VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"JnanaSangama", Belgaum -590014, Karnataka.

LAB REPORT

on

COMPUTER NETWORKS

Submitted by

JITENDAR EY(1BM21CS268)

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

JUNE-2023 to SEPTEMBER-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 **COMPUTER NETWORKS**" carried out by **JITENDAR EY(1BM21CS268),** who is a bonafide student of **B. M. S. College ofEngineering.** 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 year2023. The Lab report has been approved as it satisfies the academic requirements in respect of a **Computer Networks (22CS4PCCON)** work prescribed for the said degree.

Dr. Shyamala G
Assistant Professor
Department of CSE
BMSCE, Bengaluru
`

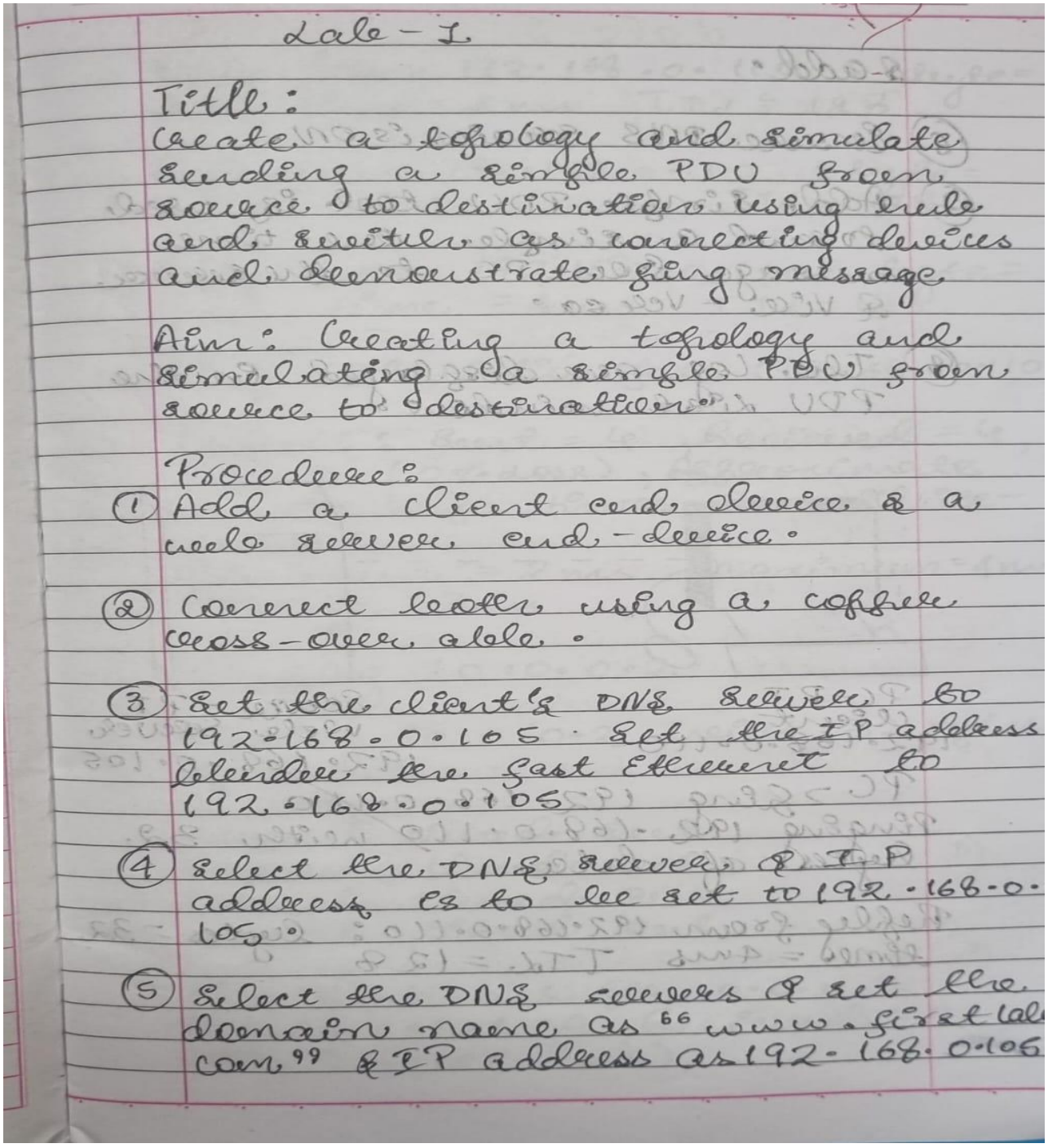
Dr. Jyothi S Nayak
Professor and Head
Department of CSE
BMSCE, Bengaluru

Sl. No.	Experiment Title
1.	Create a topology and simulate sending a simple PDU from source to destination using hub and switch asconnecting devices and demonstrate ping message.
2	Configure IP address to routers in packet tracer. Explore the following messages: ping responses, destinationunreachable, request timed out, reply
3	Configure default route, static route to the Router
4	Configure DHCP within a LAN and outside LAN. ,.
5	Configure Web Server, DNS within a LAN.
6	Configure RIP routing Protocol in Routers.
7	Configure OSPF routing protocol
8	To construct simple LAN and understand the concept and operation of Address Resolution Protocol (ARP)
9	Demonstrate the TTL/ Life of a Packet
10	To understand the operation of TELNET by accessing therouter in server room from a PC in IT office.
11	To construct a VLAN and make the PC's communicate among a VLAN.
12	To construct a WLAN and make the nodes communicate wirelessly.
	Cycle - 2
13	Write a program for error detecting code using CRC-CCITT(16-bits).
14	Write a program for congestion control using Leakybucket algorithm.
15	Using TCP/IP sockets, write a client-server program to make client sending the file name and the server to sendback the contents of the requested file if present.
16	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.

CN LAB 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

OBSERVATION



Lab - 1

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

Aim: Creating a topology and simulating a simple PDU from source to destination.

Procedure:

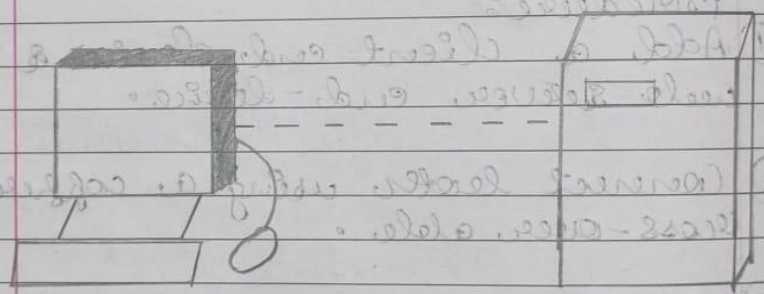
1. Add a client and device & a web server end-device.
2. Connect router using a coffee cross-over cable.
3. Set the client's DNS server to 192.168.0.105 Set the IP address devices for fast Ethernet to 192.168.0.105
4. Select the DNS server & IP address as to be set to 192.168.0.105.
5. Select the DNS servers & set the domain name as "www.firstlab.com" & IP address as 192.168.0.105

& add.

(6) Ensure DNS service is ON.

(7) Add simple PDU tool is used to send a simple one-time message from PC to the server & vice-versa.

(8) The log are displayed on the PDU list window.



PC-PT
client
192.168.0.110

Server-PT
Web Server
192.168.0.105

PC > ping 192.168.0.110

Pinging 192.168.0.110 with 32 bytes of data:

Reply from 192.168.0.110: bytes = 32 time = 4ms TTL = 128

Reply from 192.168.0.110 : bytes=
32 time=2ms TTL=128

Reply from 192.168.0.110 : bytes=
32 time=4ms TTL=128

Reply from 192.168.0.110 : bytes=
32 time=4ms TTL=128

Ping statistics for 192.168.0.110:

Packets: Sent = 4, Received = 4,
Loss = 0 (0% loss), Approximate
round trip times in milliseconds:
Minimum = 2ms Maximum = 4ms,
Average = 3ms

PC> ping 10.0.0.2

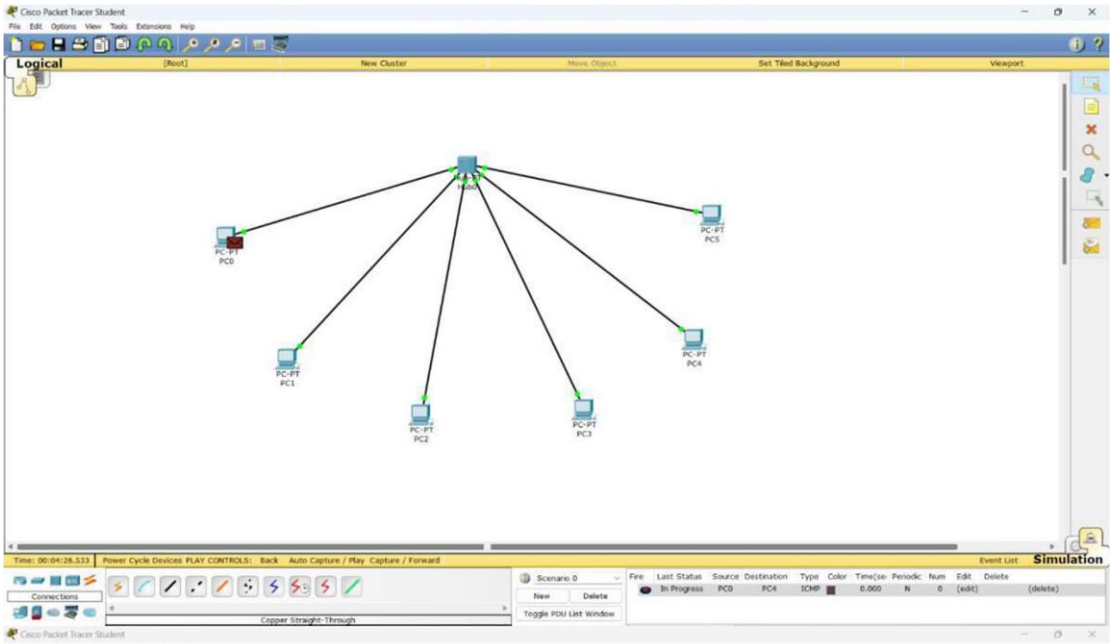
Pinging 10.0.0.2 with 32 bytes
of data:

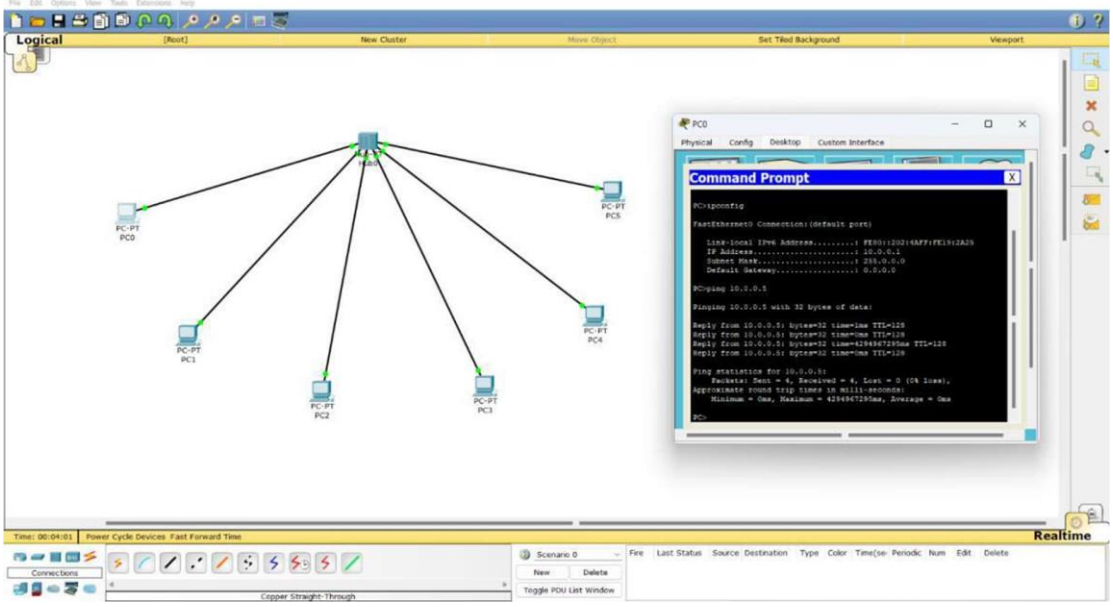
Request timed out
Request timed out
Request timed out
Request timed out

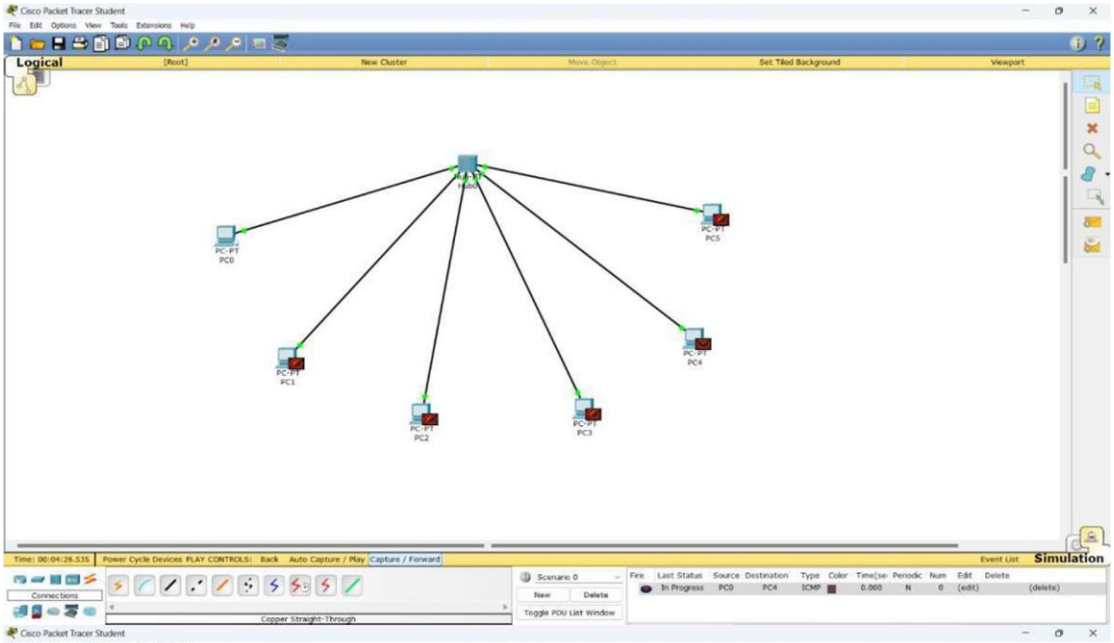
Ping statistics for 10.0.0.2
Packets: Sent = 4, Received = 0,
Lost = 4 (100% loss)

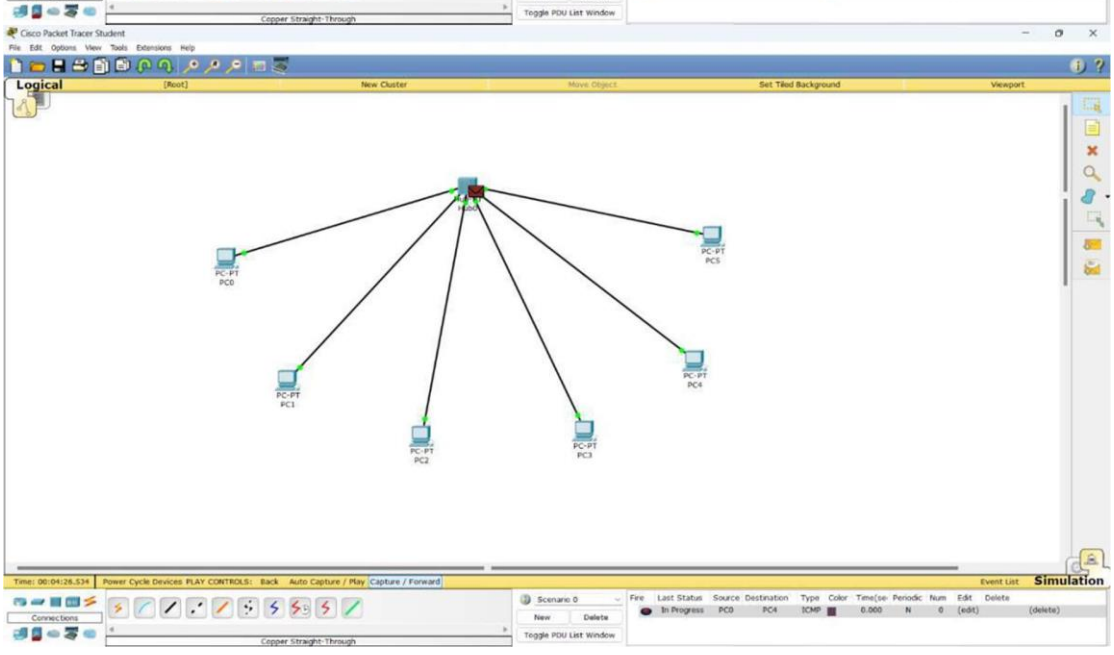
SCREENSHOTS

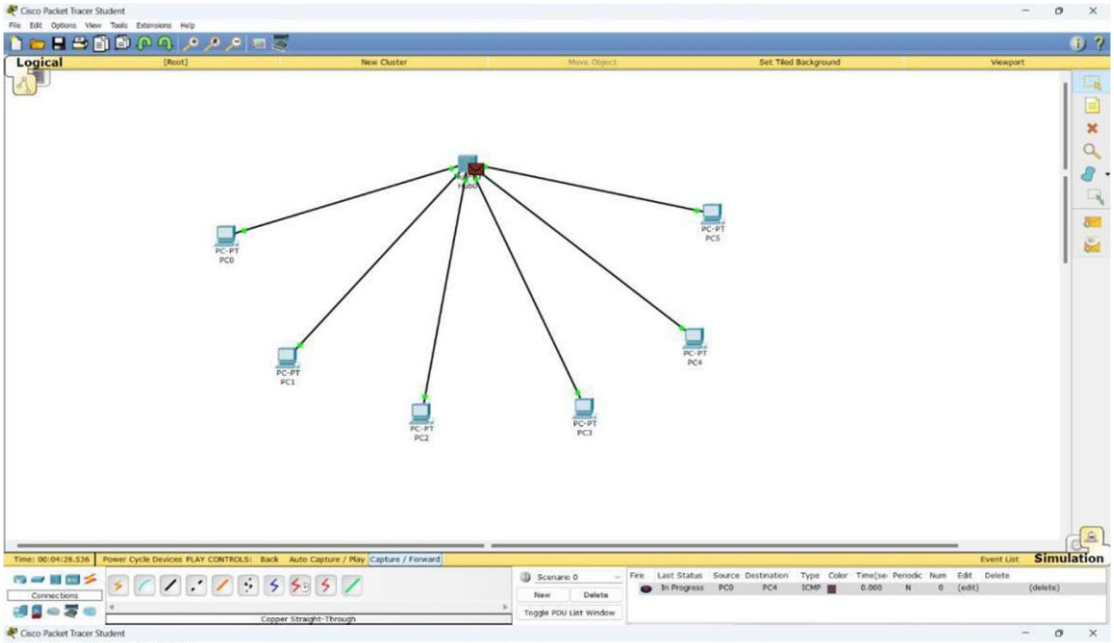
USING HUB:

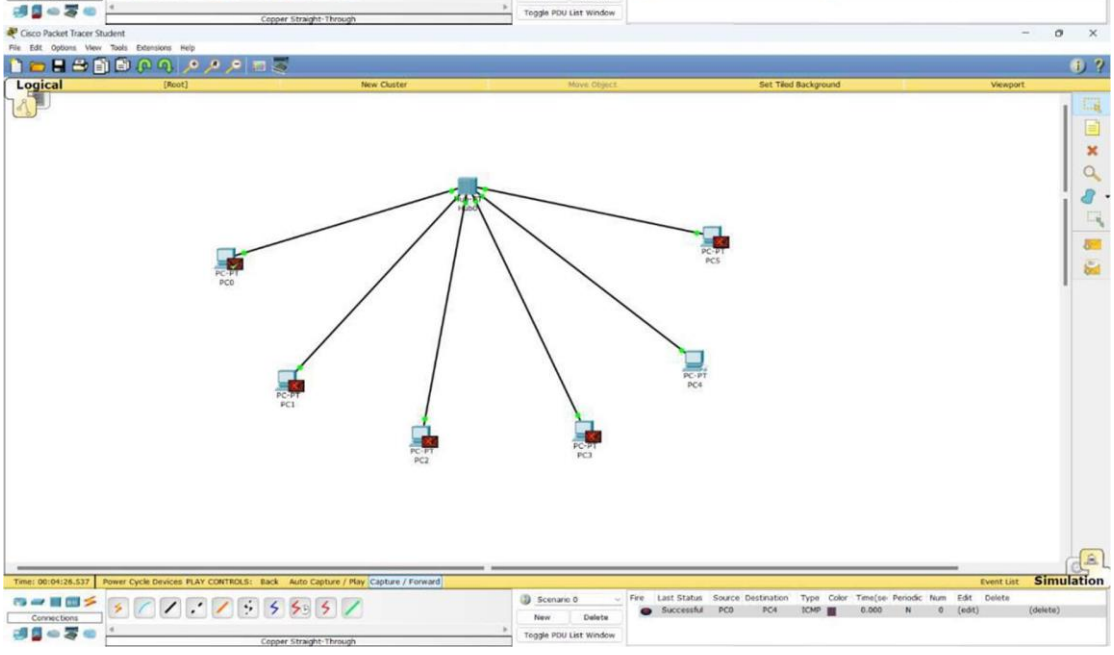




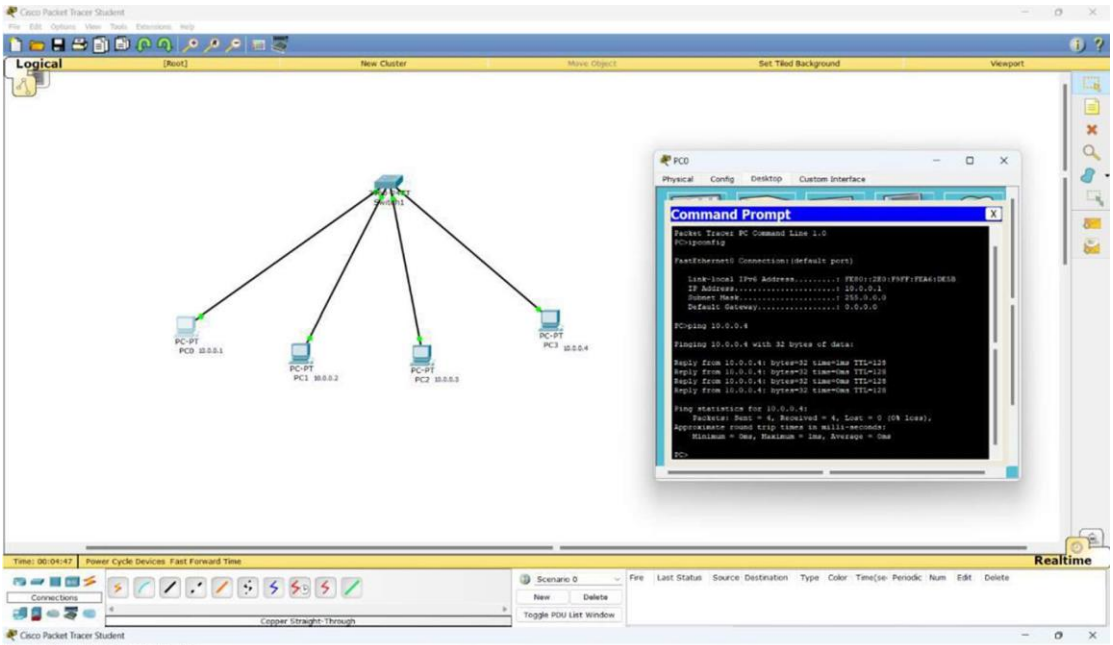

Cisco Packet Tracer Student
File Edit Options View Tools Extensions Help
Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
PC-PT PC0
PC-PT PC1
PC-PT PC2
PC-PT PC3
PC-PT PC4
PC-PT PC5
Time: 00:04:26.535
Power Cycle Devices PLAY CONTROLS: Back Auto Capture / Play Capture / Forward
Event List
Simulation
Connections
Copper Straight-Through
Scenario 0
New
Delete
Toggle PDU List Window
Fire Last Status Source Destination Type Color Time(se Periodic Num Edit Delete
In Progress PC0 PC4 ICMP 0.000 N 0 (edit) (delete)

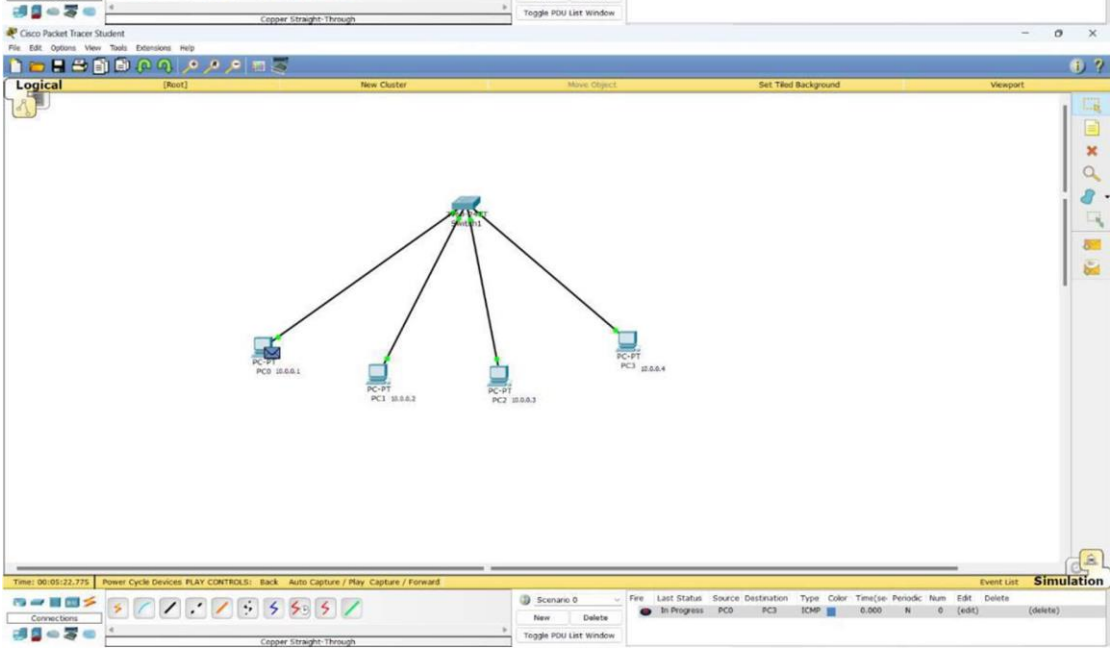

Cisco Packet Tracer Student
File Edit Options View Tools Extensions Help
Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
PC-PT PC0
PC-PT PC1
PC-PT PC2
PC-PT PC3
PC-PT PC4
PC-PT PC5
Time: 00:04:26.534
Power Cycle Devices PLAY CONTROLS: Back Auto Capture / Play Capture / Forward
Event List
Simulation
Connections
Copper Straight-Through
Scenario 0
New
Delete
Toggle PDU List Window
Fire Last Status Source Destination Type Color Time(se Periodic Num Edit Delete
In Progress PC0 PC4 ICMP 0.000 N 0 (edit) (delete)

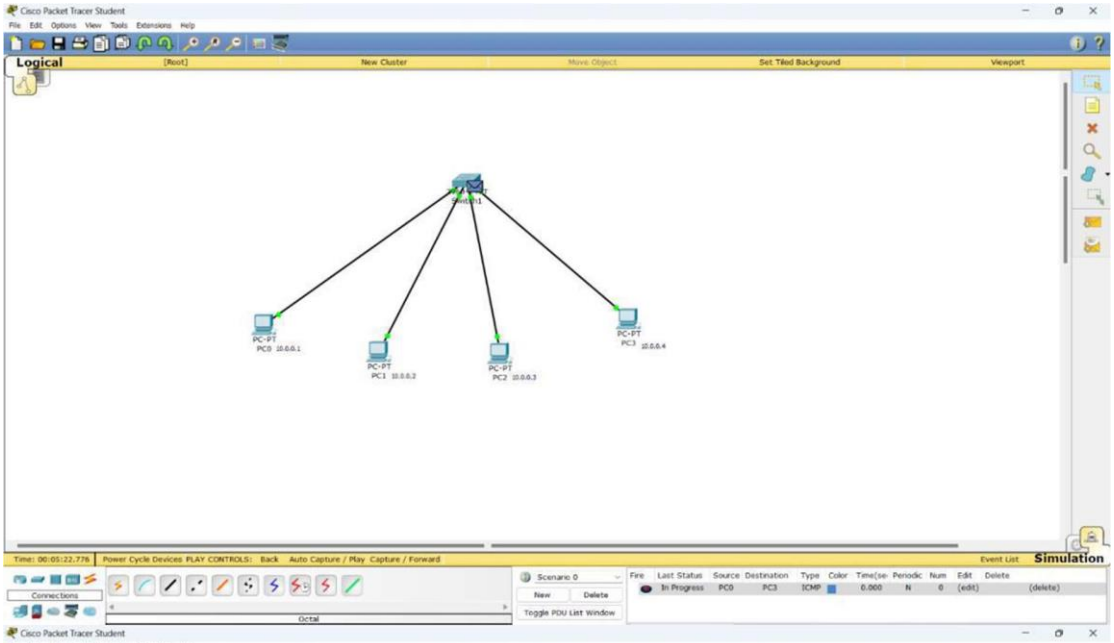

Cisco Packet Tracer Student
File Edit Options View Tools Extensions Help
Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
Hub0
PC-PT PC0
PC-PT PC1
PC-PT PC2
PC-PT PC3
PC-PT PC4
PC-PT PC5
Time: 00:04:26.536
Power Cycle Devices PLAY CONTROLS: Back Auto Capture / Play Capture / Forward
Event List
Simulation
Connections
Copper Straight-Through
Scenario 0
New
Delete
Toggle PDU List Window
Fire Last Status Source Destination Type Color Time(se Periodic Num Edit Delete
In Progress PC0 PC4 ICMP 0.000 N 0 (edit) (delete)

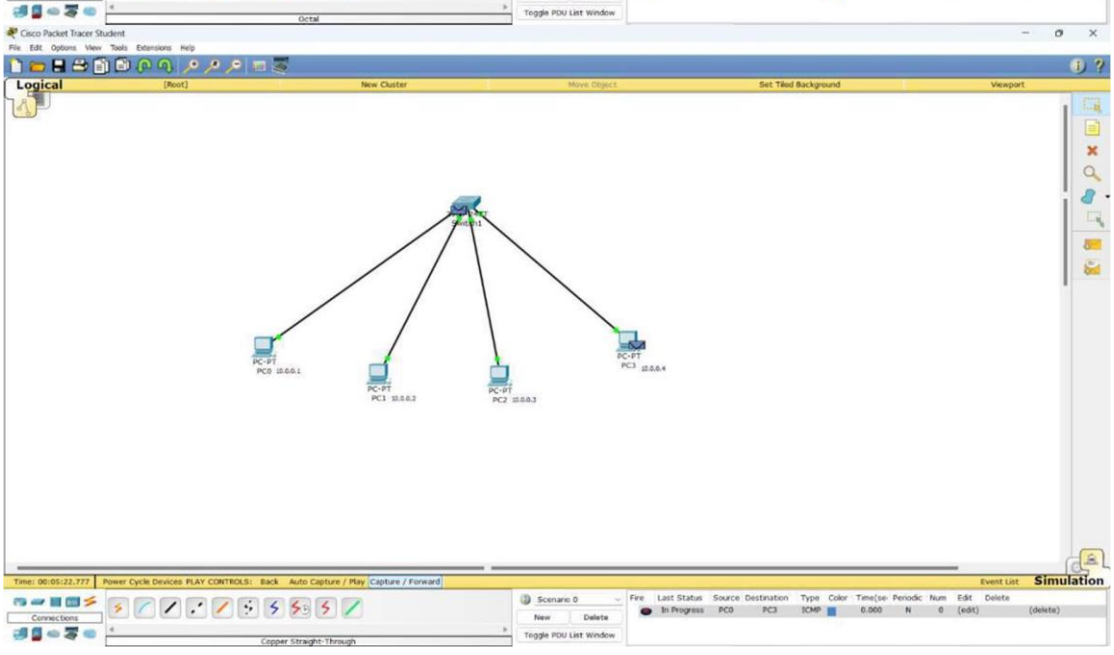

Cisco Packet Tracer Student
File Edit Options View Tools Extensions Help
Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
Hub0
PC-PT PC0
PC-PT PC1
PC-PT PC2
PC-PT PC3
PC-PT PC4
PC-PT PC5
Time: 00:04:26.537
Power Cycle Devices PLAY CONTROLS: Back Auto Capture / Play Capture / Forward
Event List
Simulation
Connections
Copper Straight-Through
Scenario 0
New
Delete
Toggle PDU List Window
Fire Last Status Source Destination Type Color Time(se Periodic Num Edit Delete
Successful PC0 PC4 ICMP 0.000 N 0 (edit) (delete)

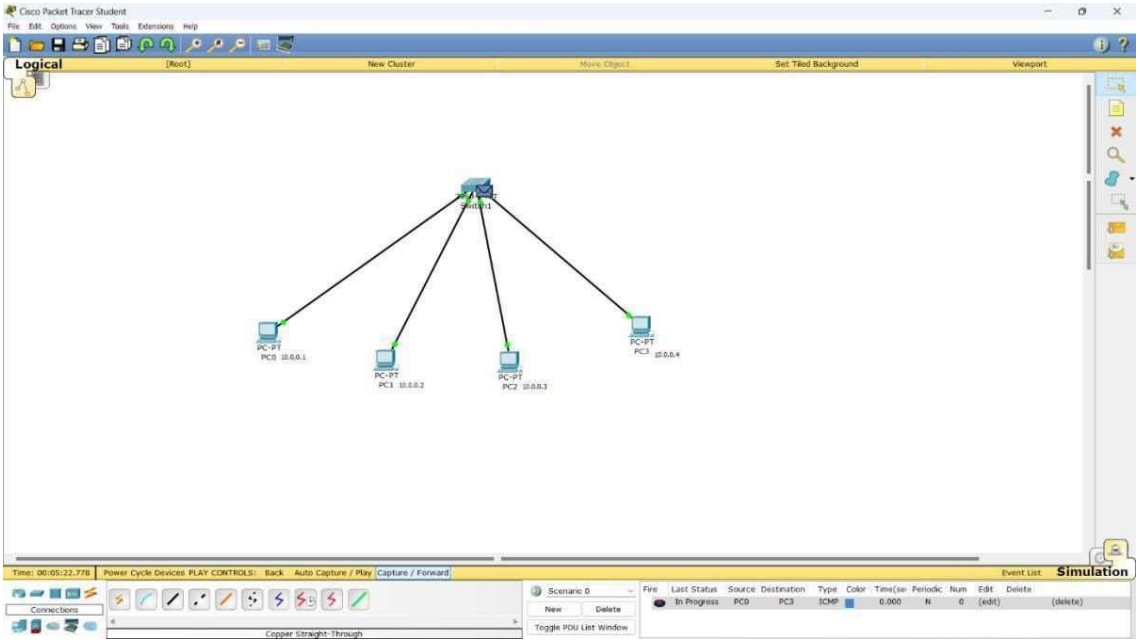
USING SWITCH:



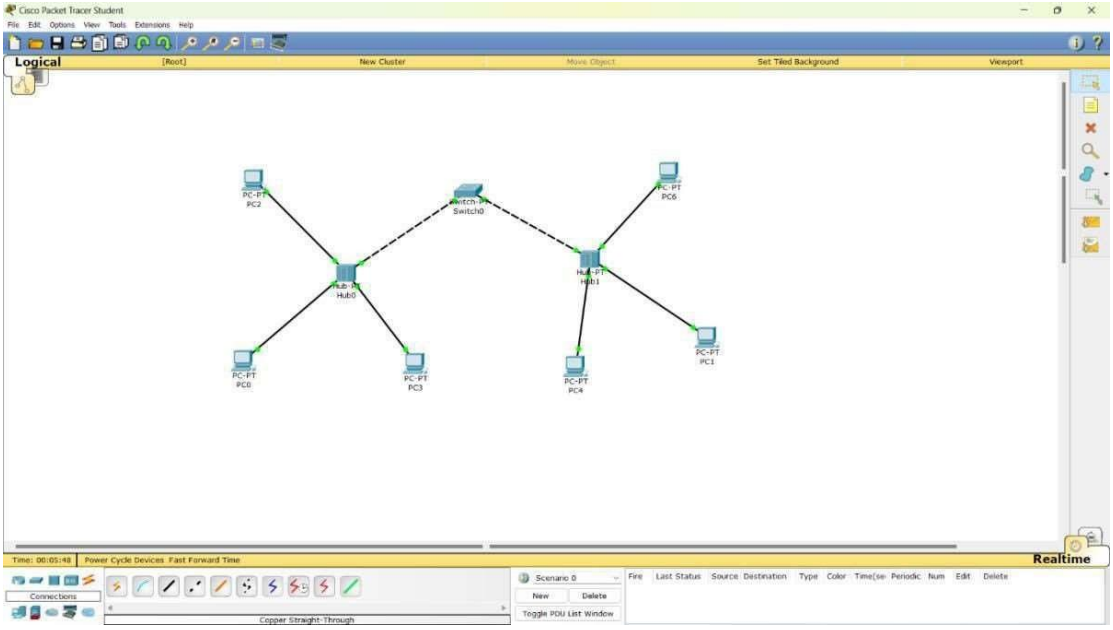


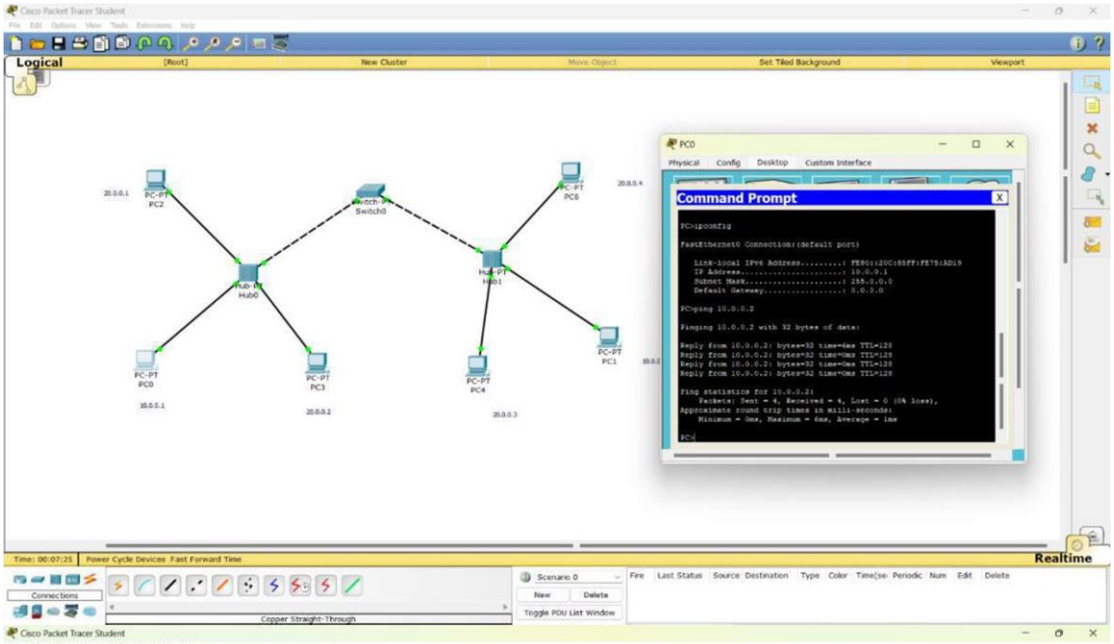

Cisco Packet Tracer Student
File Edit Options View Tools Extensions Help
Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
Switch1
PC-PT PC0 10.0.0.1
PC-PT PC1 10.0.0.2
PC-PT PC2 10.0.0.3
PC-PT PC3 10.0.0.4
Time: 00:05:22.776
Power Cycle Devices PLAY CONTROLS: Back Auto Capture / Play Capture / Forward
Event List
Simulation
Connections
Octal
Scenario 0
New
Delete
Toggle PDU List Window
Fire Last Status Source Destination Type Color Time(se Periodic Num Edit Delete
In Progress PC0 PC3 ICMP 0.000 N 0 (edit) (delete)

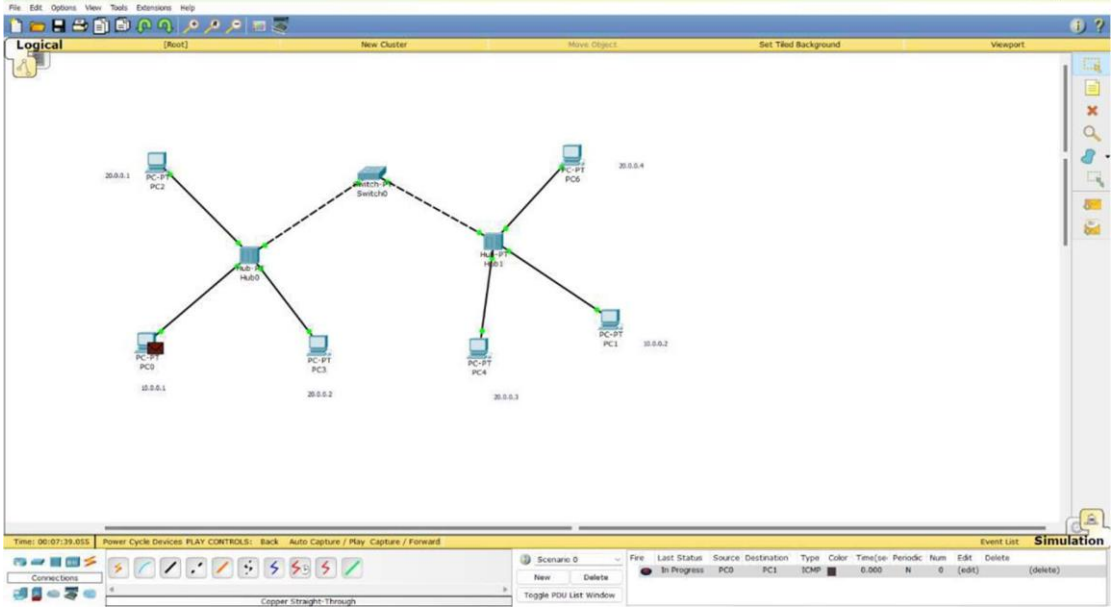

Cisco Packet Tracer Student
File Edit Options View Tools Extensions Help
Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
Switch1
PC-PT PC0 10.0.0.1
PC-PT PC1 10.0.0.2
PC-PT PC2 10.0.0.3
PC-PT PC3 10.0.0.4
Time: 00:05:22.777
Power Cycle Devices PLAY CONTROLS: Back Auto Capture / Play Capture / Forward
Event List
Simulation
Connections
Copper Straight-Through
Scenario 0
New
Delete
Toggle PDU List Window
Fire Last Status Source Destination Type Color Time(se Periodic Num Edit Delete
In Progress PC0 PC3 ICMP 0.000 N 0 (edit) (delete)

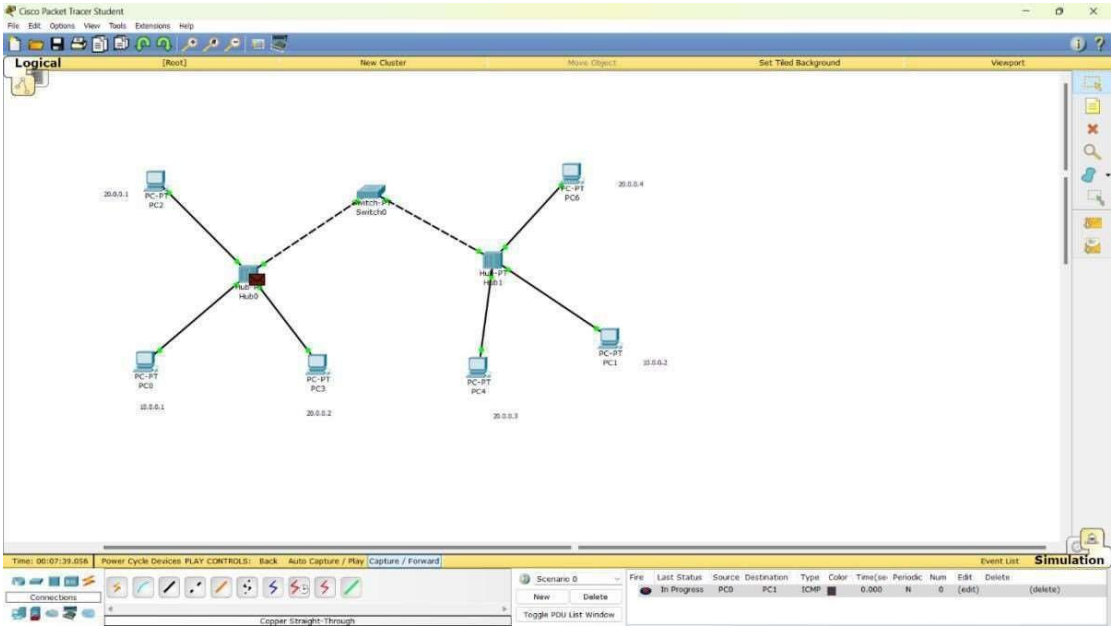


USING HUB AND SWITCH:




Cisco Packet Tracer Student
Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
PC0
Physical
Config
Desktop
Custom Interface
Command Prompt
PC>ipconfig
FastEthernet0 Connection:(default port)
Link-local IPv6 Address.....: FE80::20C:85FF:FE75:AD19
IP Address.....: 10.0.0.1
Subnet Mask.....: 255.0.0.0
Default Gateway.....: 0.0.0.0
PC>ping 10.0.0.2
Pinging 10.0.0.2 with 32 bytes of data:
Reply from 10.0.0.2: bytes=32 time=4ms 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, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 4ms, Average = 1ms
PC>
PC-PT PC2
Switch-PT Switch0
PC-PT PC6
Hub-PT Hub0
Hub-PT Hub1
PC-PT PC0
PC-PT PC3
PC-PT PC4
PC-PT PC1
Time: 00:07:25
Power Cycle Devices Fast Forward Time
Realtime
Connections
Copper Straight-Through
Scenario 0
New
Delete
Toggle PDU List Window
Fire
Last Status
Source
Destination
Type
Color
Time(se
Periodic
Num
Edit
Delete

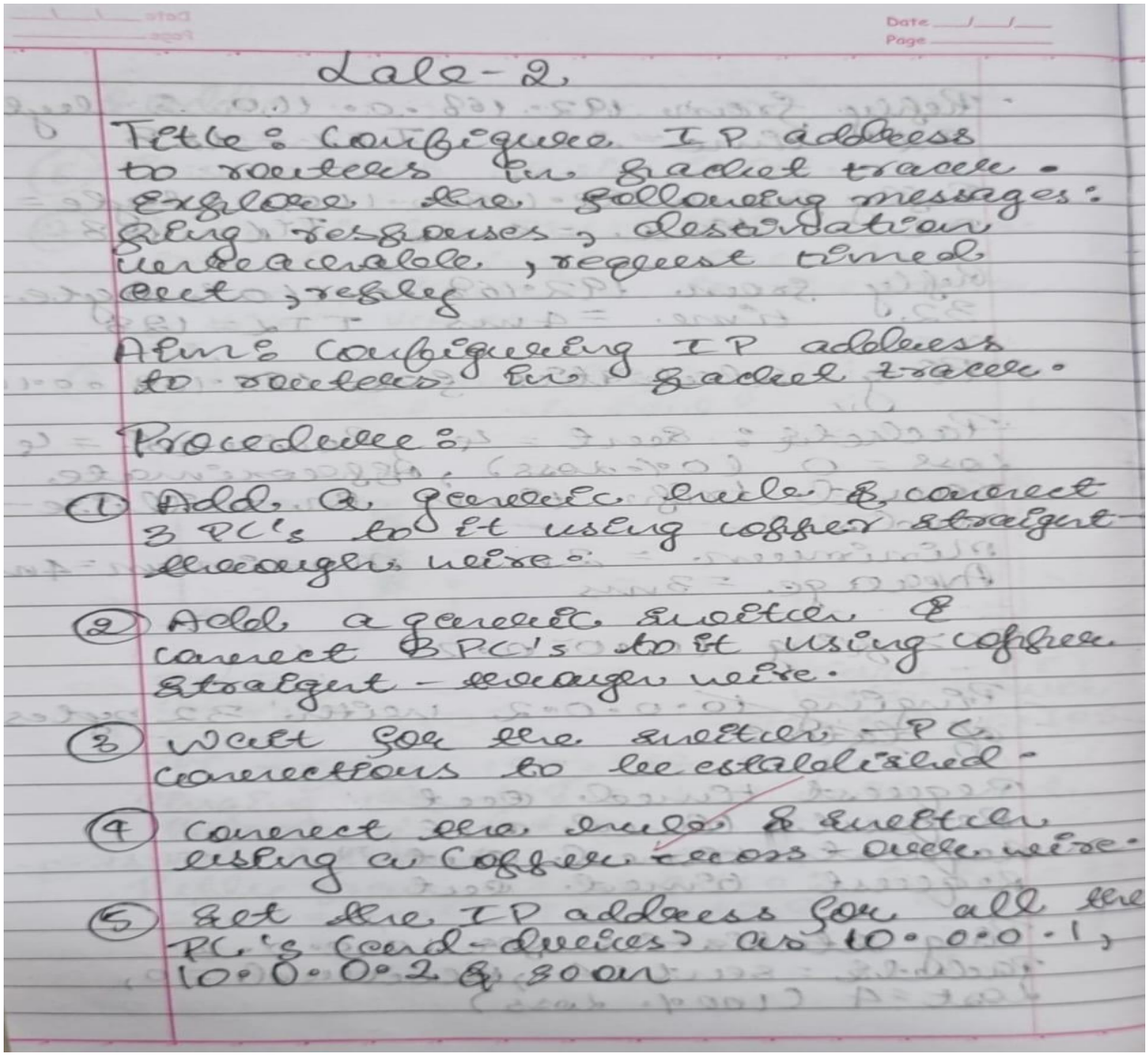

Cisco Packet Tracer Student
Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
PC-PT PC2
Switch-PT Switch0
PC-PT PC6
Hub-PT Hub0
Hub-PT Hub1
PC-PT PC0
PC-PT PC3
PC-PT PC4
PC-PT PC1
Time: 00:07:39.055
Power Cycle Devices PLAY CONTROLS: Back Auto Capture / Play Capture / Forward
Event List
Simulation
Connections
Copper Straight-Through
Scenario 0
New
Delete
Toggle PDU List Window
Fire
Last Status
Source
Destination
Type
Color
Time(se
Periodic
Num
Edit
Delete
In Progress
PC0
PC1
ICMP
0.000
N
0
(edit)
(delete)


Cisco Packet Tracer Student
File Edit Options View Tools Extensions Help
Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
PC-PT PC2
Switch-PT Switch0
PC-PT PC6
Hub-PT Hub0
Hub-PT Hub1
PC-PT PC0
PC-PT PC3
PC-PT PC4
PC-PT PC1
Time: 00:07:39.056
Power Cycle Devices PLAY CONTROLS: Back Auto Capture / Play Capture / Forward
Event List
Simulation
Connections
Copper Straight-Through
Scenario 0
New
Delete
Toggle PDU List Window
Fire Last Status Source Destination Type Color Time(se Periodic Num Edit Delete
In Progress PC0 PC1 ICMP 0.000 N 0 (edit) (delete)

CN LAB 2

AIM: Configure IP address to routers (one and three) in packet tracer. Explore the following messages: ping responses, destination unreachable, request timed out, reply**.**

OBSERVATION



Lab-2.

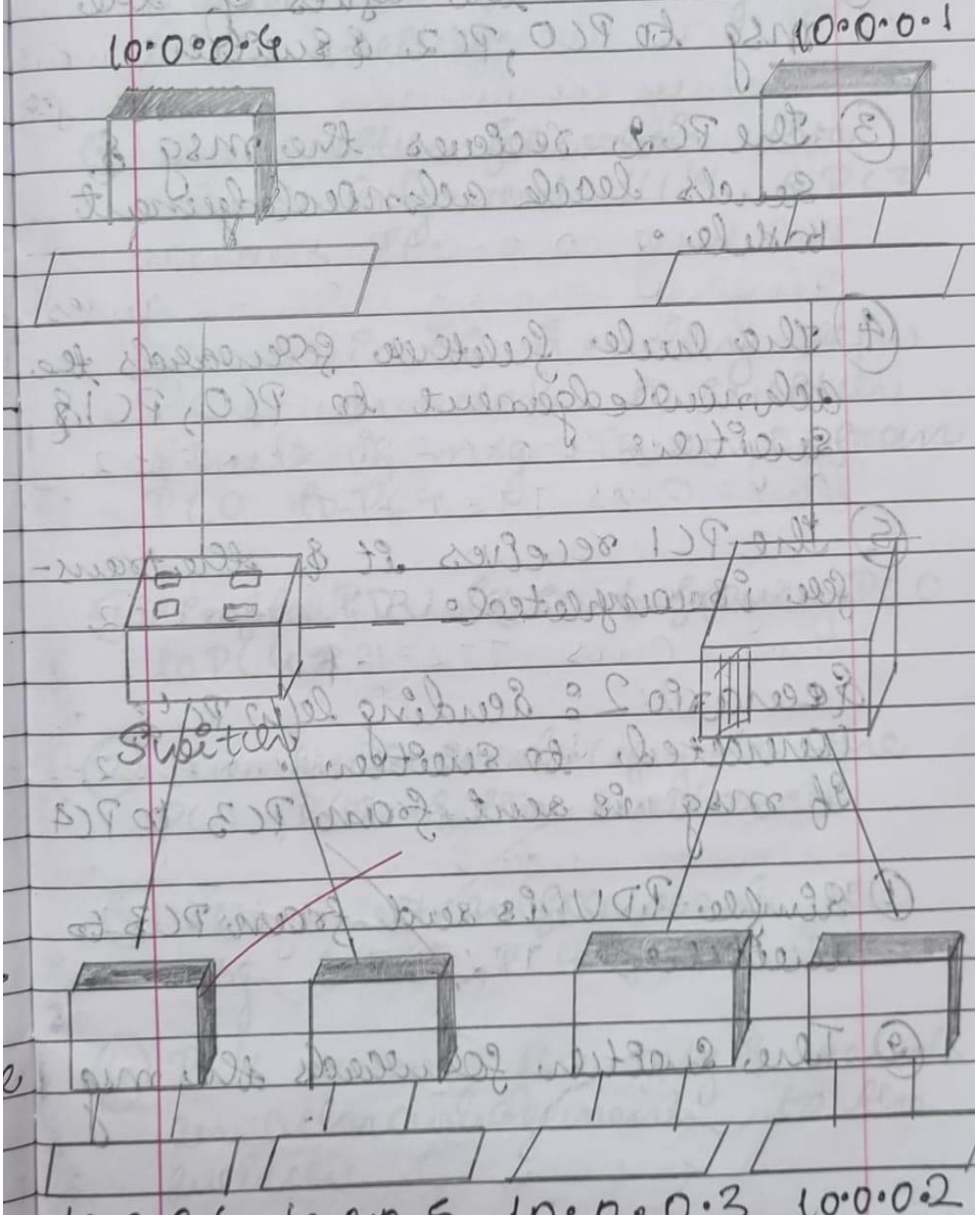
Title: Configure IP address to routers in packet tracer. Explore the following messages: ping responses, destination unreachable, request timed out, reply

Aim: Configuring IP address to routers in packet tracer.

Procedure:

1. Add a generic router & connect 3 PC's to it using copper straight through wire.
2. Add a general switch & connect 3 PC's to it using copper straight - through wire.
3. Wait for the switch - PC connections to be established.
4. Connect the router & switch using a copper cross - over wire.
5. Set the IP address for all the PC's (end-devices) as 10.0.0.1, 10.0.0.2 & so on

⑥ Send a simple PDU message from one PC to another & wait for the simulation to run after clicking Auto capture.



to PC5, PC5 & the hub.

(3) The hub forwards it to the PC0, PC1 & PC2.

(4) The PC4 accepts the message & sends an acknowledgment to the switch.

(5) The switch forwards the acknowledgment to PC3 & PC3 receives it.

Scenario 5: Sending msg PC's connected to switch & hub if msg is sent from PC0 to PC4.

(1) Simple PDU is sent from PC0 to PC4.

(2) The hub sends copy to the PC1, PC2 & the switch.

(3) The switch forwards the msg to PC3, PC4 & PC5.

(4) PC4 recieves the msg & sends an acknowledgement to the switch.

(5) The switch forwards the acknowledgement to the PC3 & PC5 & the hub.

(6) Hub sends copy of the acknowledgement to PC0, PC1, & UV

(7) PC0 receives the acknowledgement thereby completing the transfer of msg.

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, Recieved = 4, Lost = 0 (0% loss). Approximate round trip time in milli-seconds:

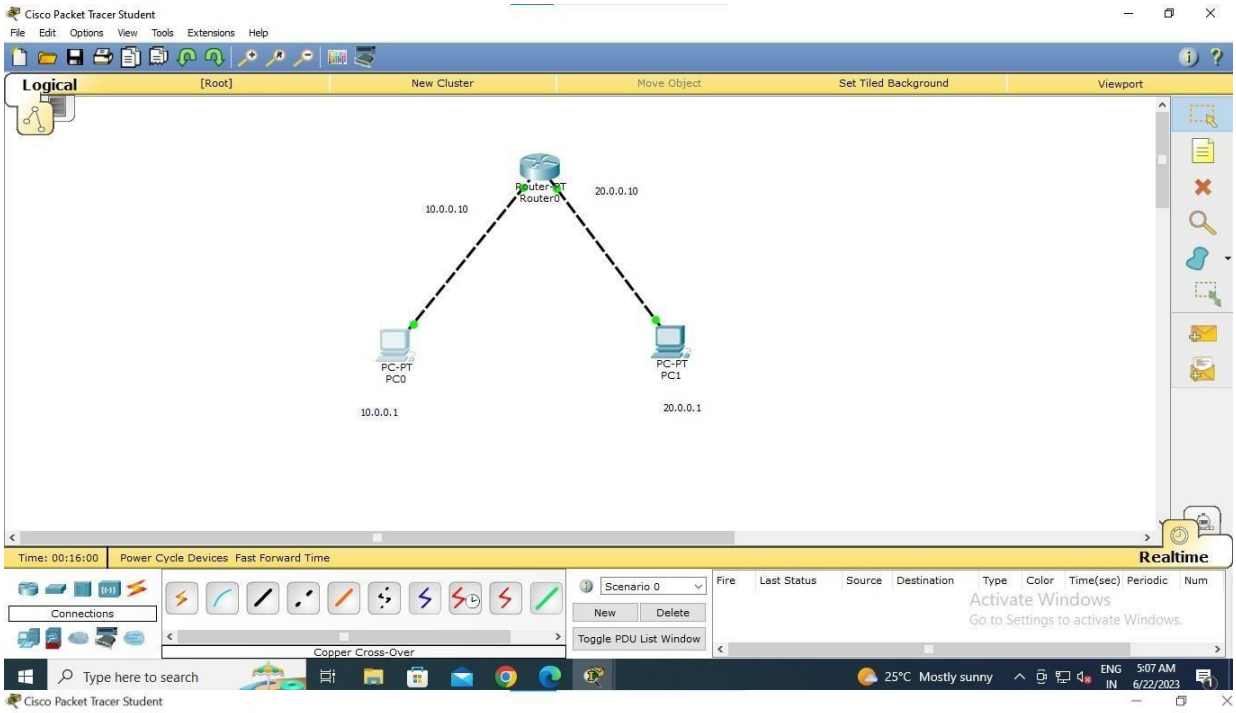
Minimum = 0ms, Maximum = 0 ms, Average = 0ms

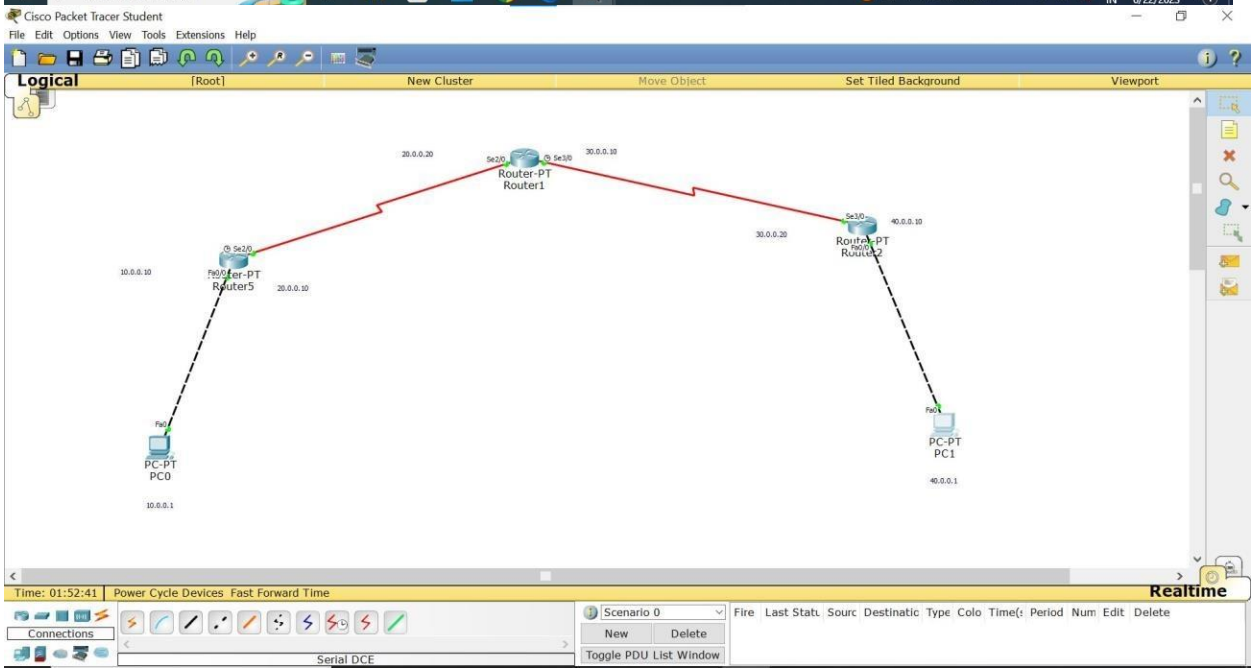
Command Prompt

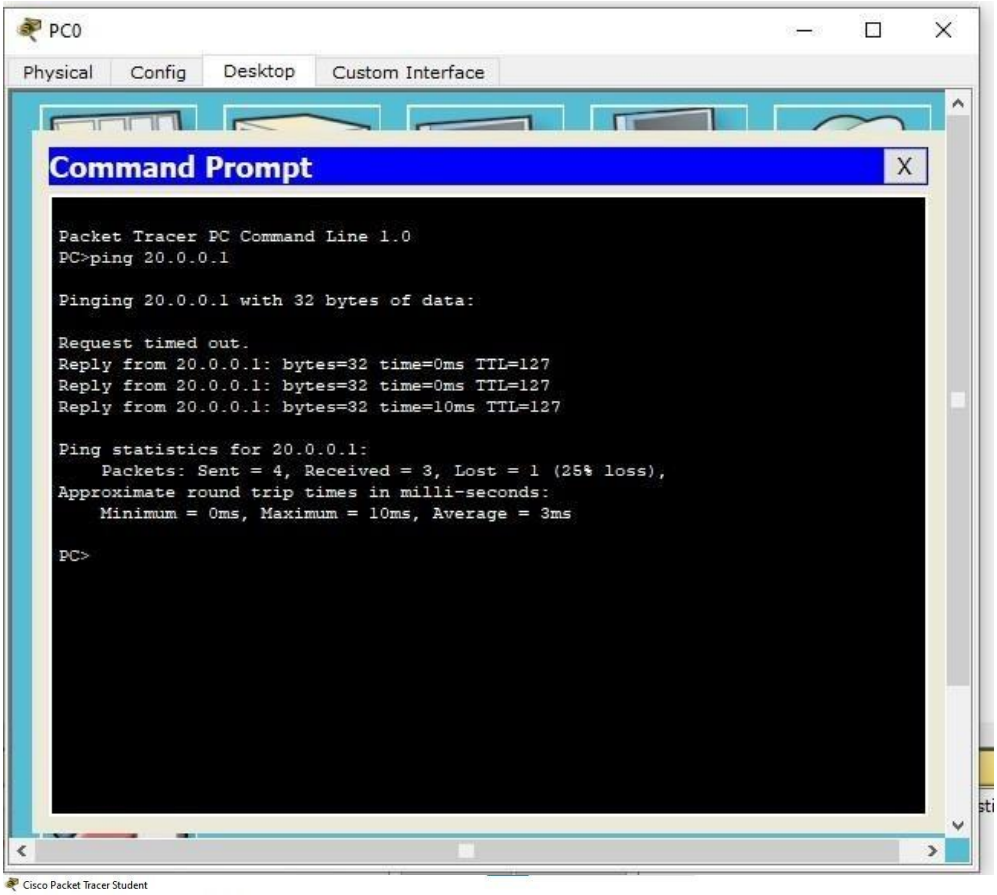
PC > ping 10.0.0.6

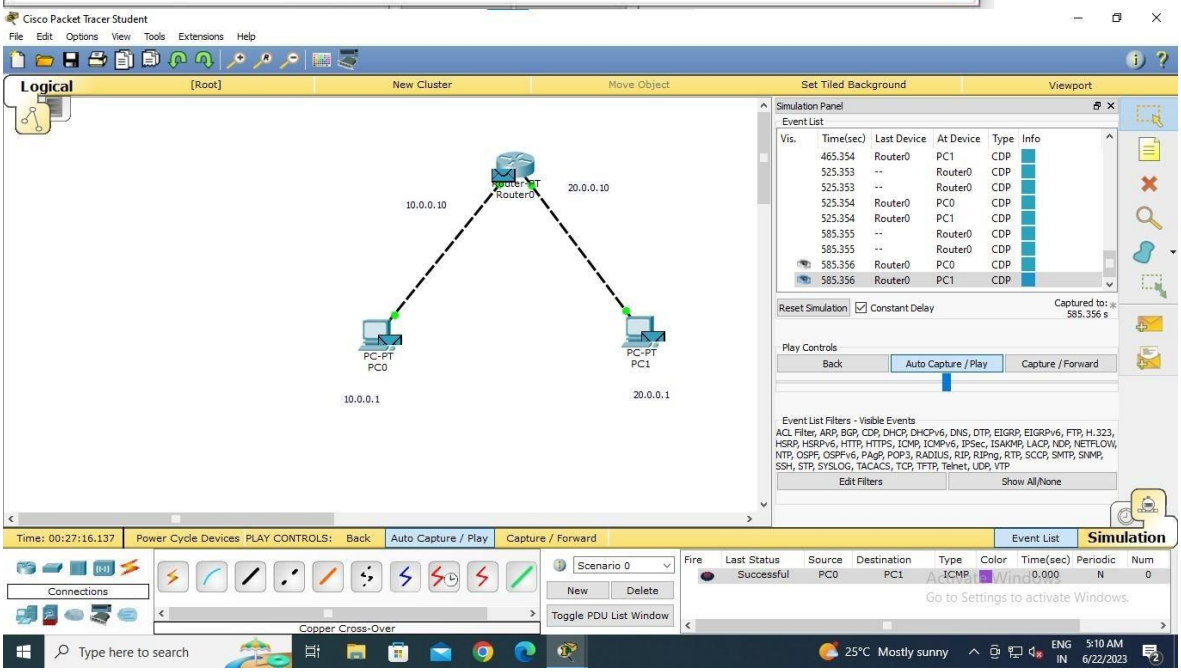
Pinging

SCREENSHOTS






PC0
Physical
Config
Desktop
Custom Interface
Command Prompt
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=10ms 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 = 10ms, Average = 3ms
PC>


Cisco Packet Tracer Student
Logical
10.0.0.10
20.0.0.10
Router0
PC0
PC1
10.0.0.1
20.0.0.1
Simulation Panel
Event List
Reset Simulation
Constant Delay
Play Controls
Back
Auto Capture / Play
Capture / Forward
Edit Filters
Show All/None
Time: 00:27:16.137
Power Cycle Devices
PLAY CONTROLS:
Simulation
Connections
Copper Cross-Over
New
Delete
Toggle PDU List Window
Successful
ICMP
Type here to search
25°C Mostly sunny
5:10 AM
6/22/2023

CN LAB 3

AIM: Configure default route, static route to the Router.

OBSERVATION:

enter no for continue user configuration dialouge.

iv) Type enable

v) Type 'configure terminal'

router(config)# interface Fa0/0

router(config-if)# ip address 10.0.0.2 255.0.0.0

router(config-if)# no shutdown

router(config-if)# exit

vi) Repeat steps for Fa1/0 interface

vii) Ping PC1 from PC0

viii) Set default gateway for PC0 as 10.0.0.2 and 20.0.0.2 for PC1

ix) Ping PC1 again from PC0

R. ESULT

Before setting the default gateway

Ping 20.0.0.1
Request timed out
Request timed out
Request timed out
Request timed out

Ping statistics for 10.0.0.2

Packets sent = 4 recieved = 0
loss = 4

After setting default gateway
Ping 20.0.0.1

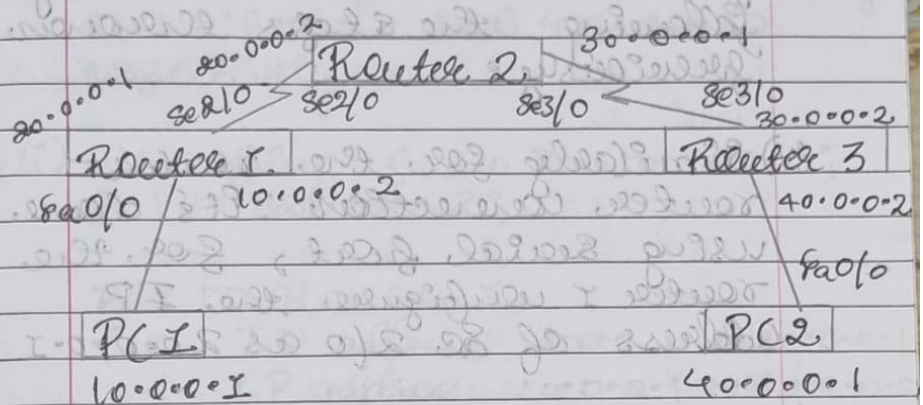
Reply from 20.0.0.1 time = 0 ms
TTL 128

Reply from 20.0.0.1 time = 0ms
TTL = 128

Reply from 20.0.0.1 time = 0ms
TTL = 128

Reply from 20.0.0.1 time = 0ms
TTL = 128
Ping statistics for 10.0.0.2
Packets: sent = 4, recieved = 0,
loss = 0

3 routers together and end devices



i) Place 3 generic router and two end devices. The end devices are connected to Router 1 and Router 3. The Router 2 is connected to Router 1 and 3 acting as an intermediate.

ii) Set the IP address of end devices. Here each end devices is considered to be a seperate network.

iii) Go to command line interface of Router 1 and config the

IP address of the fast ethernet port (Fa0/0) to 10.0.0.2 by following the steps mentioned previously.

iv) Similarly for the router-router connection it's done using serial port, for the router 1 configure the IP address of Se 2/0 as 20.0.0.1.

v) Repeat step 3 and 4 for the router 2 and router 3

Following are the required IP address

Router 1 Fa0/0 → 10.0.0.1
Se 2/0 → 20.0.0.1

Router 2 Se 2/0 → 20.0.0.2
Se 3/0 → 30.0.0.1

Router 3 Se 3/0 → 30.0.0.2
Fa 0/0 → 40.0.0.2

vi) Now in PC1 and PC2 set the IP address of the default gateway as 10.0.0.2 and 40.0.0.2 respectively.

vii) Now ping from PC1 to PC2 ; PC1 to Router 2;

PC1 to Router 1
i.e, IP address 10.0.0.1 to 20.0.0.
IP address 10.0.0.1 to 20.0.0.
IP address 10.0.0.1 to 30.0.0.
IP address 10.0.0.1 to 40.0.0.

viii) Note down the ping outputs you get

RESULT :

a) Ping 20.0.0.2
Request Timed out
Request Timed out
Request Timed out
Request Timed out

Ping statistics for 20.0.0.2
Packets sent = 4, recieved = 0
loss = 4

b) Ping 20.0.0.1
Reply from 20.0.0.1 time=0m
TTL = 128

Reply from 20.0.0.1 time=0ms
TTL=128

Reply from 20.0.0.1 time=0ms
TTL=128

Reply from 20.0.0.1 time=0ms
TTL=128

c) Ping 30.0.0.1
Reply from 20.0.0.2 : Destination host unreachable

Reply from 20.0.0.2 : Destination host unreachable

Reply from 20.0.0.2 : Destination host unreachable

Reply from 20.0.0.2 : Destination host unreachable

Ping statistics from 30.0.0.1

Packets: Sent=4, recieved=0, lost=4

→ Ping 10.0.0.10

Pinging 10.0.0.10 with 32 bytes of data

Reply from 10.0.0.10 : bytes = 32 : time = 9ms TTL = 125

Reply from 10.0.0.10 : bytes = 32 : time = 8ms TTL = 125

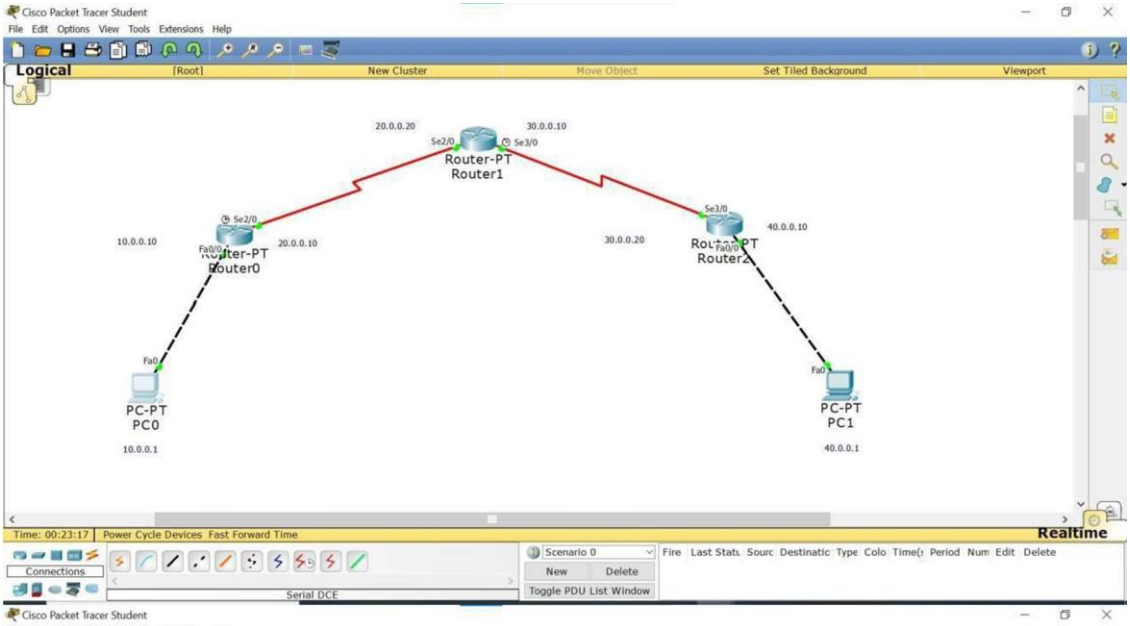
Reply from 10.0.0.10 : bytes = 32 : time = 4ms TTL = 125

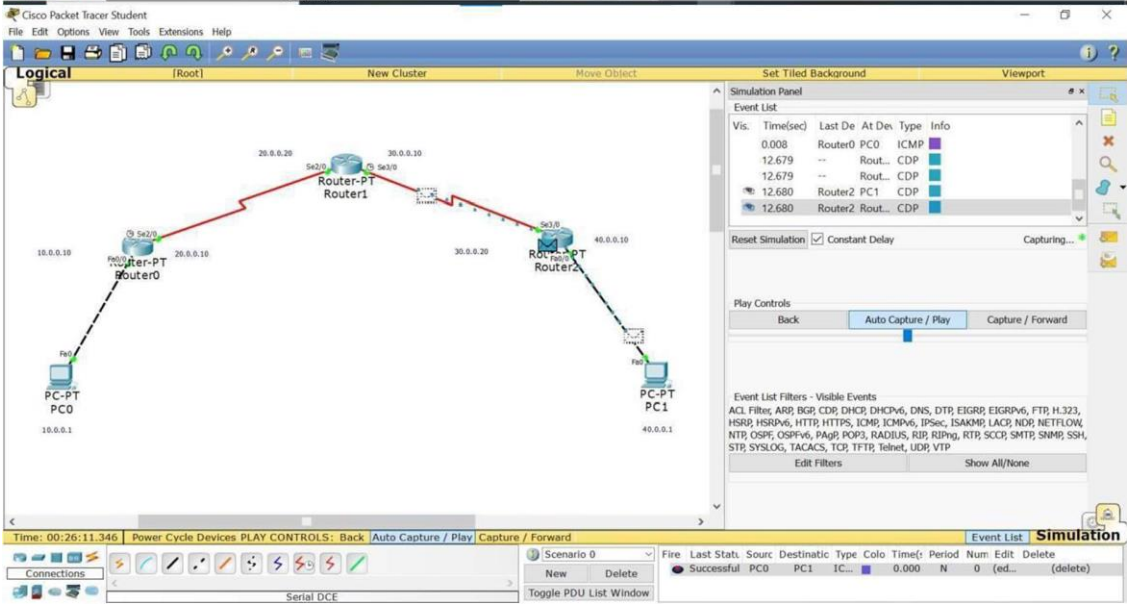
Reply from 10.0.0.10 : bytes = 3
time = 4ms TTL = 125

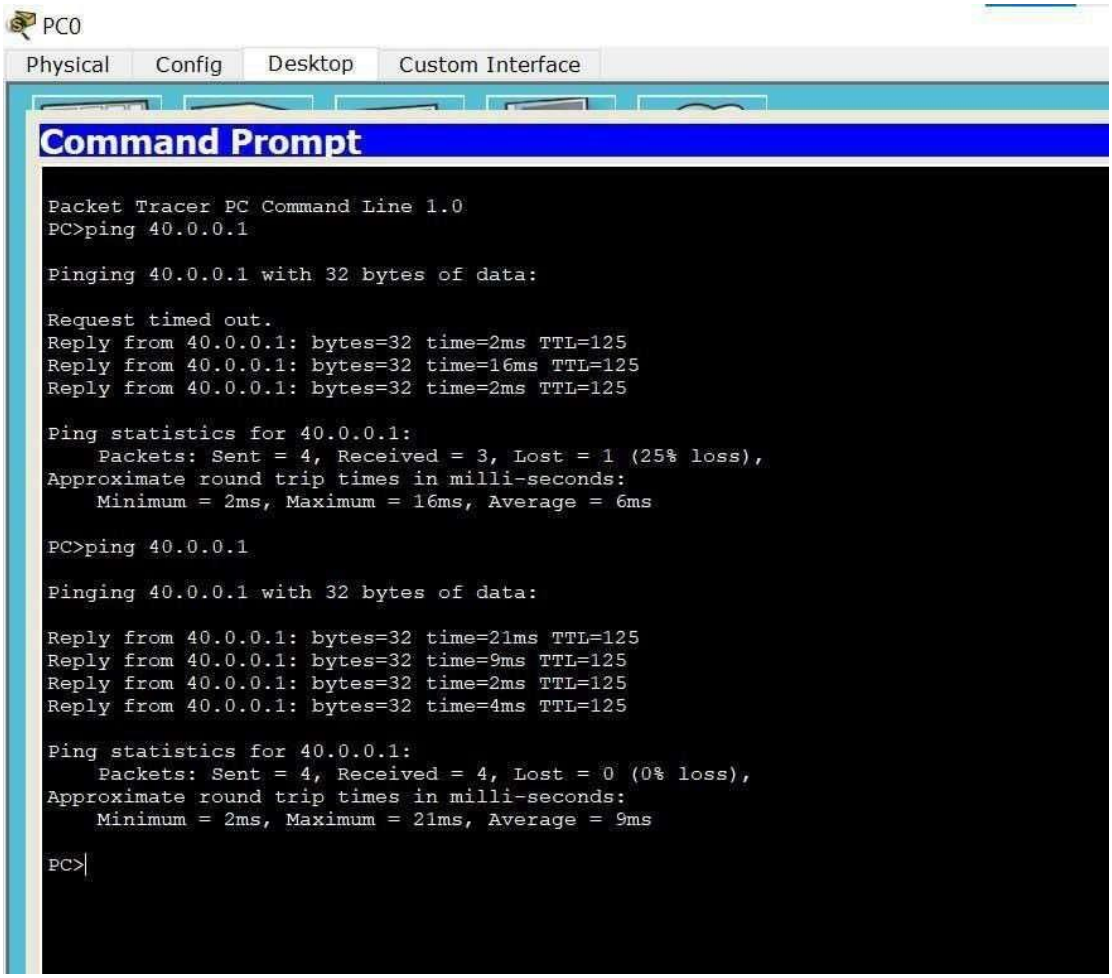
Ping statistics for 10.0.0.10

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

SCREENSHOTS






PC0
Physical
Config
Desktop
Custom Interface
Command Prompt
Packet Tracer PC Command Line 1.0
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=2ms TTL=125
Reply from 40.0.0.1: bytes=32 time=16ms 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 = 16ms, Average = 6ms
PC>ping 40.0.0.1
Pinging 40.0.0.1 with 32 bytes of data:
Reply from 40.0.0.1: bytes=32 time=21ms TTL=125
Reply from 40.0.0.1: bytes=32 time=9ms TTL=125
Reply from 40.0.0.1: bytes=32 time=2ms TTL=125
Reply from 40.0.0.1: bytes=32 time=4ms TTL=125
Ping statistics for 40.0.0.1:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 2ms, Maximum = 21ms, Average = 9ms
PC>

CN LAB 4

AIM: Configure DHCP within a LAN and outside LAN.

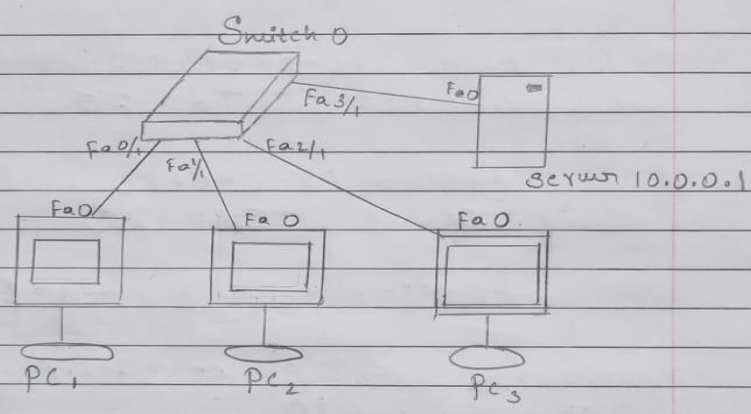
OBSERVATION:

Week 5: Configure DHCP within a LAN and outside LAN

Aim: Configure DHCP within a LAN and outside LAN

Topology

Within a LAN



Procedure:

→ Drag and drop 3 PC's, 1 Switch, and 1 server.

→ Connect all the PC's and server to the switch

→ Set the IP address of the server manually say 10.0.0.1

→ Click on Server Services, then select DHCP

→ Set the start IP address say 10.0.0.2

→ Add the configuration.

→ Now click on each PC and go to config and select IP Configuration as DHCP

Observation:

→ All the PC's connected to the switch and server gets automatically assigned IP address starting from 10.0.0.2.

→ PC1 is assigned 10.0.0.2

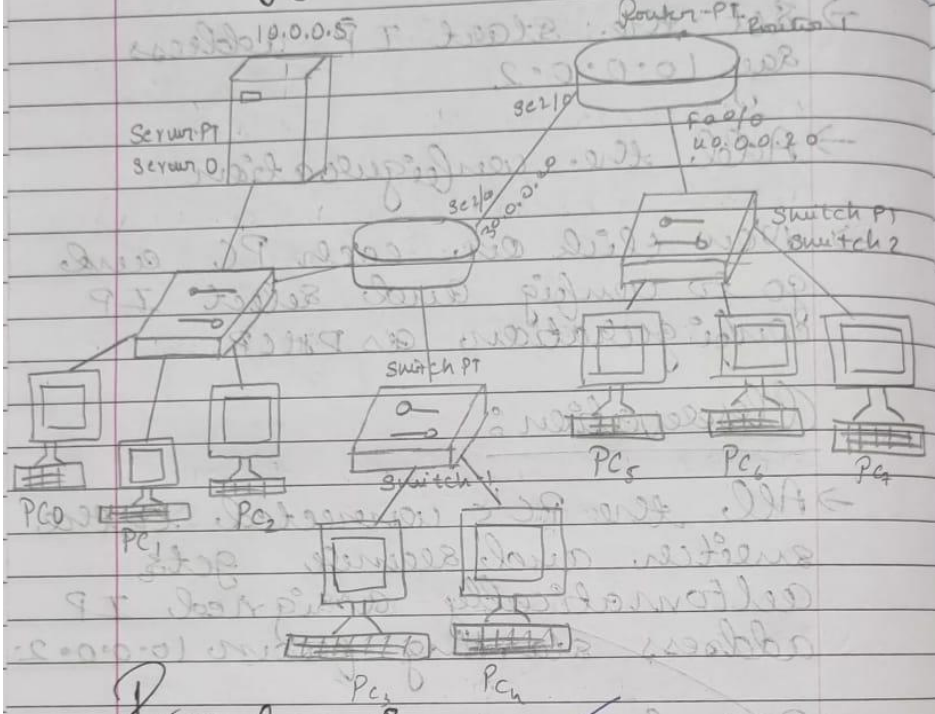
→ PC2 is assigned 10.0.0.3

→ PC3 is assigned 10.0.0.4

Result:
All the PC's are assigned an IP address automatically / Dynamically.

Aim: To configure DHCP outside a LAN

Topology:



Procedure:

1) Refer the procedure we did for the LAN (DHCP) within a LAN.

2) Now add 2 router, a router, set of PC's and switch 1 (LAN 2).

3) Config the ip address of ports Fa4/0 and Fa0/0 of Router 0.

Router 7 enable

Router # config terminal

Router(config)# interface fa4/0

Router(config-if) # ip address 10.0.0.20 255.0.0.0

Router(config-if) # ip helper-address 10.0.0.5

Router(config-if) # no shutdown

Router(config-if) # exit

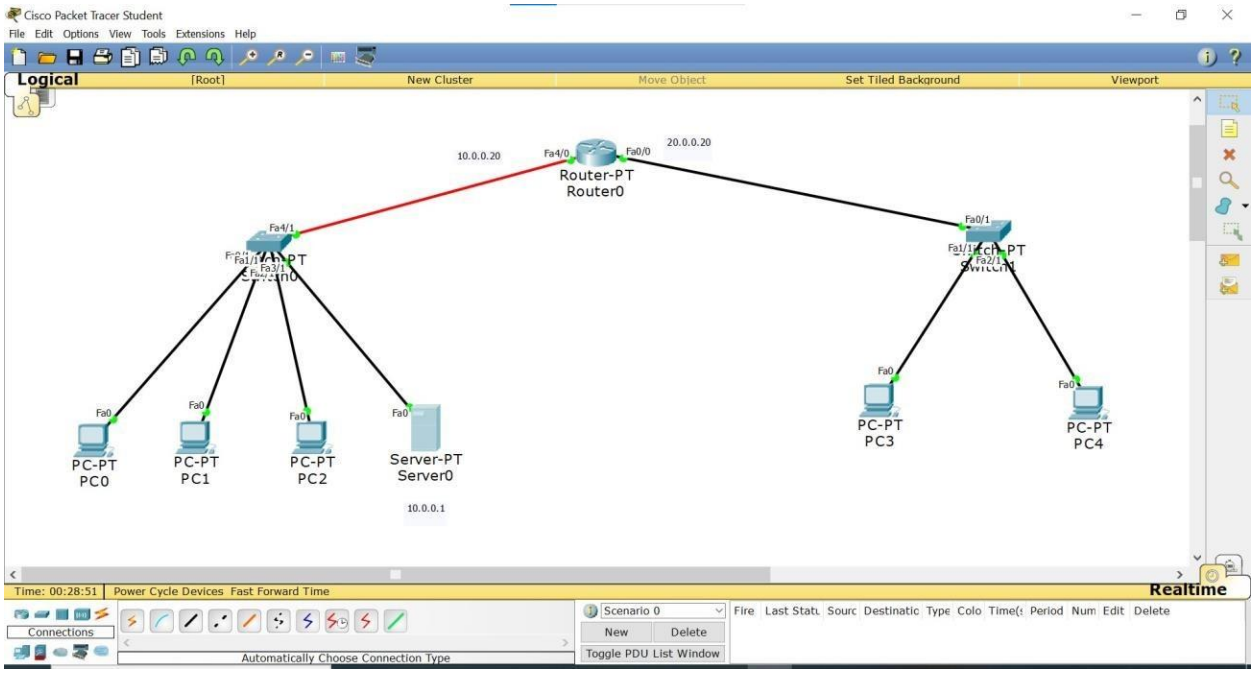
Similarly for Fa 0/0

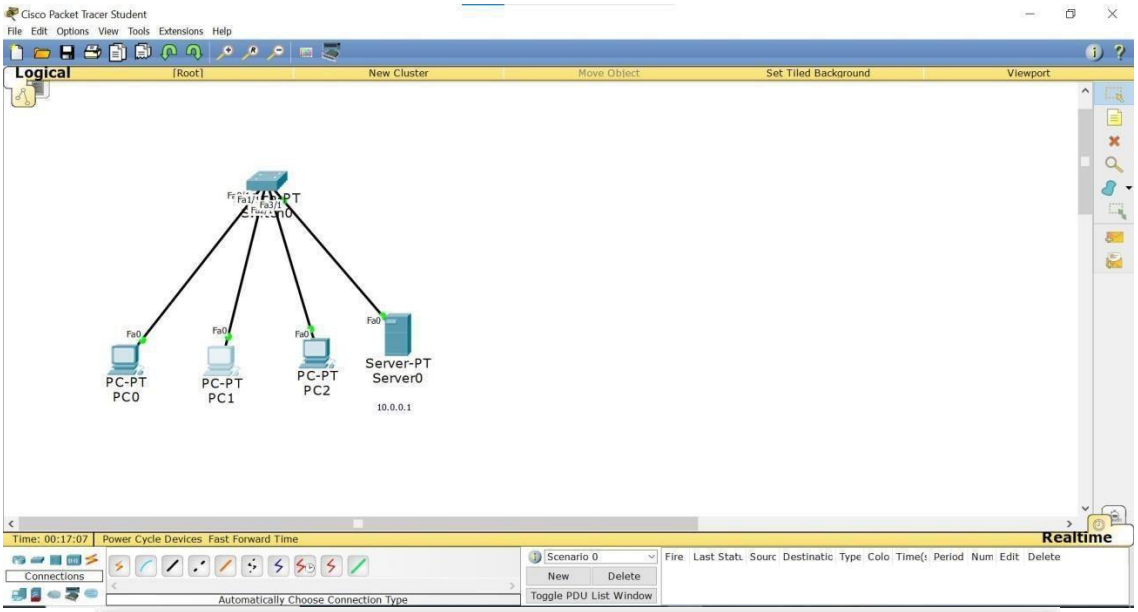
Here ip helper address is the ip address of the server 0.

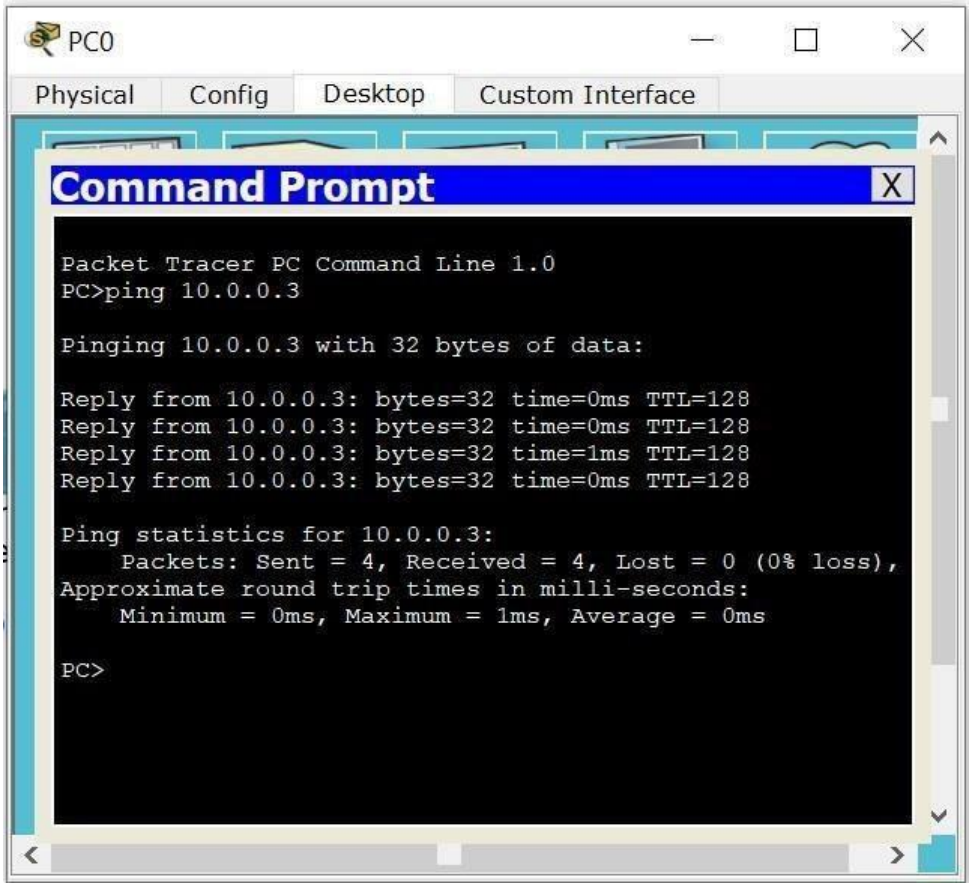
4) Go to PC5 and PC4 and switch to DHCP in

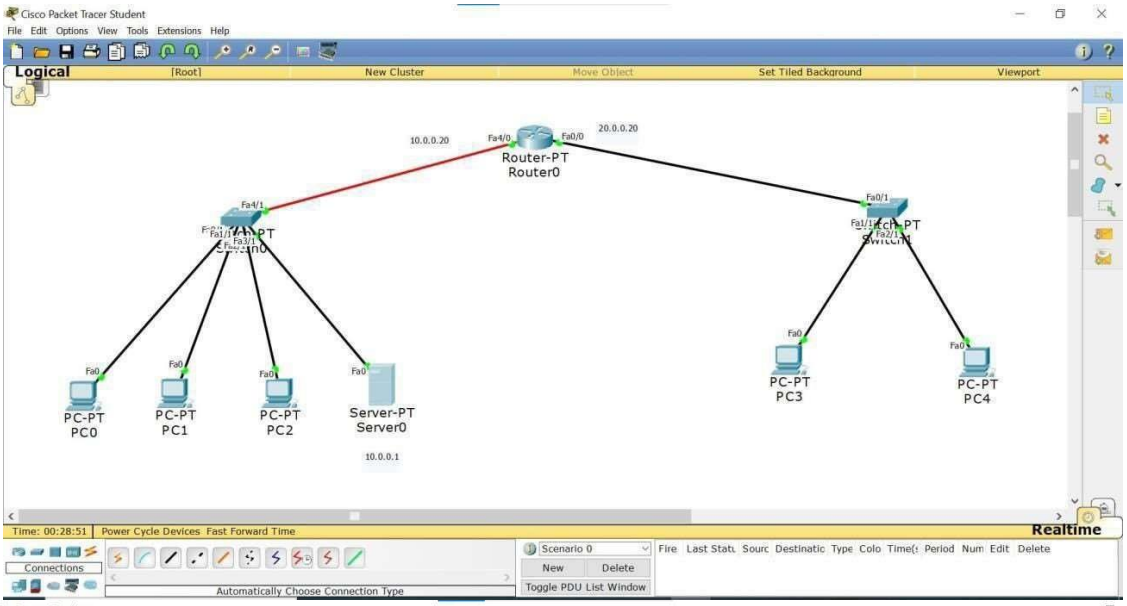
System → IP configuration

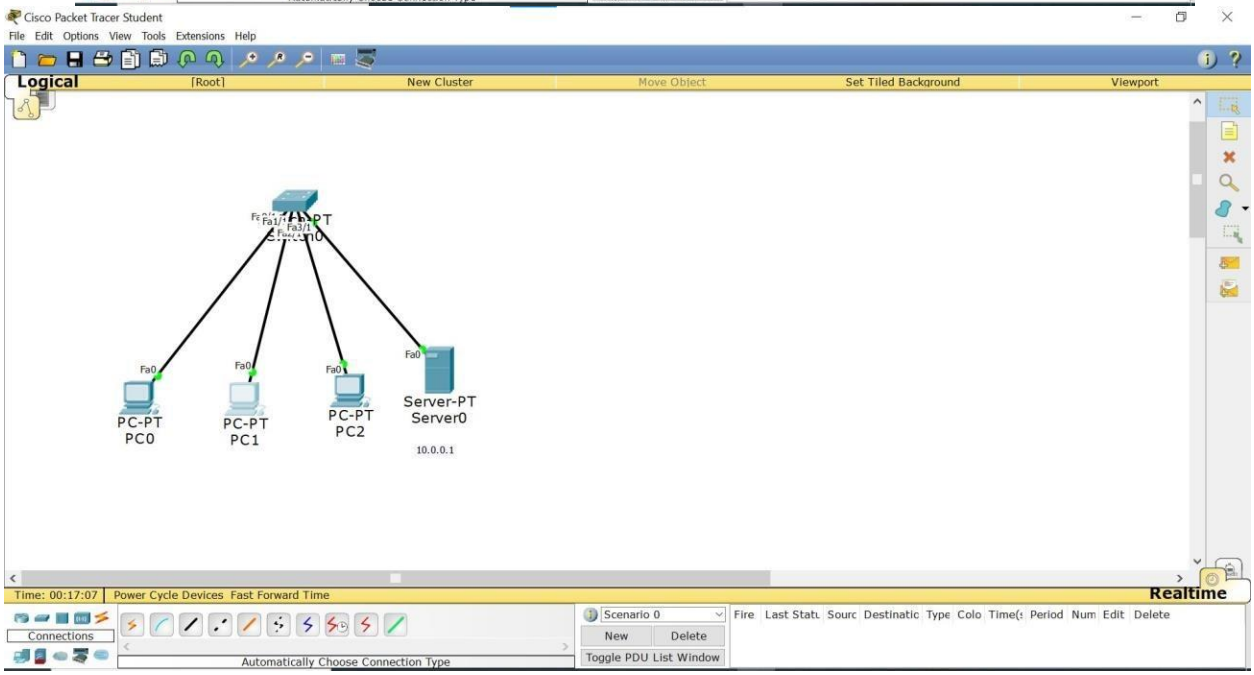
SCREENSHOTS

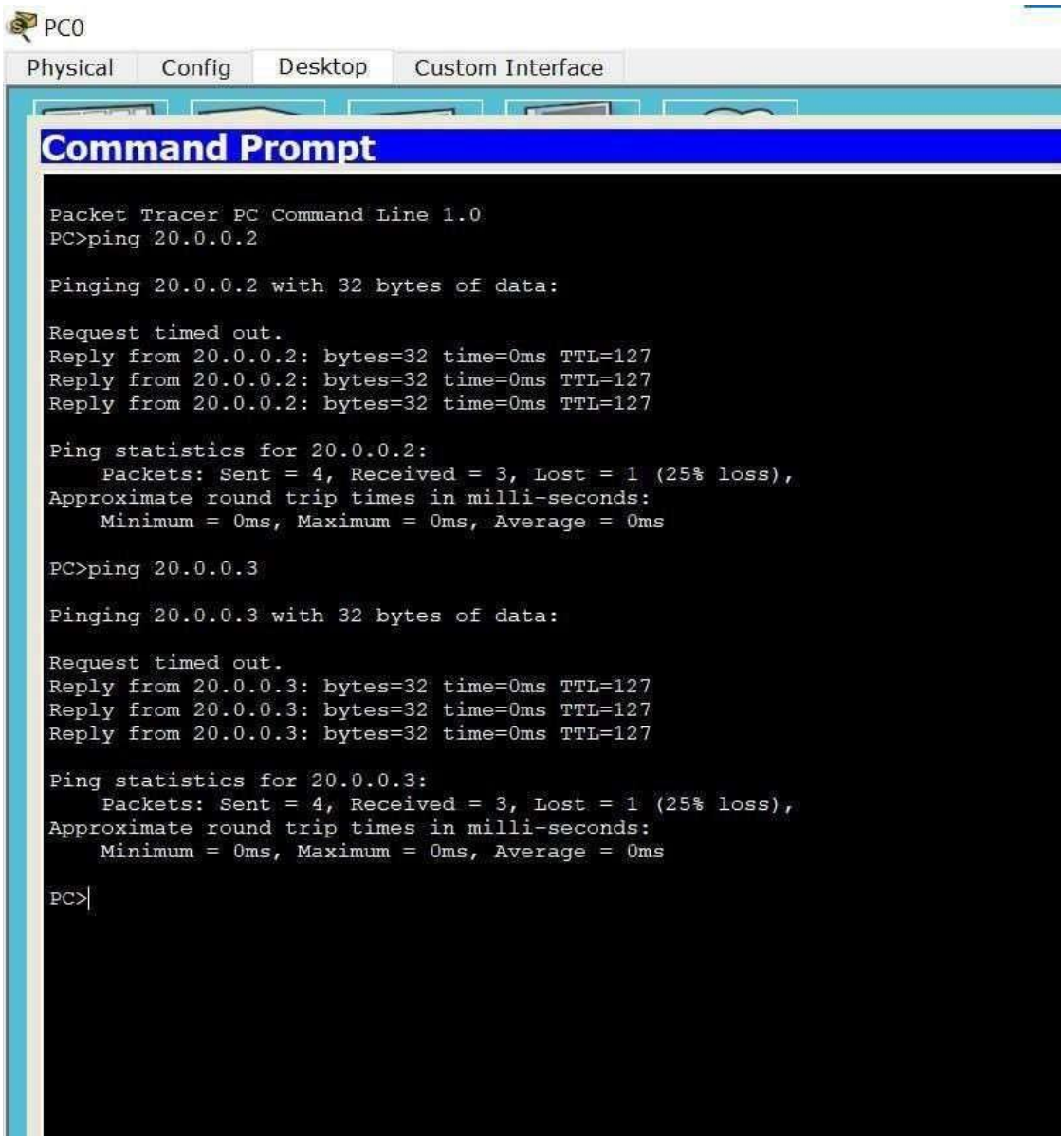



Cisco Packet Tracer Student
File Edit Options View Tools Extensions Help
Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
PC-PT
PC0
PC-PT
PC1
PC-PT
PC2
Server-PT
Server0
10.0.0.1
Time: 00:17:07
Power Cycle Devices Fast Forward Time
Realtime
Connections
Automatically Choose Connection Type
Scenario 0
New
Delete
Toggle PDU List Window


PC0
Physical
Config
Desktop
Custom Interface
Command Prompt
X
Packet Tracer PC Command Line 1.0
PC>ping 10.0.0.3
Pinging 10.0.0.3 with 32 bytes of data:
Reply from 10.0.0.3: bytes=32 time=0ms TTL=128
Reply from 10.0.0.3: bytes=32 time=0ms TTL=128
Reply from 10.0.0.3: bytes=32 time=1ms 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 = 1ms, Average = 0ms
PC>


Cisco Packet Tracer Student
File Edit Options View Tools Extensions Help
Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
10.0.0.20
20.0.0.20
Router-PT
Router0
PC-PT
PC0
PC-PT
PC1
PC-PT
PC2
Server-PT
Server0
10.0.0.1
PC-PT
PC3
PC-PT
PC4
Time: 00:28:51
Power Cycle Devices Fast Forward Time
Realtime
Connections
Automatically Choose Connection Type
Scenario 0
New
Delete
Toggle PDU List Window

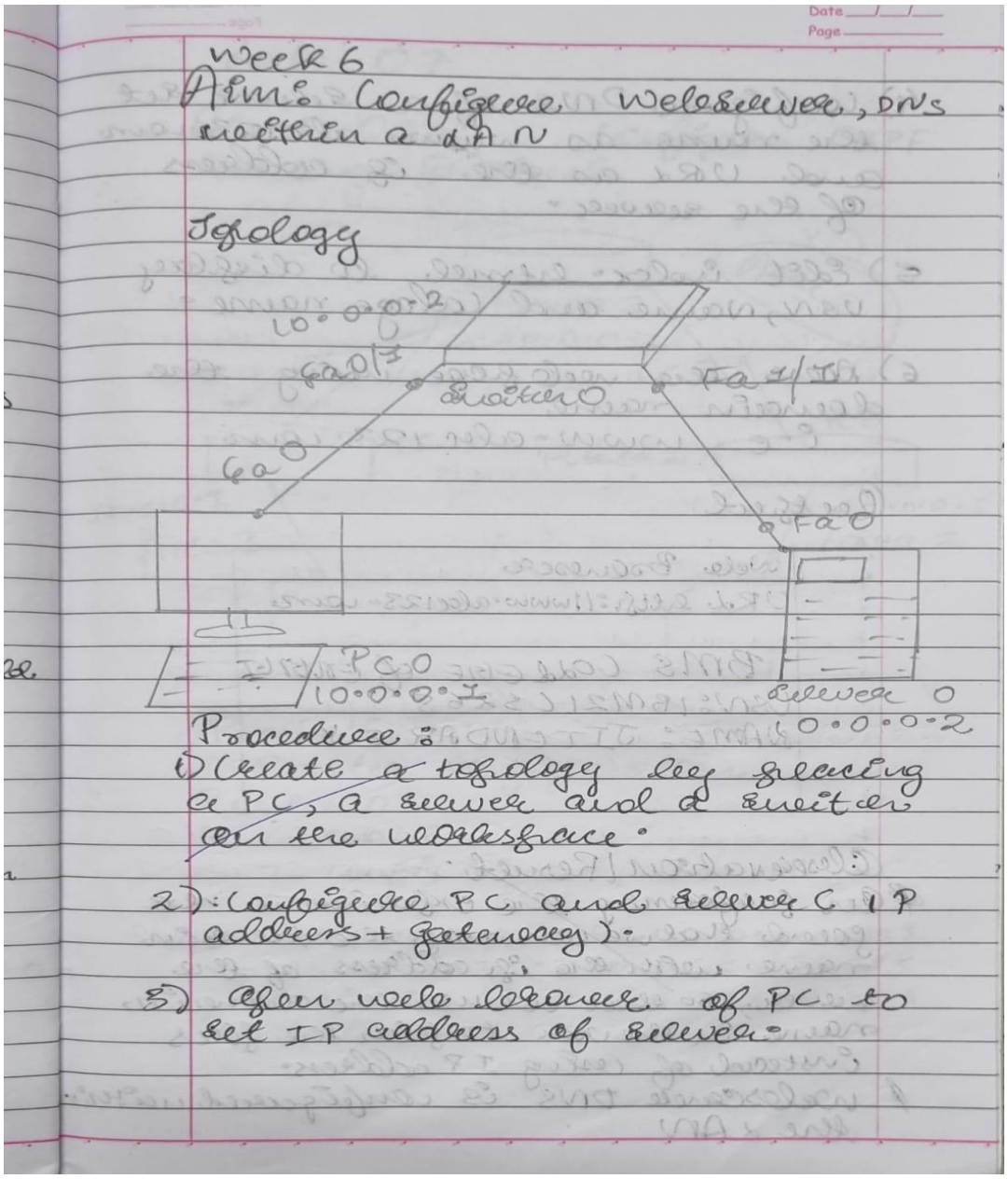

Cisco Packet Tracer Student
File Edit Options View Tools Extensions Help
Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
PC-PT
PC0
PC-PT
PC1
PC-PT
PC2
Server-PT
Server0
10.0.0.1
Time: 00:17:07
Power Cycle Devices Fast Forward Time
Realtime
Connections
Automatically Choose Connection Type
Scenario 0
New
Delete
Toggle PDU List Window


PC0
Physical
Config
Desktop
Custom Interface
Command Prompt
Packet Tracer PC Command Line 1.0
PC>ping 20.0.0.2
Pinging 20.0.0.2 with 32 bytes of data:
Request timed out.
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 = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 0ms, Average = 0ms
PC>ping 20.0.0.3
Pinging 20.0.0.3 with 32 bytes of data:
Request timed out.
Reply from 20.0.0.3: bytes=32 time=0ms TTL=127
Reply from 20.0.0.3: bytes=32 time=0ms TTL=127
Reply from 20.0.0.3: bytes=32 time=0ms TTL=127
Ping statistics for 20.0.0.3:
Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 0ms, Average = 0ms
PC>

CN LAB 5

AIM: Configure Web Server, DNS within a LAN.

OBSERVATION:



4) Configure DNS of a server set the name as www.abc123.com and URL as the IP address of the server.

5) Edit index.html to display USN, Name and College name.

6) Access the webpage using the domain name.
i.e www.abc123.com.

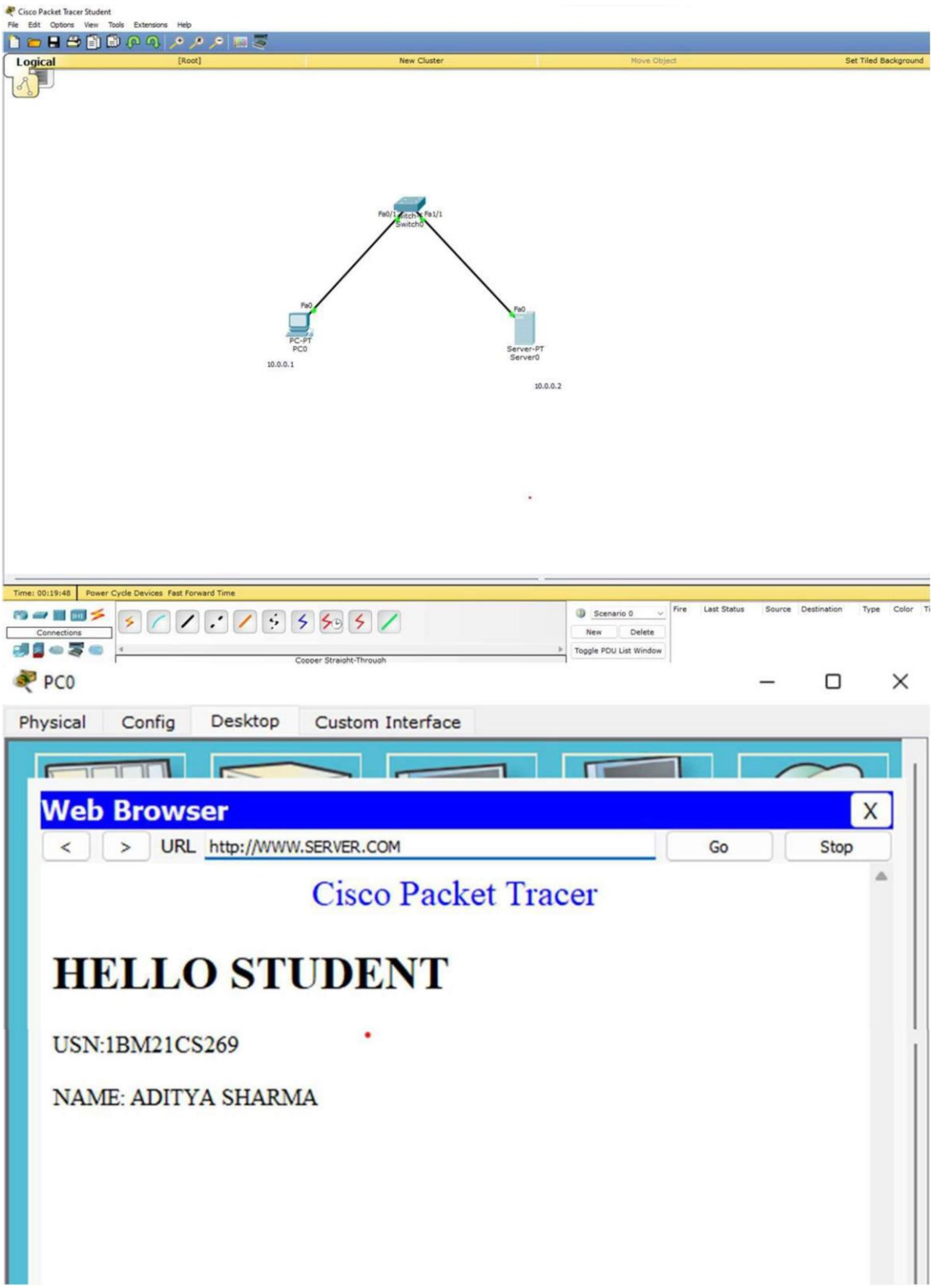
Output

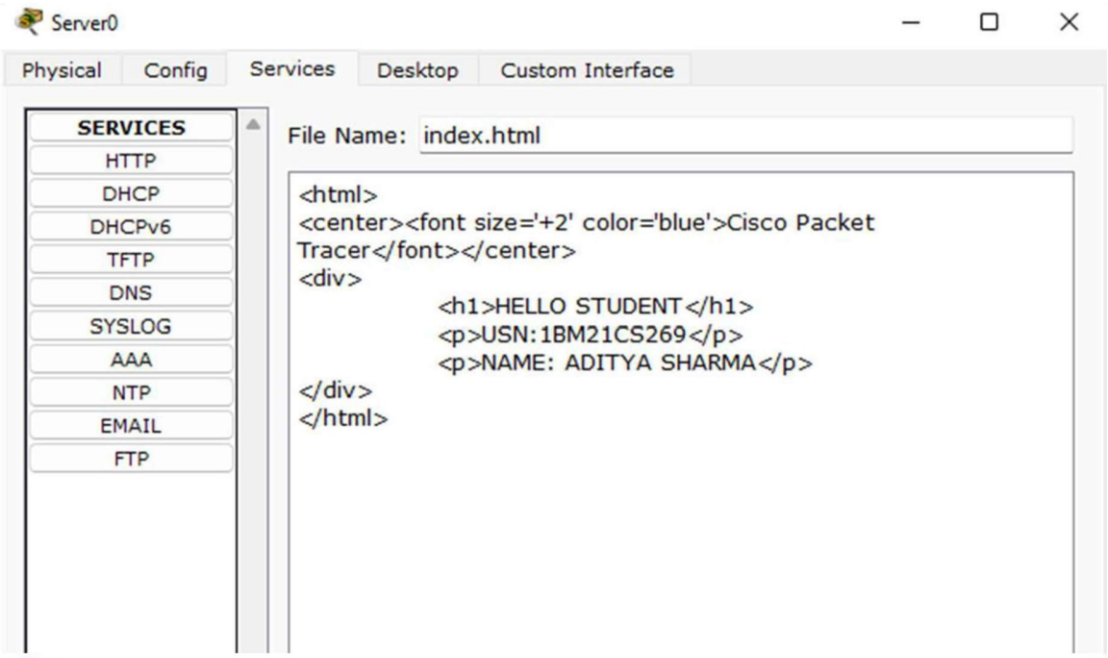
Web Browser
URL http://www.abc123.com
BMS COLLEGE of ENGG USN: 1BM21CS268 NAME: JITENDAR E Y

Observation / Result:

- On performing the experiment we found that we can map the domain name with the IP address of the server, so that we use the domain name in the URL to fetch the files instead of using IP address.
- Webscarole DNS is configured within the LAN

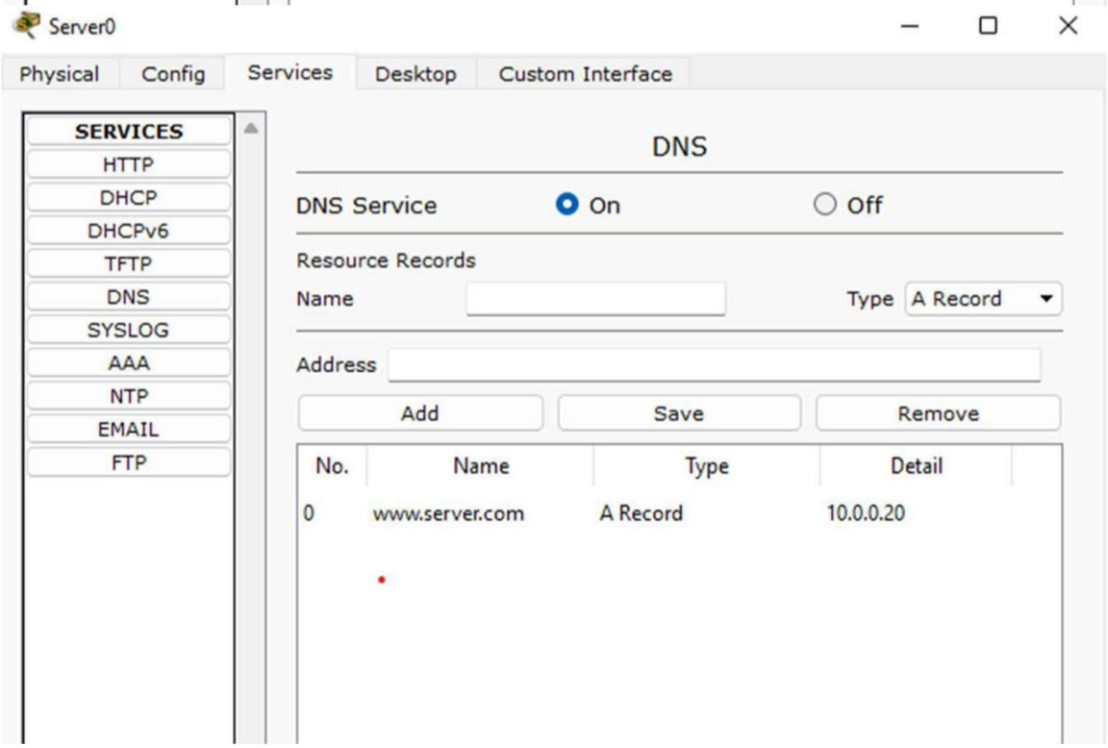
SCREENSHOTS




Server0
Physical
Config
Services
Desktop
Custom Interface
SERVICES
HTTP
DHCP
DHCPv6
TFTP
DNS
SYSLOG
AAA
NTP
EMAIL
FTP
File Name: index.html
<html>
Cisco Packet Tracer

<h1>HELLO STUDENT</h1>
USN:1BM21CS269
NAME: ADITYA SHARMA

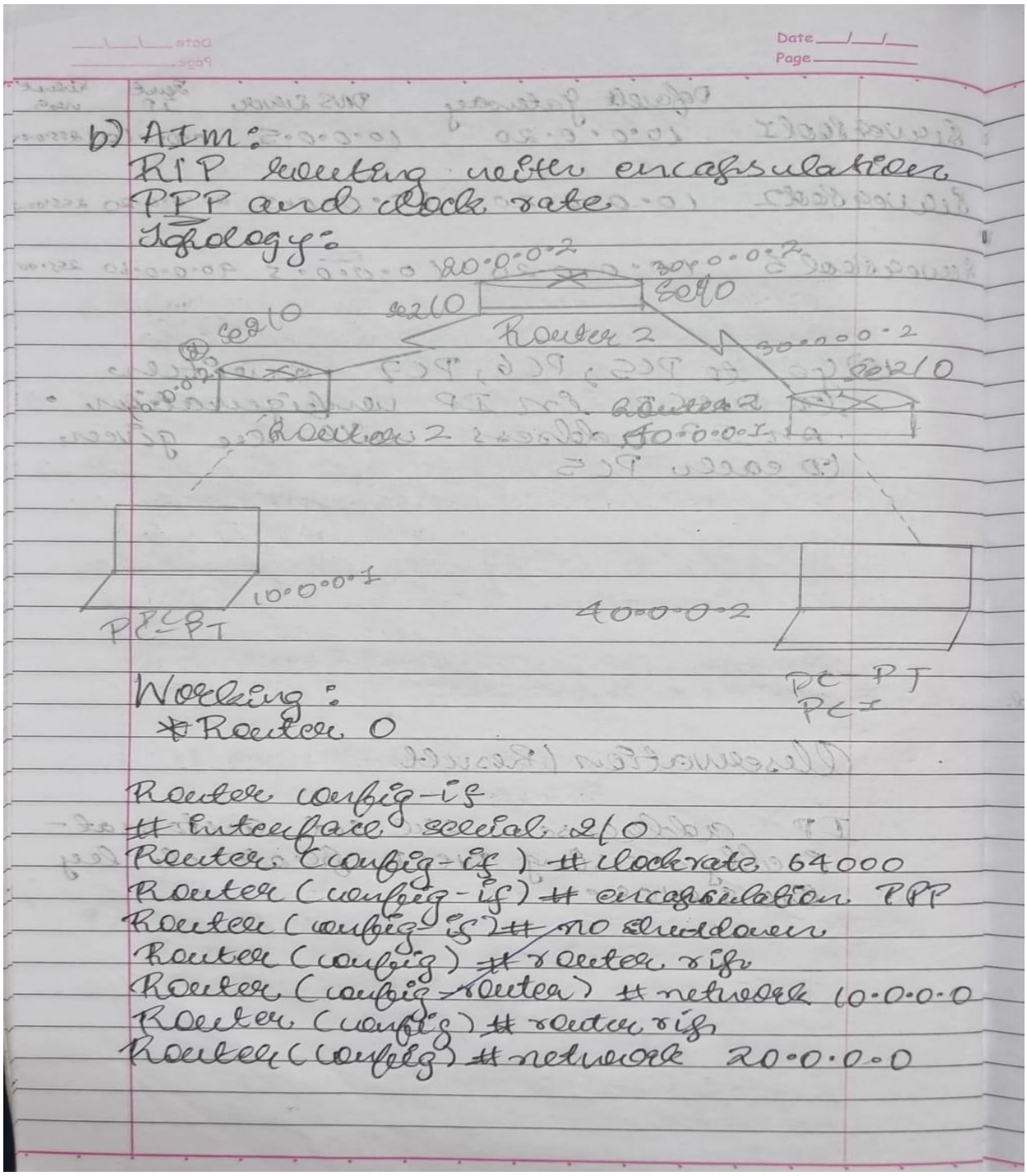
</html>


Server0
Physical
Config
Services
Desktop
Custom Interface
SERVICES
HTTP
DHCP
DHCPv6
TFTP
DNS
SYSLOG
AAA
NTP
EMAIL
FTP
DNS
DNS Service
On
Off
Resource Records
Name
Type
A Record
Address
Add
Save
Remove
No.
Name
Type
Detail
0
www.server.com
A Record
10.0.0.20

CN LAB 6

AIM: Configure RIP routing Protocol in Routers.

OBSERVATION:



Date __/__/__
Page _____

@Router 1
Router(config)# interface serial 3/0
Router(config-if)# clock rate 64000
Router(config-if)# encapsulation PPP
Router(config-if)# no shutdown
Router(config)# router rip
Router(config-if)# network 30.0.0.0

@Router 2
Router(config)# router rip
Router(config-router)# network 40.0.0.0
Router(config)# router rip
Router(config-router)# network 30.0.0.0

Output: Ping from PC1 to PC2 10.0.0.1 to 40.0.0.2

PC> ping 40.0.0.2
Pinging 40.0.0.2 with 32 bytes of data

Reply from 40.0.0.2 bytes=32 time=2ms TTL=125
Reply from 40.0.0.2 bytes=32 time=2ms TTL=125
Reply from 40.0.0.2 bytes=32

time = 2ms TTL = 125
Reply from 40.0.0.2 bytes = 32
time = 3ms TTL = 125

Ping statistics for 40.0.0.2

Packet: Sent = 4, Received = 4
Lost = 0 (0% loss), ~~Approximate~~
recent trip time in milli seconds

Minimum = 2ms, Maximum = 0ms,
Average = 3ms

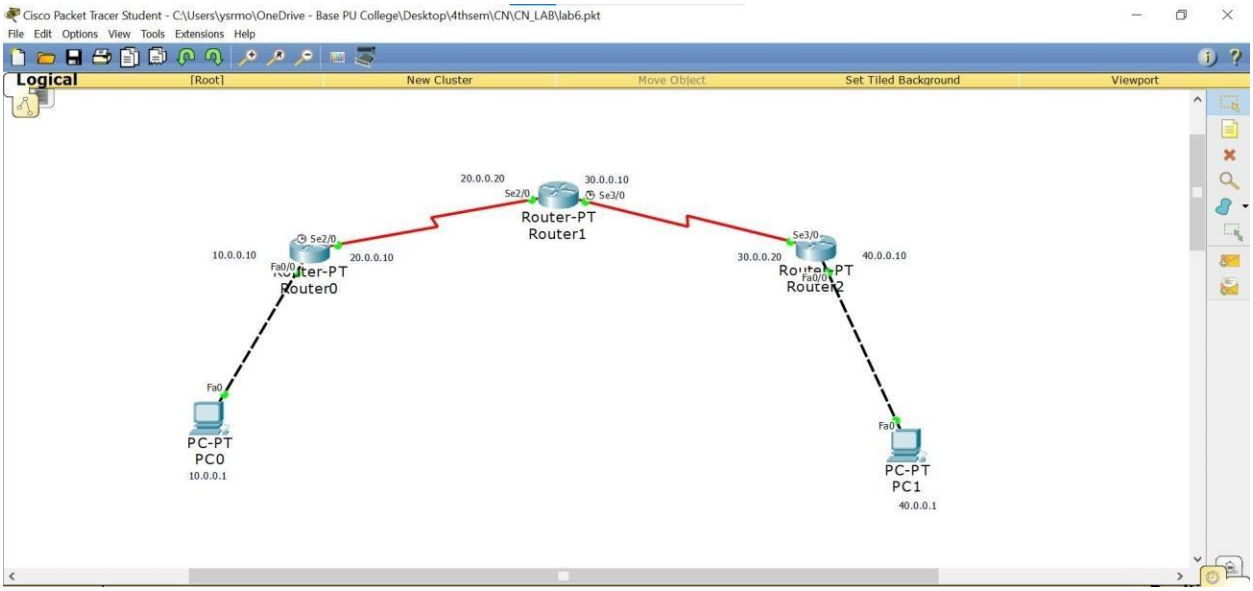
PC>

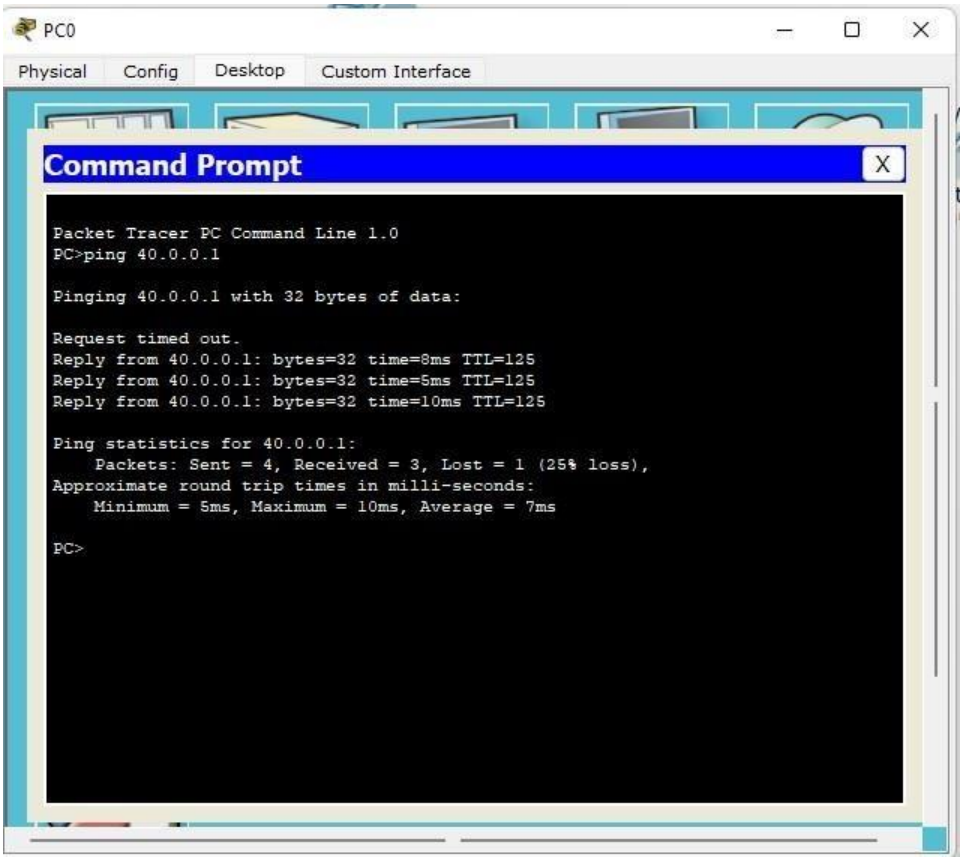
Observation:

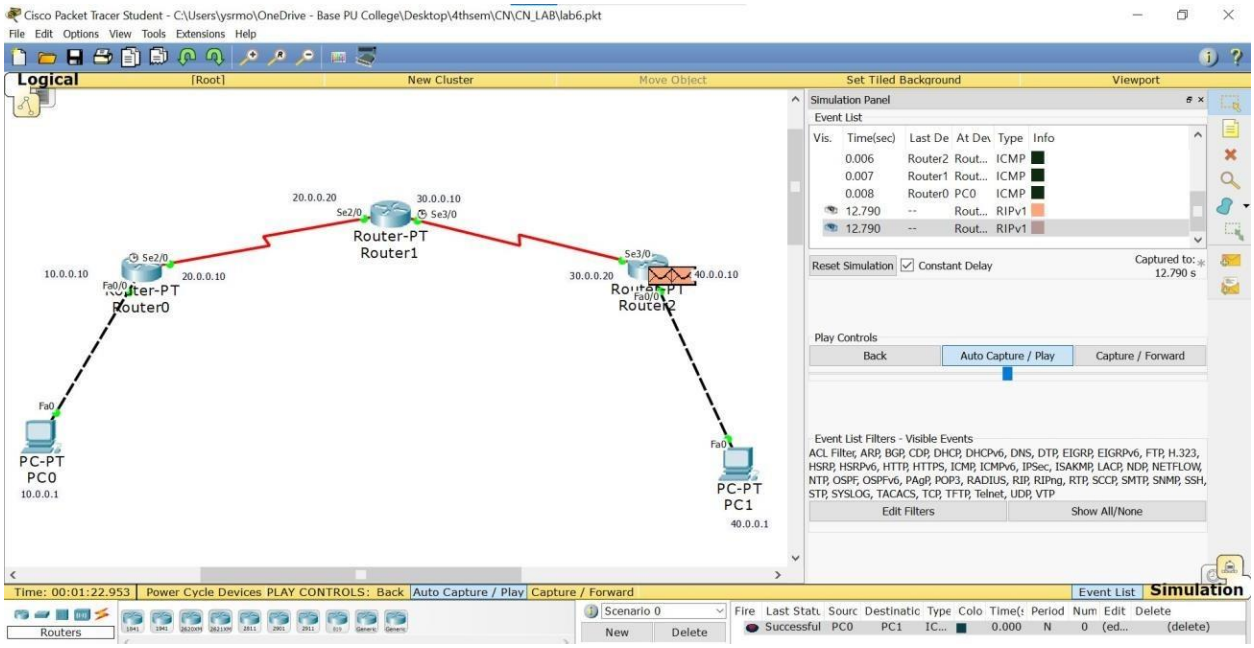
1) RIP defines how routers should share information before moving traffic among an interconnected group of the network

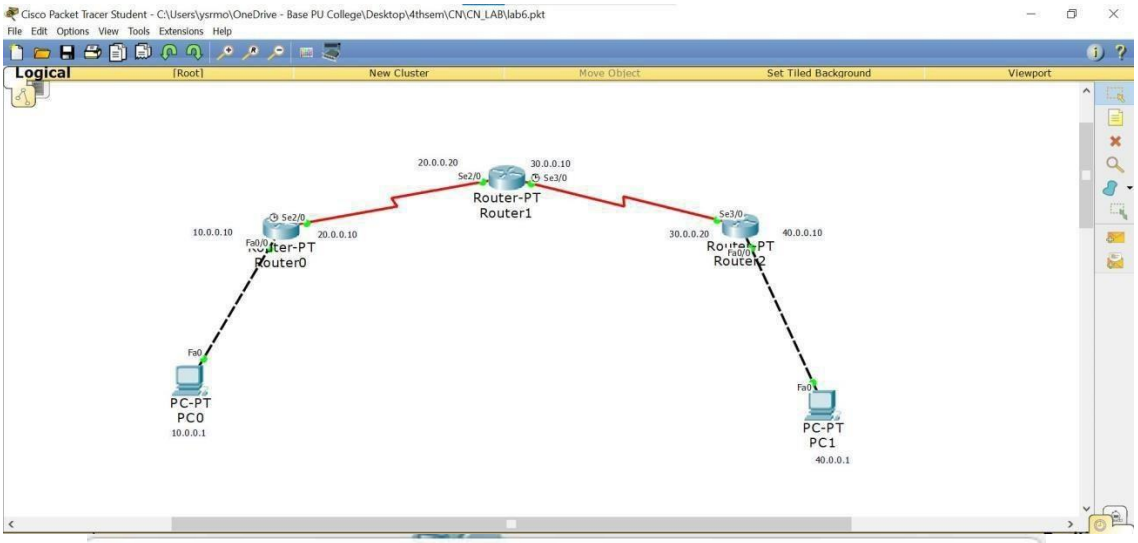
2 And the enrollment is done using commands such as encapsulation PPP and rip commands and the outcomes are noted.

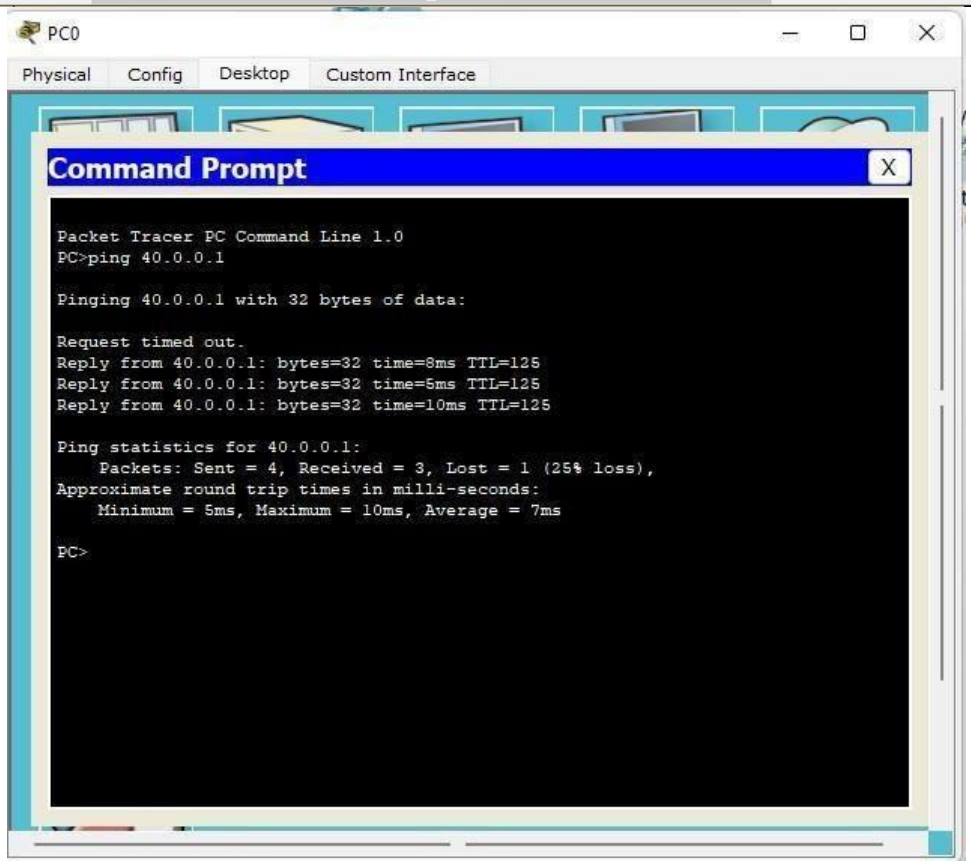
SCREENSHOTS

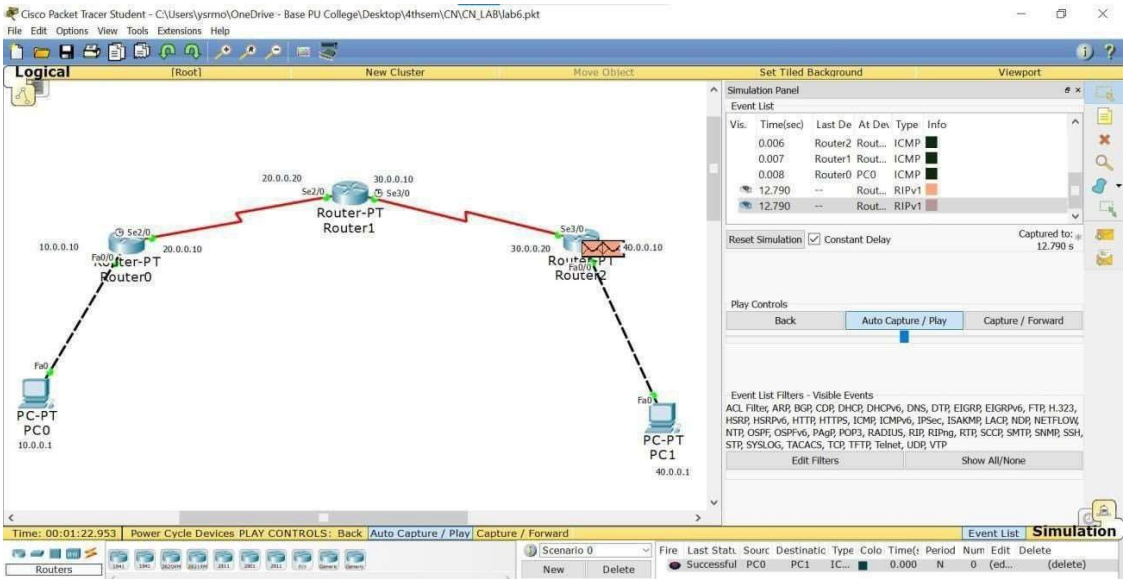





Cisco Packet Tracer Student - C:\Users\ysrmo\OneDrive - Base PU College\Desktop\4thsem\CN\CN_LAB\lab6.pkt
File Edit Options View Tools Extensions Help
Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
Router-PT
Router1
Router-PT
Router0
Router-PT
Router2
PC-PT
PC0
10.0.0.1
PC-PT
PC1
40.0.0.1
20.0.0.20
30.0.0.10
10.0.0.10
20.0.0.10
30.0.0.20
40.0.0.10
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 Statu Sourc Destinatic Type Colo Time(s Period Num Edit Delete
Successful PC0 PC1 IC... 0.000 N 0 (ed... (delete)


Cisco Packet Tracer Student - C:\Users\ysrmo\OneDrive - Base PU College\Desktop\4thsem\CN\CN_LAB\lab6.pkt
File Edit Options View Tools Extensions Help
Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
20.0.0.20
Se2/0
30.0.0.10
Se3/0
Router-PT
Router1
Se2/0
10.0.0.10
Fa0/0
20.0.0.10
Router0
Se3/0
30.0.0.20
40.0.0.10
Fa0/0
Router2
Fa0
PC-PT
PC0
10.0.0.1
Fa0
PC-PT
PC1
40.0.0.1

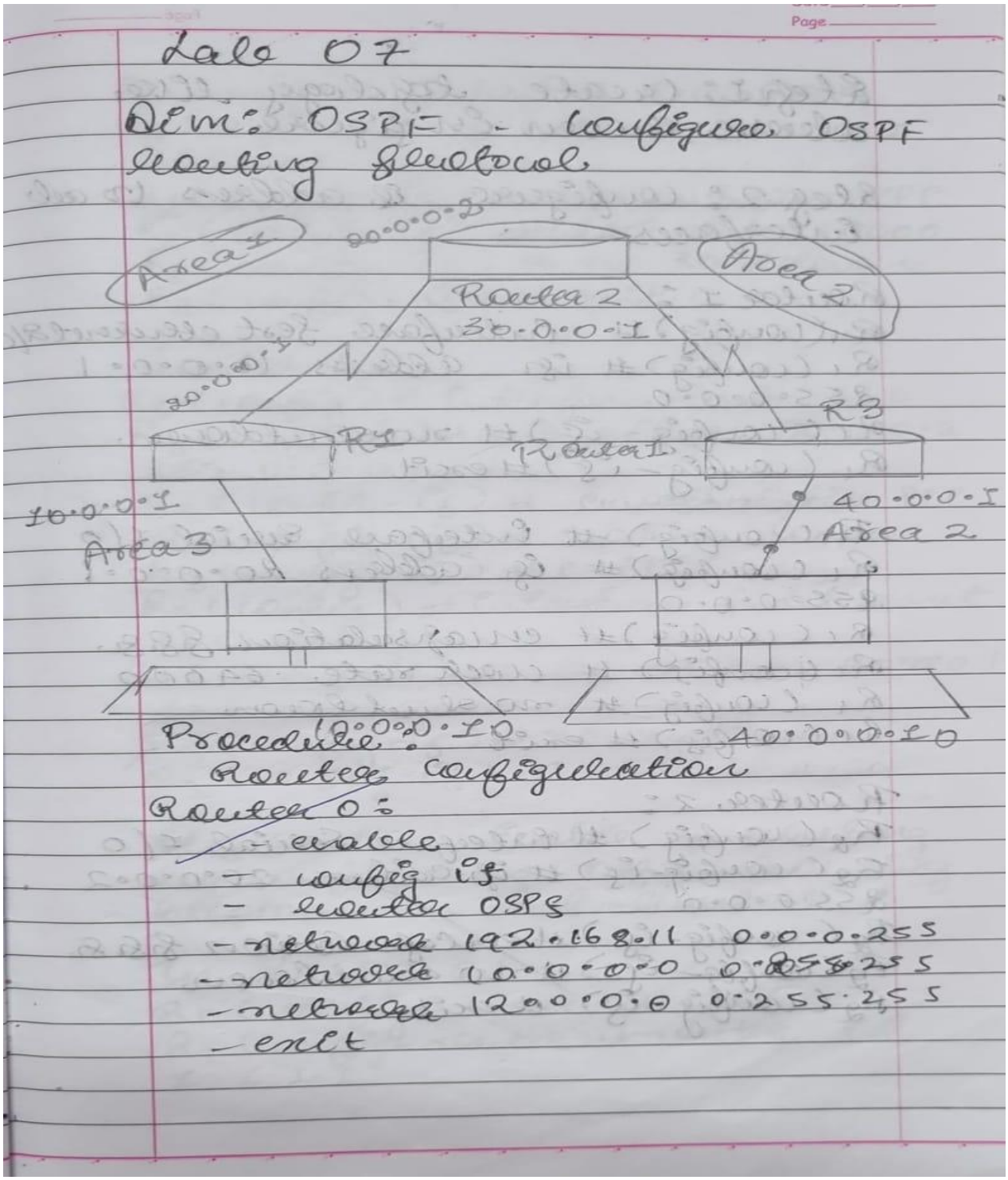

PC0
Physical
Config
Desktop
Custom Interface
Command Prompt
X
Packet Tracer PC Command Line 1.0
PC>ping 40.0.0.1
Pinging 40.0.0.1 with 32 bytes of data:
Request timed out.
Reply from 40.0.0.1: bytes=32 time=8ms TTL=125
Reply from 40.0.0.1: bytes=32 time=5ms TTL=125
Reply from 40.0.0.1: bytes=32 time=10ms 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 = 5ms, Maximum = 10ms, Average = 7ms
PC>


Cisco Packet Tracer Student - C:\Users\ysrmo\OneDrive - Base PU College\Desktop\4thsem\CN\CN_LAB\lab6.pkt
File Edit Options View Tools Extensions Help
Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
Router-PT
Router1
Router-PT
Router0
Router-PT
Router2
PC-PT
PC0
PC-PT
PC1
20.0.0.20
30.0.0.10
Se2/0
Se3/0
10.0.0.10
20.0.0.10
30.0.0.20
40.0.0.10
Fa0/0
Fa0
10.0.0.1
40.0.0.1
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(s Period Num Edit Delete
Successful PC0 PC1 IC... 0.000 N 0 (ed... (delete)

CN LAB 7

AIM: Configure OSPF routing protocol.

OBSERVATION:



Step 1: Create topology like above shown in figure

Step 2: Configure IP address to all interfaces

Router 1:
R1(config)# interface fast ethernet 0/0
R1(config)# ip address 10.0.0.1 255.0.0.0
R1(config-if)# no shutdown
R1(config-if)# exit

R1(config)# interface serial 1/0
R1(config)# ip address 20.0.0.1 255.0.0.0
R1(config)# encapsulation PPP
R1(config)# clock rate 64000
R1(config)# no shutdown
R1(config)# exit

Router 2:
R2(config)# interface serial 1/0
R2(config-if)# ip address 20.0.0.2 255.0.0.0
R2(config-if)# encapsulation PPP
R2(config-if)# no shutdown
R2(config-if)# exit

R2(config)# interface serial 1/0
R2(config-if)# ip address 30.0.0.1 255.0.0.0
R2(config-if)# encapsulation PPP
R2(config-if)# clock rate 64000
R2(config-if)# no shutdown
R2(config-if)# exit

Router R3
R3(config)# interface serial 1/0
R3(config-if)# ip address 30.0.0.2 255.0.0.0
R3(config-if)# encapsulation PPP
R3(config-if)# no shutdown
R3(config-if)# exit
R3(config)# interface fastethernet 2/0
R3(config-if)# ip address 40.0.0.1 255.0.0.0
R3(config-if)# no shutdown
R3(config-if)# exit

Step 3: Now, Enable ip routing and configuring ospf routing protocol in all routers

In Router R1
R1(config)# router ospf 1
R1(config-router)# router-id 1.1.1.1

R1(config-if router) #

Process 1, Nbr 2.2.2 on OSPF_VL0 from Loading to Full Loading Done.

Repeat the procedure for R2 and R3

Step 6: check connectivity b/w host 10.0.0.1 to 40.0.0.10

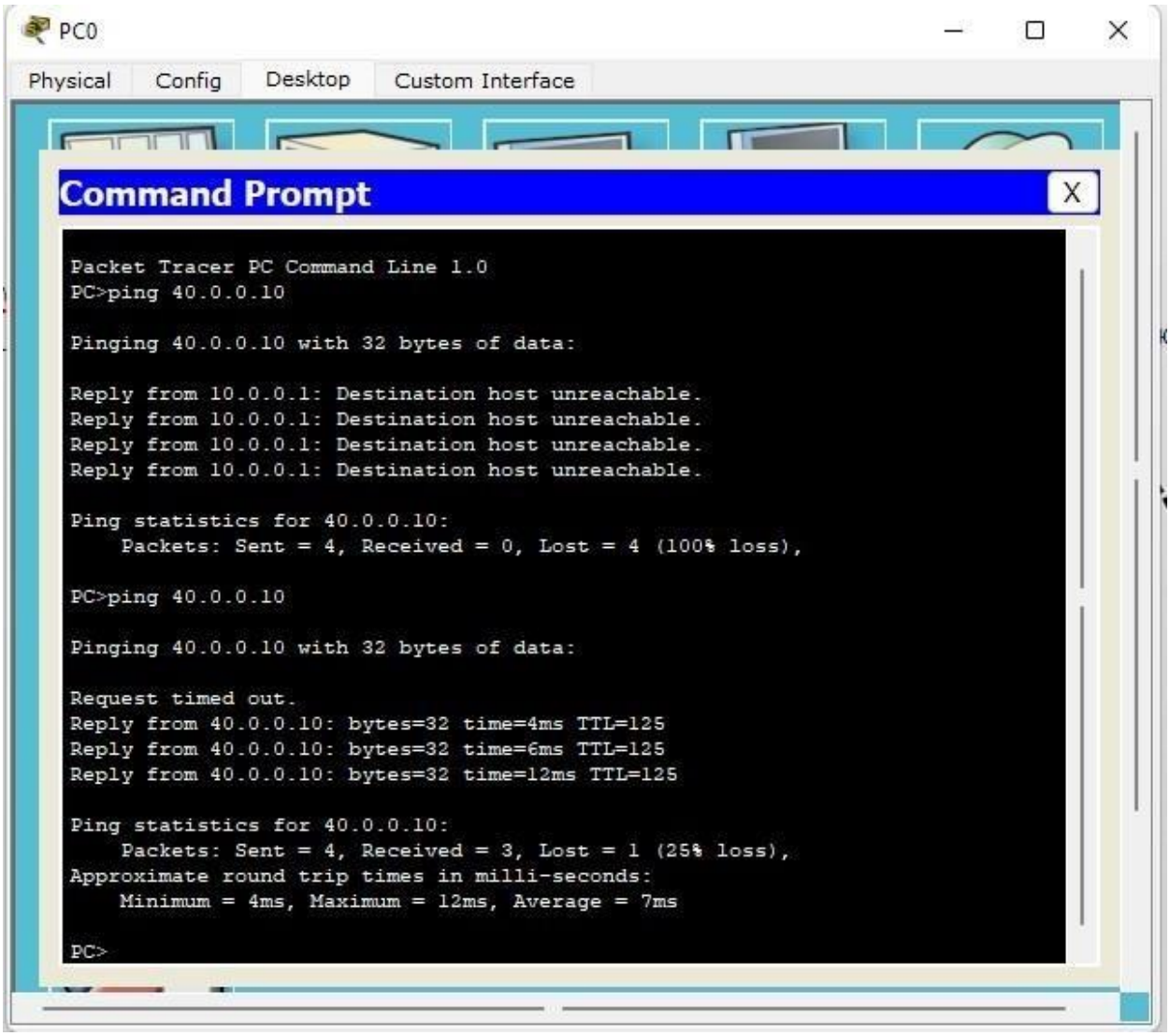
OUTPUT/OBSERVATION

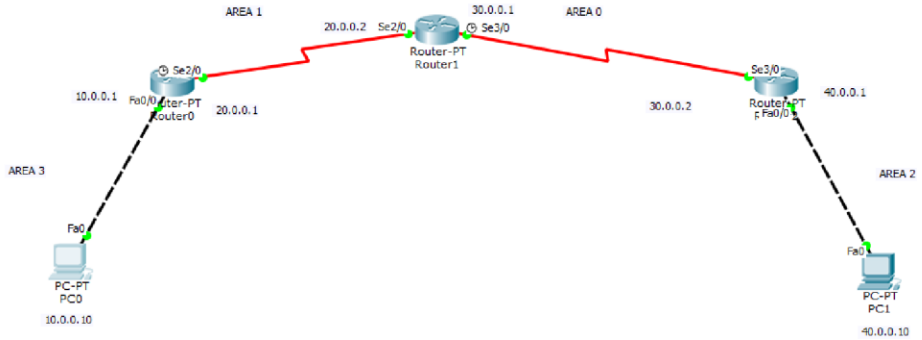
Ping 40.0.0.10

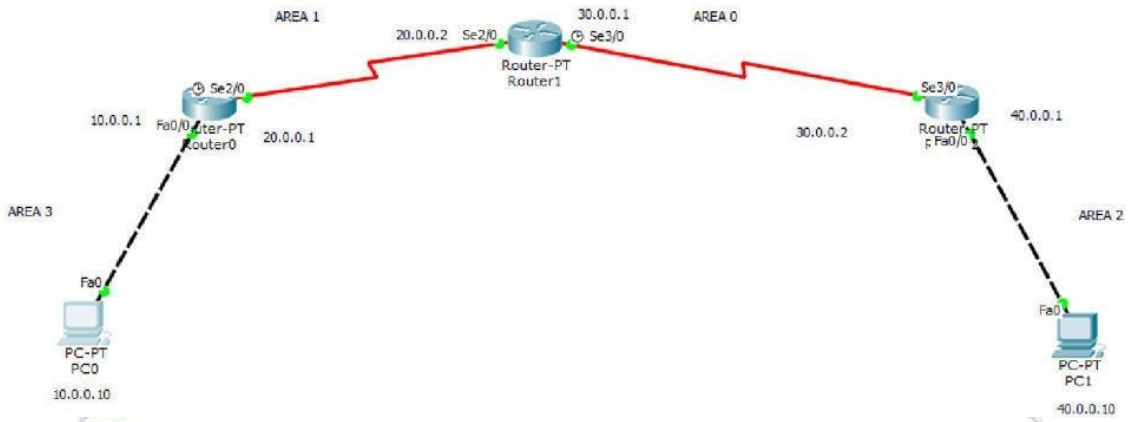
Ping 40.0.0.10 :56 bytes
64 bytes from 40.0.0.10 seq = 1
TTL = 61 time = 73 ms
64 bytes from 40.0.0.10 seq = 1
TTL = 61 time = 64 ms
64 bytes from 40.0.0.10 seq = 1
TTL = 61 time = 66 ms
64 bytes from 40.0.0.10 seq = 1
TTL = 61 time = 60 ms
64 bytes from 40.0.0.10 seq = 1
TTL = 61 time = 96 ms

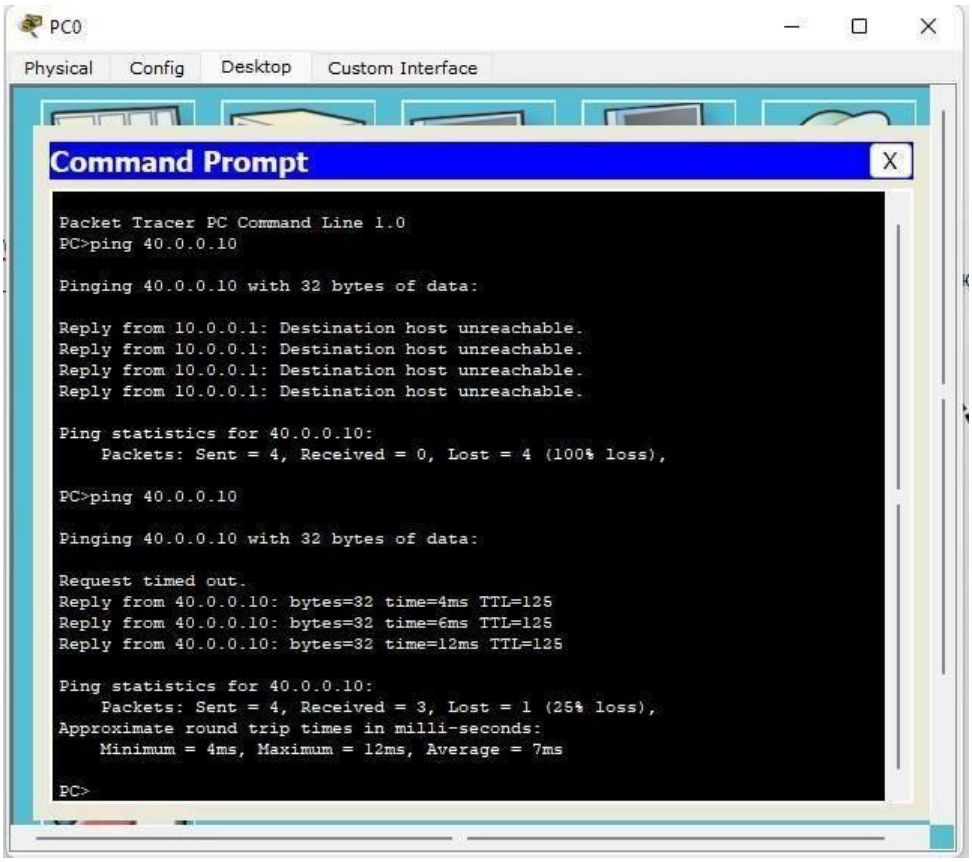
6 packets transmitted, 5 received, 16
packet loss round-trip min/avg/ma
60.829/92.438/173.753 ms

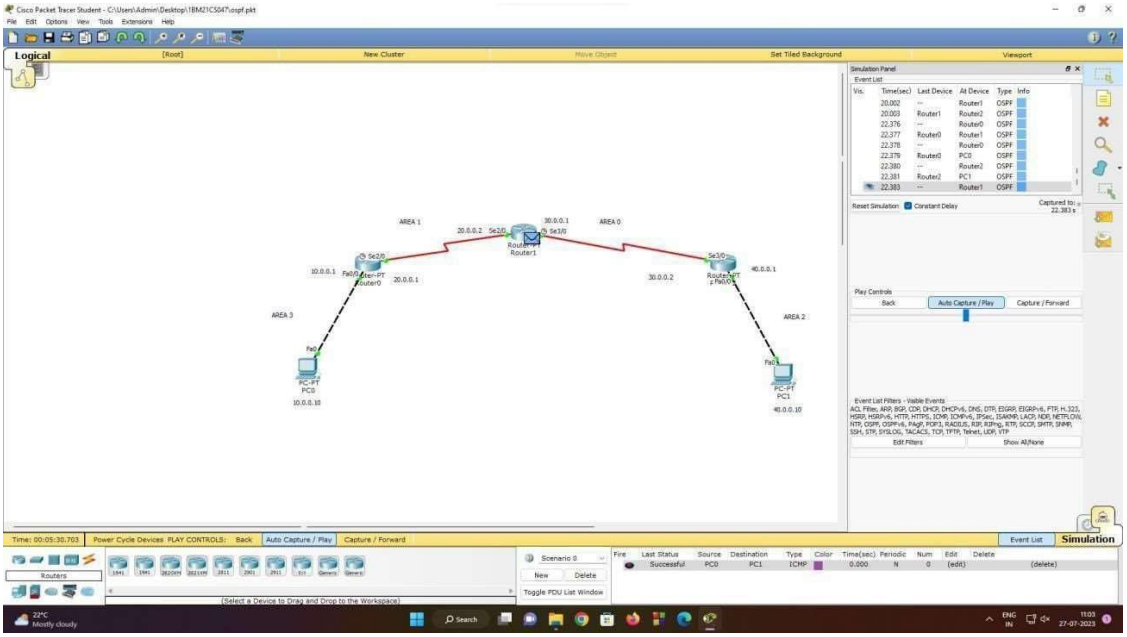
SCREENSHOTS:






AREA 1
20.0.0.2
Se2/0
30.0.0.1
Se3/0
AREA 0
Router-PT
Router1
Se2/0
10.0.0.1
Fa0/0
Router-PT
Router0
20.0.0.1
30.0.0.2
Se3/0
Router-PT
Fa0/0
40.0.0.1
AREA 3
AREA 2
Fa0
PC-PT
PC0
10.0.0.10
Fa0
PC-PT
PC1
40.0.0.10

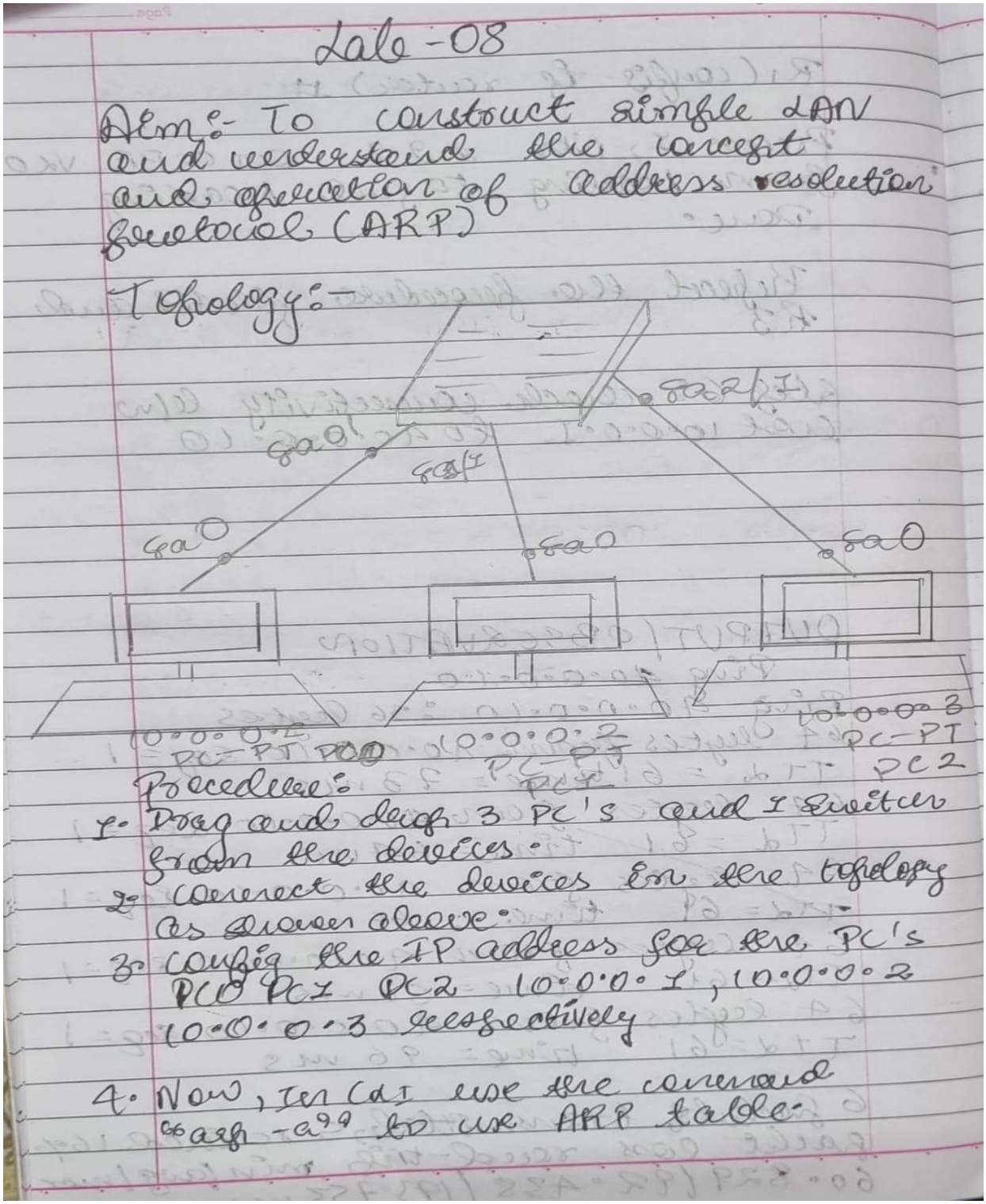

PC0
Physical
Config
Desktop
Custom Interface
Command Prompt
X
Packet Tracer PC Command Line 1.0
PC>ping 40.0.0.10
Pinging 40.0.0.10 with 32 bytes of data:
Reply from 10.0.0.1: Destination host unreachable.
Reply from 10.0.0.1: Destination host unreachable.
Reply from 10.0.0.1: Destination host unreachable.
Reply from 10.0.0.1: Destination host unreachable.
Ping statistics for 40.0.0.10:
Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
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=4ms TTL=125
Reply from 40.0.0.10: bytes=32 time=6ms TTL=125
Reply from 40.0.0.10: bytes=32 time=12ms TTL=125
Ping statistics for 40.0.0.10:
Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
Minimum = 4ms, Maximum = 12ms, Average = 7ms
PC>


Logical
AREA 1
AREA 0
AREA 3
AREA 2
20.0.0.2
30.0.0.1
10.0.0.1
20.0.0.1
30.0.0.2
40.0.0.1
10.0.0.10
40.0.0.10
Router1
Router0
PC0
PC1
Simulation

CN LAB 8

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

OBSERVATION:



Approximate round trip time in milliseconds

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

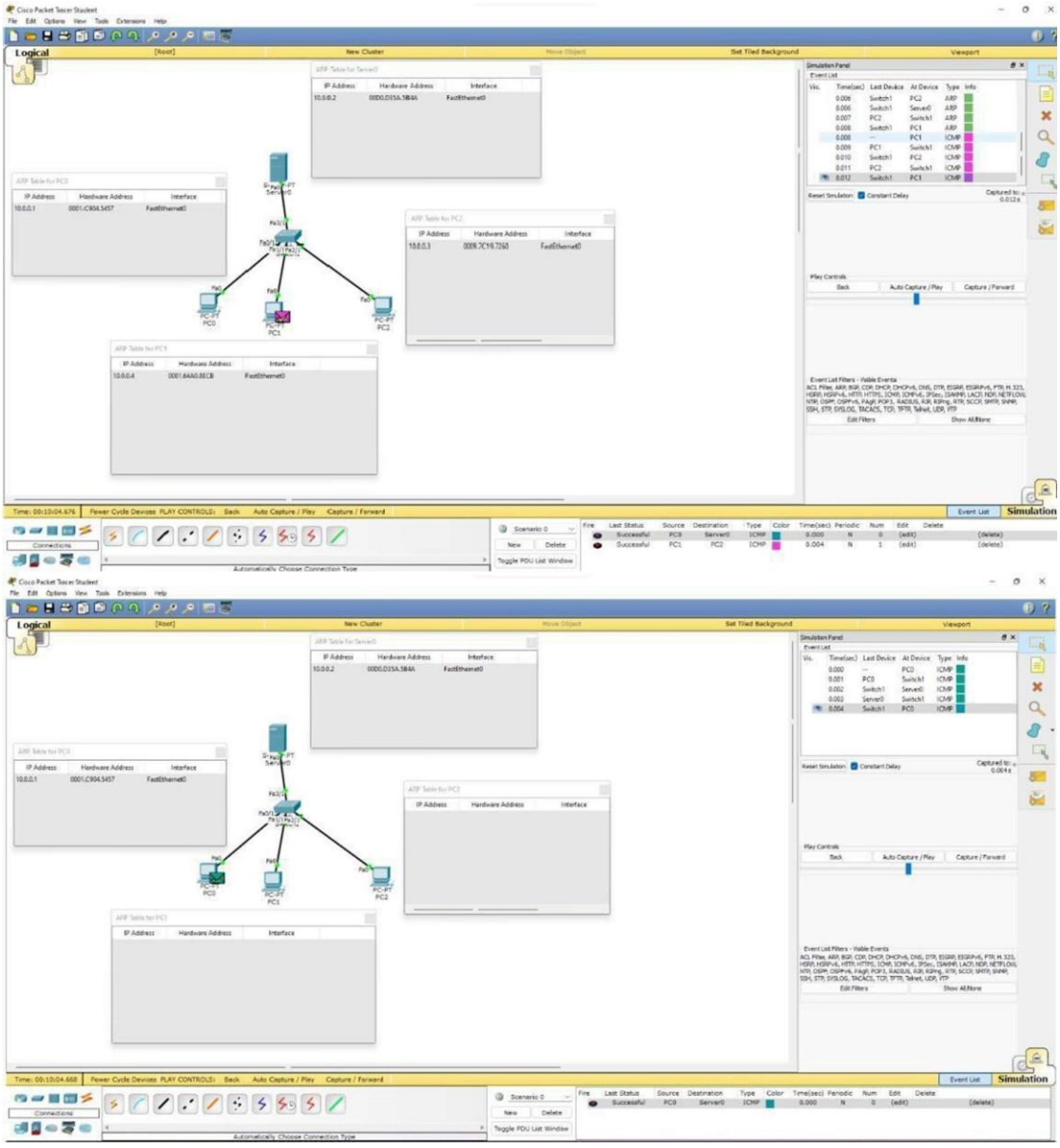
(1) Again check router arp -a command

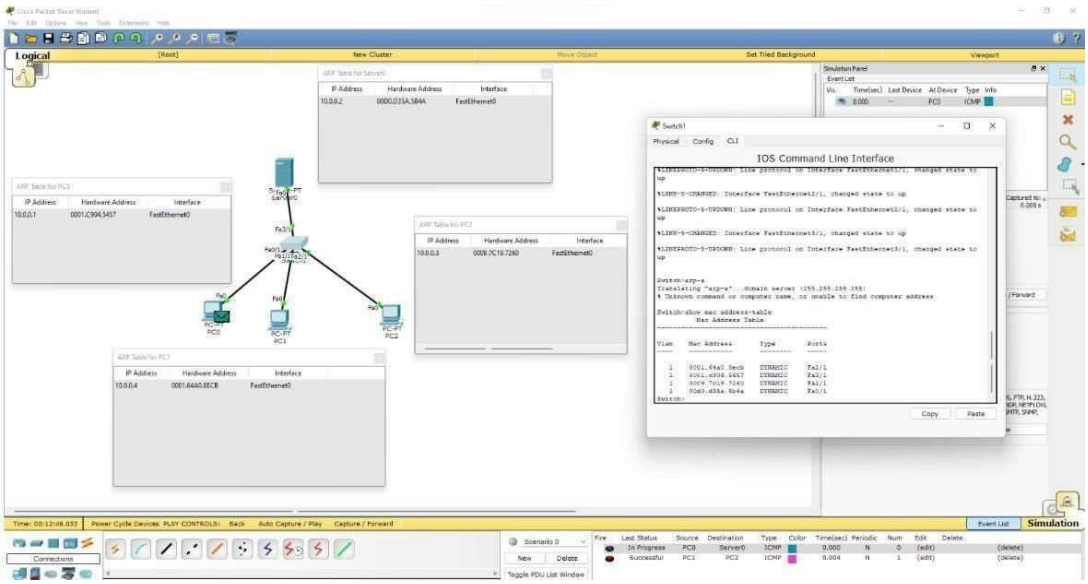
PC > arp -a

Internet address	Physical address	Type
10.0.0.3	0090.2b7c.1589	dynamic

(2) arp -a command is used to clear the table.

SCREENSHOTS:

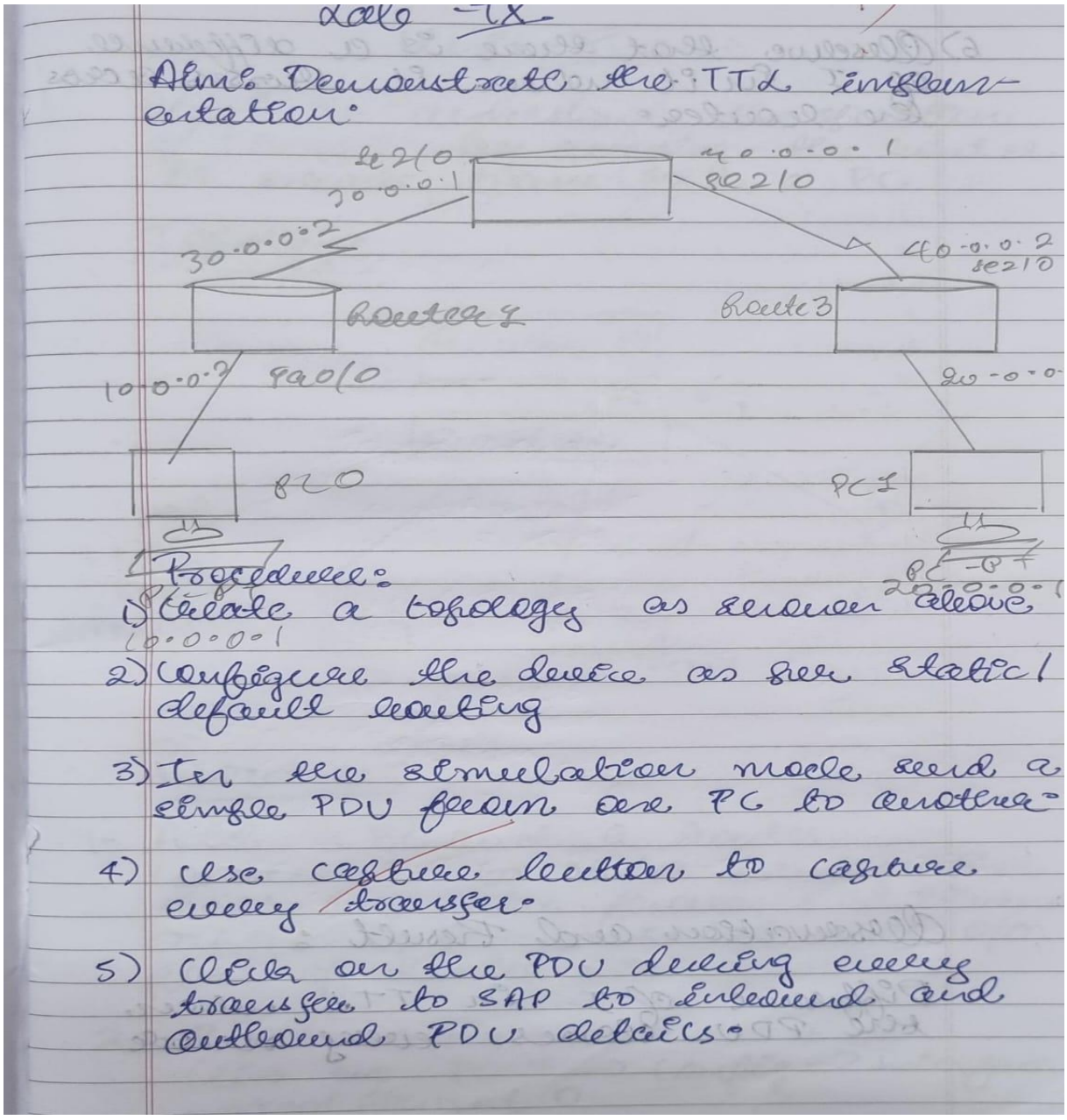



Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
Switch1
Physical
Config
CLI
IOS Command Line Interface
Copy
Paste
Time: 00:12:06.033
Power Cycle Devices
PLAY CONTROLS:
Back
Auto Capture / Play
Capture / Forward
Event List
Simulation
Connections
New
Delete
Toggle PDU List Window

CN LAB 9

AIM: Demonstrate the TTL/ Life of a Packet.

OBSERVATION:

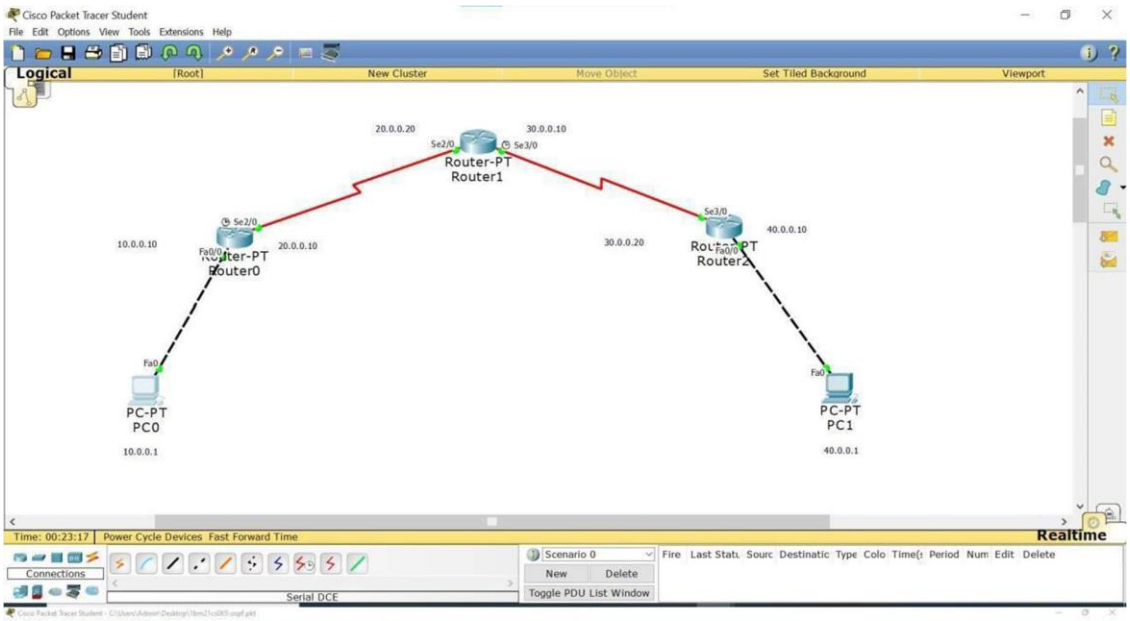


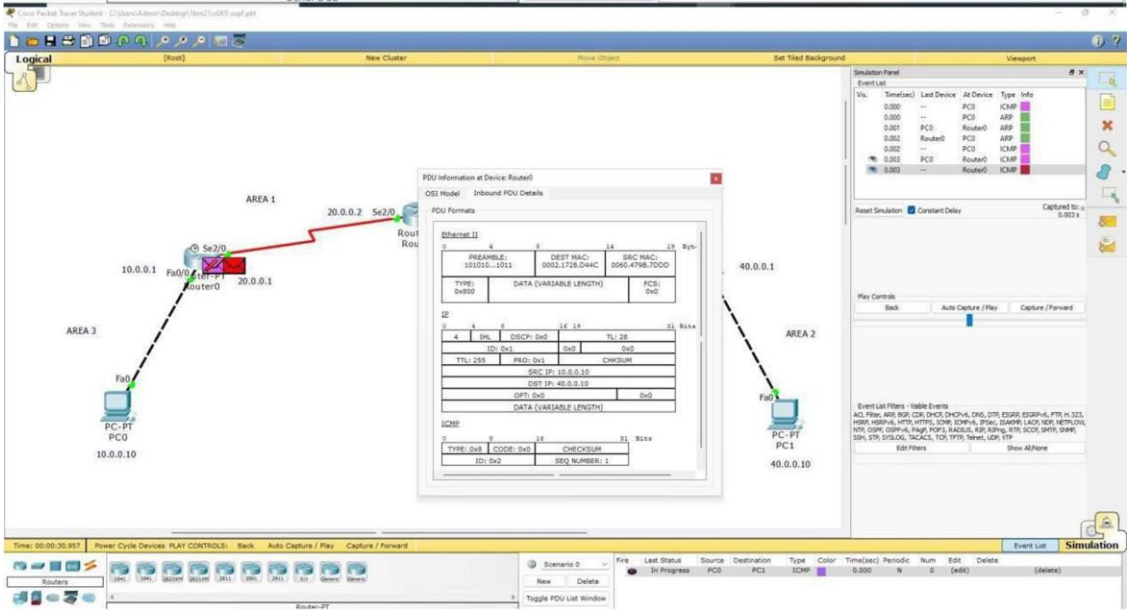
6) Observe that there is a difference of 1 in it when it crosses each router.

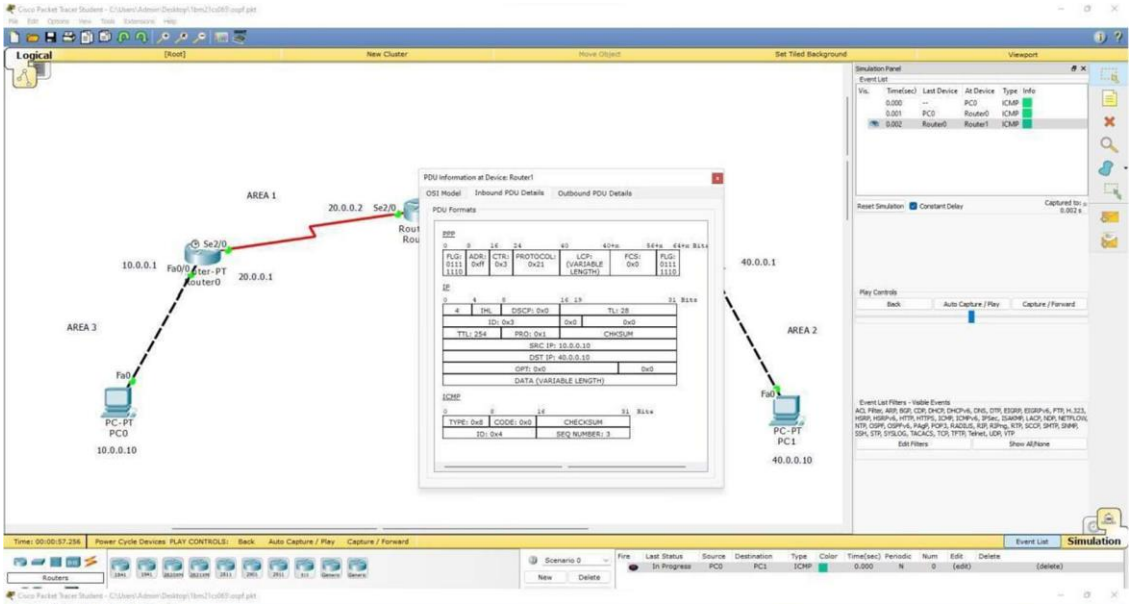
Observation and Result :

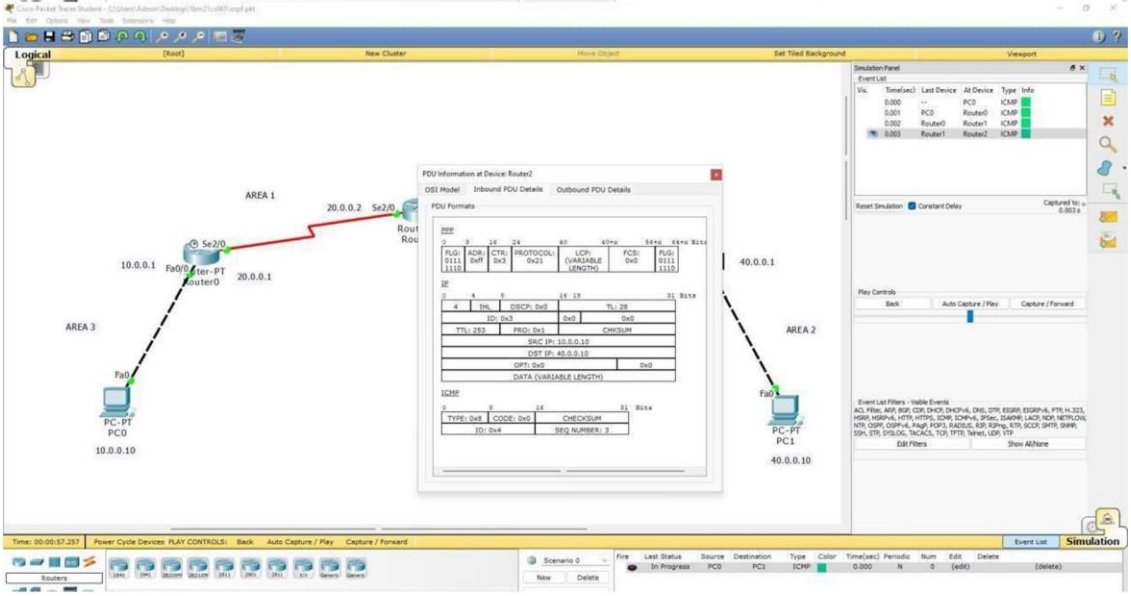
Difference of 1 in TTL when the PDU crosses every router

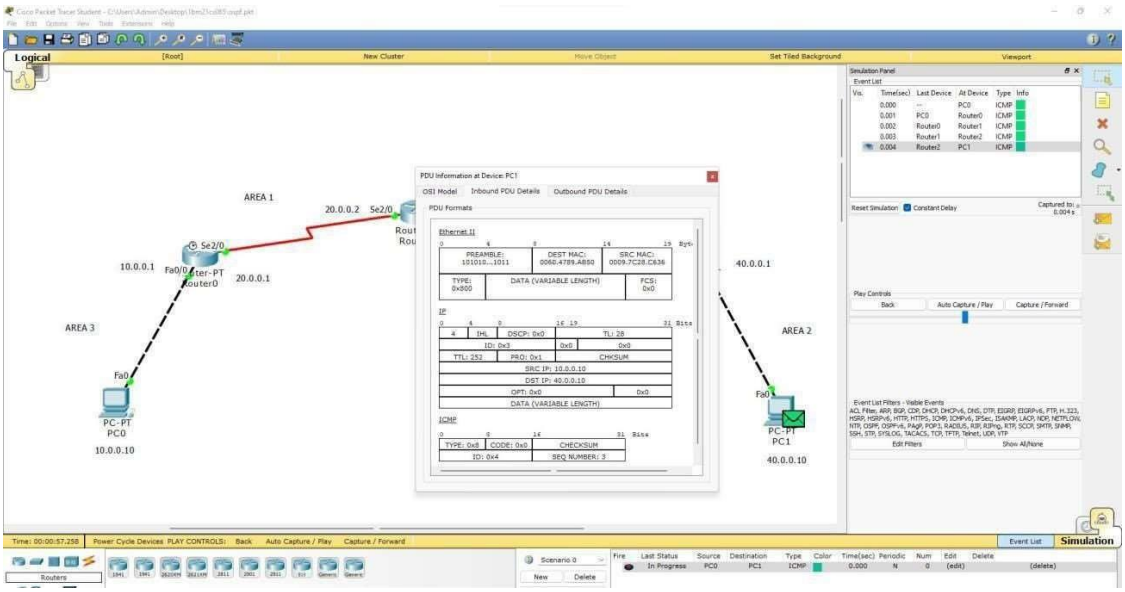
SCREENSHOTS:






AREA 1
20.0.0.2
Se2/0
10.0.0.1
Fa0/0
Router0
20.0.0.1
40.0.0.1
AREA 3
AREA 2
Fa0
PC-PT
PC0
10.0.0.10
Fa0
PC-PT
PC1
40.0.0.10
PDU Information at Device: Router1
OSI Model
Inbound PDU Details
Outbound PDU Details
PDU Formats
PPP
IP
ICMP
SRC IP: 10.0.0.10
DST IP: 40.0.0.10
DATA (VARIABLE LENGTH)
TYPE: 0x8
CODE: 0x0
CHECKSUM
ID: 0x4
SEQ NUMBER: 3
Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
Simulation Panel
Event List
Play Controls
Back
Auto Capture / Play
Capture / Forward
Edit Filters
Show All/None
Time: 00:00:57.256
Power Cycle Devices
PLAY CONTROLS:
Event List
Simulation
Routers
Scenario 0
New
Delete
Fire
Last Status
Source
Destination
Type
Color
Time(sec)
Periodic
Num
Edit
Delete
In Progress
PC0
PC1
ICMP
0.000
N
0
(edit)
(delete)

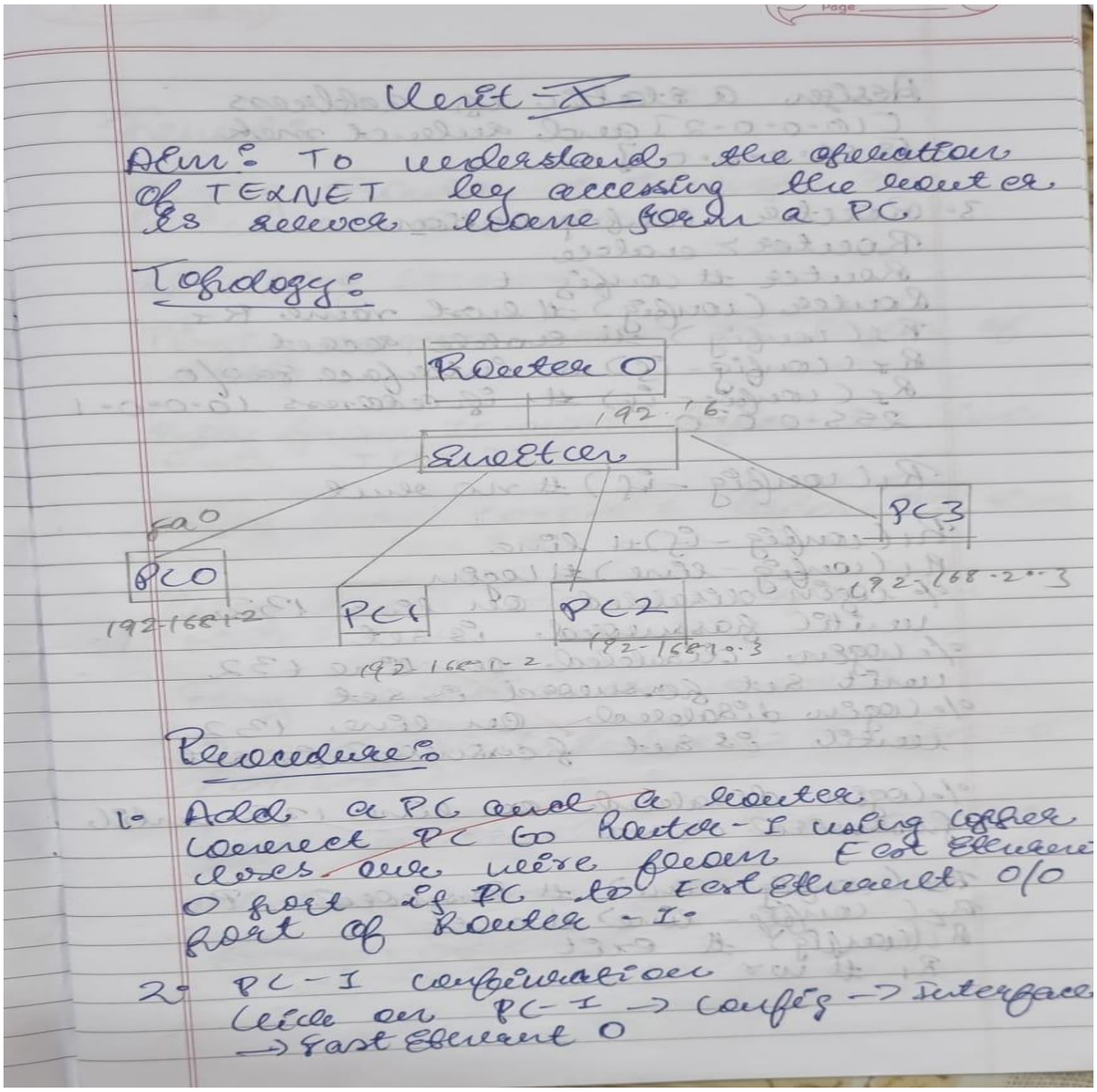

AREA 1
20.0.0.2
Se2/0
10.0.0.1
Fa0/0
Router0
20.0.0.1
40.0.0.1
AREA 3
AREA 2
Fa0
PC-PT
PC0
10.0.0.10
Fa0
PC-PT
PC1
40.0.0.10
PDU Information at Device: Router2
OSI Model
Inbound PDU Details
Outbound PDU Details
PDU Formats
PPP
IP
ICMP
SRC IP: 10.0.0.10
DST IP: 40.0.0.10
DATA (VARIABLE LENGTH)
TYPE: 0x8
CODE: 0x0
CHECKSUM
ID: 0x4
SEQ NUMBER: 3
Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
Simulation Panel
Event List
Play Controls
Back
Auto Capture / Play
Capture / Forward
Edit Filters
Show All/None
Time: 00:00:57.257
Power Cycle Devices
PLAY CONTROLS:
Event List
Simulation
Routers
Scenario 0
New
Delete
Fire
Last Status
Source
Destination
Type
Color
Time(sec)
Periodic
Num
Edit
Delete
In Progress
PC0
PC1
ICMP
0.000
N
0
(edit)
(delete)


Logical
AREA 1
20.0.0.2
Se2/0
10.0.0.1
Fa0/0
20.0.0.1
AREA 3
AREA 2
40.0.0.1
Fa0
PC-PT
PC0
10.0.0.10
PC-PT
PC1
40.0.0.10
PDU Information at Device: PC1
OSI Model
Inbound PDU Details
Outbound PDU Details
PDU Formats
Ethernet II
IP
ICMP
Simulation Panel
Event List
Simulation

CN LAB 10

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

OBSERVATION:



Assign a static ip address [10.0.0.2] and subnet mask [255.0.0.0].

3. Router - Configuration

Router > enable
Router # config t
Router (config) # host name R1
R1 (config) # enable secret
R1 (config-if) # interface fa 0/0
R1 (config-if) # ip address 10.0.0.1 255.0.0.0

R1 (config-if) # no shut

R1 (config-if) # line
R1 (config-line) # login
% login disabled on line 132 until password is set
% login disabled on line 132 until set password is set
% login disabled on line 132 until is set password is set

% login disabled on line 132 until password is set

R1 (config-line) # Password P
R1 (config-line) # exit
R1 (config) # exit
R1 # w

Result:

Pinging Router I through PC in command prompt

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 = 0ms TTL = 255

Reply from 10.0.0.1 bytes = 32 time = 0ms TTL = 255

Ping statistics for 10.0.0.1

Packets = sent = Received = 4 lost = 0 (0% loss)

Approximate round trip time in ms

Minimum = 0ms, maximum = 0ms

Average = 0ms

SCREENSHOTS:

PC0

Physical | Config | Desktop | Custom Interface

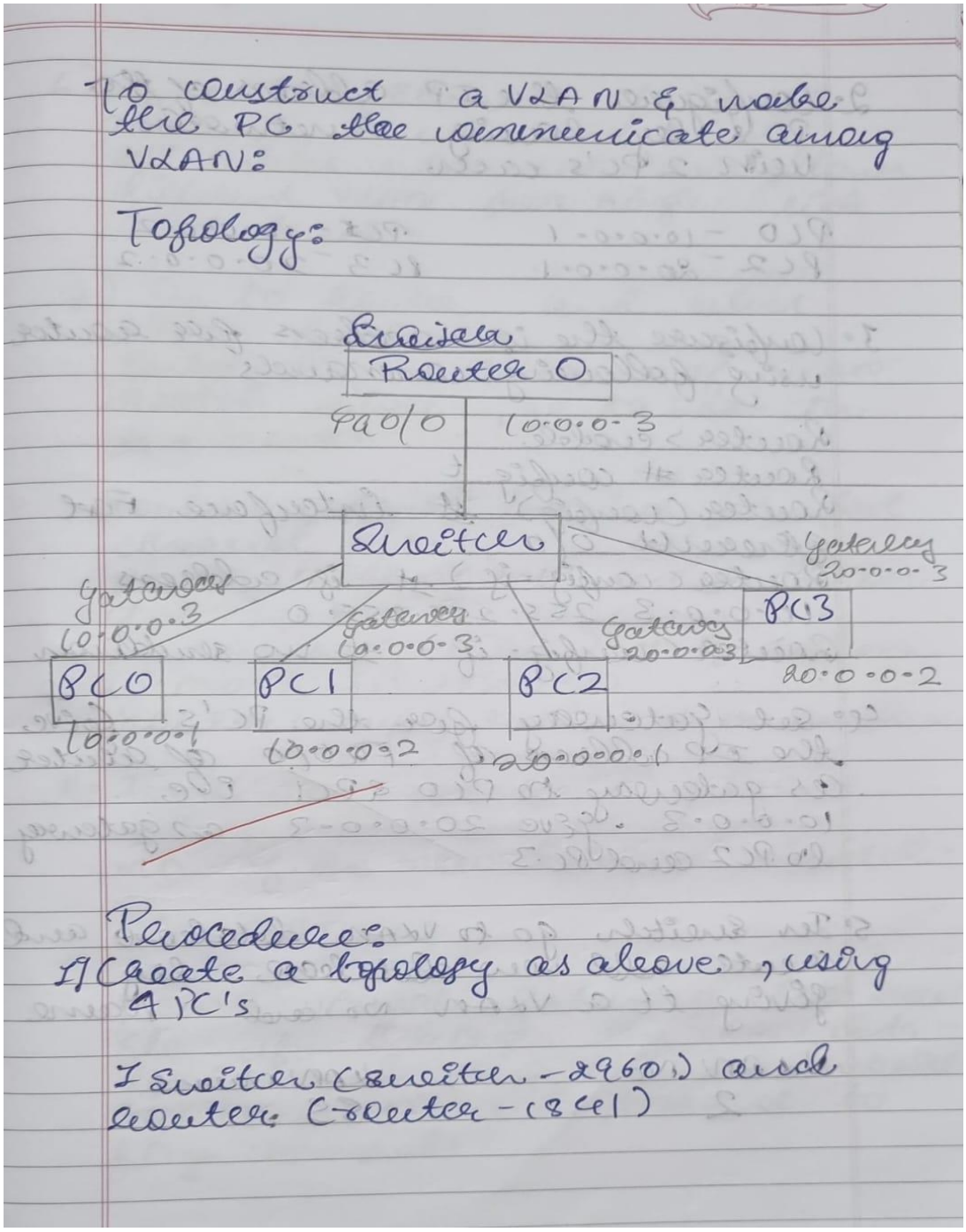
Command Prompt

```
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=1ms TTL=255
Reply from 10.0.0.1: bytes=32 time=0ms TTL=255
Reply from 10.0.0.1: bytes=32 time=0ms TTL=255
Reply from 10.0.0.1: bytes=32 time=0ms 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 = 1ms, Average = 0ms

PC>telnet 10.0.0.1
Trying 10.0.0.1 ...Open


User Access Verification

Password:
% Password:  timeout expired!

[Connection to 10.0.0.1 closed by foreign host]
PC>telnet 10.0.0.1
Trying 10.0.0.1 ...Open


User Access Verification

Password:
Password:
Password:

[Connection to 10.0.0.1 closed by foreign host]
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, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

C    10.0.0.0/8 is directly connected, FastEthernet0/0
r1#
```

CN LAB 11

AIM: To construct a VLAN and make a pc communicate among VLAN.

OBSERVATION:



2. Configure the IP address of the PCs by having 2 networks with 2 PC's each

PC0 - 10.0.0.1 PC1 - 10.0.0.2
PC2 - 20.0.0.1 PC3 - 20.0.0.2

3. Configure the ip address for router using following commands

Router > enable
Router # config t
Router (config) # interface Fast Ethernet 0/0
Router (config-if) # ip address 10.0.0.3 255.255.255.0
Router (config-if) # no shutdown

4. Set gateway for the PC's. Give the IP address of Fa0/0 of router as gateway to PC0 & PC1 i.e. 10.0.0.3. Give 20.0.0.3 as gateway to PC2 and PC3

5. In switch go to VLAN database and create add new database by giving it a VLAN No and Name

VLAN No. VLAN name
2 SS.

Result:

PC2:

Ping 20.0.0.1

Pinging 20.0.0.1 with 32 bytes of data:

Reply from 20.0.0.1 bytes = 32 time=2ms TTL = 128

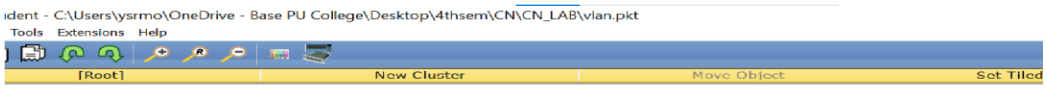
Reply from 20.0.0.1 bytes = 32 time=2ms TTL = 128

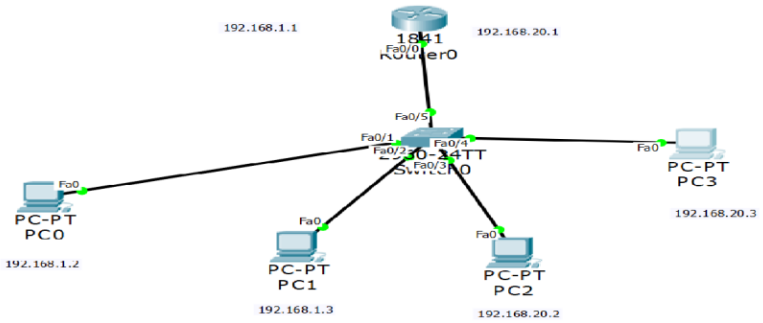
Reply from 20.0.0.1 bytes = 32 time=4ms TTL = 128

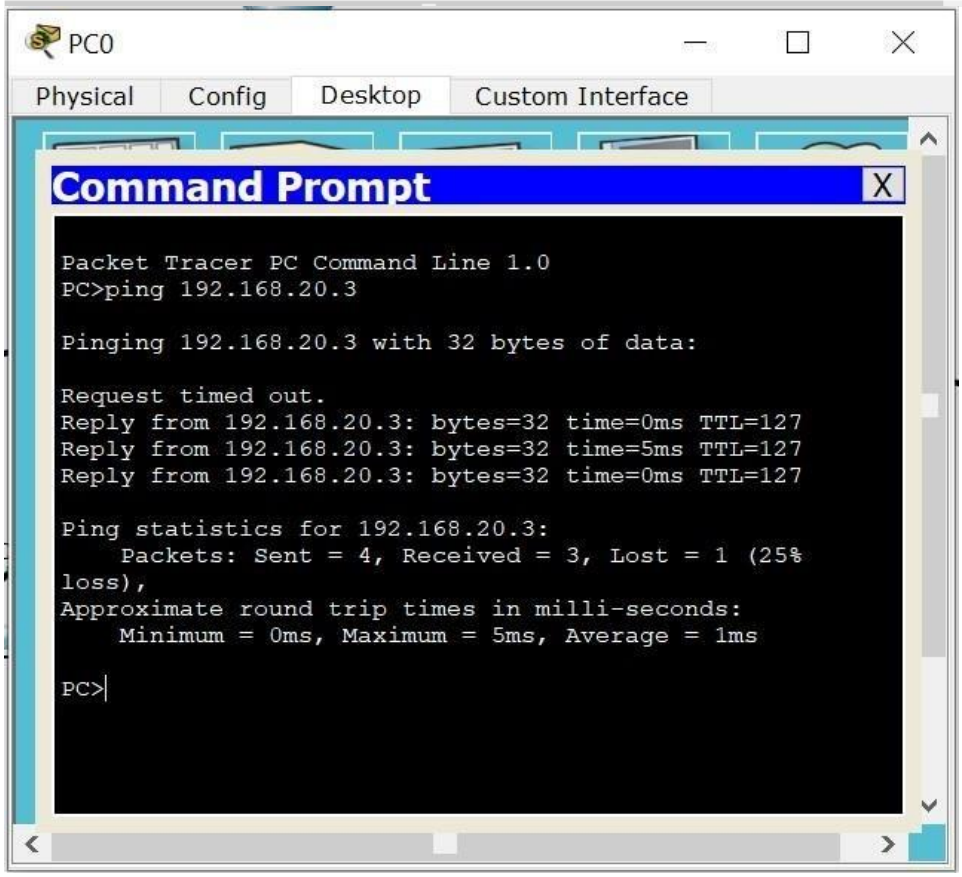
Ping statistics for 20.0.0.1

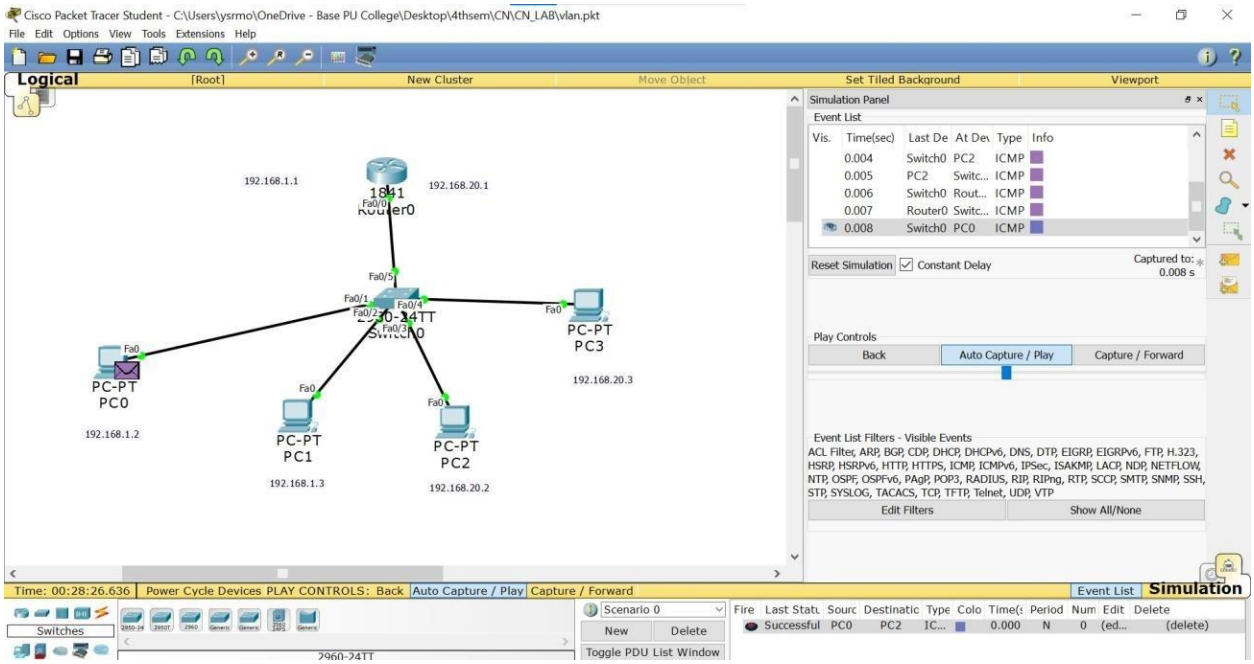
Packets Sent = 4 Received = 4 Lost = 0 (0% loss)

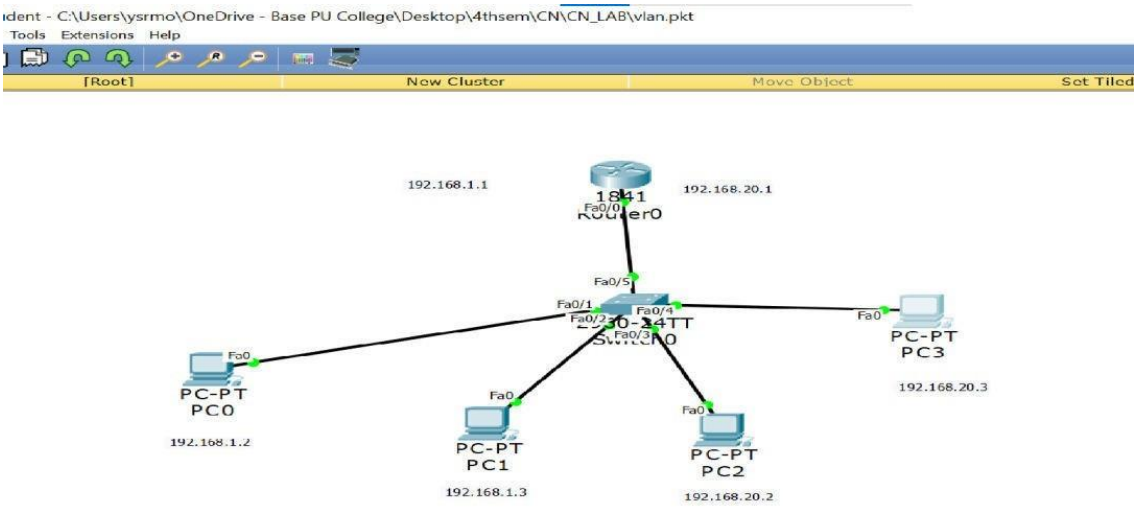
SCREENSHOTS:

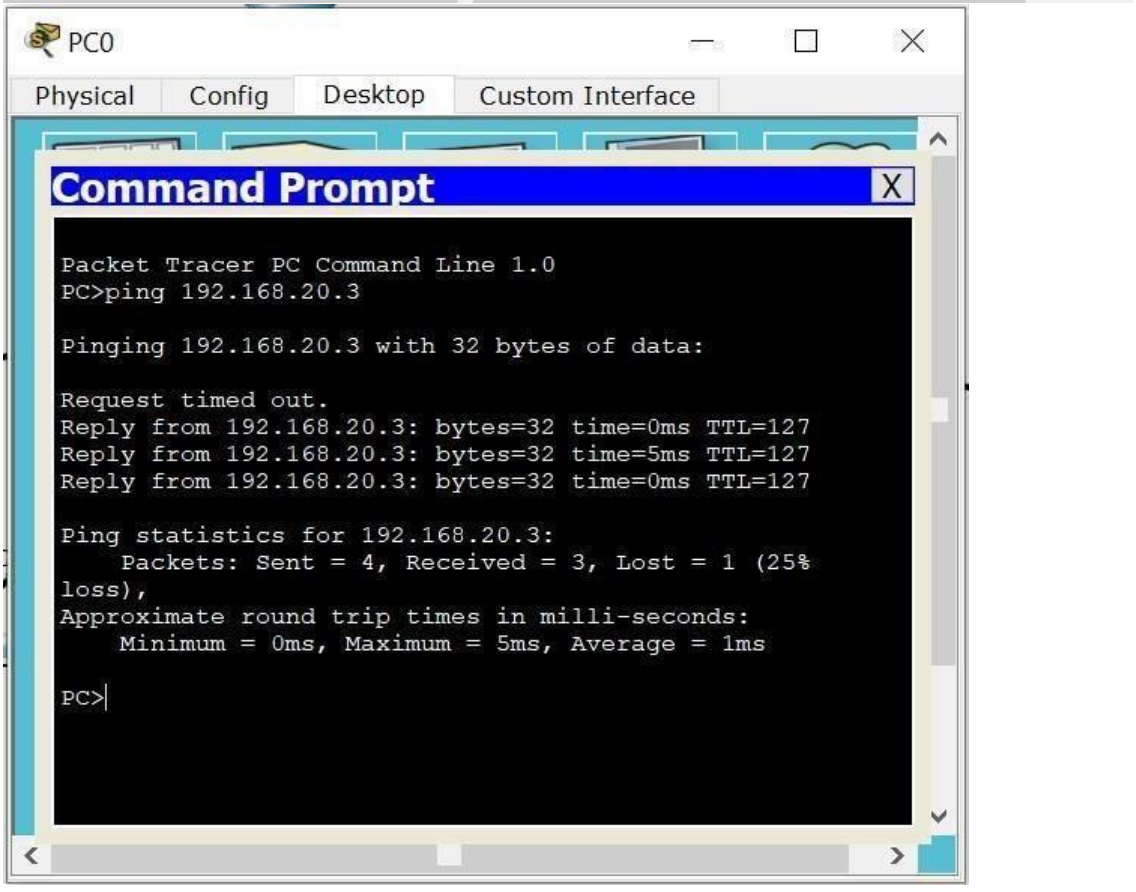







Cisco Packet Tracer Student - C:\Users\ysrmo\OneDrive - Base PU College\Desktop\4thsem\CN\CN_LAB\vlan.pkt
File Edit Options View Tools Extensions Help
Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
192.168.1.1
192.168.20.1
1841
Router0
Fa0/0
Fa0/5
Fa0/1
Fa0/2
Fa0/3
Fa0/4
2960-24TT
Switch0
Fa0
PC-PT
PC0
192.168.1.2
PC-PT
PC1
192.168.1.3
PC-PT
PC2
192.168.20.2
PC-PT
PC3
192.168.20.3
Simulation Panel
Event List
Vis. Time(sec) Last De At Dev Type Info
0.004 Switch0 PC2 ICMP
0.005 PC2 Switc... ICMP
0.006 Switch0 Rout... ICMP
0.007 Router0 Switc... ICMP
0.008 Switch0 PC0 ICMP
Reset Simulation
Constant Delay
Captured to:
0.008 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: 00:28:26.636
Power Cycle Devices PLAY CONTROLS: Back
Auto Capture / Play
Capture / Forward
Event List
Simulation
Switches
2960-24TT
Scenario 0
New
Delete
Toggle PDU List Window
Fire Last Statu Sourc Destinatio Type Colo Time(s Period Num Edit Delete
Successful PC0 PC2 IC... 0.000 N 0 (ed... (delete)

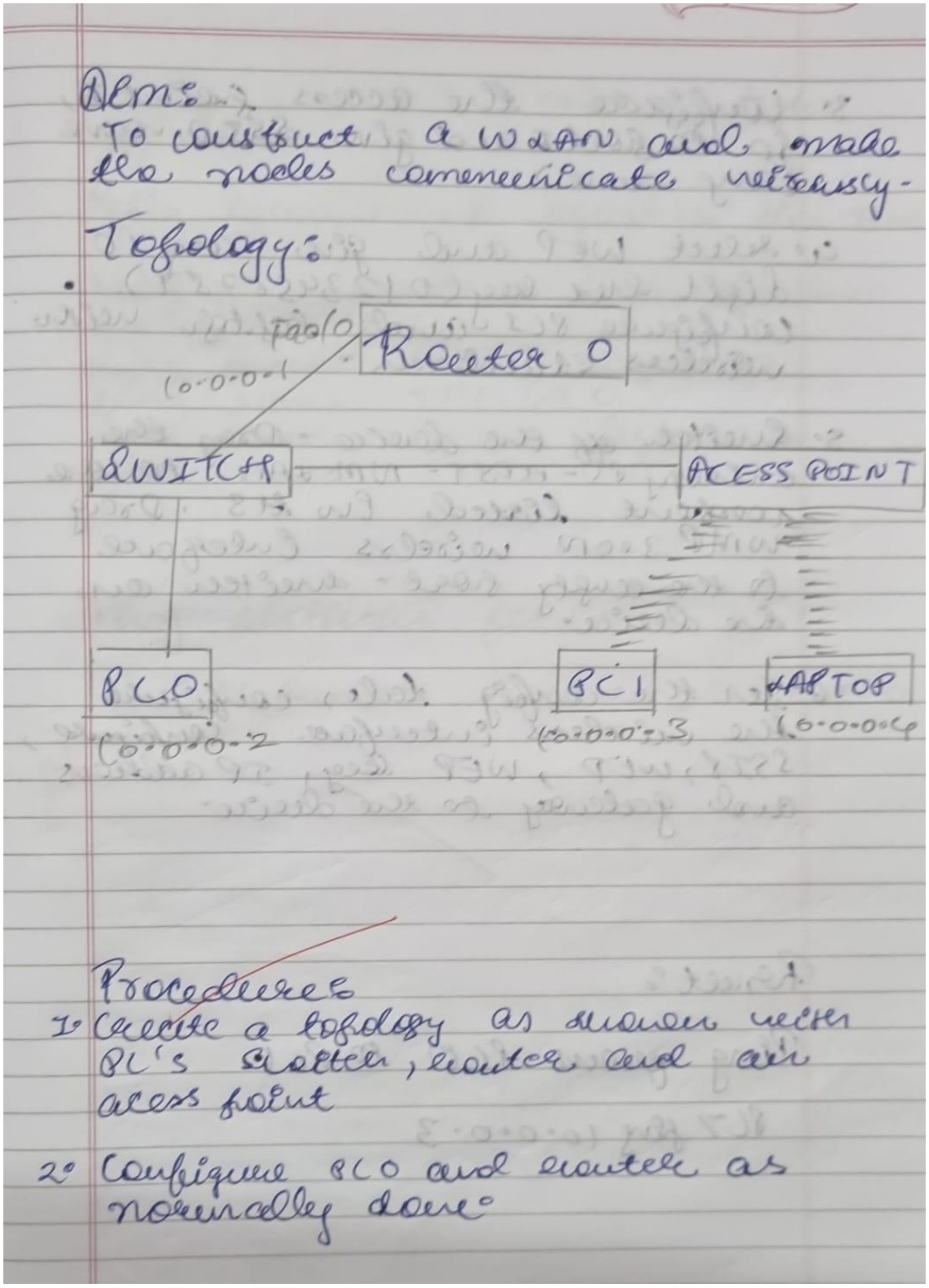

ident - C:\Users\ysrmo\OneDrive - Base PU College\Desktop\4thsem\CN\CN_LAB\vlan.pkt
Tools Extensions Help
[Root]
New Cluster
Move Object
Set Tiled
192.168.1.1
1841
Router0
192.168.20.1
Fa0/0
Fa0/5
Fa0/1
Fa0/2
Fa0/4
Fa0/3
Switch0
Fa0
PC-PT
PC0
192.168.1.2
PC-PT
PC1
192.168.1.3
PC-PT
PC2
192.168.20.2
PC-PT
PC3
192.168.20.3


PC0
Physical
Config
Desktop
Custom Interface
Command Prompt
X
Packet Tracer PC Command Line 1.0
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=5ms TTL=127
Reply from 192.168.20.3: bytes=32 time=0ms TTL=127
Ping statistics for 192.168.20.3:
Packets: Sent = 4, Received = 3, Lost = 1 (25%
loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 5ms, Average = 1ms
PC>

CN LAB 12

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

OBSERVATION:



3. Configure the access point I, go to port I and give SSID name (any name)

4. Select WEP and give any 10 digit hex key (0123456789). Configure PC I and laptop with wireless standards.

5. Switch off the device. Drag the existing PT-HOST-NM-1AM to the empty slot. Drag WMP 300N wireless interface to the empty port. Switch on the device.

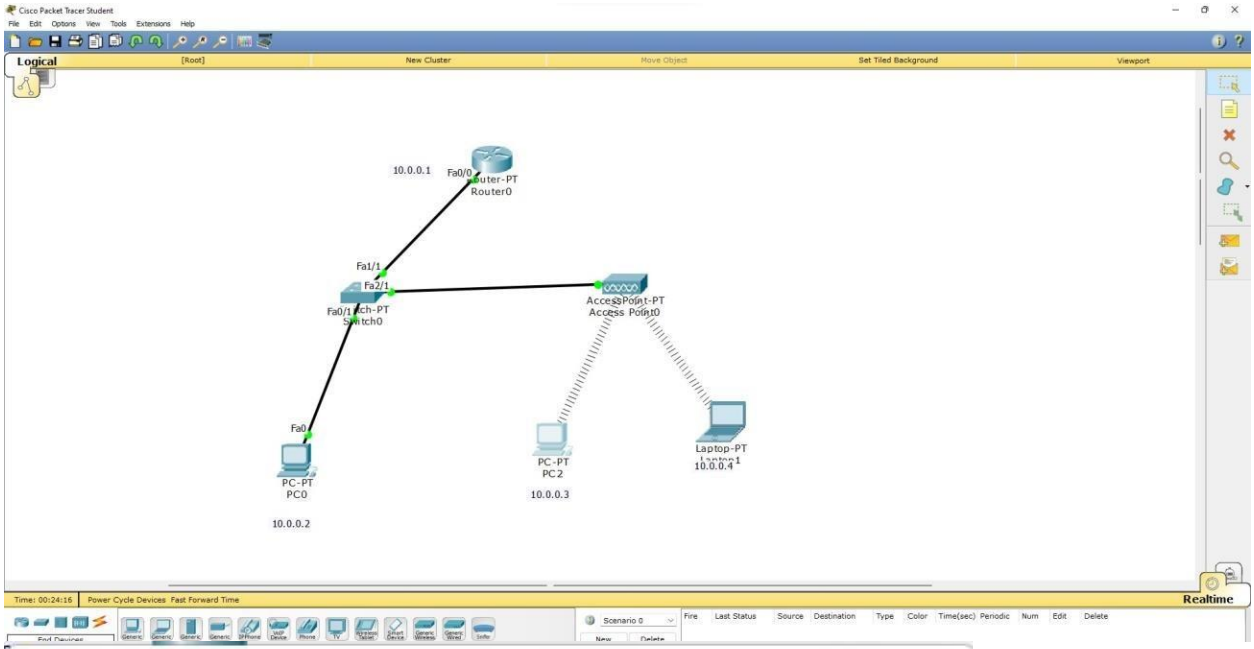
6. Go to the config tab, configure the wireless interface. Configure, SSID, WEP, WEP key, IP address and gateway to the device.

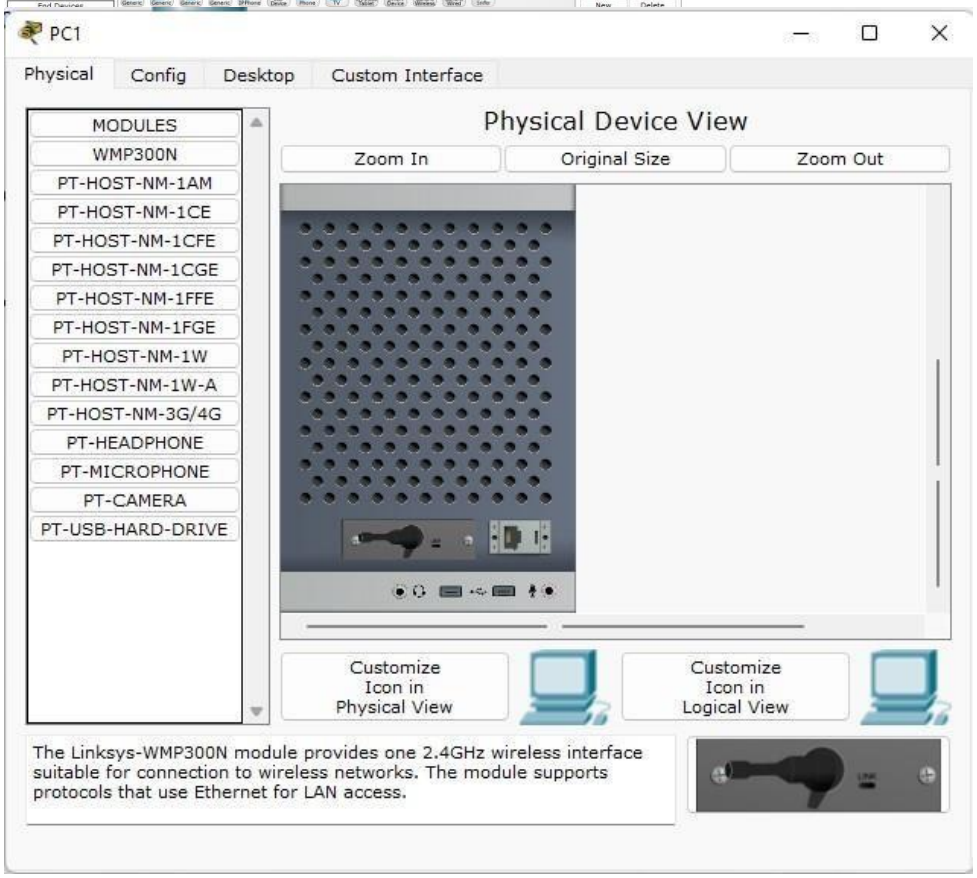
Result:

Ping from PC0 to PC1.

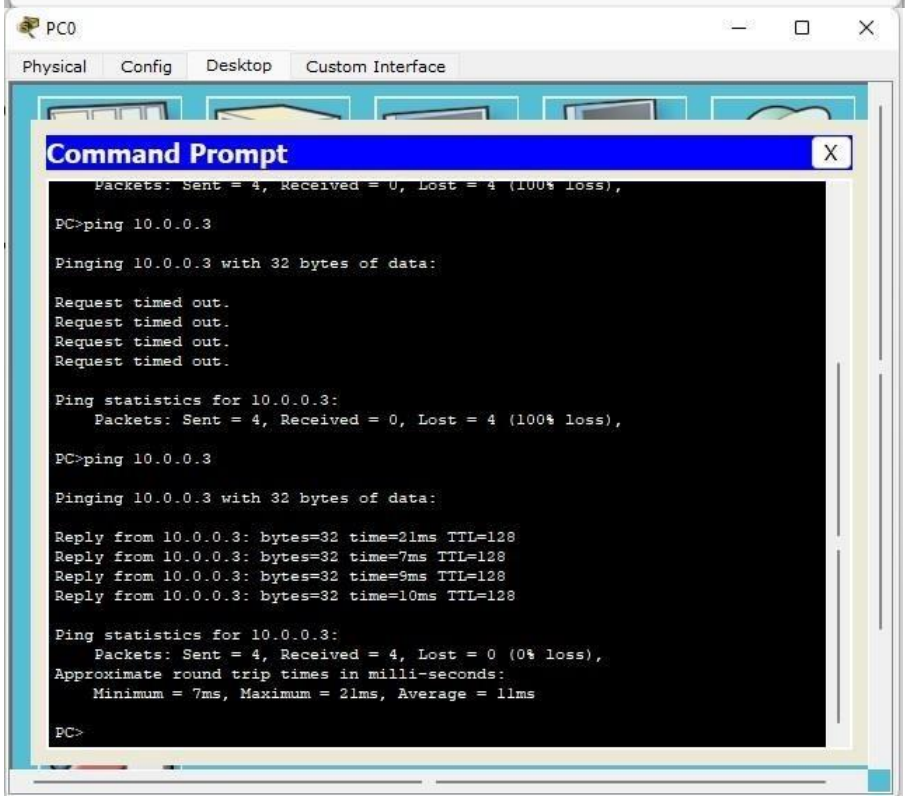
PC1 ping 10.0.0.3

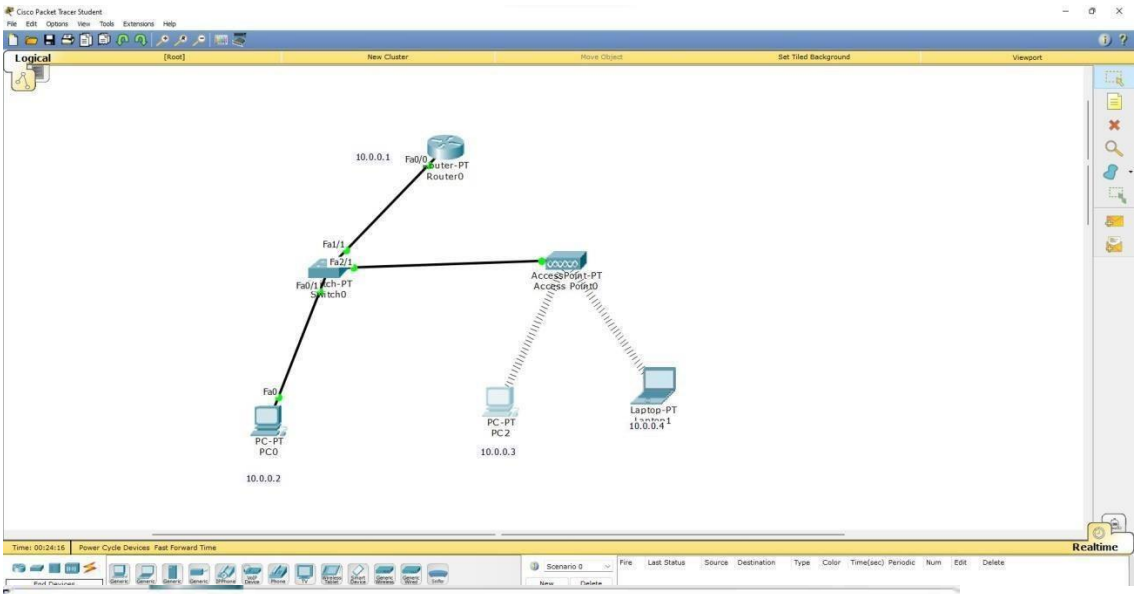
SCREENSHOTS:

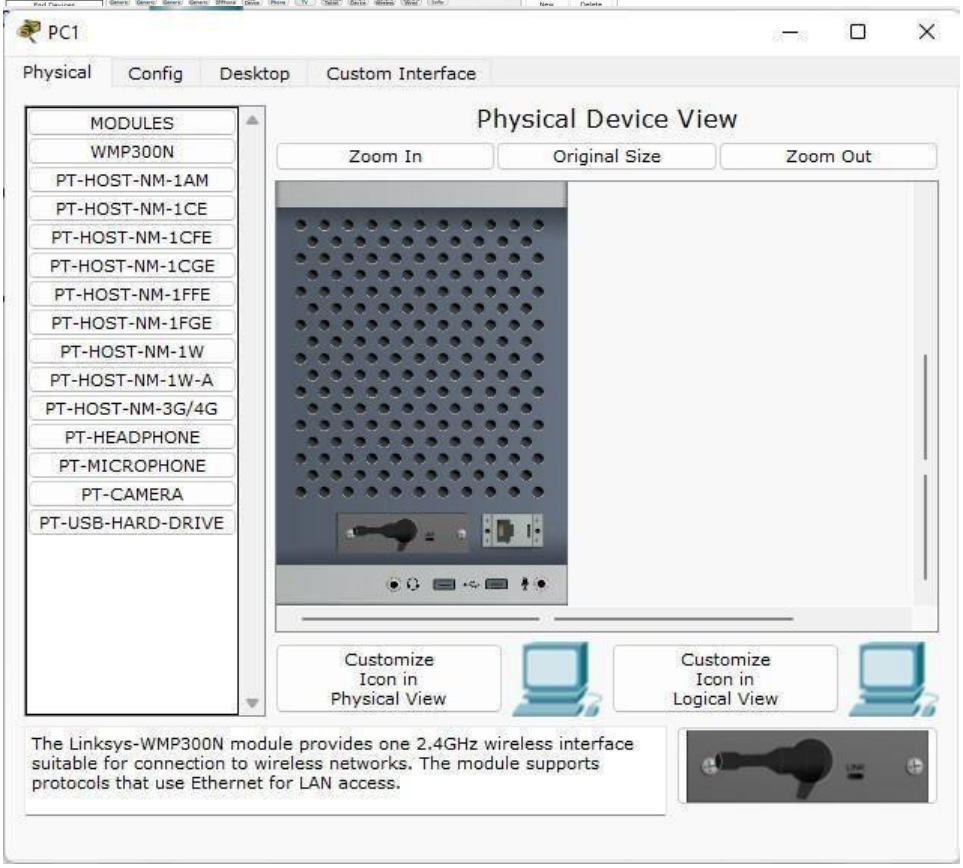




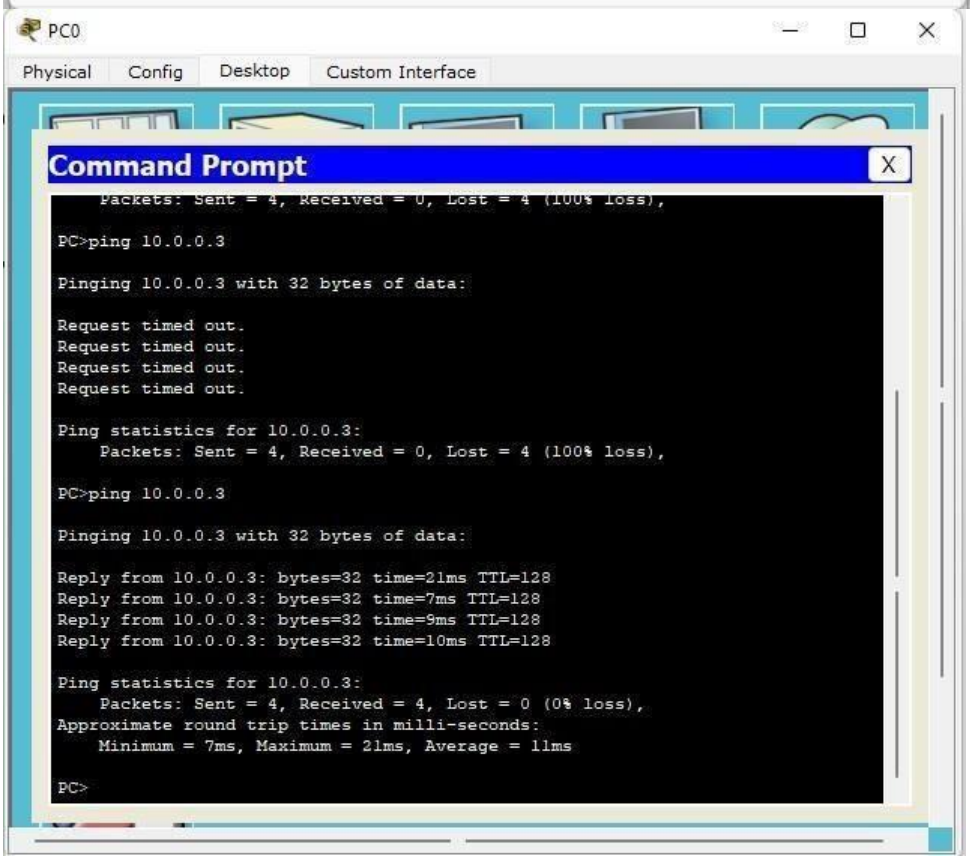

Laptop0
Physical
Config
Desktop
Custom Interface
MODULES
WPC300N
PT-LAPTOP-NM-1AM
PT-LAPTOP-NM-1CE
PT-LAPTOP-NM-1CFE
PT-LAPTOP-NM-1CGE
PT-LAPTOP-NM-1FFE
PT-LAPTOP-NM-1FGE
PT-LAPTOP-NM-1W
PT-LAPTOP-NM-1W-A
PT-LAPTOP-NM-3G/4G
PT-HEADPHONE
PT-MICROPHONE
PT-CAMERA
PT-USB-HARD-DRIVE
Physical Device View
Zoom In
Original Size
Zoom Out
Customize Icon in Physical View
Customize Icon in Logical View
The Linksys-WPC300N module provides one 2.4GHz wireless interface suitable for connection to wireless networks. The module supports protocols that use Ethernet for LAN access.


PC0
Physical
Config
Desktop
Custom Interface
Command Prompt
X
Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
PC>ping 10.0.0.3
Pinging 10.0.0.3 with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.
Ping statistics for 10.0.0.3:
Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
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=21ms TTL=128
Reply from 10.0.0.3: bytes=32 time=7ms TTL=128
Reply from 10.0.0.3: bytes=32 time=9ms 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 = 7ms, Maximum = 21ms, Average = 11ms
PC>


Logical
10.0.0.1
Fa0/0
Router0
Fa1/1
Fa2/1
Access Point-PT
Access Point0
Fa0
PC-PT
PC0
10.0.0.2
PC-PT
PC2
10.0.0.3
Laptop-PT
10.0.0.4
Realtime


PC1
Physical
Config
Desktop
Custom Interface
MODULES
WMP300N
PT-HOST-NM-1AM
PT-HOST-NM-1CE
PT-HOST-NM-1CFE
PT-HOST-NM-1CGE
PT-HOST-NM-1FFE
PT-HOST-NM-1FGE
PT-HOST-NM-1W
PT-HOST-NM-1W-A
PT-HOST-NM-3G/4G
PT-HEADPHONE
PT-MICROPHONE
PT-CAMERA
PT-USB-HARD-DRIVE
Physical Device View
Zoom In
Original Size
Zoom Out
Customize Icon in Physical View
Customize Icon in Logical View
The Linksys-WMP300N module provides one 2.4GHz wireless interface suitable for connection to wireless networks. The module supports protocols that use Ethernet for LAN access.

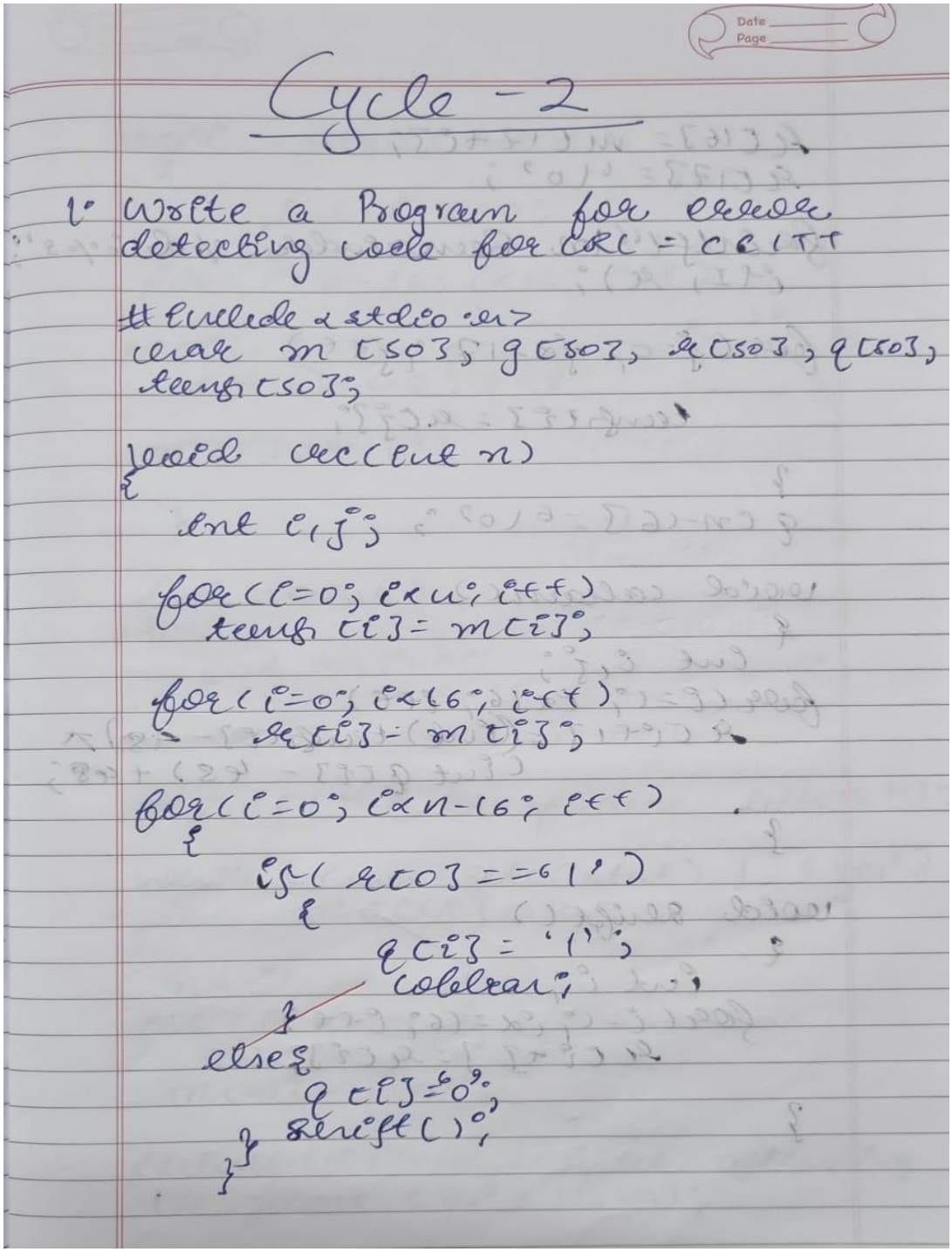

Laptop0
Physical
Config
Desktop
Custom Interface
MODULES
WPC300N
PT-LAPTOP-NM-1AM
PT-LAPTOP-NM-1CE
PT-LAPTOP-NM-1CFE
PT-LAPTOP-NM-1CGE
PT-LAPTOP-NM-1FFE
PT-LAPTOP-NM-1FGE
PT-LAPTOP-NM-1W
PT-LAPTOP-NM-1W-A
PT-LAPTOP-NM-3G/4G
PT-HEADPHONE
PT-MICROPHONE
PT-CAMERA
PT-USB-HARD-DRIVE
Physical Device View
Zoom In
Original Size
Zoom Out
Customize Icon in Physical View
Customize Icon in Logical View
The Linksys-WPC300N module provides one 2.4GHz wireless interface suitable for connection to wireless networks. The module supports protocols that use Ethernet for LAN access.


PC0
Physical
Config
Desktop
Custom Interface
Command Prompt
X
Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
PC>ping 10.0.0.3
Pinging 10.0.0.3 with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.
Ping statistics for 10.0.0.3:
Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
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=21ms TTL=128
Reply from 10.0.0.3: bytes=32 time=7ms TTL=128
Reply from 10.0.0.3: bytes=32 time=9ms 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 = 7ms, Maximum = 21ms, Average = 11ms
PC>

CN LAB 13

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

OBSERVATION:



Cycle - 2

1. Write a Program for error detecting code for CRC - CCITT

```
#include<stdio.h>
char m[50], g[50], a[50], q[50], temp[50];

void crc(int n)
{
  int i, j;

  for(i=0; i<n; i++)
    temp[i] = m[i];

  for(i=0; i<16; i++)
    a[i] = m[i];

  for(i=0; i<n-16; i++)
  {
    if(a[0]=='1')
    {
      q[i] = '1';
      colshift();
    }
    else{
      q[i]='0';
      shift();
    }
  }
```

```
void caesar(int n)
{
    int i, k=0;
for(i=n-16; i<n; i++)
m[i] = ((int)m[i]-48)^((int)
c[k+1]-48)+48;
m[i] = '\0';
}
int main()
{
  int n, i=0;
char ch, flag=0;
printf("Enter plaintext: ");
while((ch=getch(stdin)) != '\r')
  m[i++] = ch;
n=i;
for(i=0; i<16; i++)
  m[n++] = '0';
m[n] = '\0';
printf("Message after appending
16 zeroes: %s", m);
```

```
for (i=0; i<16; i++)
    g[i]='0';
g[0]=g[4]=g[11]=g[16]='1';
g[17]='1';
printf("\n generator:");
for(i=0;i<=17;i++)
printf("%d",g[i]);
crc(n);

printf("\n\n Transmitted frame:");
for(i=0;i<n;i++)
printf("%d",m[i]);

printf("\n Enter transmitted frame:");
for(i=0;i<n;i++)
scanf("%d",&m[i]);

printf("\n CRC checking\n");
crc(n);

printf("\n\n Error Remainder:");
for(i=0;i<16;i++)
printf("%d",g[i]);

for(i=0;i<16;i++)
    if(g[i]!='0')
        flag=1;
    else
        continue;
```

```
if(flag==1)
  printf("Error during
  Transmission");

else
printf("Recieved frame is
correct");
}
```

Output:

Enter frame bits: 1011

Message after appending 16 zeroes

1011 0000 0000 0000 0000

Generator: 1000 1000 0000 00001

Quotient: 1011

Transmitted: 1011 1011 0001 0110 1011

Enter transmitted frame

1011 1011 0001 0110 1011

Last Remainder: 000 0000 0000 0000

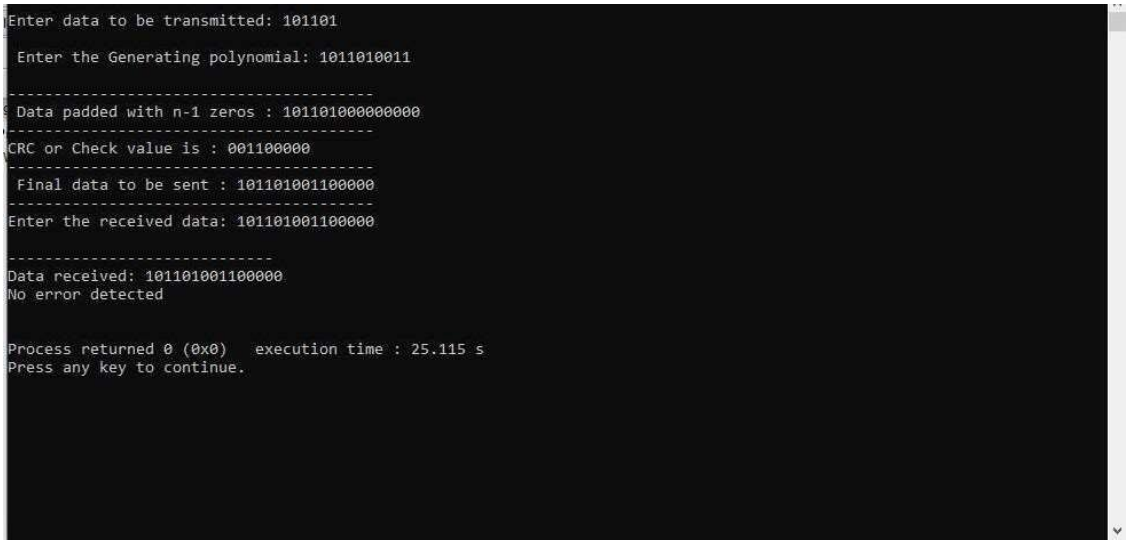
Recieved frame is correct

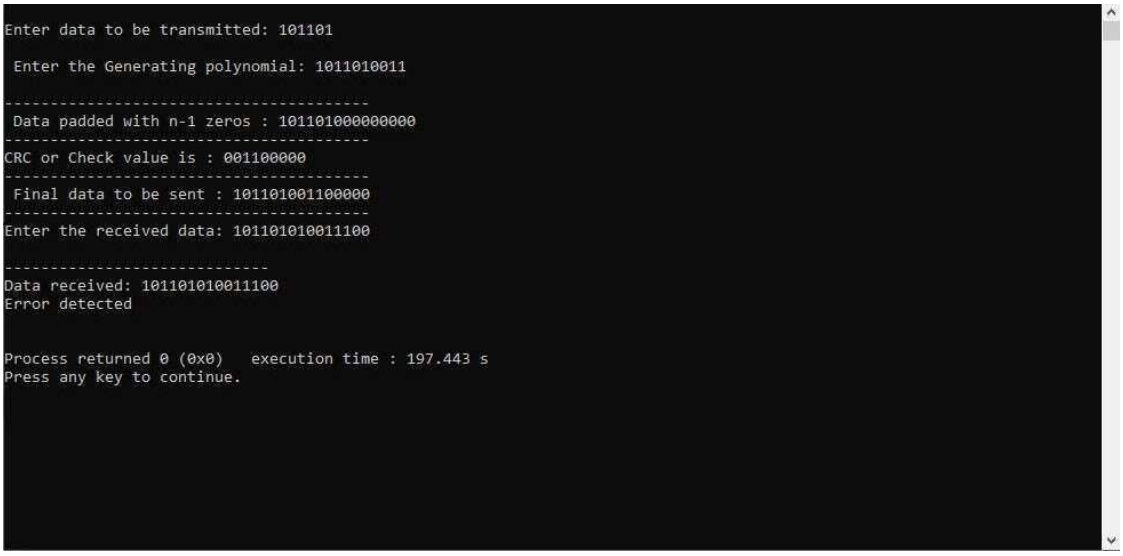
CODE:

```
 #include<stdio.h>
#include<string.h>
#define N strlen(gen_poly)
char data[28];
char check_value[28];
char   gen_poly[10];
int data_length,i,j;
void XOR(){
for(j = 1;j < N; j++)
check_value[j] = (( check_value[j] == gen_poly[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)  &&  (check_value[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++)
check_value[i]=data[i];
do{
if(check_value[0]=='1') XOR();
for(j=0;j<N-1;j++) check_value[j]=check_value[j+1];
check_value[j]=data[i++];
}while(i<=data_length+N-1);
}
int main()
```

```
{
printf("\nEnter data to be transmitted: ");
 scanf("%s",data);
printf("\n Enter the Generating polynomial: ");
scanf("%s",gen_poly);
data_length=strlen(data);
for(i=data_length;i<data_length+N-1;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",check_value);
for(i=data_length;i<data_length+N-1;i++)
data[i]=check_value[i-data_length];
printf("\n ");
 printf("\n Final data to be sent : %s",data);
printf("\n \n");
receiver();
return 0;
}
```

OUTPUT

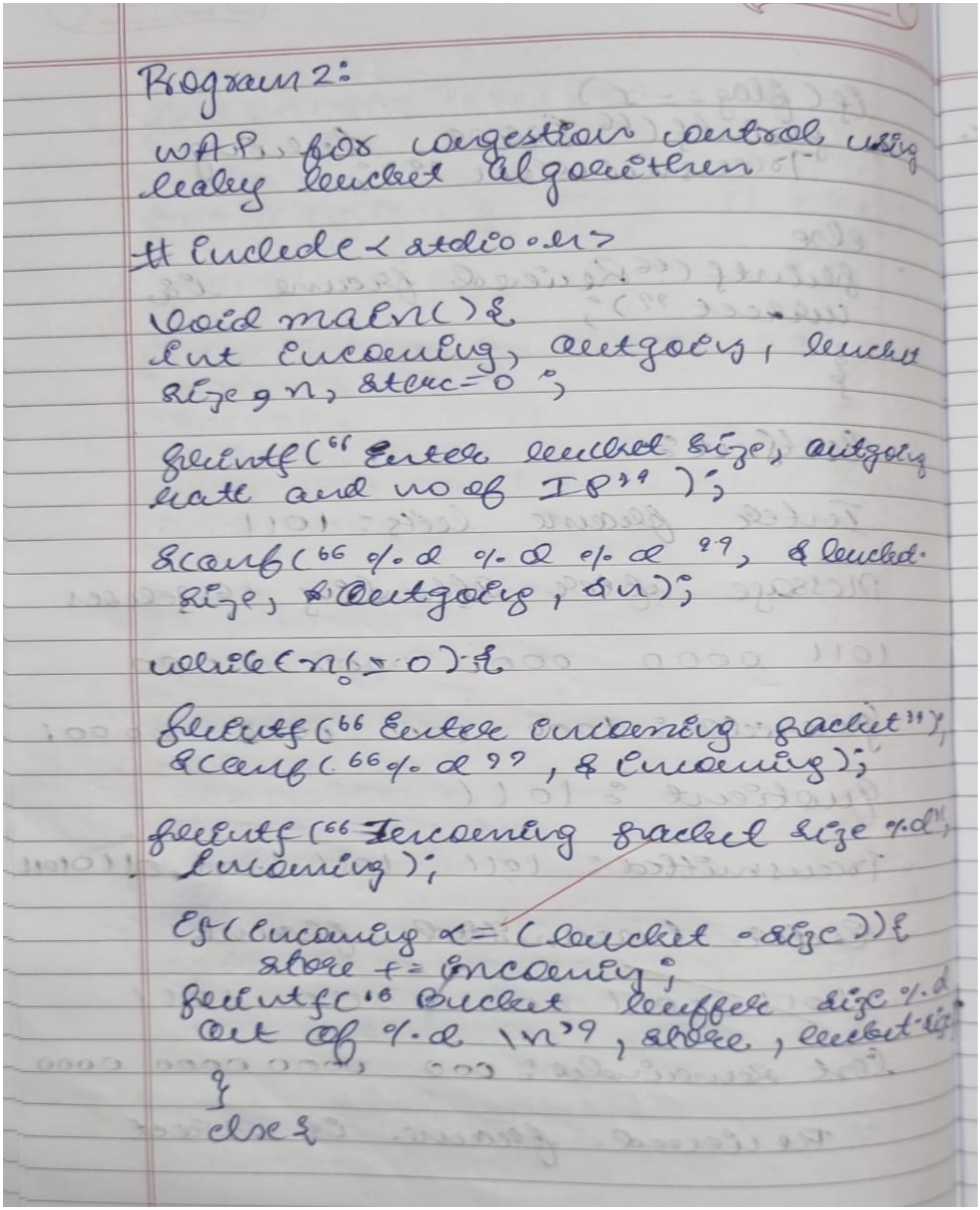

Enter data to be transmitted: 101101
Enter the Generating polynomial: 1011010011
Data padded with n-1 zeros : 101101000000000
CRC or Check value is : 001100000
Final data to be sent : 101101001100000
Enter the received data: 101101001100000
Data received: 101101001100000
No error detected
Process returned 0 (0x0) execution time : 25.115 s
Press any key to continue.


Enter data to be transmitted: 101101
Enter the Generating polynomial: 1011010011
Data padded with n-1 zeros : 101101000000000
CRC or Check value is : 001100000
Final data to be sent : 101101001100000
Enter the received data: 101101010011100
Data received: 101101010011100
Error detected
Process returned 0 (0x0) execution time : 197.443 s
Press any key to continue.

CNLAB 14

AIM: Write a program for congestion control using Leaky bucket algorithm**.**

OBSERVATION:



Program 2:

WAP. for congestion control using leaky bucket algorithm

```
#include<stdio.h>
void main(){
int incoming, outgoing, bucket size, n, store=0;
printf("Enter bucket size, outgoing rate and no of IP");
scanf("%d %d %d", &bucket size, &outgoing, &n);
while(n!=0){
printf("Enter incoming packet");
scanf("%d", &incoming);
printf("Incoming packet size %d", incoming);
if(incoming<=(bucket size)){
store+=incoming;
printf("Bucket buffer size %d out of %d \n", store, bucket size);
}
else{
```

printf("Dropped %d no of packets\n", incoming - (bucket_size));

printf("Bucket buffer size %d out of %d \n", store, bucket_size);

store = bucket_size;

}

store = store - outgoing;

printf("After outgoing %d packets left out %d in buffer \n", store, bucket_size);

n--;

}

}

Output:

Enter bucket size, outgoing rate and no of I/P:

20 10 2

Enter incoming packet size = 30

Dropped 10 no of packets

Bucket buffer size 0 out of 20

After outgoing 10 packet left out 20 in buffer.

Enter incoming packet size = 10

Bucket buffer size 10 out of 20

After outgoing 10 packets left and 20 in buffer.

CODE:

```
#include<stdio.h> void main()
{ int b_size,d_rate,in_d_rate,rem_b_size;
printf("Enter the bucket size:\n");
scanf("%d",&b_size);
rem_b_size=b_size;
printf("Enter the outgoing data rate:\n");
scanf("%d",&d_rate); while(1) {
printf("Enter the size of incoming packet\n");
scanf("%d",&in_d_rate);
if(in_d_rate<=b_size)
{
if(in_d_rate<=rem_b_size)
{
 rem_b_size=rem_b_size-in_d_rate;
 rem_b_size=rem_b_size+d_rate;
 printf("Data packet is accepted\n");
printf("Remaining space in bucket is..... %d\n",rem_b_size);
printf("\n");
}
 else
{
printf("Data packet is dropped because the bucket size is less than the packet
size\n");
printf("\n");
}
}
}
}
```

OUTPUT

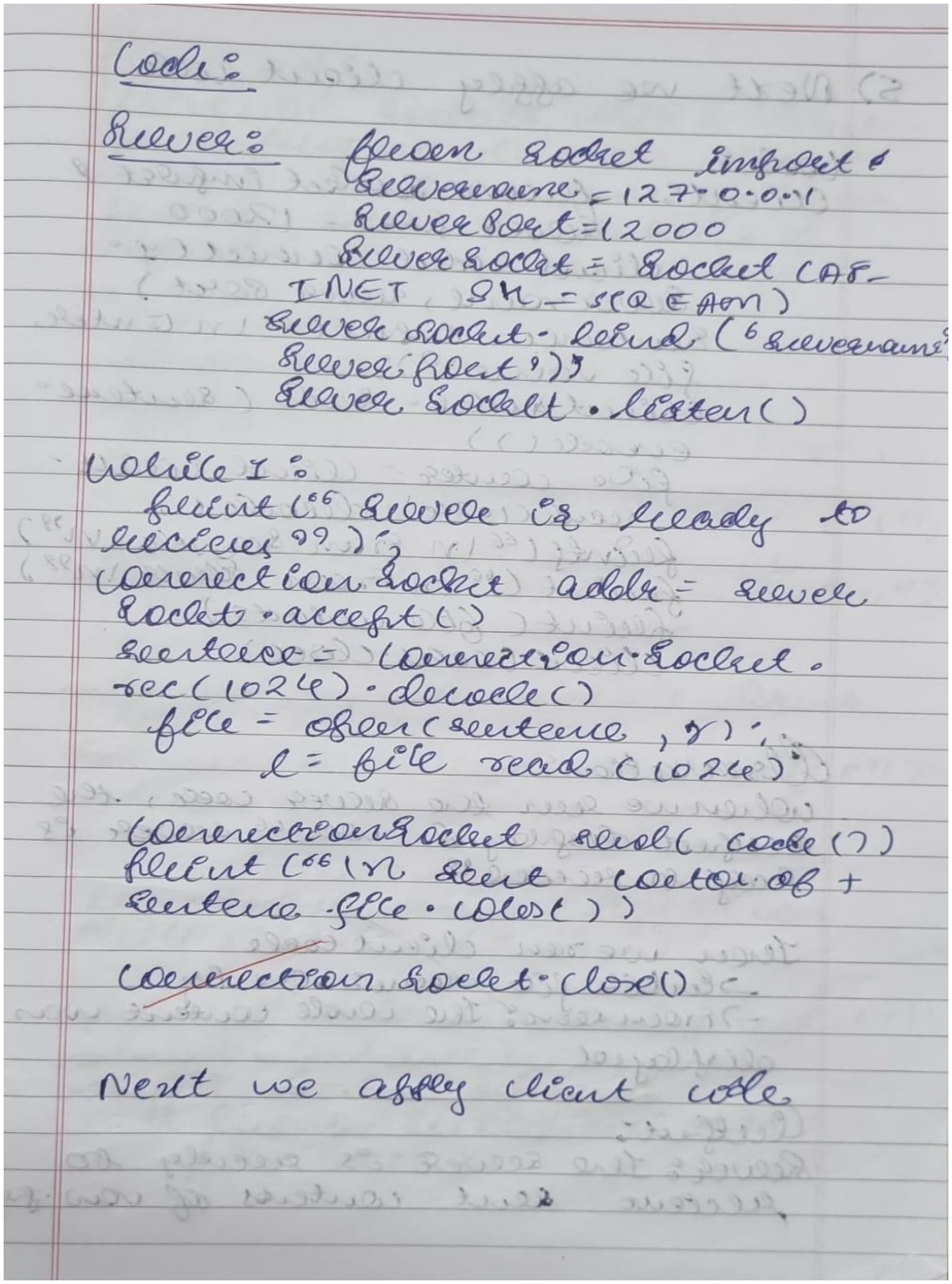
```
Enter the bucket size:
5000
Enter the outgoing data rate:
200
Enter the size of incoming packet
3000
Data packet is accepted
Remaining space in bucket is.... 2200

Enter the size of incoming packet
2500
Data packet is dropped because the bucket size is less than the packet size

Enter the size of incoming packet
```

CN LAB 14

Program 1

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.

Observation:



5) Next we apply client code

Code:
Client:
from socket import *
server port = 12000
clientsocket.connect((server name, server port))
sentence = input("\n Enter file name");
clientsocket.send(sentence.encode())
file contents = clientsocket.recv(1024).decode()
print("\n From Server \n")
print(file contents)
clientsocket.close()

Observation:
When we run the server code, the output displayed was "server is ready to receive"

Then we run client code
→ Enter file name: new.py
→ From server: The code content was displayed

Output:
Server: The server is ready to receive sent contents of new.py

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

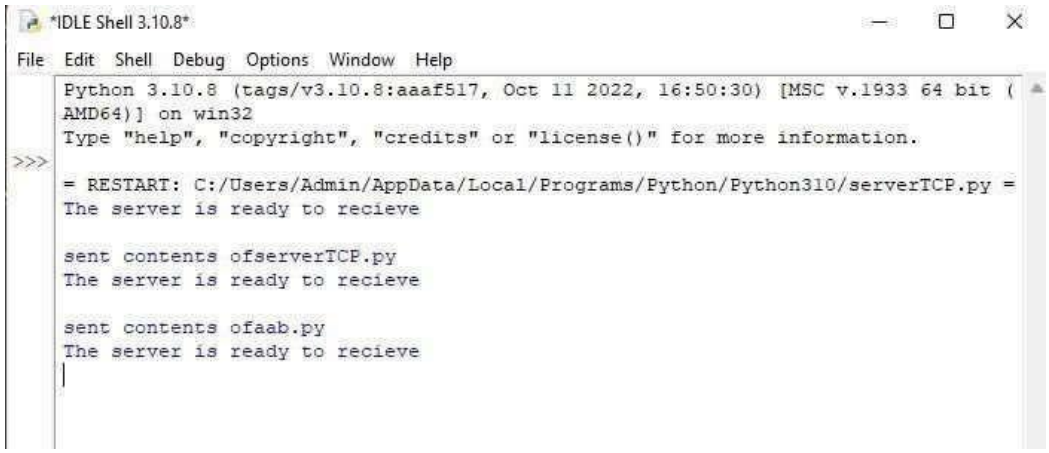
```
from socket import *
serverName="127.0.0.1"
serverPort = 12000
serverSocket = socket(AF_INET,SOCK_STREAM)
serverSocket.bind((serverName,serverPort))
serverSocket.listen(1)
while 1:
print ("The server is ready to receive")
connectionSocket, addr = serverSocket.accept()
sentence = connectionSocket.recv(1024).decode()
file=open(sentence,"r")
l=file.read(1024)
connectionSocket.send(l.encode()) print ('\nSent contents of ' + sentence)
file.close()
connectionSocket.close()
```

OUTPUT:

Client:

```
= RESTART: C:/Users/Admin/AppData/Local/Programs/Python/Python310/clientUDP.py =

Enter file name:  serverUDP.py

Reply from Server:

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")
     file=open(sentence,"r")
     con=file.read(2048)

     serverSocket.sendto(bytes(con,"utf-8"),clientAddress)

     print ('\nSent contents of ', end = ' ')
     print (sentence)
    #  for i in sentence:
        #  print (str(i), end = '')
     file.close()

>>>
```

SERVER:

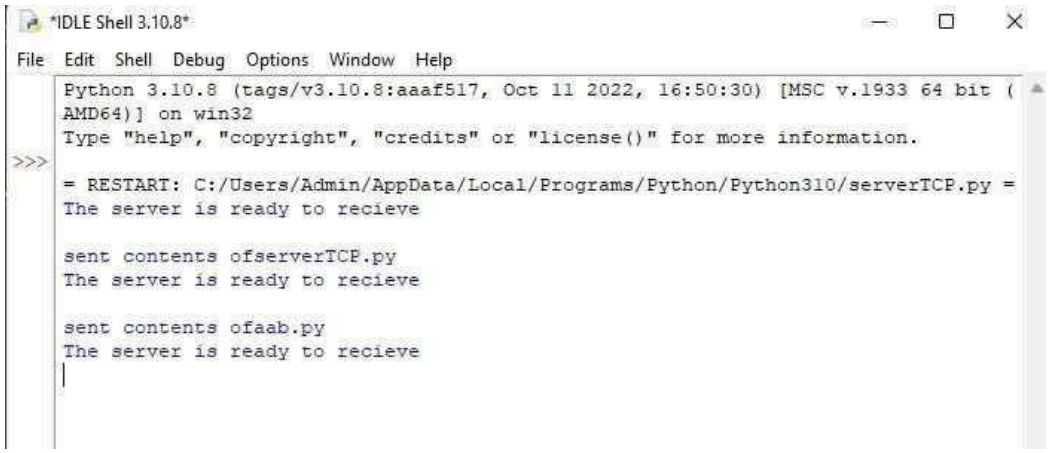


IDLE Shell 3.10.8

File Edit Shell Debug Options Window Help

```
Python 3.10.8 (tags/v3.10.8:aaaf517, Oct 11 2022, 16:50:30) [MSC v.1933 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/Admin/AppData/Local/Programs/Python/Python310/clientTCP.py =

Enter file name:serverTCP.py

From Server:

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("The server is ready to recieve")
    connectionSocket,addr=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()


>>>
= RESTART: C:/Users/Admin/AppData/Local/Programs/Python/Python310/clientTCP.py =

Enter file name:aab.py

From Server:

Python 3.10.8 (tags/v3.10.8:aaaf517, Oct 11 2022, 16:50:30) [MSC v.1933 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
class Node:
    def _init_(self,data):
        self.data=data
        self.left=None
        self.right=None
        self.height=1

class AVL Tree:
    def getHeight(self,root):
        if not root:
            return 0
        return root.height

    def getBalance(self,root):
        if not root:
            return 0
        return self.getHeight(root.left)-self.getHeight(root.right)

    def rightRotate(self,z):
        y=z.left
        T3=y.right

        y.right=z
        z.left=T3

        z.height=1+max(self.getHeight(z.left),self.getHeight(z.right))
        y.height=1+max(self.getHeight(y.left),self.getHeight(y.right))

        return y

    def insert(self,root,data):
        if not root:
            return Node(data)
        if data < root.data
            root.left=self.insert(root.left,data)
        else:
            root.right=se
>>>
```

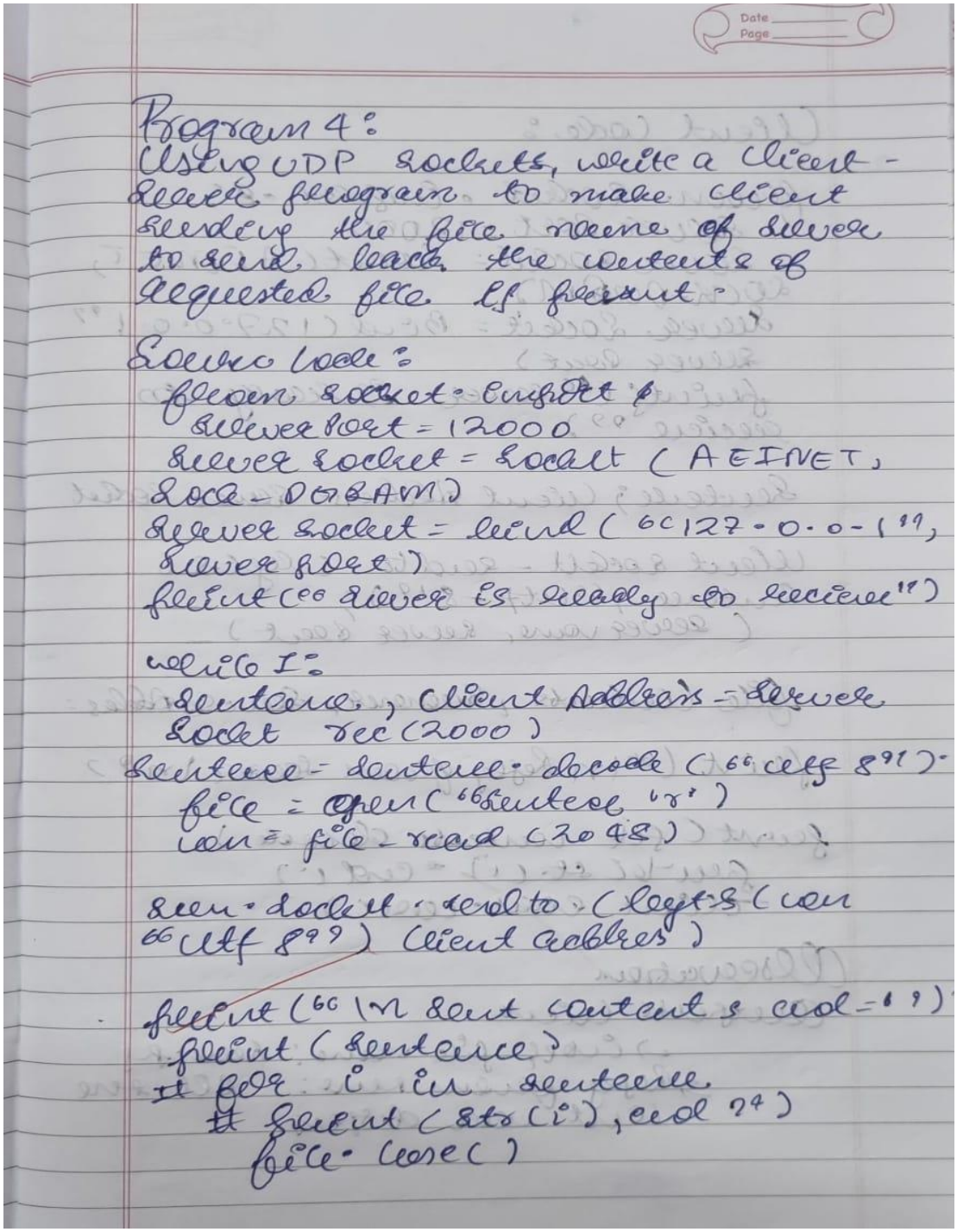
Server:


IDLE Shell 3.10.8
File Edit Shell Debug Options Window Help
Python 3.10.8 (tags/v3.10.8:aaaf517, Oct 11 2022, 16:50:30) [MSC v.1933 64 bit (
AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/Admin/AppData/Local/Programs/Python/Python310/serverTCP.py =
The server is ready to recieve
sent contents ofserverTCP.py
The server is ready to recieve
sent contents ofaab.py
The server is ready to recieve

Program 2

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.

Observation:



Program 4:
Using UDP sockets, write a client-server program to make client sending the file name of server to send back the contents of requested file if present.

Server code:
from socket import *
server port = 12000
server socket = socket (AF_INET, SOCK_DGRAM)
server socket = bind ('127.0.0.1', server port)
print ("server is ready to receive")

while 1:
sentence, client Address = server socket rec (2000)
sentence = sentence. decode ("utf 8").
file = open ("sentence", 'r')
con = file. read (2048)

scon. socket. send to (bytes (con "utf 8") client address)

print ("\n sent content : end = '')
print (sentence)
for i in sentence
print (str (i), end '')
file. close ()

Client code:

```
from socket import *
server_port = 12000
server_socket = socket(AF_INET,
SOCK_DGRAM)
server_socket = bind(('127.0.0.1',
server_port))
print("server is ready to
receive")

sentence, client_address = server_socket

file_contents, filename, server_address =

print("In Reply from server \n")

print(file contents decode)
    file[str(i)] = end(i)
client_socket.close()
```

Observation

Output screen

→ Enter file name: test.txt

→ Reply from server: All the text were displayed

CODE:

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

OUTPUT:

Client:

```
= RESTART: C:/Users/Admin/AppData/Local/Programs/Python/Python310/clientUDP.py =

Enter file name:  serverUDP.py

Reply from Server:

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")
     file=open(sentence,"r")
     con=file.read(2048)

     serverSocket.sendto(bytes(con,"utf-8"),clientAddress)

     print ('\nSent contents of ', end = ' ')
     print (sentence)
    #  for i in sentence:
        #  print (str(i), end = '')
     file.close()

>>>
```

Server:

```
>>>
= RESTART: C:/Users/Admin/AppData/Local/Programs/Python/Python310/serverUDP.py =
The server is ready to receive

Sent contents of  serverUDP.py
```