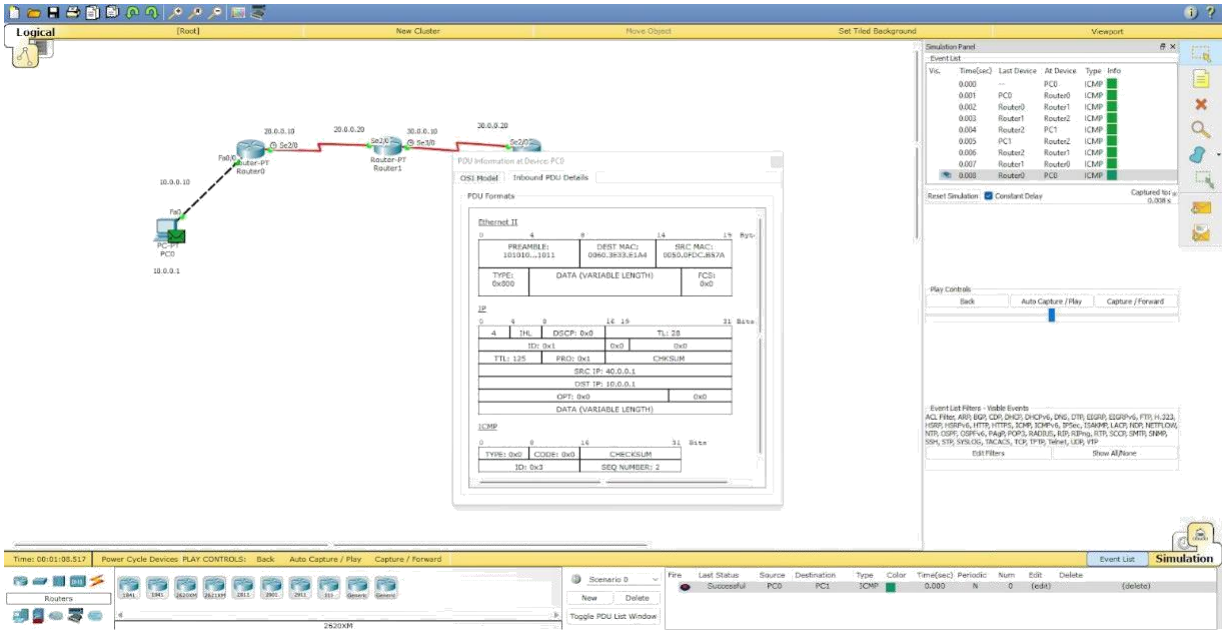

Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
Simulation Panel
Event List
PDU Information at Device: Router1
OSI Model
Inbound PDU Details
Outbound PDU Details
PDU Formats
HDLC
IP
ICMP
Play Controls
Back
Auto Capture / Play
Capture / Forward
Event List Filters - Visible Events
Edit Filters
Show All/None
Time: 00:01:08.511
Power Cycle Devices
PLAY CONTROLS:
Back
Auto Capture / Play
Capture / Forward
Event List
Simulation
Routers
Scenario 0
New
Delete
Toggle PDU List Window
2620XM


Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
Router-PT Router0
Router-PT Router1
Router-PT Router2
PC-PT PC0
PC-PT PC1
10.0.0.10
20.0.0.10
20.0.0.20
30.0.0.10
30.0.0.20
40.0.0.10
10.0.0.1
40.0.0.1
PDU Information at Device Router2
OSI Model
Inbound PDU Details
Outbound PDU Details
PDU Formats
HDLC
IP
ICMP
Simulation Panel
Event List
Play Controls
Back
Auto Capture / Play
Capture / Forward
Edit Filters
Show All/None
Time: 00:01:08.512
Power Cycle Devices PLAY CONTROLS:
Event List
Simulation
Routers
Scenario 0
New
Delete
Toggle PDU List Window


Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
Router-PT Router0
Router-PT Router1
Router-PT Router2
PC-PT PC0
PC-PT PC1
10.0.0.10
20.0.0.10
20.0.0.20
30.0.0.10
30.0.0.20
40.0.0.10
10.0.0.1
40.0.0.1
PDU Information at Device Router2
OSI Model
Inbound PDU Details
Outbound PDU Details
PDU Formats
HDLC
IP
ICMP
Simulation Panel
Event List
Play Controls
Back
Auto Capture / Play
Capture / Forward
Time: 00:01:08.514
Power Cycle Devices PLAY CONTROLS:
Event List
Simulation
Routers
Scenario 0
New
Delete
Toggle PDU List Window


Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
Router0
Router1
PC-PT
PC0
Router-PT
Simulation Panel
Event List
PDU Information at Device Router0
OSI Model
Inbound PDU Details
Outbound PDU Details
PDU Formats
HDLC
IP
ICMP
Play Controls
Back
Auto Capture / Play
Capture / Forward
Event List Filters - Visible Events
Edit Filters
Show All/None
Simulation


Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
Router0
Router1
PC-PT
PC0
Router-PT
Simulation Panel
Event List
PDU Information at Device Router0
OSI Model
Inbound PDU Details
Outbound PDU Details
PDU Formats
Ethernet II
IP
ICMP
Play Controls
Back
Auto Capture / Play
Capture / Forward
Event List Filters - Visible Events
Edit Filters
Show All/None
Simulation

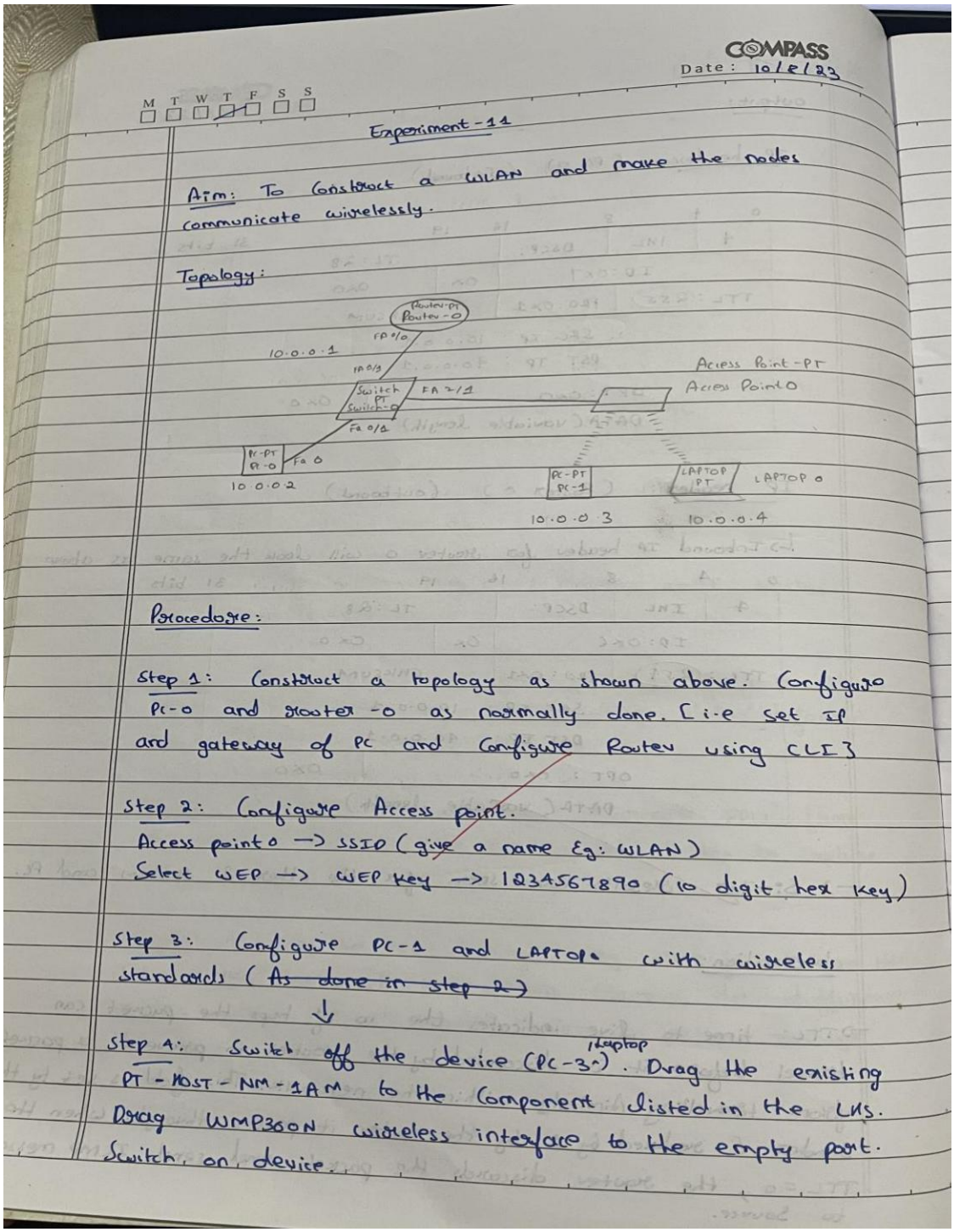

Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
Router-PT
Router0
Router-PT
Router1
PC-PT
PC0
10.0.0.10
20.0.0.20
10.0.0.1
PDU Information at Device: PC0
OSI Model
Inbound PDU Details
PDU Formats
Ethernet II
IP
ICMP
Simulation Panel
Event List
Play Controls
Back
Auto Capture / Play
Capture / Forward
Time: 00:01:05.517
Power Cycle Devices
PLAY CONTROLS:
Routers
Scenario 0
New
Delete
Toggle PDU List Window
Simulation

LAB PROGRAM – 11

Aim :

To construct a WLAN and make the nodes communicate wirelessly

Procedure :



M T W T F S S

COMPASS
Date: 10/8/23

Step 5: In the Config tab of (PC-1 & Laptop 0) a new wireless interface would have been added. Configure SSID, WEP key, IP Address, Gateway (10.0.0.1) to the device.

Result:

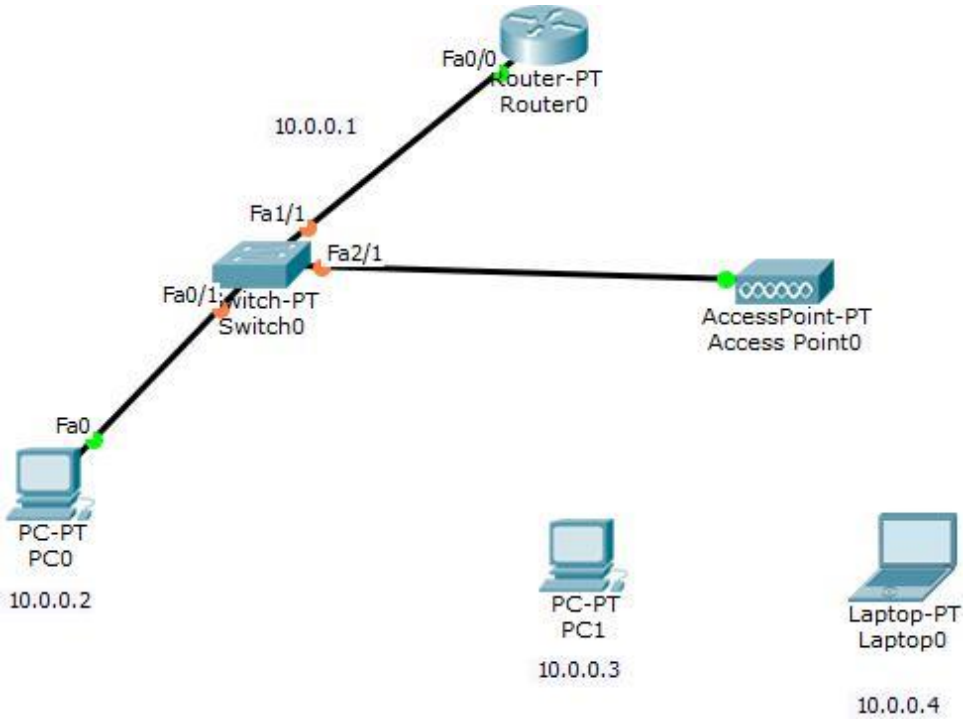
PC > Ping 10.0.0.3

pinging 10.0.0.3 with 32 bytes of data:
Reply from 10.0.0.3 : bytes = 32 time = 19ms TTL = 128
Reply from 10.0.0.3 : bytes = 32 time = 8ms TTL = 128
Reply from 10.0.0.3 : bytes = 32 time = 10ms TTL = 128
Reply from 10.0.0.3 : bytes = 32 time = 10ms 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 = 8ms, Maximum = 19ms, Average = 11ms

Observation:

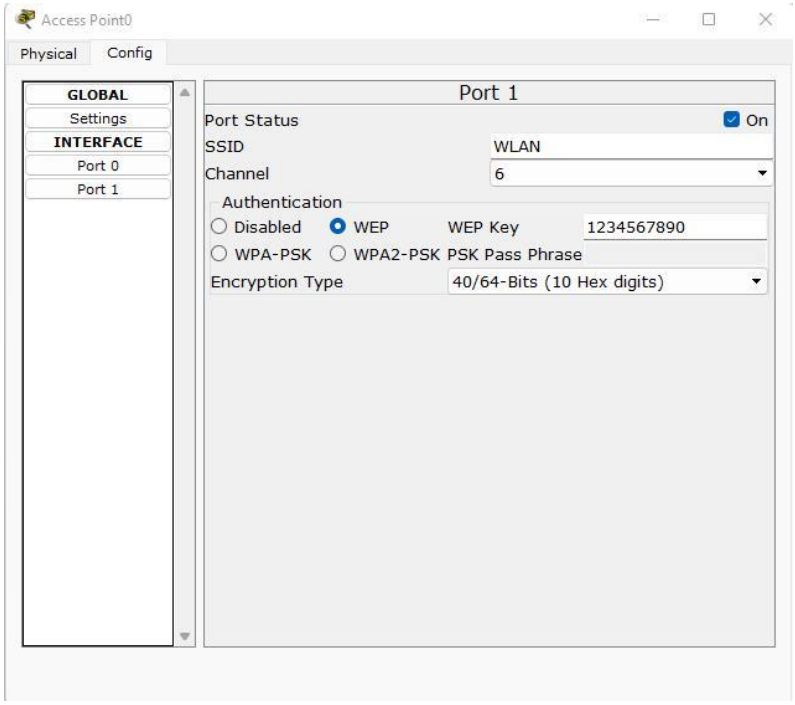
WLAN - (wireless local area network) is colocated devices that form a network based on radio transmissions. WLAN transmits information over radio waves. Data is sent in form of packets. Each packet consists of layers, labels and instructions with unique MAC address assigned to end points.
Access point is a base station that serves as a bridge between wired and wireless networks. Through Access points we can connect to multiple devices wirelessly and transmit data.

Topology :

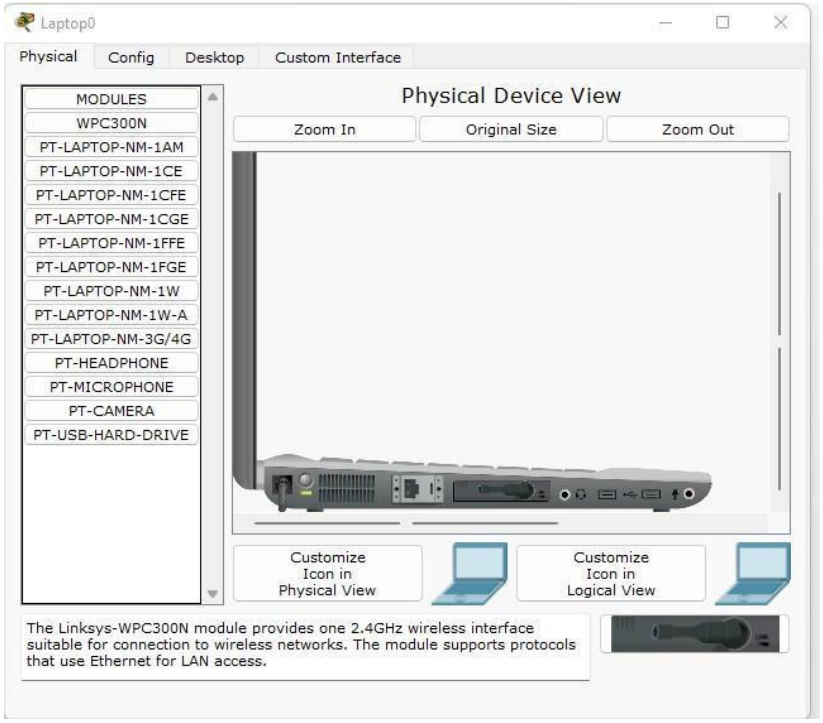


Configurations :

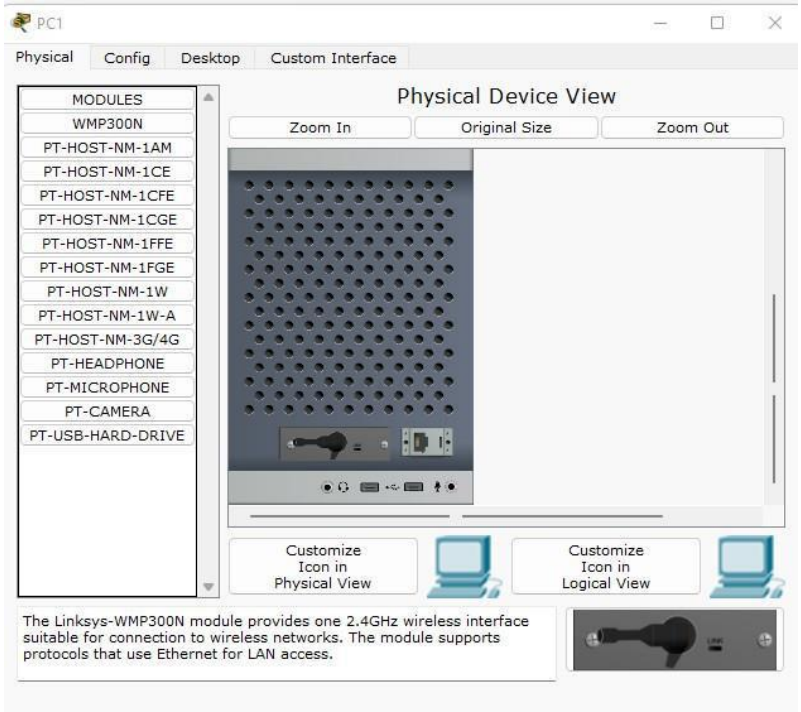
Access Point0:



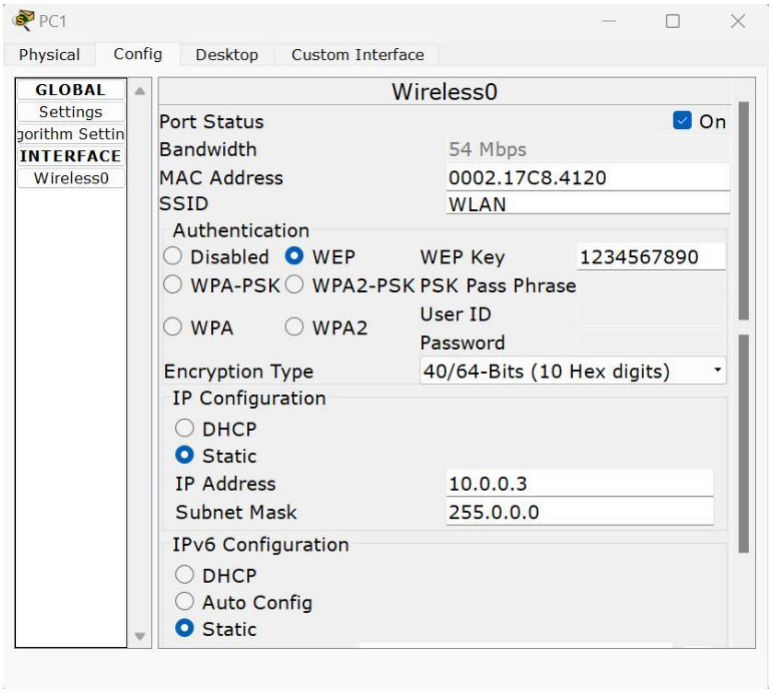
Laptop0 Physical port change



PC0 Physical port change:



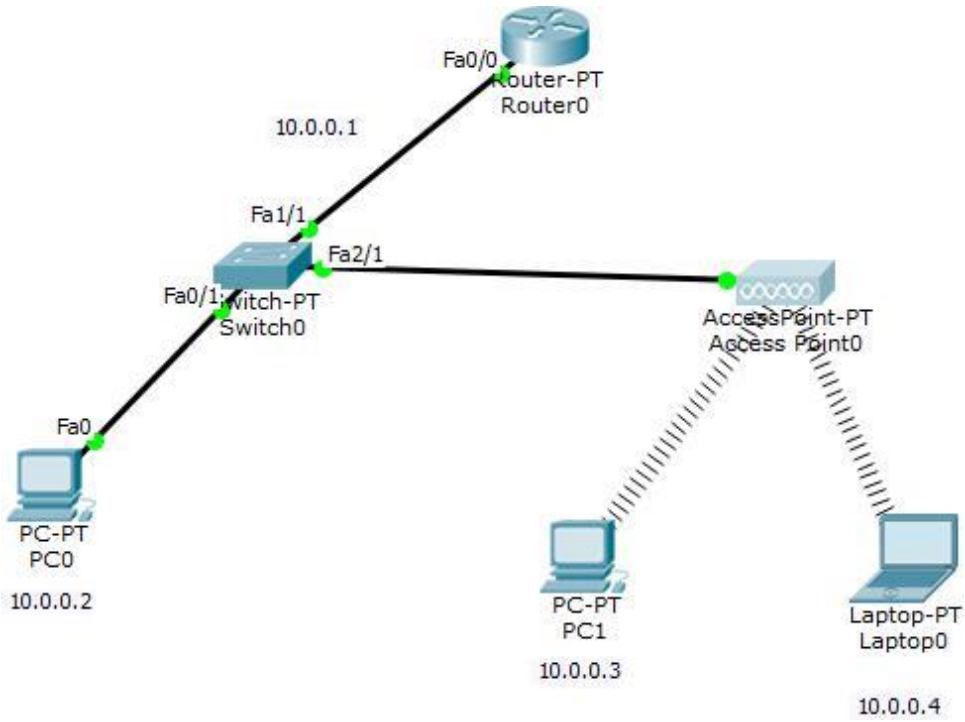
PC0 and Laptop0 Wireless configuration:



Router Configuration :

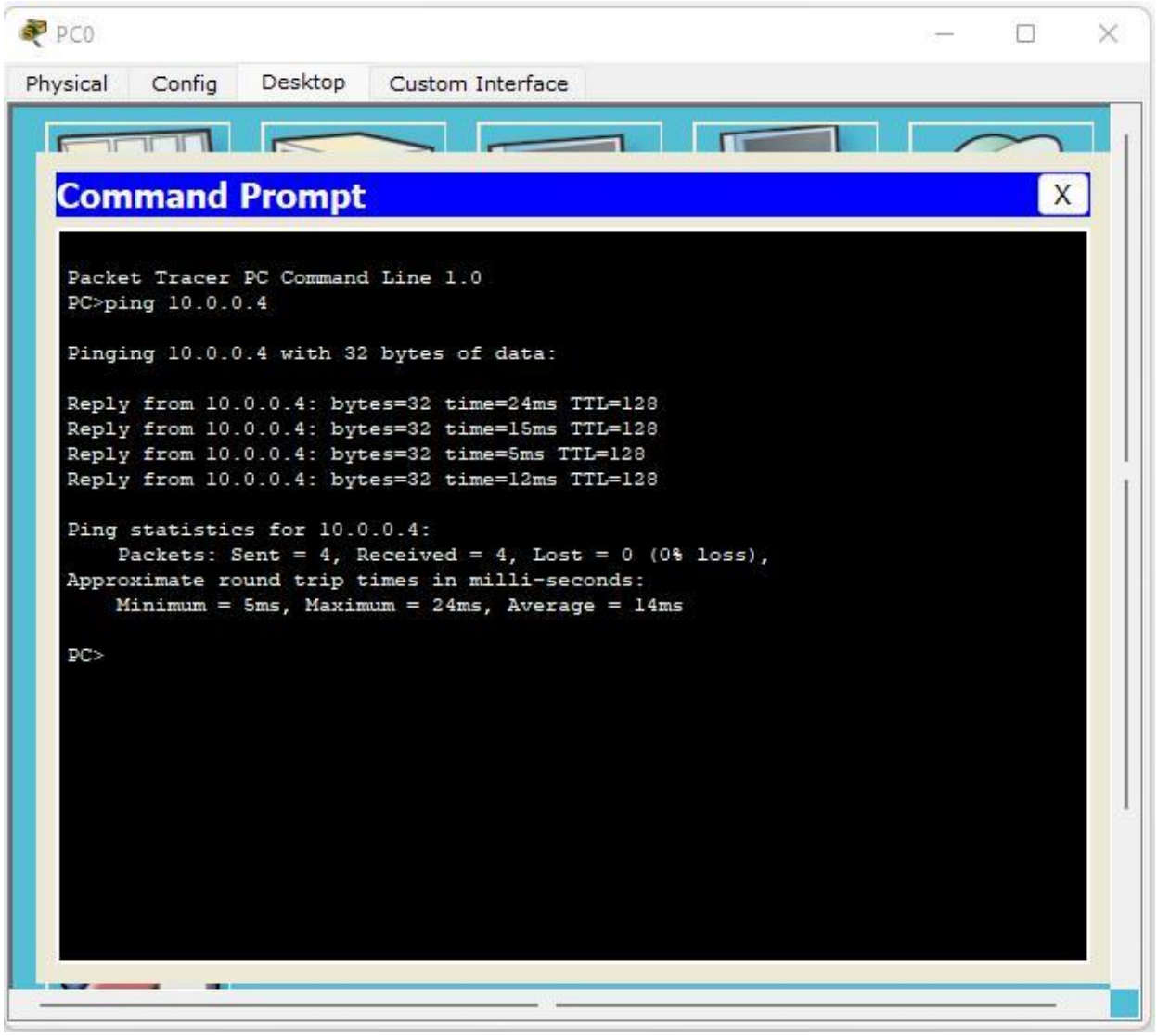
Router 0 CLI :

Final Topology:

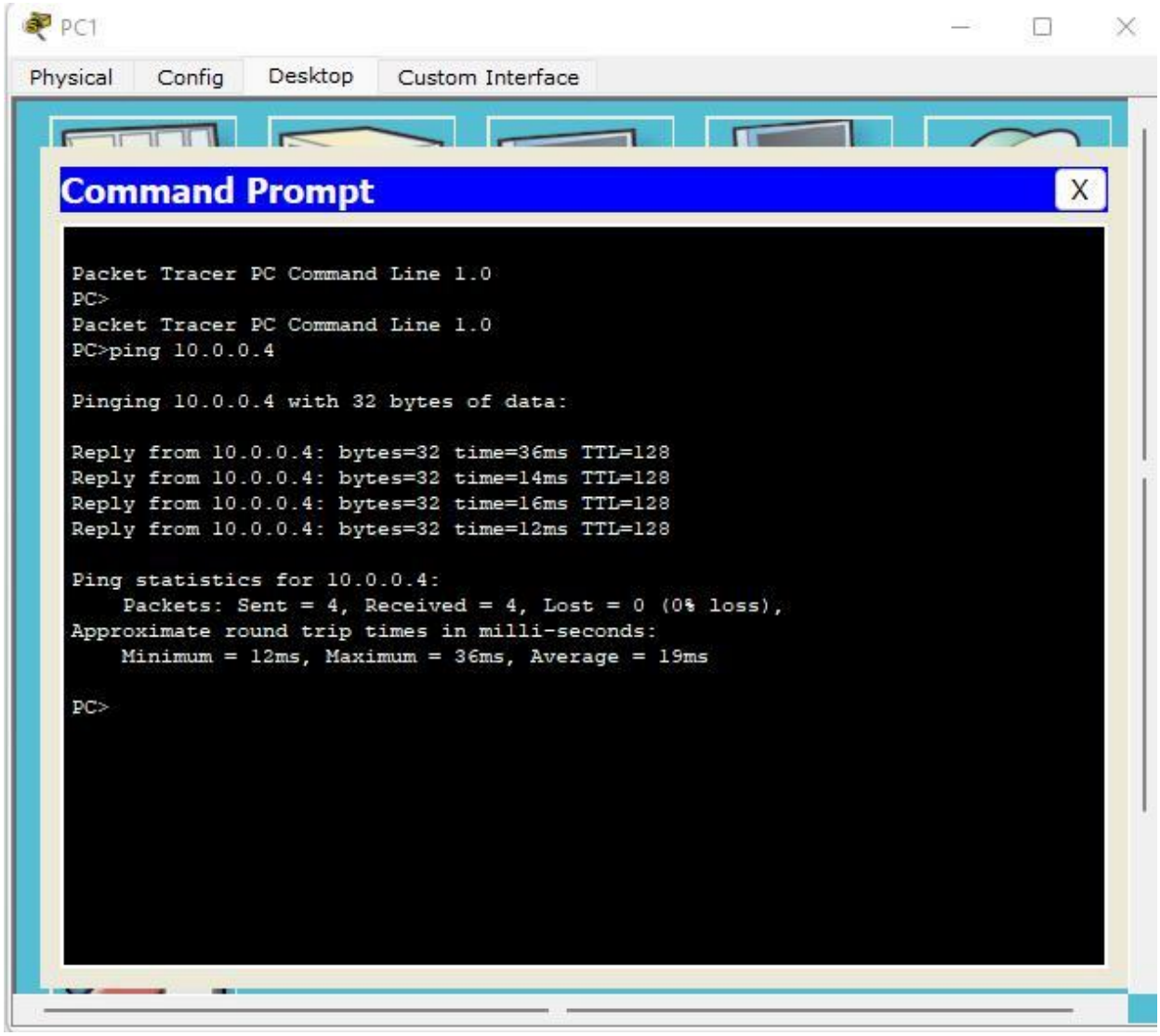


Ping Results :

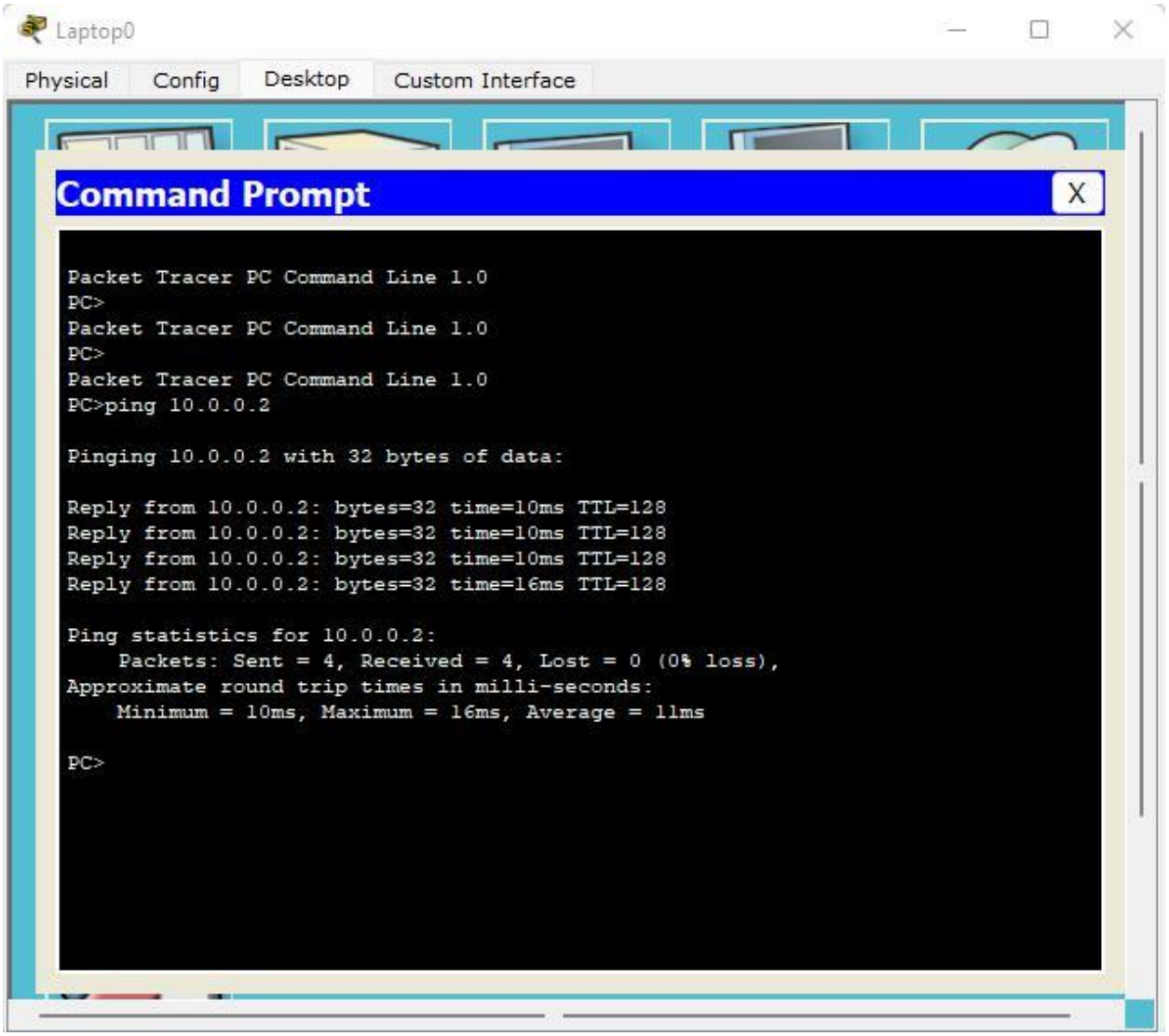
PC0 to Laptop0 :



PC1 to Laptop0 :



Laptop0 to PC0:

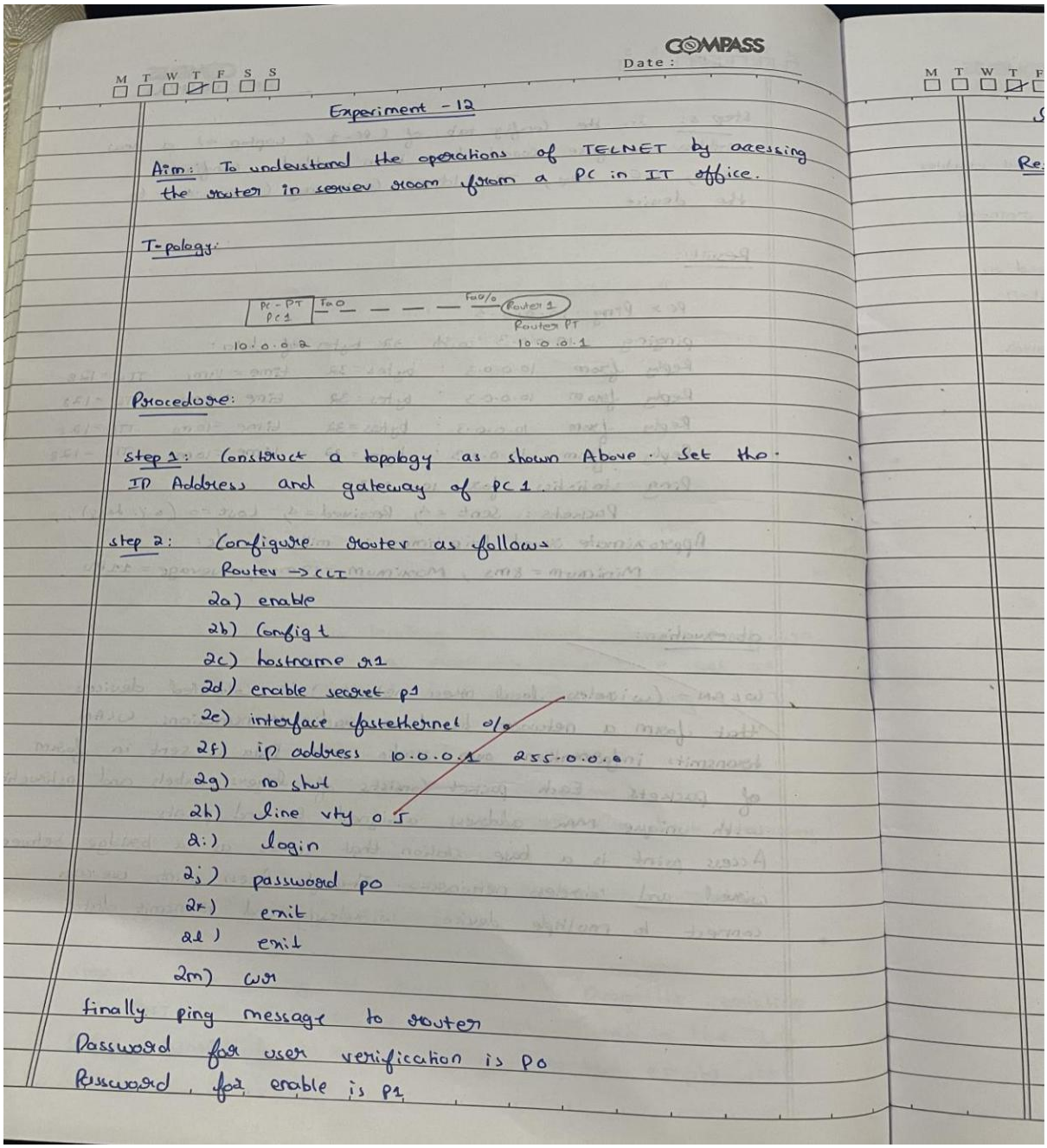


LAB PROGRAM – 12

Aim :

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

Procedure :



Experiment - 12

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

Topology:

PC-PT PC1 — Fa0 — — — Fa0/0 Router1 Router PT

10.0.0.2 10.0.0.1

Procedure:

step 1: Construct a topology as shown Above. Set the IP Address and gateway of PC 1.

step 2: Configure router as follows

Router -> CLI

2a) enable
2b) Config t
2c) hostname r1
2d) enable secret p1
2e) interface fastethernet 0/0
2f) ip address 10.0.0.1 255.0.0.0
2g) no shut
2h) line vty 0 5
2i) login
2j) password p0
2k) exit
2l) exit
2m) wr

finally ping message to router

Password for user verification is p0

Password for enable is p1

M T W T F S S

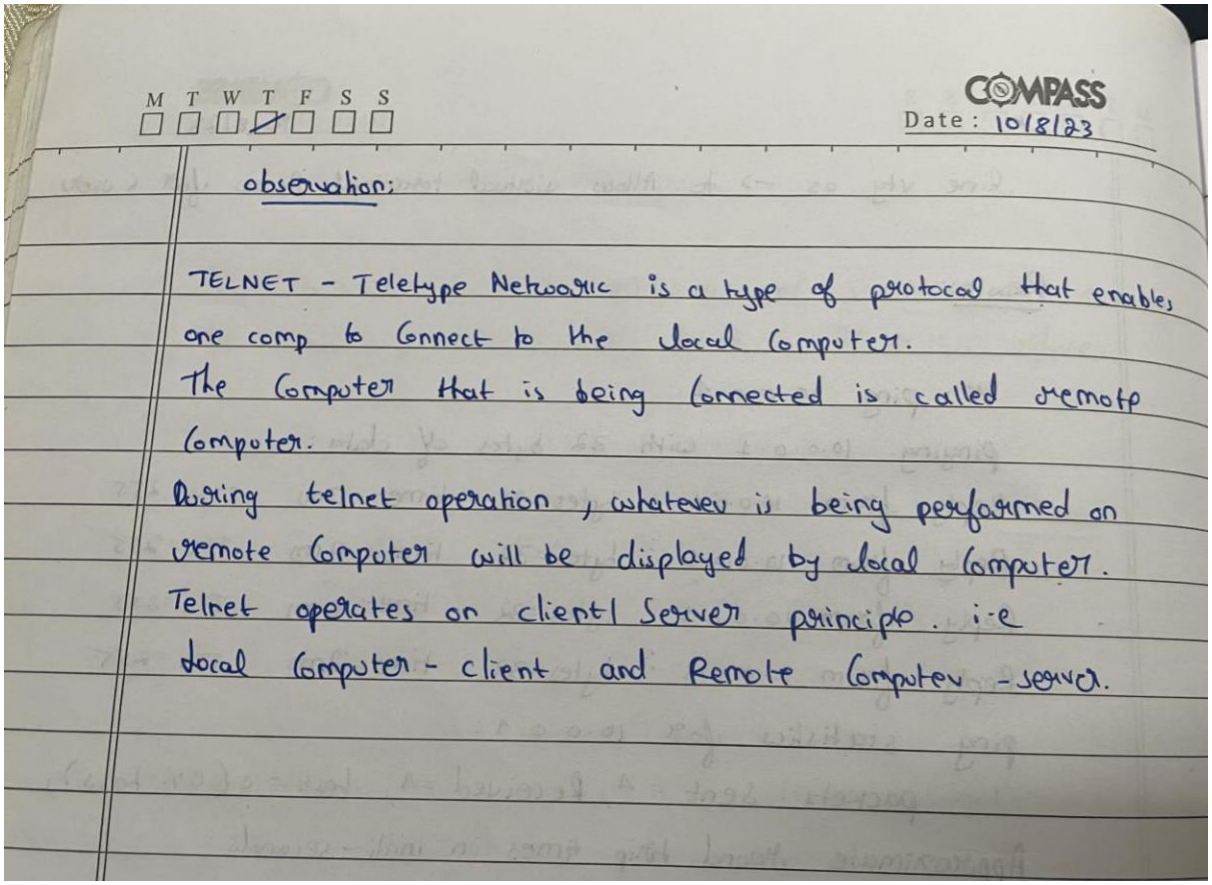
COMPASS
Date: 10/8/23

line vty 05 -> to Allow virtual terminal Access for 6 users

Result:

PC> ping 10.0.0.1
Pinging 10.0.0.1 with 32 bytes of data:
Reply from 10.0.0.1 : bytes=32 time= 0ms TTL = 255
Reply from 10.0.0.1 : bytes=32 time= 0 ms TTL=255
Reply from 10.0.0.1 : bytes=32 time= 0ms TTL=255
Reply from 10.0.0.1 : bytes=32 time= 2ms TTL=255
ping statistics for 10.0.0.1:
packets : Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 2ms, Average = 0ms

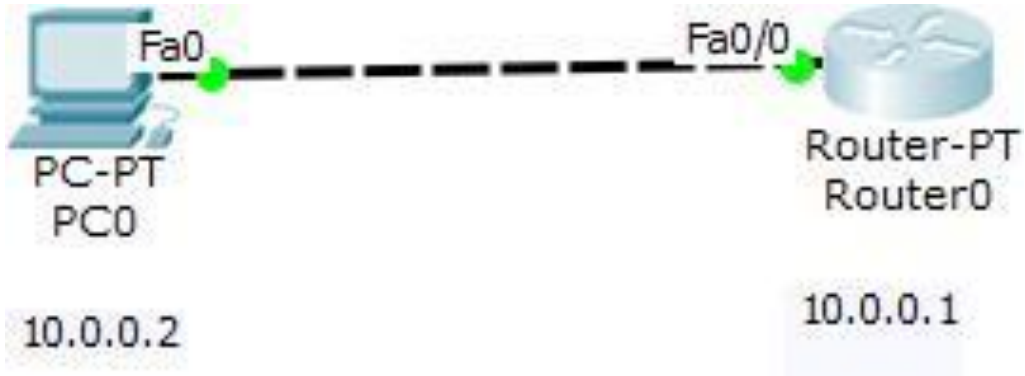
PC > telnet 10.0.0.1
Trying 10.0.0.1 .... open
User Access Verification
Password:
r1>enable
Password:
r1 # show ip route
Codes : C - connected, S- static, I-IGRP, R-RIP, M-mobile
D - EIGRP, EX-EIGRP external
N1 - OSPF NSSA external type 1
E1 - OSPF external type 1
i - IS-IS, L1-IS-IS
* - candidate default
P - periodic downloaded static route
Gateway of last resort is not set
C 10.0.0.0/8 is directly Connected, FastEthernet 0/0



observation:

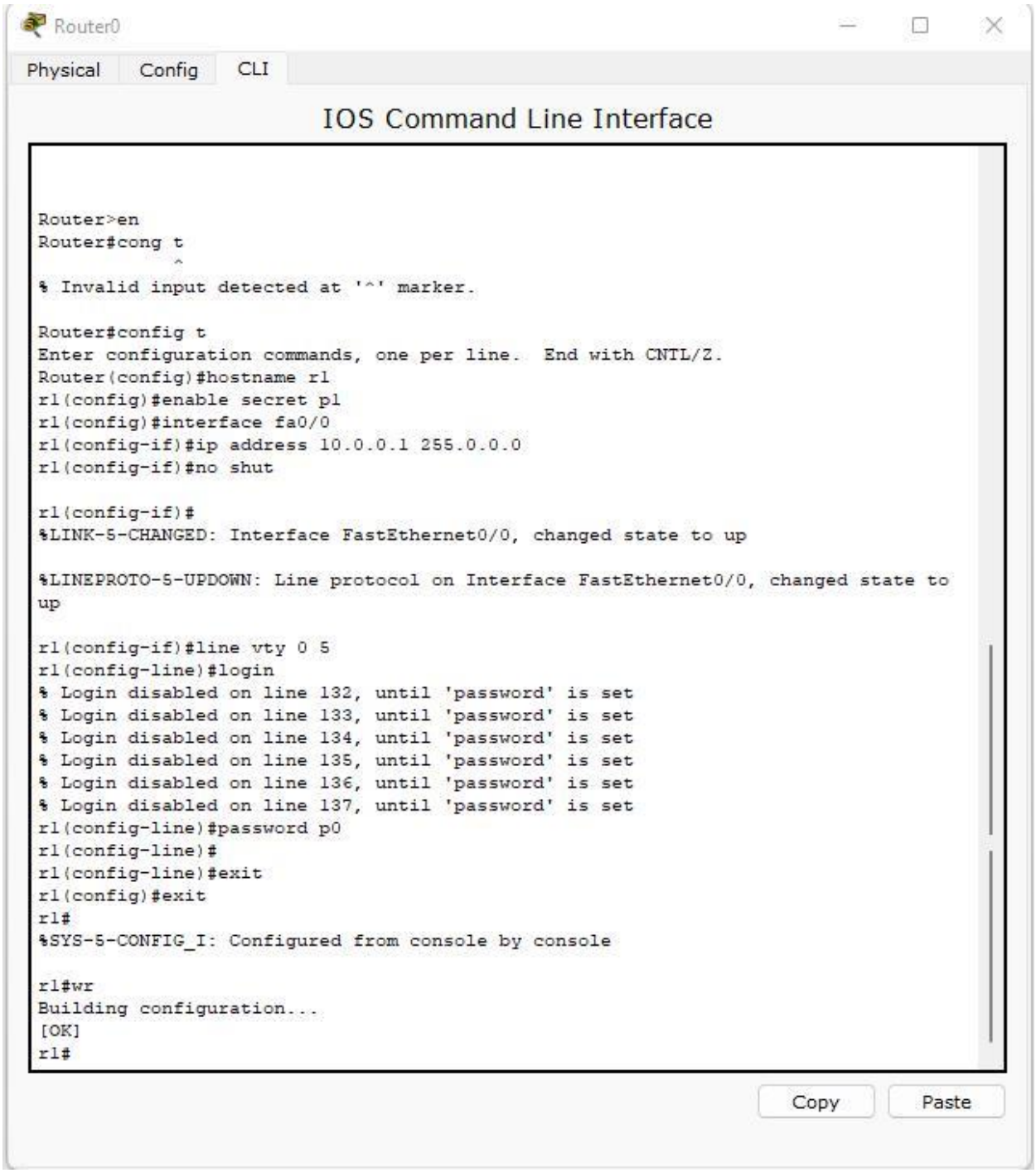
TELNET - Teletype Network is a type of protocol that enables one comp to connect to the local computer.
The computer that is being connected is called remote computer.
During telnet operation, whatever is being performed on remote computer will be displayed by local computer.
Telnet operates on client/server principle. i.e local computer - client and Remote computer - server.

Topology :



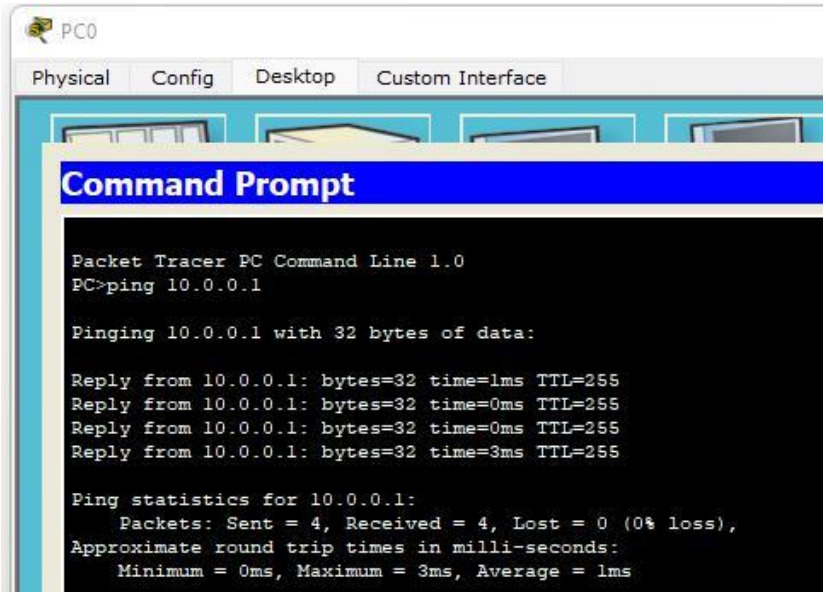
Router Configuration :

Router 0 CLI:

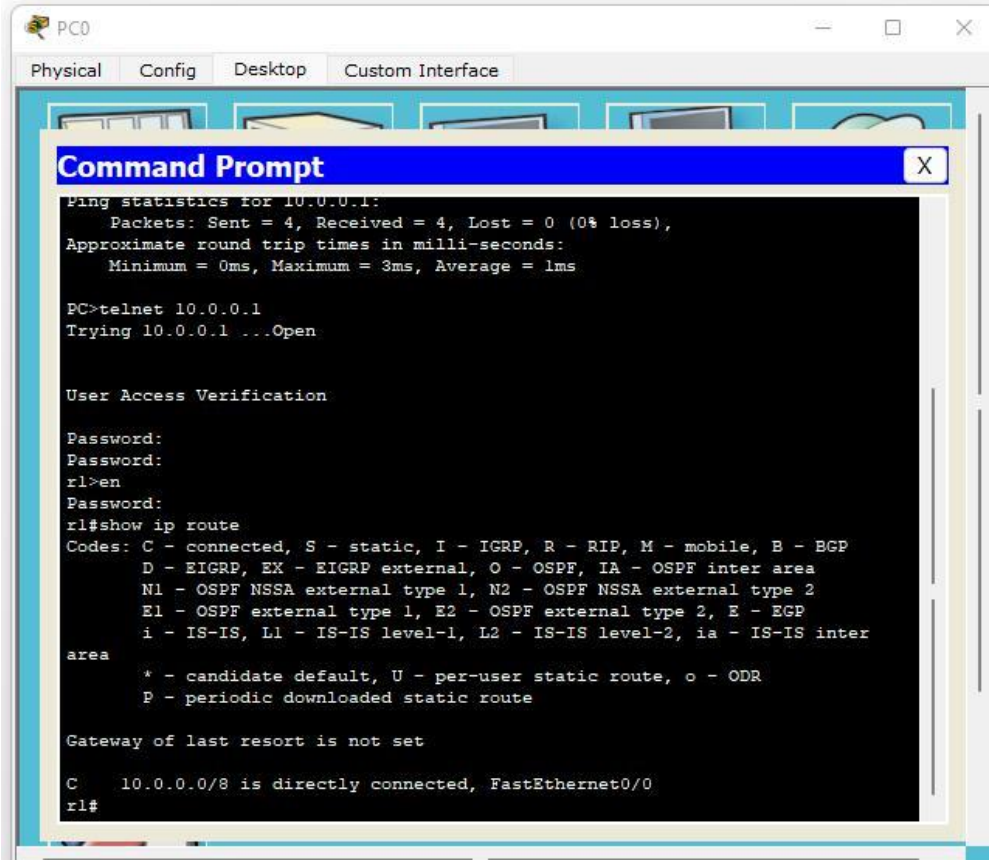


Ping Results :

PC0 to Router:



Accessing the router in server room from a PC in IT office.



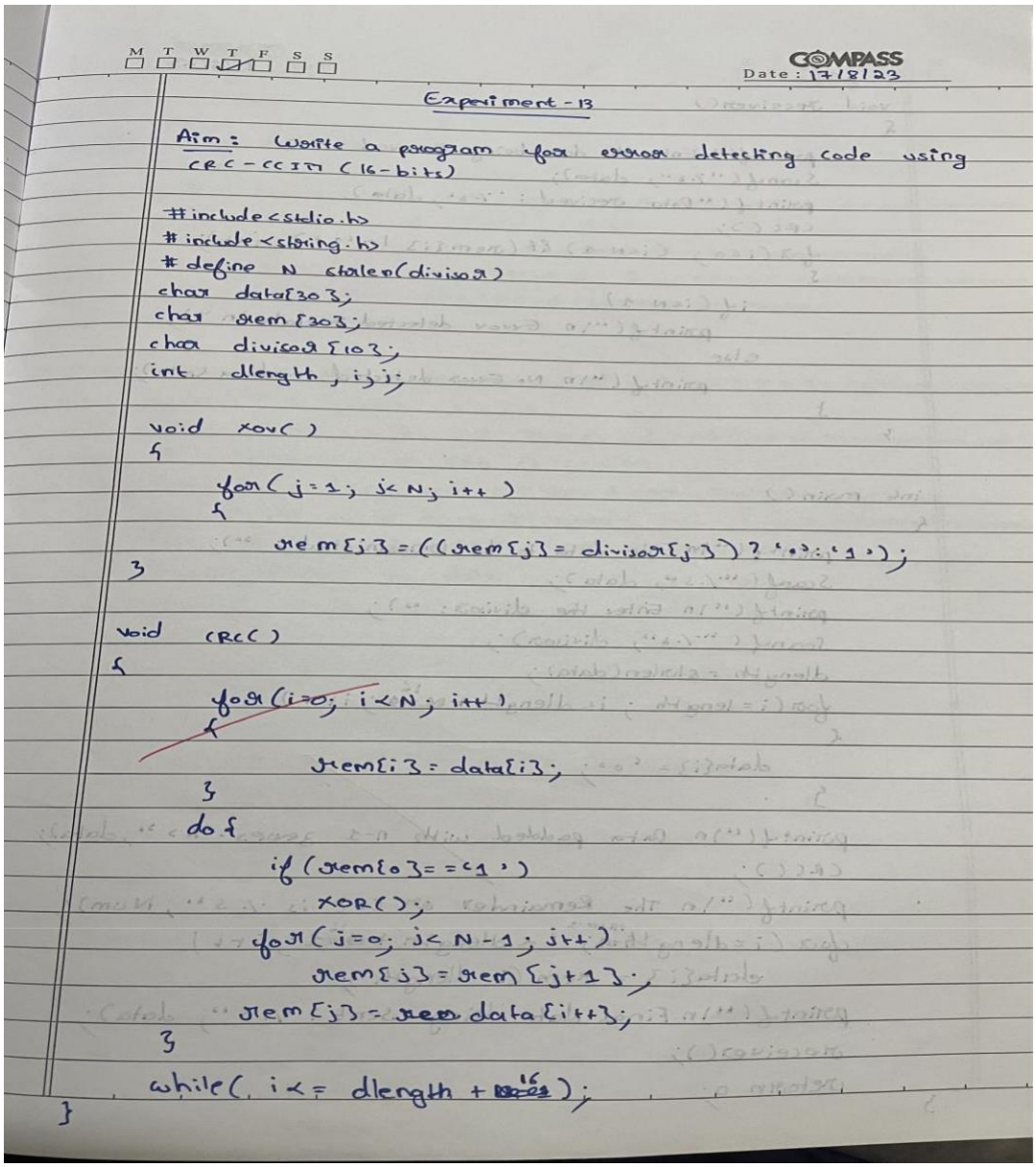
CYCLE II

LAB PROGRAM – 13

Aim :

Write a program for error detecting code using CRCCCITT (16-bits)

Procedure :



Experiment - 13

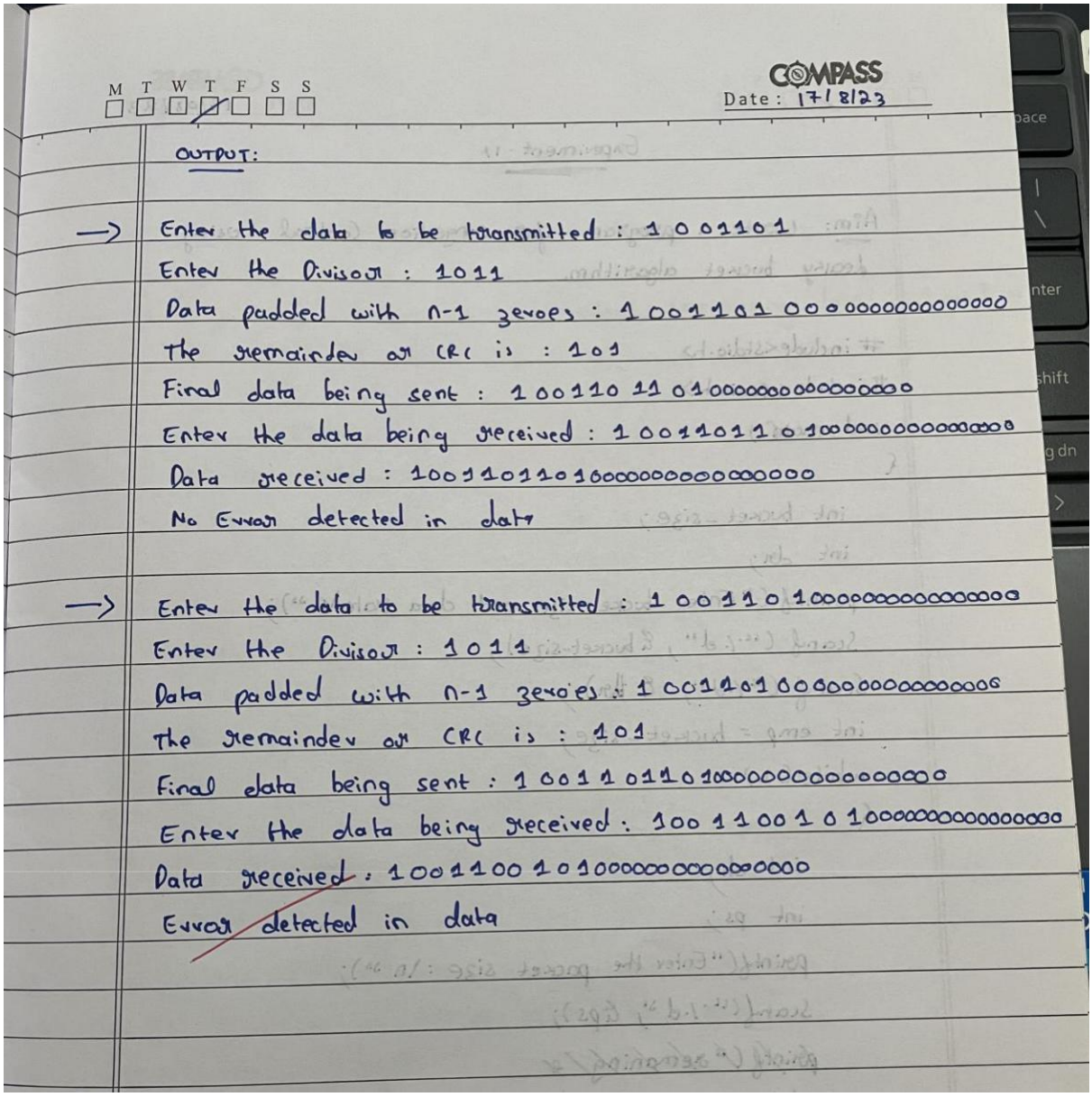
Aim: Write a program for error detecting code using CRC-CCITT (16-bits)

```
#include<stdio.h>
#include<string.h>
#define N strlen(divisor)
char data[30];
char rem[30];
char divisor[10];
int dlength, i, j;

void xor()
{
    for(j=1; j<N; i++)
    {
        rem[j] = ((rem[j] = divisor[j])? '0':'1');
}

void crc()
{
    for(i=0; i<N; i++)
    {
        rem[i] = data[i];
    }
    do{
        if(rem[0]=='1')
            XOR();
        for(j=0; j<N-1; j++)
            rem[j] = rem[j+1];
        rem[j] = data[i++];
    }
    while(i<= dlength + 16);
}
```

COMPASS
Date: 17/8/23

M T W T F S S

```
void receiver()
{
    printf("Enter the data being recived : ");
    scanf("%s", data);
    printf("Data recived : %s", data);
    CRC();
    for(i=0; (i<N-1) && (rem[i] != '1'); i++);
    {
        if(i<N-1)
            printf("\n Error detected in data\n");
        else
            printf("\n No Error detected in data\n");
    }
}

int main()
{
    printf("\n Enter data to be transmitted : ");
    scanf("%s", data);
    printf("\n Enter the divisor : ");
    scanf("%s", divisor);
    dlength = strlen(data);
    for(i=length; i<dlength+16; i++)
    {
        data[i] = '0';
    }
    printf("\n Data padded with n-1 zeroes : %s", data);
    CRC();
    printf("\n The Remainder or CRC is %s", Num);
    for(i=dlength+16-1; i<dlength+16; i++)
        data[i] = rem[i-dlength];
    printf("\n Final data being sent : %s", data);
    receiver();
    return 0;
}
```



OUTPUT:

Enter the data to be transmitted : 1001101
Enter the Divisor : 1011
Data padded with n-1 zeroes : 1001101000
The remainder or CRC is : 101
Final data being sent : 1001101101000
Enter the data being received : 1001101101000
Data received : 1001101101000
No Error detected in data

Enter the data to be transmitted : 100110100
Enter the Divisor : 1011
Data padded with n-1 zeroes : 1001101000
The remainder or CRC is : 101
Final data being sent : 1001101101000
Enter the data being received : 1001100101000
Data received : 1001100101000
Error detected in data

Code :

```
#include<stdio.h>
#include<string.h>
#define N strlen(divisor)
char data[28];
char rem[28];
char divisor[10];
int dlength,i,j;
void XOR(){
    for(j = 1;j < N; j++)
    rem[j] = (( rem[j] == divisor[j])?'0':'1');

}
```

```
void receiver(){

    printf("Enter the received data: ");
    scanf("%s", data);
    printf("\n\n");
    printf("Data received: %s", data);

    crc();

    for(i=0;(i<N-1) && (rem[i]!='1');i++);
        if(i<N-1)
            printf("\nError detected\n\n");
        else
            printf("\nNo error detected\n\n");
}

void crc(){

    for(i=0;i<N;i++)
        rem[i]=data[i];
    do{

        if(rem[0]=='1')
            XOR();

        for(j=0;j<N-1;j++)
            rem[j]=rem[j+1];

        rem[j]=data[i++];
    }
    while(i<=dlength+16);

}

int main()
{  int c=0;

    printf("\nEnter data to be transmitted: ");
    scanf("%s",data);
    printf("\n Enter the Divisor: ");
    scanf("%s",divisor);
    dlength=strlen(data);
    for(i=dlength;i<dlength+16;i++)
        data[i]='0';
    printf("\n");
    printf("\n Data padded with n-1 zeros : %s",data);
    printf("\n");
```

```
    crc();
    printf("\nCRC or Check value is : %s",rem);
    printf("\n rem strlen is : %d ", strlen(rem));
    for(i=dlength+13;i<dlength+16;i++)
      {   printf("\n %s",data);
          data[i]= rem[c++];
          }
    printf("\n");

    printf("\n Final data to be sent : %s",data);
    printf("\n\n");

    receiver();
        return 0;
}
```

Output :

```
Enter data to be transmitted: 1001101

 Enter the Divisor: 1011


 Data padded with n-1 zeros : 100110100000000000000000

CRC or Check value is : 111
 rem strlen is : 3
 100110100000000000000000
 100110100000000000000100
 100110100000000000000110

 Final data to be sent : 100110100000000000000111

Enter the received data: 100110100000000000000111


Data received: 100110100000000000000111
No error detected
```

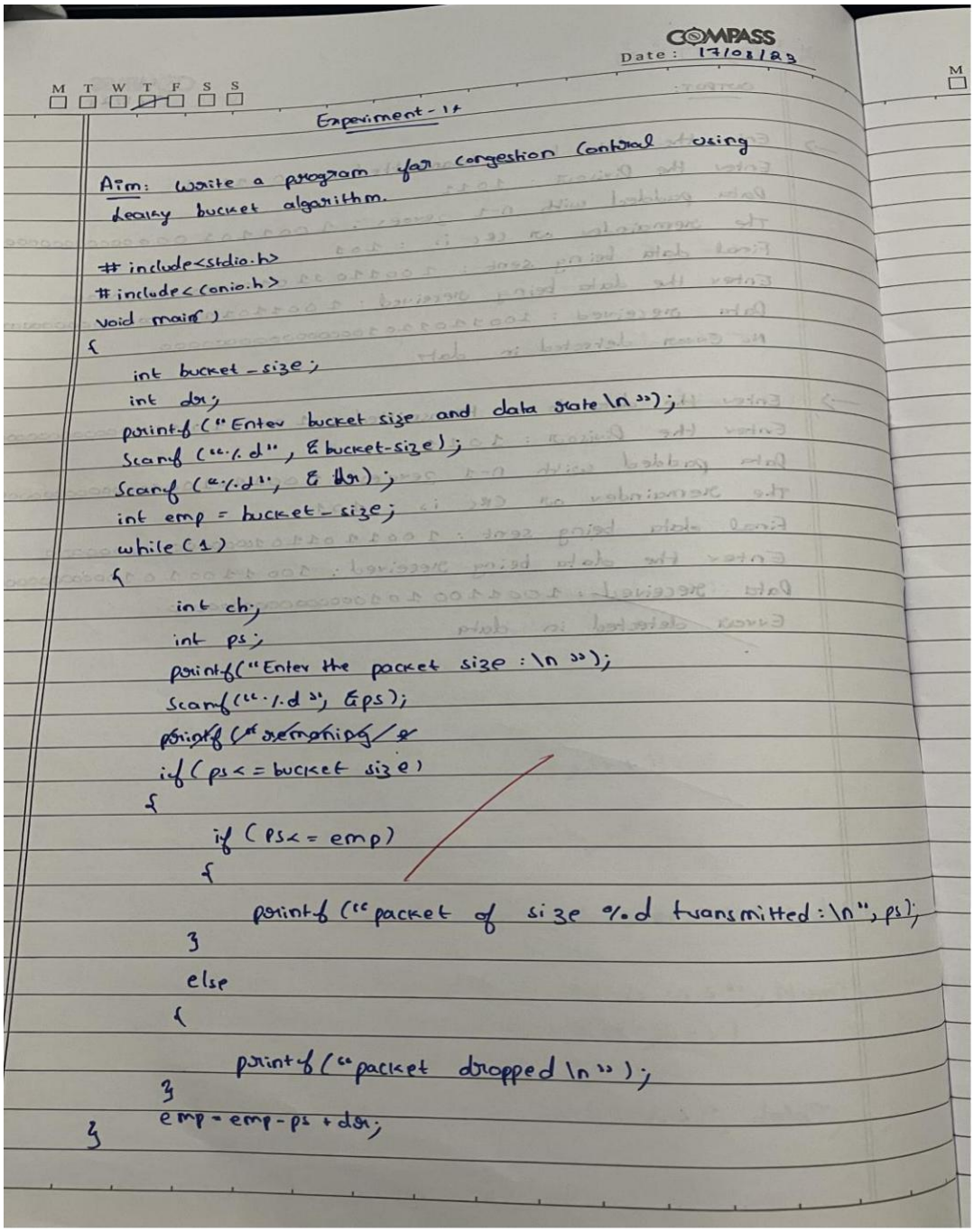
```
Enter data to be transmitted: 101101

 Enter the Divisor: 1011


 Data padded with n-1 zeros : 10110100000000000000

CRC or Check value is : 100
 rem strlen is : 3
 10110100000000000000
 10110100000000000100
 10110100000000000100

 Final data to be sent : 10110100000000000100

Enter the received data: 10110100000000000000


Data received: 10110100000000000000
Error detected
```

LAB PROGRAM – 14

Aim :

Write a program for congestion control using Leaky bucket algorithm.

Procedure :



Experiment - 14

Aim: Write a program for congestion control using leaky bucket algorithm.

```
#include<stdio.h>
#include<conio.h>
void main()
{
  int bucket_size;
  int dr;
  printf("Enter bucket size and data rate \n");
  Scanf("%d", &bucket_size);
  Scanf("%d", &dr);
  int emp = bucket_size;
  while(1)
  {
    int ch;
    int ps;
    printf("Enter the packet size : \n");
    Scanf("%d", &ps);
    printf("remaining %d
    if(ps<=bucket size)
    {
      if(ps<=emp)
      {
        printf("packet of size %d transmitted: \n", ps);
      }
      else
      {
        printf("packet dropped \n");
      }
      emp = emp - ps + dr;
    }
```

M T W T F S S

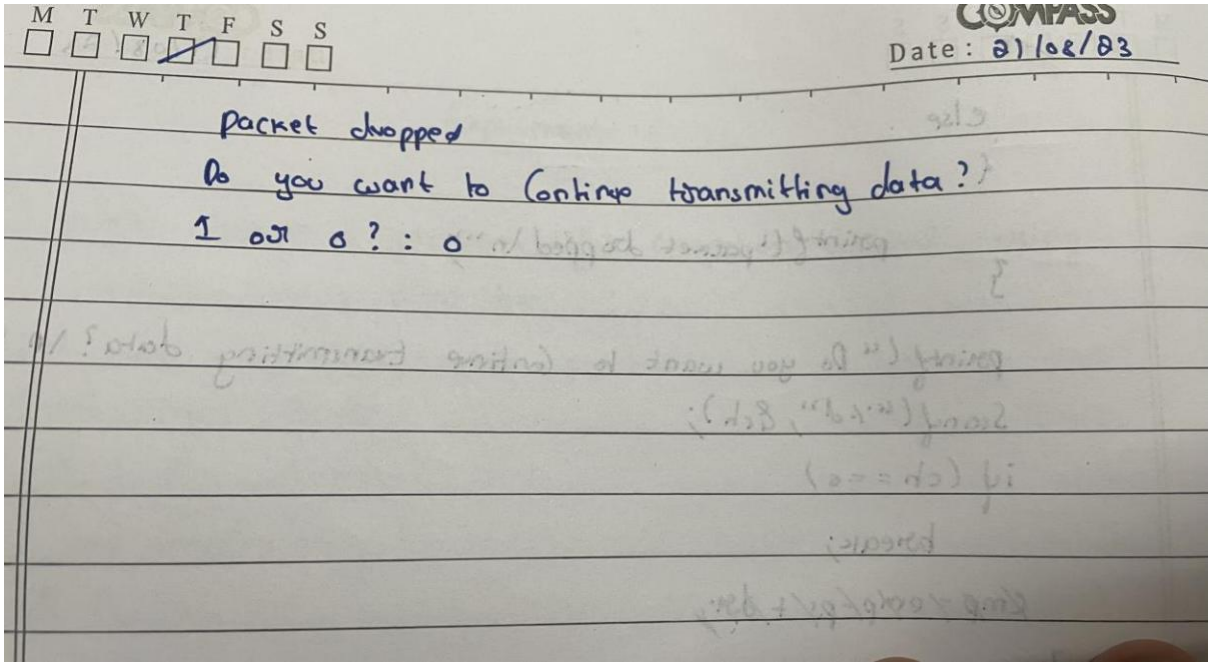
COMPASS

Date : 17/08/23

```
        else
        {
            printf("packet dropped \n");
        }
        printf("Do you want to continue transmitting data? \n 1 or 0 ? :");
        scanf("%d", &ch);
        if (ch == 0)
            break;
        ~~temp = temp - pt + dr;~~
    }
}
```

OUTPUT:

```
Enter bucket size and data rate
 5000
 200
Enter the packet size:
 6000
packet dropped
Do you want to continue transmitting data?
 1 or 0 ? : 1
Enter the packet size :
 3000
packet of size 3000 transmitted :
Do you want to continue transmitting data?
 1 or 0 ? : 1
Enter the packet size :
 2000
packet of size 2000 transmitted :
Do you want to continue transmitting data?
1 or 0 ? : 1
Enter the packet size :
 1000
```



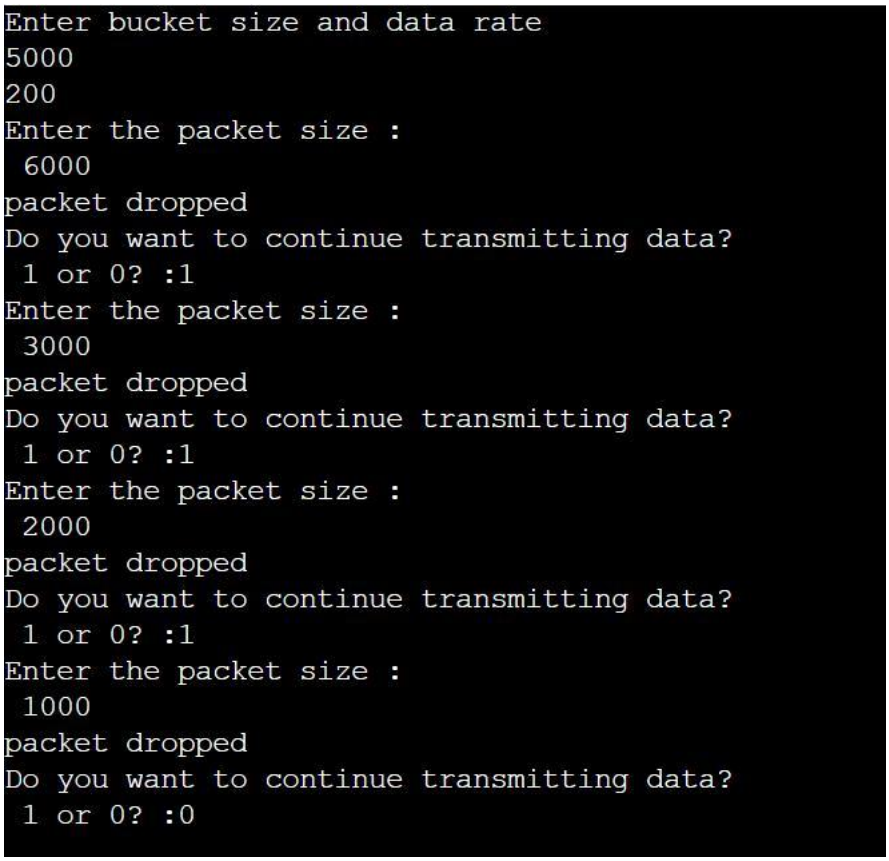
Code :

```
#include<stdio.h>
#include<conio.h>
void main()
{
    int bucket_size;
    int dr;
    printf("Enter bucket size and data
    rate\n"); scanf("%d",&bucket_size);
    scanf("%d",&dr);
    int emp = bucket_size;
    while(1)
    {
        int ch;
```

```
int ps;
printf("Enter the packet size :\n ");
scanf("%d",&ps);
if(ps<=bucket_size)
{
   if(ps<=emp)
   {
      printf("packet of size %d transmitted :\n",ps);
   }
   else
   {
      printf("packet dropped\n");
   }

}
else
{
   printf("packet dropped\n");
}
printf("Do you want to continue transmitting data?\n 1 or 0? :");
scanf("%d",&ch);
if(ch==0)
{
```

```
            break;
        }
        emp =emp-ps+dr;
    }
}
```

Output :



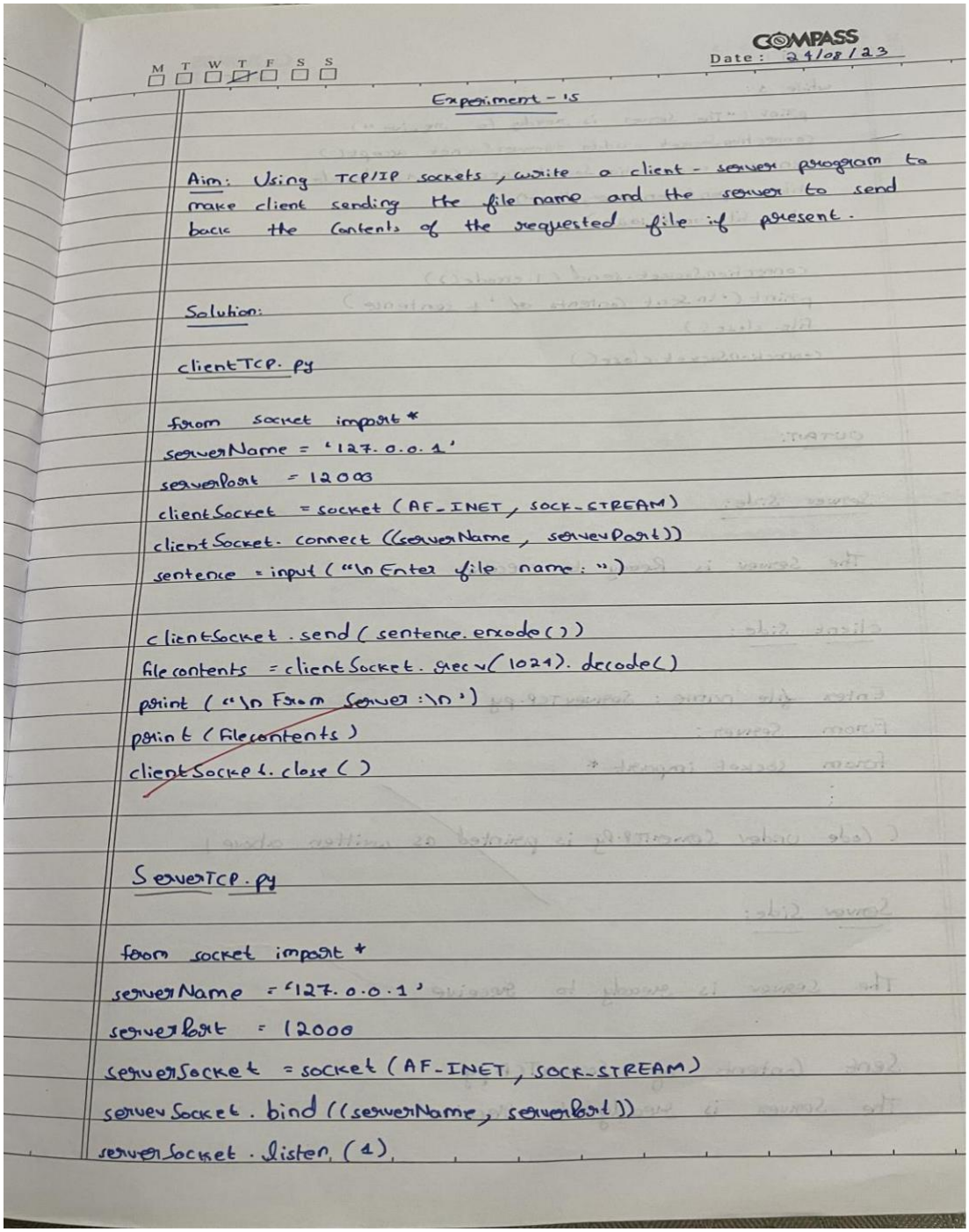
```
Enter bucket size and data rate
5000
200
Enter the packet size :
 6000
packet dropped
Do you want to continue transmitting data?
 1 or 0? :1
Enter the packet size :
 3000
packet dropped
Do you want to continue transmitting data?
 1 or 0? :1
Enter the packet size :
 2000
packet dropped
Do you want to continue transmitting data?
 1 or 0? :1
Enter the packet size :
 1000
packet dropped
Do you want to continue transmitting data?
 1 or 0? :0
```

LAB PROGRAM – 15

Aim :

Using TCP/IP 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.

Procedure :



Experiment - 15

Aim: Using TCP/IP 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.

Solution:

clientTCP.py

```
from socket import *
serverName = '127.0.0.1'
serverPort = 12003
clientSocket = socket(AF_INET, SOCK_STREAM)
clientSocket.connect((serverName, serverPort))
sentence = input("\nEnter file name: ")

clientSocket.send(sentence.encode())
fileContents = clientSocket.recv(1024).decode()
print("\nFrom Server:\n")
print(fileContents)
clientSocket.close()
```

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)
```

COMPASS
Date : 24/08/23

M T W T F S S

```
while 1:
print ("The Server is ready to receive")
connectionSocket, addr = serverSocket.accept()
sentence = connectionSocket.recv(1024).decode()
file = open(sentence, "r")
l = file.read(1024)

connectionSocket.send(l.encode())
print('\n Sent Contents of ' + sentence)
file.close()
connectionSocket.close()
```

OUTPUT:

Server Side:

The Server is Ready to receive

client Side:

Enter file name : ServerTCP.py
From Server :
from socket import *
:

(Code under ServerTCP.py is printed as written above)

Server Side:

The Server is ready to receive

Sent Contents of ServerTCP.py
The Server is ready to receive

Code :

ClientTCP.py

```
from socket import *
serverName = "127.0.0.1"
serverPort = 12000
clientSocket=socket(AF_INET , SOCK_STREAM)
clientSocket.connect((serverName , serverPort))
sentence = input("\nEnter File Name  : ")
clientSocket.send(sentence.encode())
filecontents = clientSocket.recv(1024).decode()
print("\nFrom Server:\n")
print(filecontents)
clientSocket.close()
```

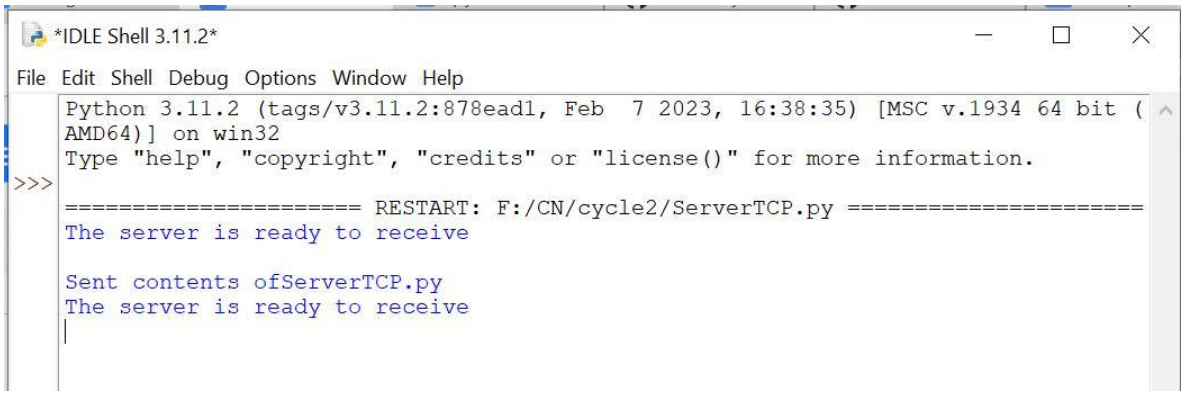
ServerTCP.py

```
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("Server is ready to recieve")
connectionSocket , address = serverSocket.accept()
sentence = connectionSocket.recv(1024).decode()
file = open(sentence,"r")
l=file.read(1024)
connectionSocket.send(l.encode())
print("\nsent contents of " + sentence)
file.close()
connectionSocket.close()
```

Output :

Server Instance :



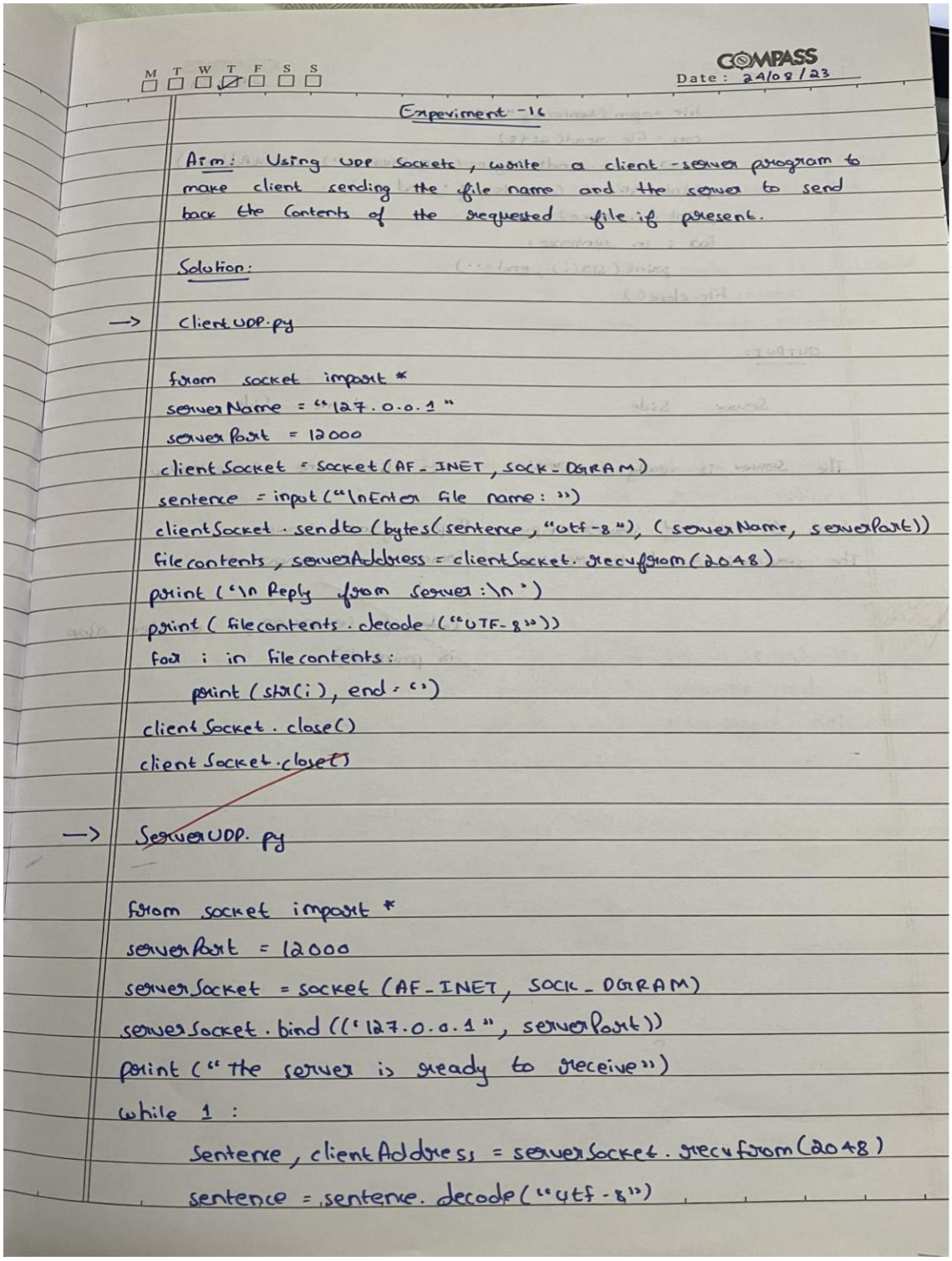
Client Instance :

LAB PROGRAM – 16

Aim :

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.

Procedure :



Experiment - 16

Aim: 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.

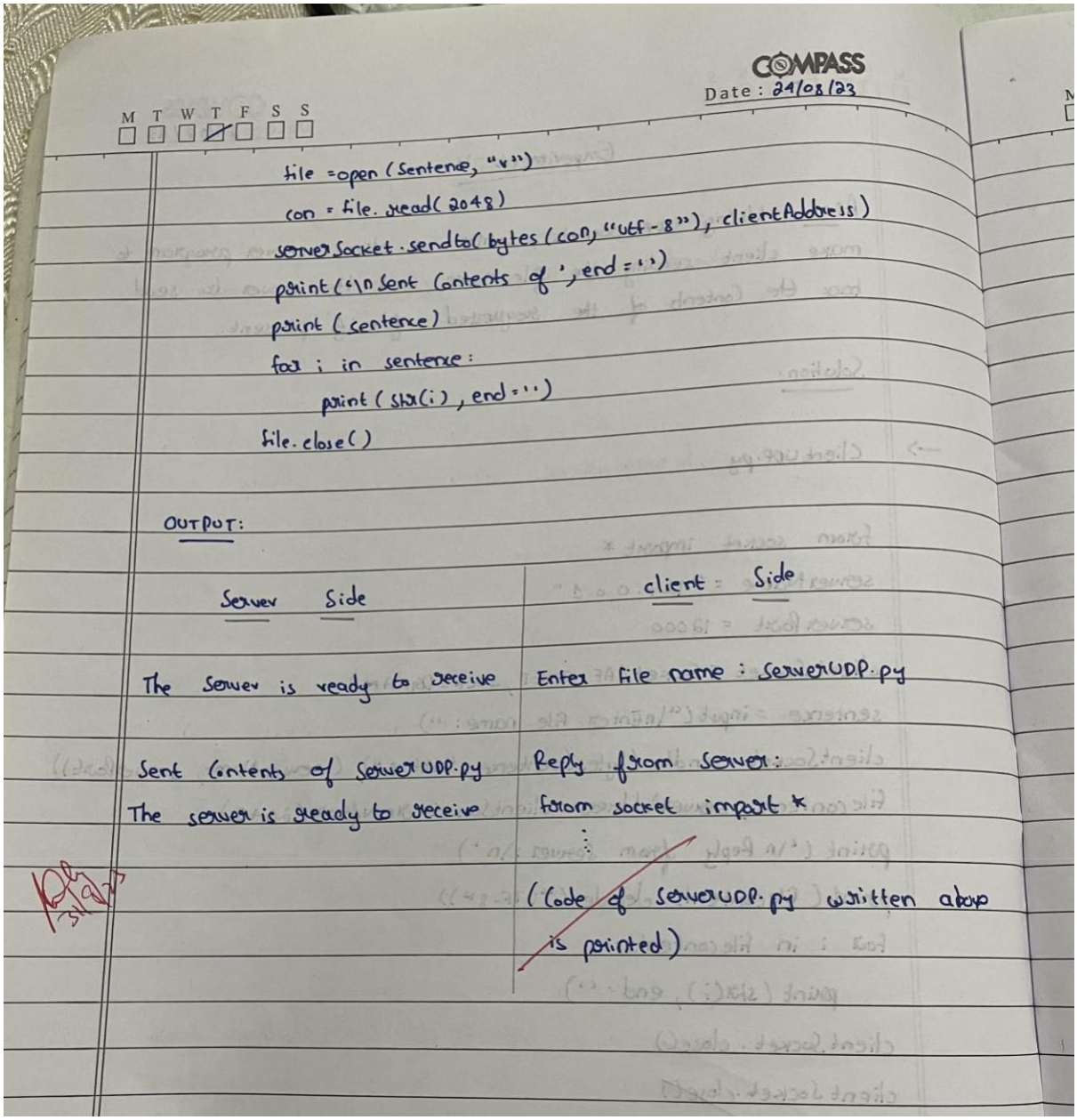
Solution:

→ ClientUDP.py

```
from socket import *
serverName = "127.0.0.1"
serverPort = 12000
clientSocket = socket(AF_INET, SOCK_DGRAM)
sentence = input("\nEnter file name: ")
clientSocket.sendto(bytes(sentence, "utf-8"), (serverName, serverPort))
fileContents, serverAddress = clientSocket.recvfrom(2048)
print('\n Reply from Server:\n')
print(fileContents.decode("UTF-8"))
for i in fileContents:
    print(str(i), end='')
clientSocket.close()
clientSocket.close()
```

→ ServerUDP.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, clientAddress = serverSocket.recvfrom(2048)
    sentence = sentence.decode("utf-8")
```



COMPASS
Date : 24/08/23
M T W T F S S

```
file =open (Sentence, "r")
con = file. read( 2048)
serverSocket . sendto( bytes (con, "utf - 8"), clientAddress )
print ('\n Sent Contents of ', end = ' ')
print ( sentence)
for i in sentence :
    print ( str(i) , end = '')
file. close()
```

OUTPUT:

Server Side	client Side
The Server is ready to receive	Enter file name : serverUDP.py
Sent Contents of serverUDP.py The server is ready to receive	Reply from Server: foram socket import * : (Code of serverUDP.py written above is printed)

Code :

ServerUDP.py

```
from socket import *
serverPort = 12000
serverSocket = socket(AF_INET , SOCK_DGRAM)
serverSocket.bind(("127.0.0.1",serverPort))
```

```
print("The server is read to recieve")
while 1 :
    sentence,clientAddress=serverSocket.recvfrom(2048)
    sentence=sentence.decode("utf-8")
    file=open(sentence,"r")
    con=file.read(2048)
    serverSocket.sendto(bytes(con,"utf-8"),clientAddress)
    print("\n Sent contents of " , end= "")
    print(sentence)
    for i in sentence :
        print(str(i),end="")
    file.close()
```

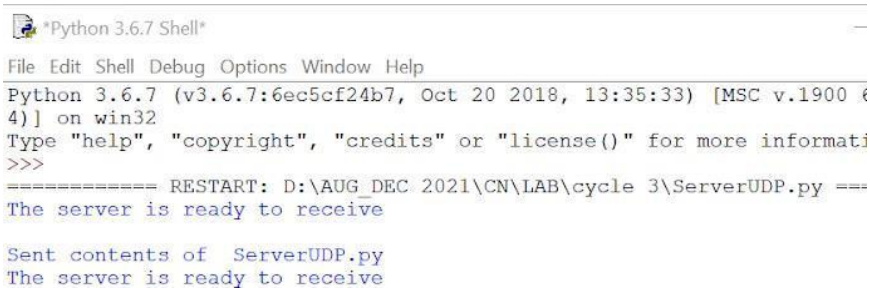
ClientUDP.py

```
from socket import *
serverName = "127.0.0.1"
serverPort = 12000
clientSocket = socket(AF_INET,SOCK_DGRAM)
sentence = input("\nEnter File Name : ")
clientSocket.sendto(bytes(sentence,"utf-
8"),(serverName,serverPort))
filecontents,serverAddress=clientSocket.recvfrom(2048)
print("\nReply from servere:\n")
```

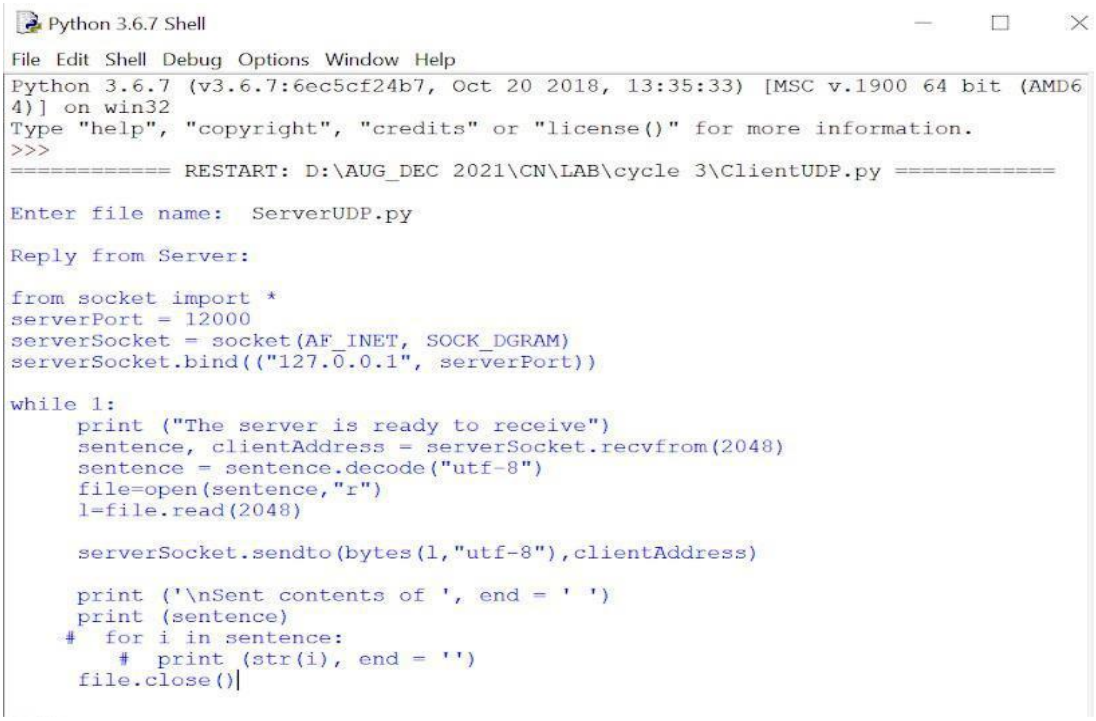
```
print(filecontents.decode("utf-8"))
for i in filecontents:
    print(str(i),end="")
clientSocket.close()
clientSocket.close()
```

Output :

Server Instance :



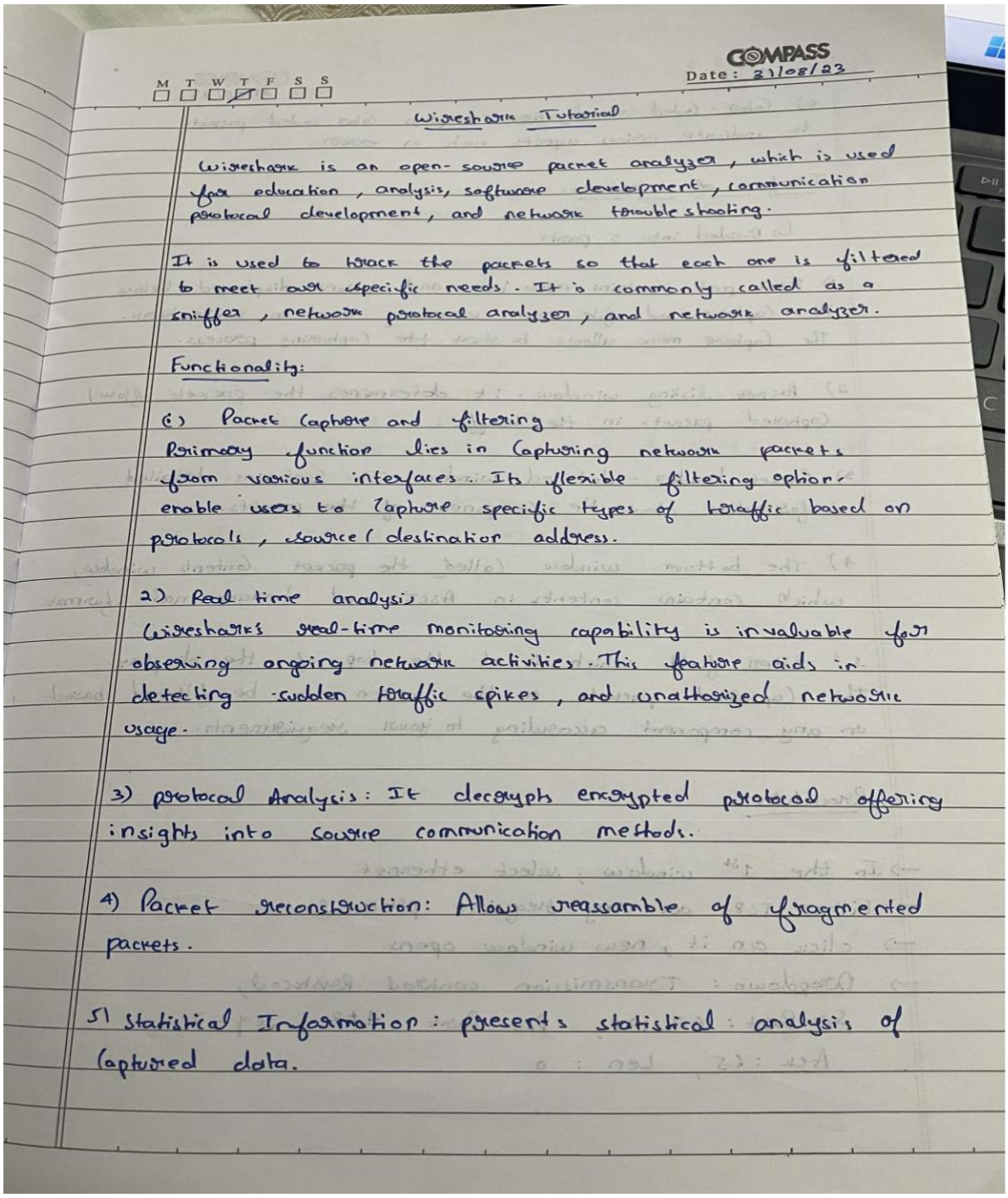
Client Instance :



LAB PROGRAM – 17

Aim :
Tool Exploration –Wireshark

Procedure :



M T W T F S S

COMPASS
Date: 21/08/23

Wireshark Tutorial

Wireshark is an open-source packet analyzer, which is used for education, analysis, software development, communication protocol development, and network troubleshooting.

It is used to track the packets so that each one is filtered to meet our specific needs. It is commonly called as a sniffer, network protocol analyzer, and network analyzer.

Functionality:

(i) Packet capture and filtering
Primary function lies in capturing network packets from various interfaces. Its flexible filtering options enable users to capture specific types of traffic based on protocols, source/destination address.

2) Real time analysis
Wireshark's real-time monitoring capability is invaluable for observing ongoing network activities. This feature aids in detecting sudden traffic spikes, and unauthorized network usage.

3) protocol Analysis: It decrypts encrypted protocol offering insights into source communication methods.

4) Packet reconstruction: Allows reassamble of fragmented packets.

5) Statistical Information: presents statistical analysis of captured data.

COMPASS
Date : 31/08/23

M T W T F S S

c) Color-Coded visualization : employs color-coded packets to indicate various aspects such as error.

Interface of wireshark:

↳ Divided into 5 parts

1) First part contains menu bar and options displayed below it. Capture and file menu commonly used in wireshark. The Capture menu allows to start the capturing process.

2) Packet listing window. It determines the packet flow / captured packets in the traffic.

3) Packet header - detailed window. It contains detailed information about the components of the packets.

4) The bottom window called the packet contents window, which contain contents in ASCII and hexadecimal format.

5) filter field which is at the top of the display. The captured packets on the screen can be filtered based on any component according to your requirements.

Procedure:

→ In the 1st window, select ethernet
→ Filter TCP or any require protocol
→ click on it, new window opens
→ Dropdown : Transmission control Protocol,
Scr Port : 62 148, Dst Port : 443, seq : 2,
Ack : 65, Len : 0

M T W T F S S

COMPASS

Date: 31/08/23

→ This is available in the prev windows in the left split of screen.

→ clicking on dropdown of it, clicking on any of them highlights its counterpart in right split side of screen.

→ In Cmd, type > ipconfig

RESULT:

Windows IP configuration

Ethernet adapter Ethernet

connection-specific DNS suffix :

Link-local IPv6 Address fe80::be78:f609:ed25:e329%13

IPv4 Address 10.124.2.83

Subnet Mask 255.255.0.0

Default Gateway 10.127.0.11