VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"JnanaSangama", Belgaum -590014, Karnataka.

LAB REPORT

on

COMPUTER NETWORKS LAB

Submitted by

B S SWARAJ (1BM22CS403)

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 "**COMPUTER NETWORKS LAB**" carried out by **B S SWARAJ (1BM22CS403),** 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 **COMPUTER NETWORKS - (22CS4PCCON)** work prescribed for the said degree.

Name of the Lab-Incharge: LATHA N.R

Designation: Assistant Professor
Department of CSE
BMSCE, Bengaluru

Dr. Jyothi S Nayak

Professor and Head
Department of CSE
BMSCE, Bengaluru

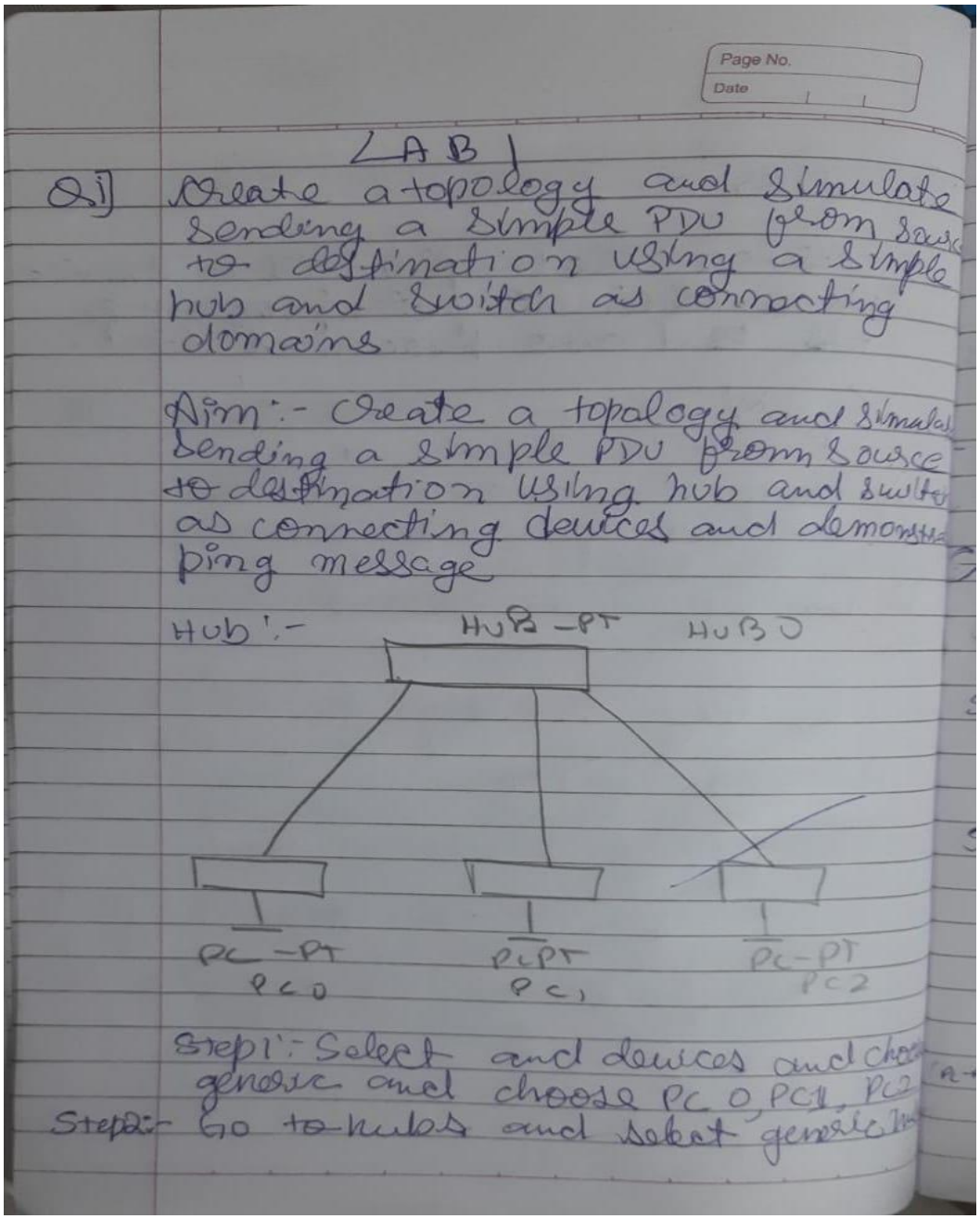
Index

Sl. No.	Experiment Title	Page No.
	CYCLE 1	
1	Create a topology and simulate sending a simple PDU from source to destination using hub and switch as connecting devices and demonstrate ping message	4
2	Configure IP address to routers in packet tracer. Explore the following messages: ping responses, destination unreachable, request timed out, reply	13
3	Configure default route, static route to the Router	25
4	Configure DHCP within a LAN and outside LAN	30
5	Configure RIP routing Protocol in Routers	41
6	Configure OSPF routing protoco	45
7	Demonstrate the TTL/ Life of a Packet	51
8	Configure Web Server, DNS within a LAN	58
9	To construct simple LAN and understand the concept and operation of Address Resolution Protocol (ARP)	63
10	To understand the operation of TELNET by accessing the router in server room from a PC in IT office	69
11	To construct a VLAN and make the PC's communicate among a VLAN	74
12	To construct a WLAN and make the nodes communicate wirelessly	80
	CYCLE 2	
13	Write a program for error detecting code using CRC CCITT (16-bits)	84
14	Write a program for congestion control using Leaky bucket algorithm	93
15	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	98
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	102
17	Tool Exploration -Wireshark	106

LAB1

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

OBSERVATION:



Page No.
Date

Step2:- Goto connection and select copper straight through wires then connect PC's to the hub (select port numbers and PC)

Step3:- Click on PC, goto config and select fast ethernet then set IP address for the 10.0.0 PC. Do the same for all the PC's 10.0.0.1, 10.0.0.2, 10.0.0.3

Step5:- Add simple PDU

Step6:- Click source and destination systems

Step7:- Then goto simulation mode, Auto capture / play. Then the packets will starts to transfer

Step8:- Click on PC goto desktop and select command prompt. Then type command ping 10.0.0.3

PC > Ping 10.0.0.3

Reply from 10.0.0.3 bytes=32 time=4ms TTL=128
Reply from 10.0.0.3 bytes=32 time=3ms 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

Page No.
Date

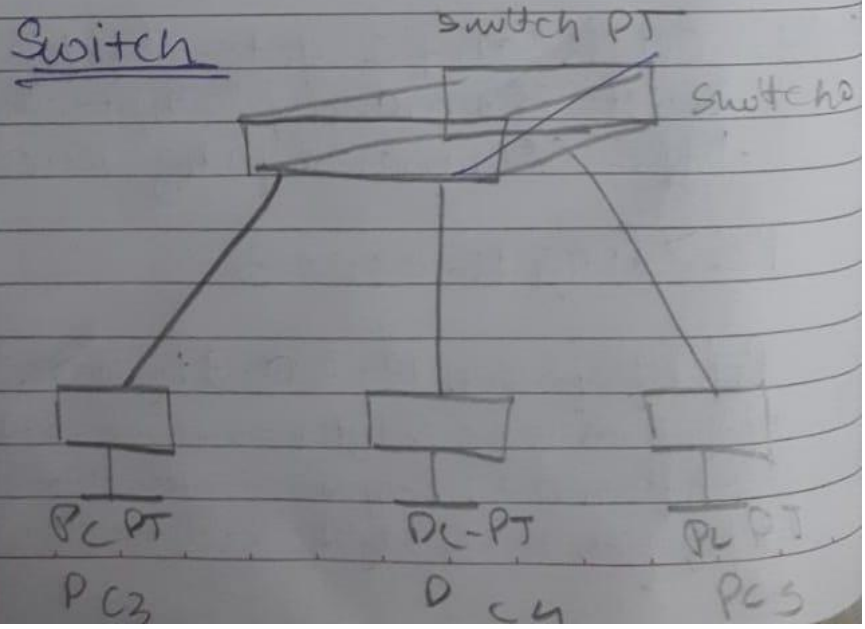
Ping Statistics for 10.0.0.3

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

Approximate round trip time in milli-seconds minimum = 0ms, maximum = 4ms
Average = 2ms

Observation :-
When the source device sent a packet to the hub it will broadcast or send the packet to all the devices which all connected to the hubs. And the destination device will receive the packet and others will reject the packet. And destination devices will send the acknowledgement and the sources will accept and other will discard.

Switch



Page No.
Date

procedure :-

Step1:- Select Switch and 8PC's

Step2:- Set IP address for all the PC's
10.0.0.4, 10.0.0.8, 10.0.0.6
PC → catalog → fast ethernet → IP address

Step3:- Connect PC's to the Switch by selecting copper straight through

Step4:- Add Simple PDU.
Select source and destination

Step5:- Go to Simulation mode & click on Auto capture/Play

Step6:- click on PC → Desktop → command prompt

Ping Message

PC> Ping 10.0.0.6
Pinging 10.0.0.6 with 32 bytes of data

Reply from 10.0.0.6 bytes = 32 time = 4ms TTL = 128
Reply from 10.0.0.6 bytes = 32 time = 4ms TTL = 128
Reply from 10.0.0.6 bytes = 32 time = 2ms TTL = 128
Reply from 10.0.0.6 bytes = 32 time = 4ms TTL = 128

ping statistics for 10.0.0.6

Page No.
Date

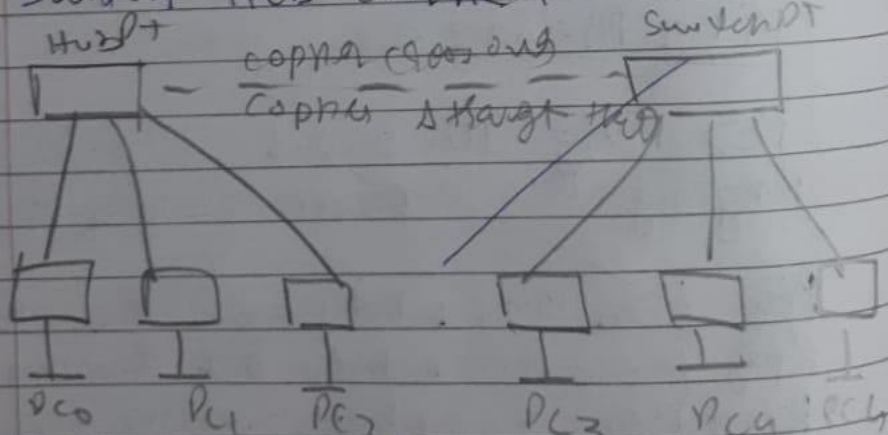
Packets sent = 4 Received = 4 cost = 0 (0% Loss)

Approximate round trip times in milliseconds minimum = 4ms Maximum = 4ms Avg = 4ms

Observation :-

When the first time the packet is sent the switch will distribute the packet with all devices. Once it changes around the IP address it will only send packet to the destination and send acknowledgement to the source

Switch - Hub connection :-



Sol :- Previously drawn hub - topology & switches topology are connected through copper cross over. In hub port 3 is used in switch

port ethernet 3/1 is used.

192- Add simple PDC from PC 0 to PC 3
ping 10.0.0.4

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

ping statistics for 10.0.0.4
packets: Sent 4 Received=4 loss=0%
Approximate round trip times in milliseconds
Minimum=4ms Maximum=4ms Average=4ms

Observation :-

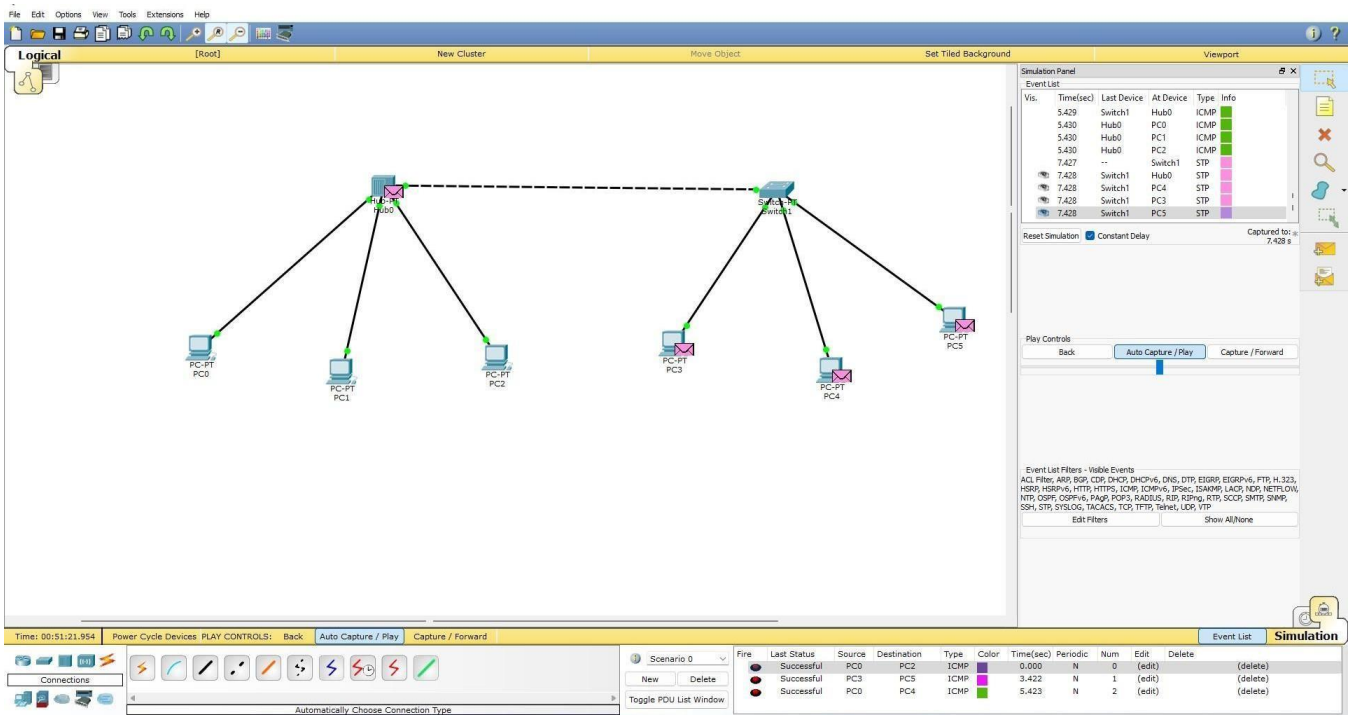
In simulation mode PC 0 sends packet to hub sends it to PC1, PC2 & Switch board casts it to PC 3, PC4 & PC 5
PC1, PC2, PC 4 & PC5 discards from PC 3 accepts & sends acknowledgement to hub through Switch.

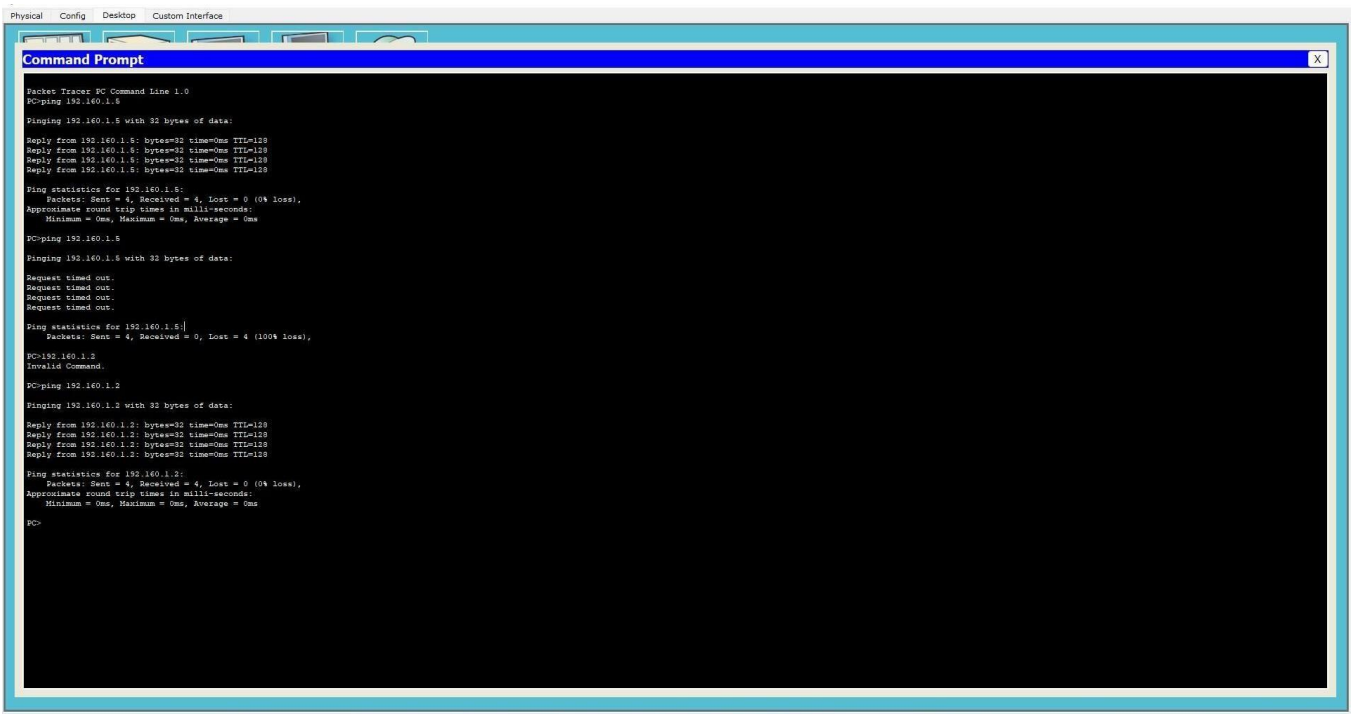
Hub is broadcast it to all 3 PCs only PC 0 accepts it and others discarded in second round PC 0.

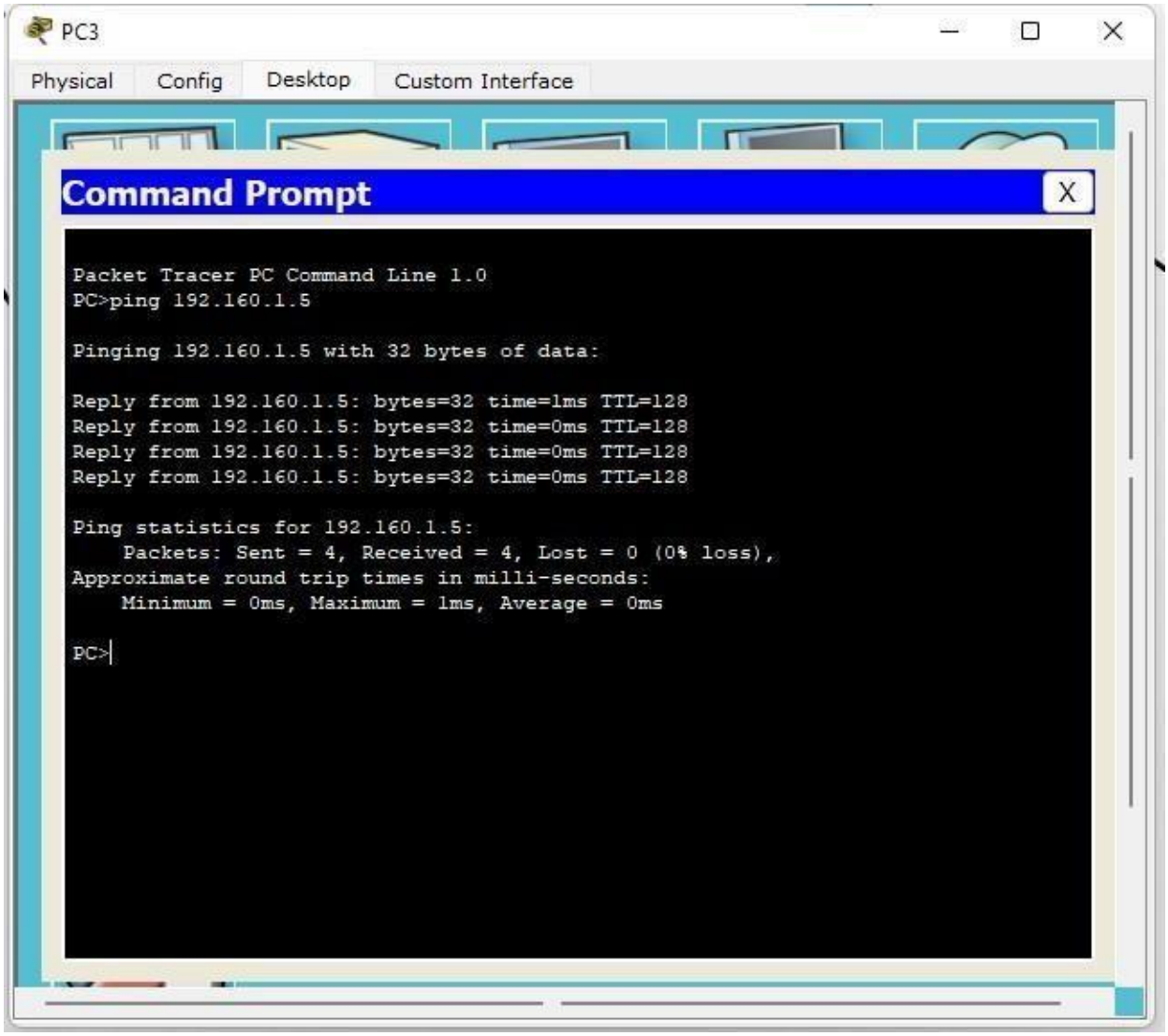
Page No.
Date

Sends packet to hub & broadcast to PC1, PC2 Switch now switch broadcast it only to PC3. Thus switch is smart devices.

OUTPUT:


File Edit Options View Tools Extensions Help
Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
Simulation Panel
Event List
Vis. Time(sec) Last Device At Device Type Info
5.429 Switch1 Hub0 ICMP
5.430 Hub0 PC0 ICMP
5.430 Hub0 PC1 ICMP
5.430 Hub0 PC2 ICMP
7.427 -- Switch1 STP
7.428 Switch1 Hub0 STP
7.428 Switch1 PC4 STP
7.428 Switch1 PC3 STP
7.428 Switch1 PC5 STP
Reset Simulation
Constant Delay
Captured to: 7.428 s
Play Controls
Back
Auto Capture / Play
Capture / Forward
Event List Filters - Visible Events
Edit Filters
Show All/None
Hub0
Switch1
PC-PT PC0
PC-PT PC1
PC-PT PC2
PC-PT PC3
PC-PT PC4
PC-PT PC5
Time: 00:51:21.954
Power Cycle Devices
PLAY CONTROLS:
Back
Auto Capture / Play
Capture / Forward
Event List
Simulation
Connections
Automatically Choose Connection Type
Scenario 0
New
Delete
Toggle PDU List Window
Fire Last Status Source Destination Type Color Time(sec) Periodic Num Edit Delete
Successful PC0 PC2 ICMP 0.000 N 0 (edit) (delete)
Successful PC3 PC5 ICMP 3.422 N 1 (edit) (delete)
Successful PC0 PC4 ICMP 5.423 N 2 (edit) (delete)

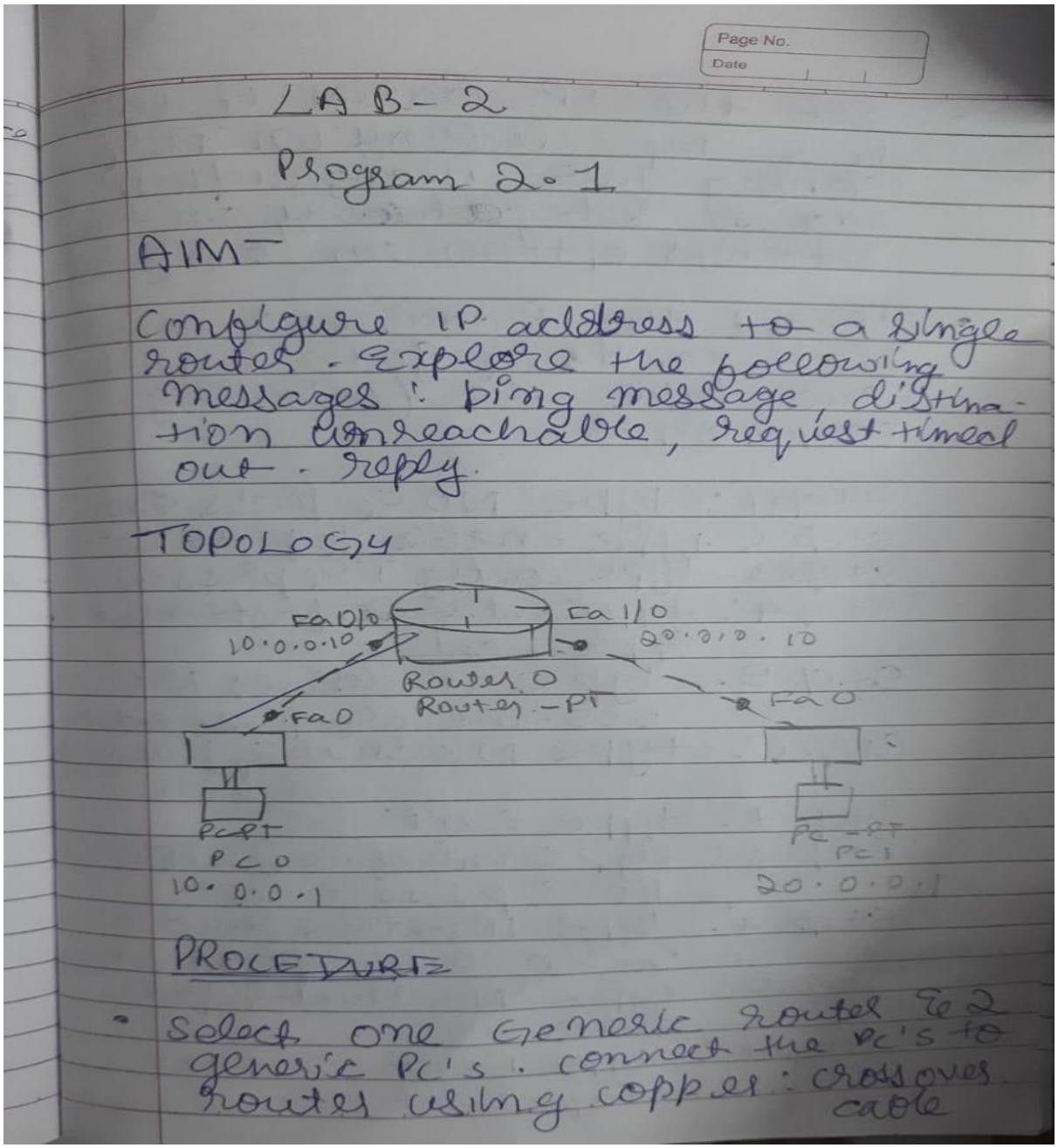

Physical
Config
Desktop
Custom Interface
Command Prompt
Packet Tracer PC Command Line 1.0
PC>ping 192.160.1.5
Pinging 192.160.1.5 with 32 bytes of data:
Reply from 192.160.1.5: bytes=32 time=0ms TTL=128
Reply from 192.160.1.5: bytes=32 time=0ms TTL=128
Reply from 192.160.1.5: bytes=32 time=0ms TTL=128
Reply from 192.160.1.5: bytes=32 time=0ms TTL=128
Ping statistics for 192.160.1.5:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 0ms, Average = 0ms
PC>ping 192.160.1.5
Pinging 192.160.1.5 with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.
Ping statistics for 192.160.1.5:
Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
PC>192.160.1.2
Invalid Command.
PC>ping 192.160.1.2
Pinging 192.160.1.2 with 32 bytes of data:
Reply from 192.160.1.2: bytes=32 time=0ms TTL=128
Reply from 192.160.1.2: bytes=32 time=0ms TTL=128
Reply from 192.160.1.2: bytes=32 time=0ms TTL=128
Reply from 192.160.1.2: bytes=32 time=0ms TTL=128
Ping statistics for 192.160.1.2:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 0ms, Average = 0ms
PC>


PC3
Physical
Config
Desktop
Custom Interface
Command Prompt
X
Packet Tracer PC Command Line 1.0
PC>ping 192.160.1.5
Pinging 192.160.1.5 with 32 bytes of data:
Reply from 192.160.1.5: bytes=32 time=1ms TTL=128
Reply from 192.160.1.5: bytes=32 time=0ms TTL=128
Reply from 192.160.1.5: bytes=32 time=0ms TTL=128
Reply from 192.160.1.5: bytes=32 time=0ms TTL=128
Ping statistics for 192.160.1.5:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 1ms, Average = 0ms
PC>

LAB-2

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

OBSERVATION:



- Set the IP address of both PC's by clicking on PC & config tab. Along with IP address set gateway in the settings option on config tab

- To set the IP address of a router click on it & go to CLI tab and type the following commands

Step 1: type NO & press Enter
Step 2 type enable & press enter
Step 3: type config T & press enter
Step 4: type interface fastEthernet 0/0 & press enter
Step 5: type IP address 10.0.0.1 255.0.0.0 & press enter
Step 6: type NO shut & press Enter

Step 7: type Exit
Step 8: type interface fastEthernet 1/0 & press enter
Step 9: type IP address 20.0.0.1 25.0.0.0 & press Enter
Step 10: type NO shut & press enter
Step 11: type Exit
Step 12: type Exit

Page No.
Date

Step 13: type show ip route
[for seeing the connection status]

- close the tab & click on PC to go to command prompt. Type ping 20.0.0.1 to send packets across

- At last send packets in simulation mode to get a successful transmission

PING OUTPUT

Packet tracer PC command line 0
PC > ping 20.0.0.1
pinging 20.0.0.1 with 32 types of data

Request timed out
Reply from 20.0.0.1 : bytes = 32 time=0ms TTL=
" " " " " " " "
" " " " " time=10ms "

Ping statistics for 20.0.0.1
packets : Sent = 4, Received = 3, lost = 1 (25% loss) Approximate round trip times in milliseconds. Minimum =0ms, Maximum =10ms, Avg = 3ms

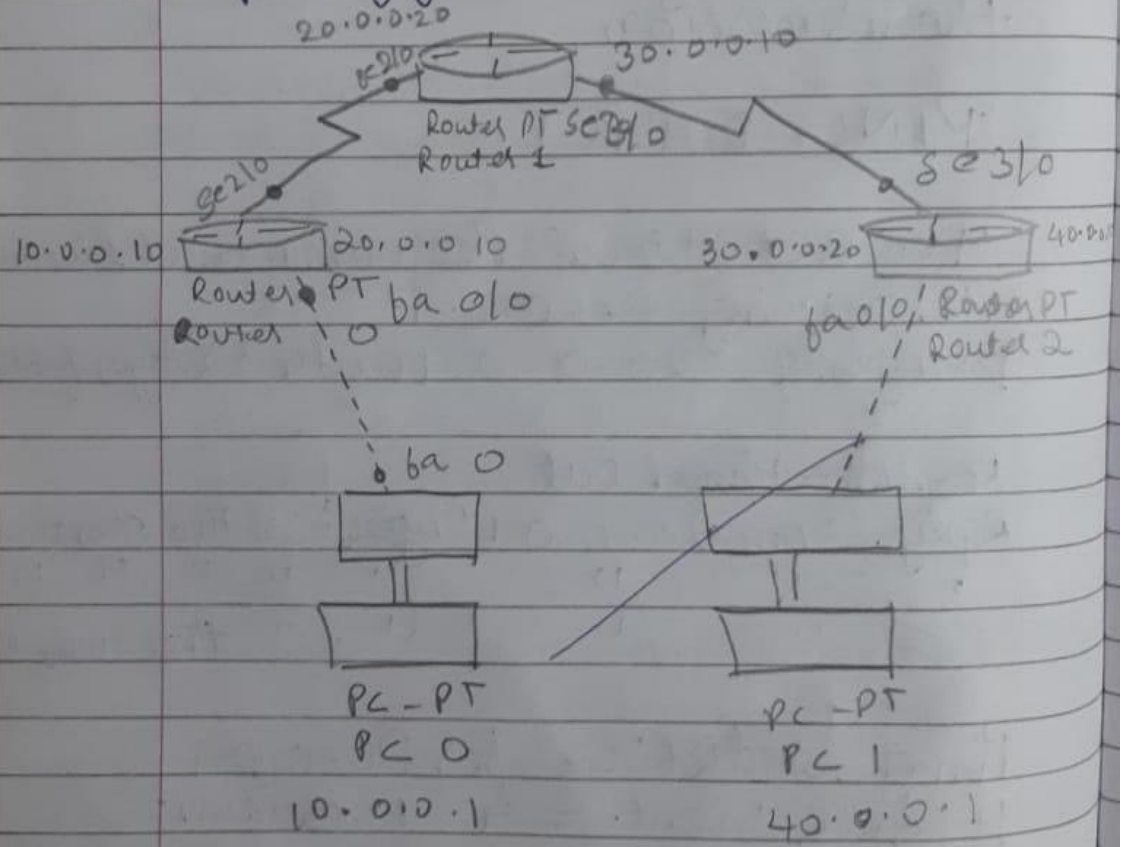
Page No.
Date

Program 2.2

AIM

configure IP address to three routers in packet tracer. Explore the following messages ping response, destination unreachable request timed out reply

Topology :-



Page No.
Date

Procedure

- Connect 2 PC's & 3 routers using copper cross over cable for PC to router and serial DCE cable to connect the routers to routers.
- Set the IP address of both PC's & gateway numbers
- Now for setting IP address & gateway numbers to routers Select one router and perform the following commands.

Step 1: type NO & press Enter
Step 2: type Enable & press enter.
Step 3: type config T & press enter
Step 4: type Interface fast Ethernet 0/0 & press Enter
Step 5: type IP address 10.0.0.10 255.0.0.0 & press Enter
Step 6: type NO shut & press enter
Step 7: type Exit
Step 8: type interface se 2/0 & press enter.
Step 9: type IP address 20.0.0.10 255.0.0.0 & press Enter
Step 10: type NO shut & press enter
Step 11: type Exit
Step 12: type Exit.

Page No.
Date

- Repeat these commands for the other two routers with respective IP/ gateway addresses

- Now to introduce the other two IP address to the first router we type the following commands.

Step1 : type config T & press enter
Step2: type IProute 30.0.0 255.0.0.0 20.0.0.20
Step3: type IProute 40.0.0.0 255.0.0.0 20.0.0.20
Step4 : Exit
Step5: Exit
Step6: type show IP route

- Repeat these steps for the other two Routers with appropriate addresses

- Go to command prompt by clicking on PC & config tab: type ping message to send packets to the destination address.

Page No.
Date

PING OUTPUT

Output 1:
packet tracer PC command line 1.0
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
" " " " " "
" " " " " "
Request timed out
ping Statistics for 40.0.0.1
packets: sent = 4, Received = 0,
lost = 4 (100% loss)

Output 2
Packet Tracer PC Command Line 1.0
PC> ping 10.0.0.1
Pinging 10.0.0.1 with 32 bytes data

Reply from 10.0.0.1: bytes = 32 time = 2ms TTL = 125
" " " " " time = 8ms "
" " " " " time = 2ms "
" " " " " time = 2ms "
Ping Statistics for 10.0.0.1
packets: Sent = 4, Received = 4, lost = 0
Approximate round trip times in milli's
seconds: Minimum = 2ms, Maximum = 8ms
Avg = 3ms

Page No.
Date

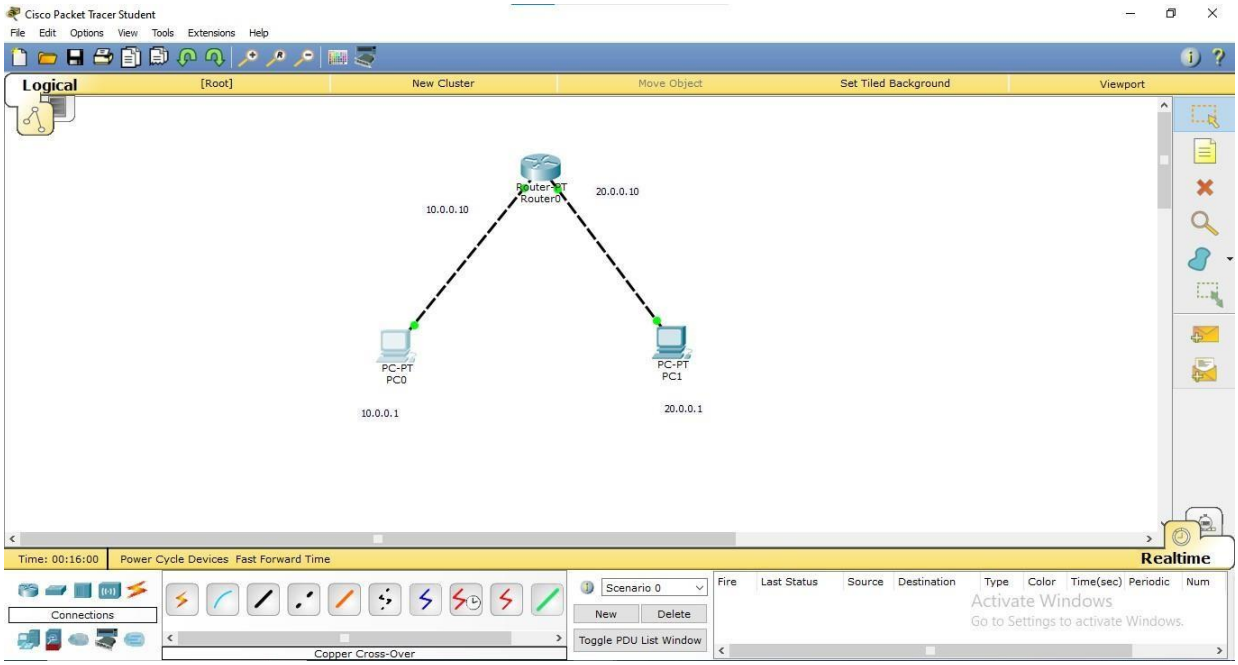
OBSERVATION

- In program 2.1 when we ping the destination address we get allocated with 32 bytes. In this first 8 bytes are used to learn about the routes and their addresses. Rest bytes are used for sending packets to destination addrss. Then, again if we ping all bytes are used for message sending and there will be no timed out message.

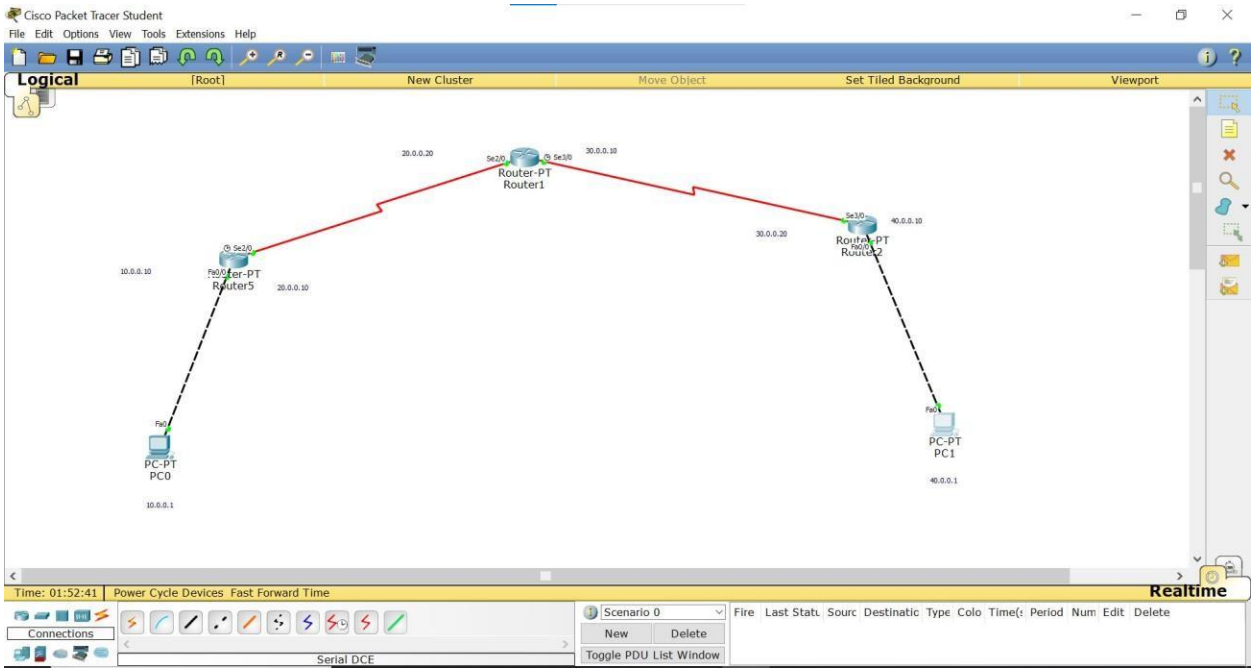
- In Program 2.2 when the routers doesnt know about the remaining address, and we ping a messag we get host unreachable message once the routers have access / knowledge about other addresses messages will be sent succesfully.

TOPOLOGY:

PROGRAM 2.1

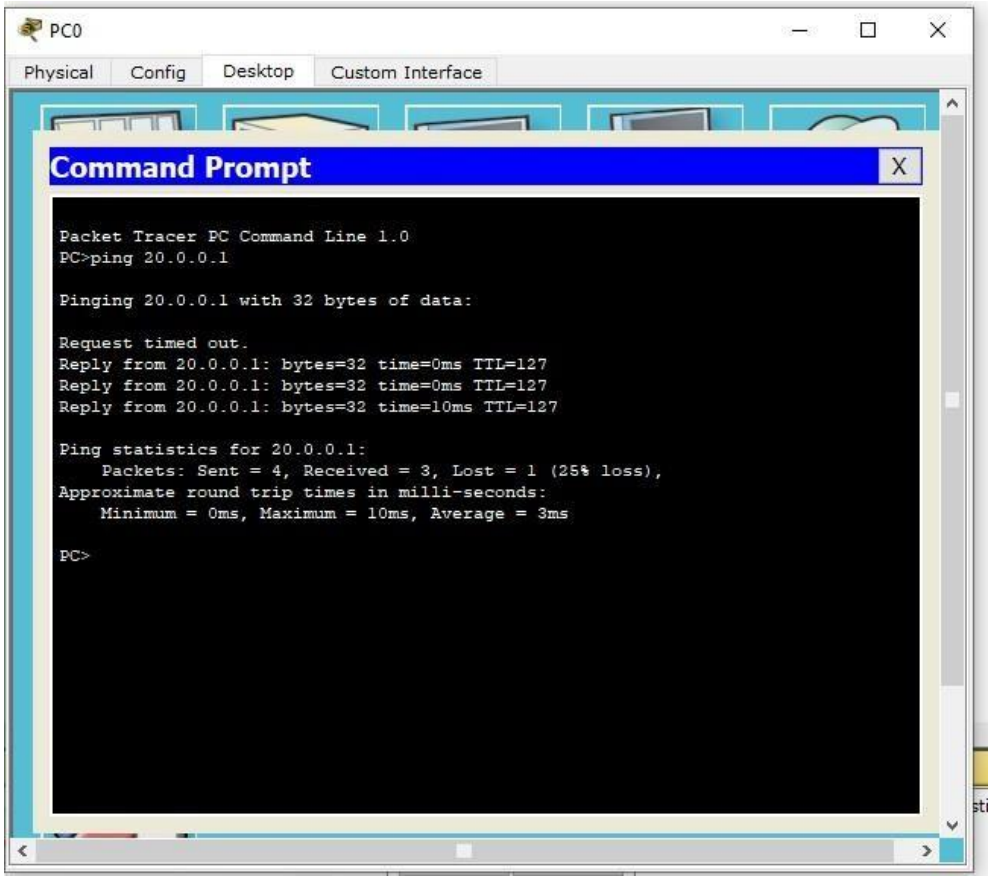


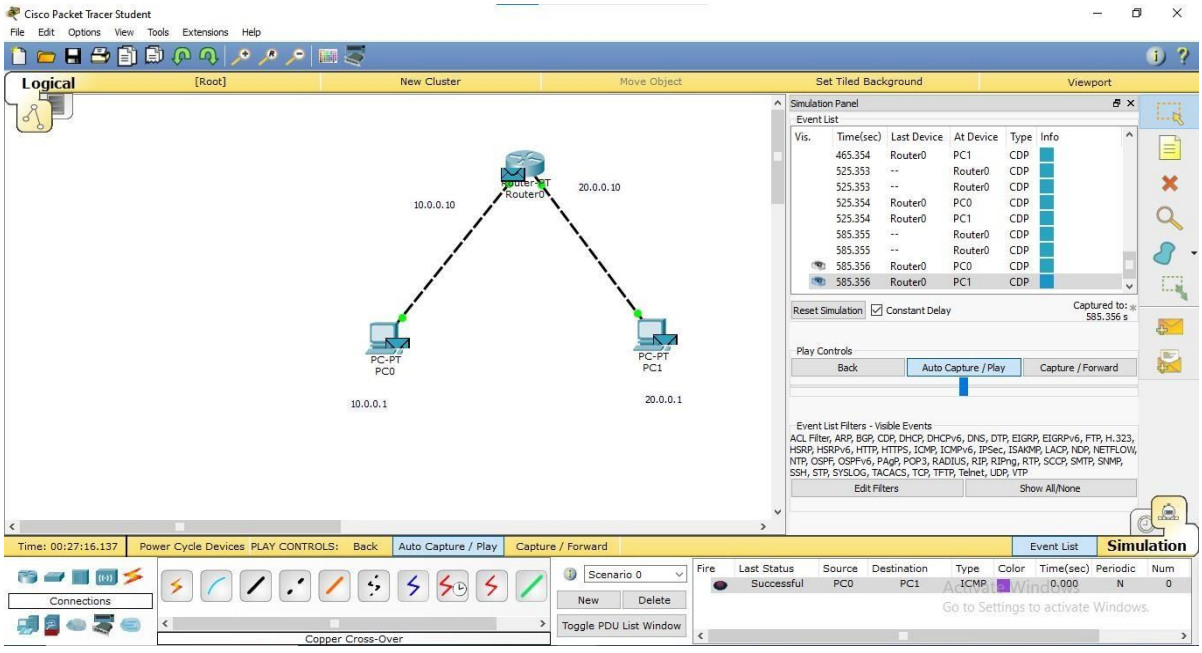
PROGRAM 2.2



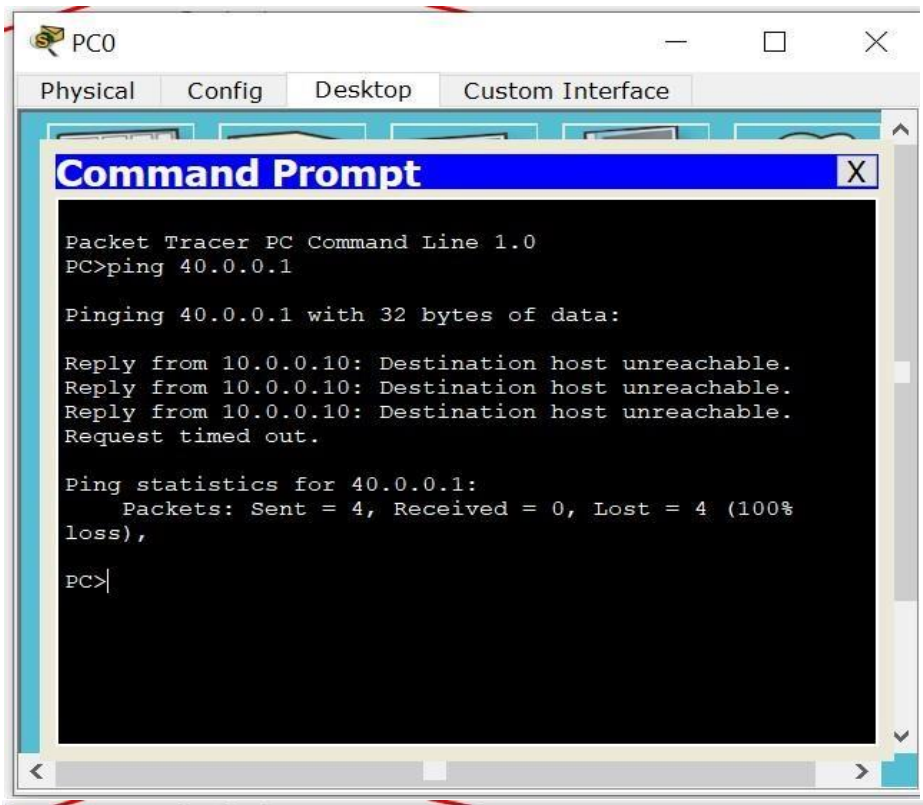
OUTPUT:

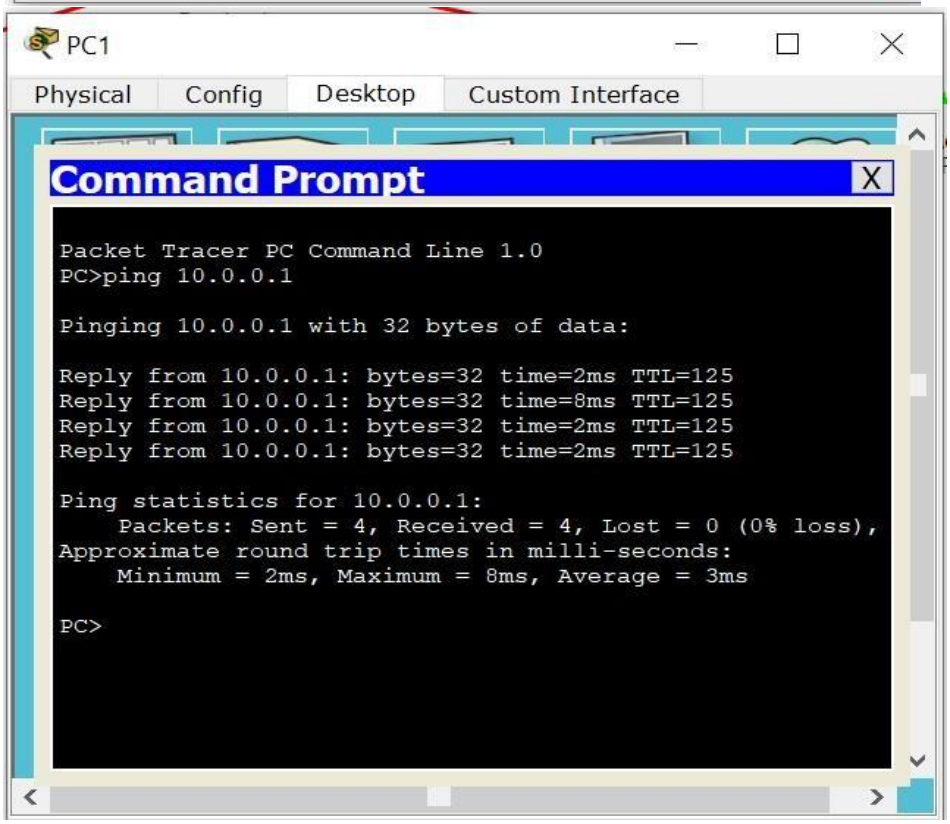
PROGRAM 2.1

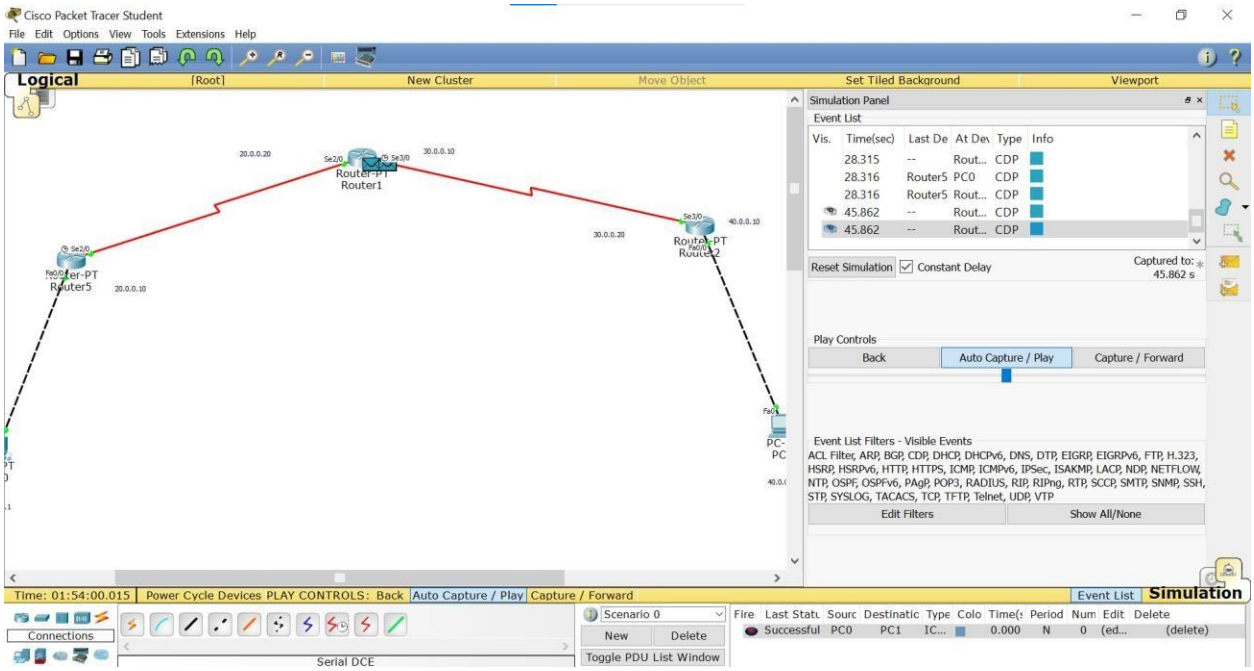

PC0
Physical
Config
Desktop
Custom Interface
Command Prompt
X
Packet Tracer PC Command Line 1.0
PC>ping 20.0.0.1
Pinging 20.0.0.1 with 32 bytes of data:
Request timed out.
Reply from 20.0.0.1: bytes=32 time=0ms TTL=127
Reply from 20.0.0.1: bytes=32 time=0ms TTL=127
Reply from 20.0.0.1: bytes=32 time=10ms 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
File Edit Options View Tools Extensions Help
Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
10.0.0.10
20.0.0.10
Router-PT
Router0
PC-PT
PC0
PC-PT
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:
Back
Auto Capture / Play
Capture / Forward
Event List
Simulation
Connections
Copper Cross-Over
Scenario 0
New
Delete
Toggle PDU List Window
Successful
PC0
PC1
ICMP
Activate Windows
Go to Settings to activate Windows.

PROGRAM 2.2


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:
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.
Request timed out.
Ping statistics for 40.0.0.1:
Packets: Sent = 4, Received = 0, Lost = 4 (100%
loss),
PC>

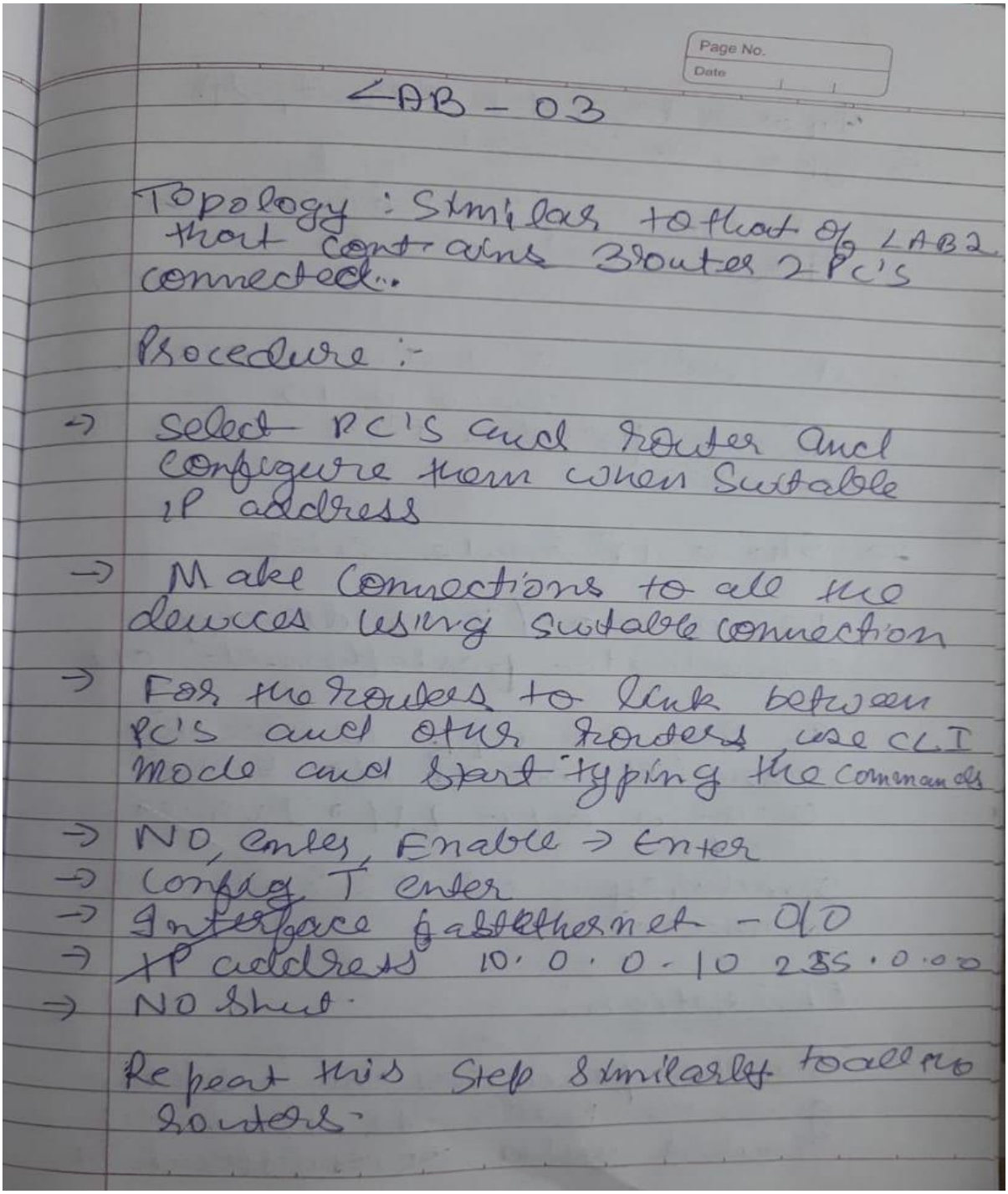

PC1
Physical
Config
Desktop
Custom Interface
Command Prompt
Packet Tracer PC Command Line 1.0
PC>ping 10.0.0.1
Pinging 10.0.0.1 with 32 bytes of data:
Reply from 10.0.0.1: bytes=32 time=2ms TTL=125
Reply from 10.0.0.1: bytes=32 time=8ms TTL=125
Reply from 10.0.0.1: bytes=32 time=2ms TTL=125
Reply from 10.0.0.1: bytes=32 time=2ms 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 = 8ms, Average = 3ms
PC>


Cisco Packet Tracer Student
File Edit Options View Tools Extensions Help
Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
Router-PT
Router1
Router-PT
Router5
Router-PT
Router2
PC-PT
PC
20.0.0.20
30.0.0.10
30.0.0.20
40.0.0.10
20.0.0.10
Simulation Panel
Event List
Vis. Time(sec) Last De At Dev Type Info
28.315 -- Rout... CDP
28.316 Router5 PC0 CDP
28.316 Router5 Rout... CDP
45.862 -- Rout... CDP
45.862 -- Rout... CDP
Reset Simulation
Constant Delay
Captured to: 45.862 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: 01:54:00.015
Power Cycle Devices PLAY CONTROLS: Back
Auto Capture / Play
Capture / Forward
Event List
Simulation
Connections
Serial DCE
Scenario 0
New
Delete
Toggle PDU List Window
Fire Last Statu Sourc Destinatic Type Colo Time(s Period Num Edit Delete
Successful PC0 PC1 IC... 0.000 N 0 (ed... (delete)

LAB-3

Configure default route, static route to the Router.

OBSERVATION:



LAB - 03

Topology : Similar to that of LAB2 that contains 3 routers 2 PC's connected..

Procedure :-

→ Select PC's and router and configure them when suitable IP address

→ Make connections to all the devices using suitable connection

→ For the routers to link between PC's and other routers use CLI mode and start typing the commands

→ NO, enter, Enable → Enter
→ Config T enter
→ Interface GabitEthernet - 0/0
→ IP address 10.0.0.10 255.0.0.0
→ NO Shut.

Repeat this step similarly to all no routers.

Page No.
Date

In order to make default path

→ Type, in config T

→ IP route Destination subnetmask Intermediate device

ie

→ IP route 0.0.0.0 0.0.0.0 20.0.0.10

similarly perform this to all the routers.

→ Show IP route enter

10.0.0.0/8 is directly connected to fastethernet 0/0

C 20.0.0.0/8 is directly connected, serial 2/0 5th 0.0.0.0/0 [1/0] via 20.0.0.10

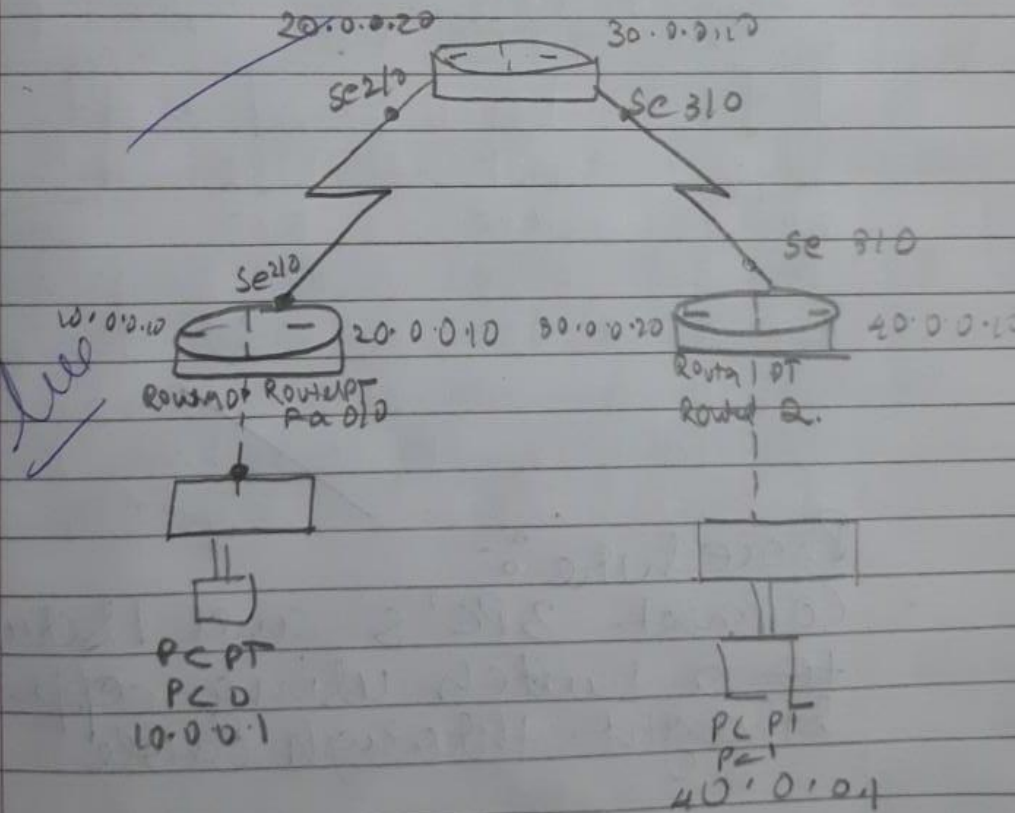
similarly all the routers are connected.

Observation:

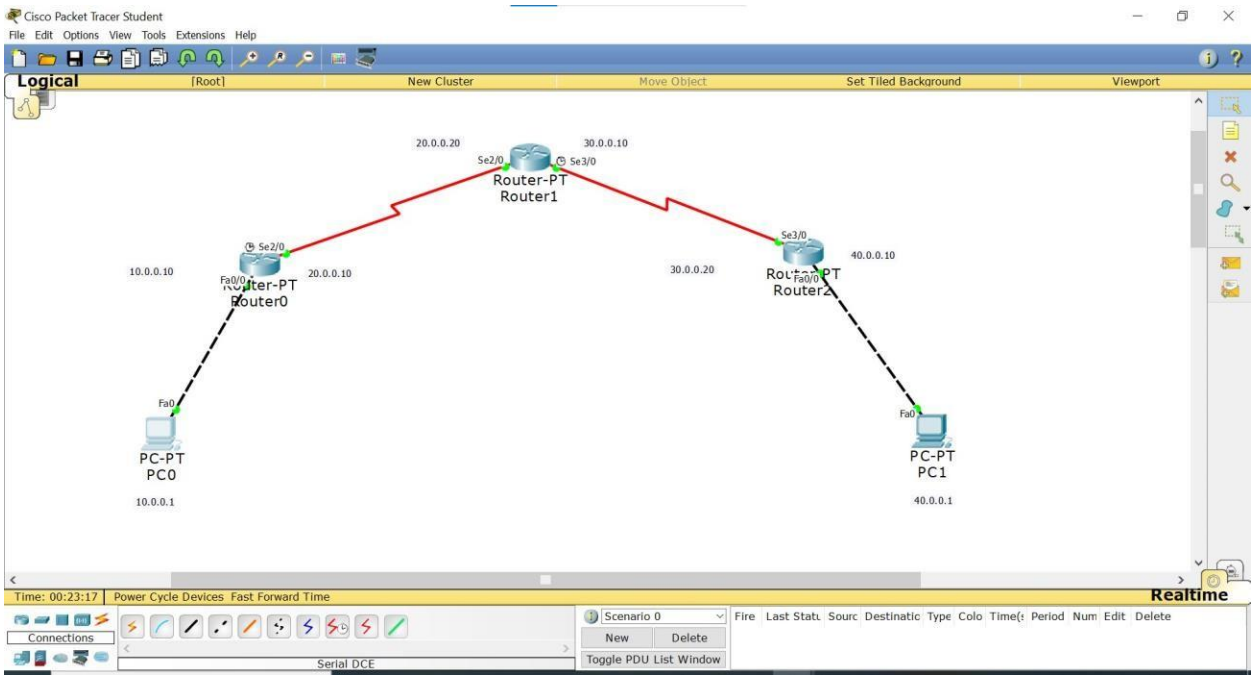
In the previous one, we have given the IP address to all the routers with destination

Page No.
Date

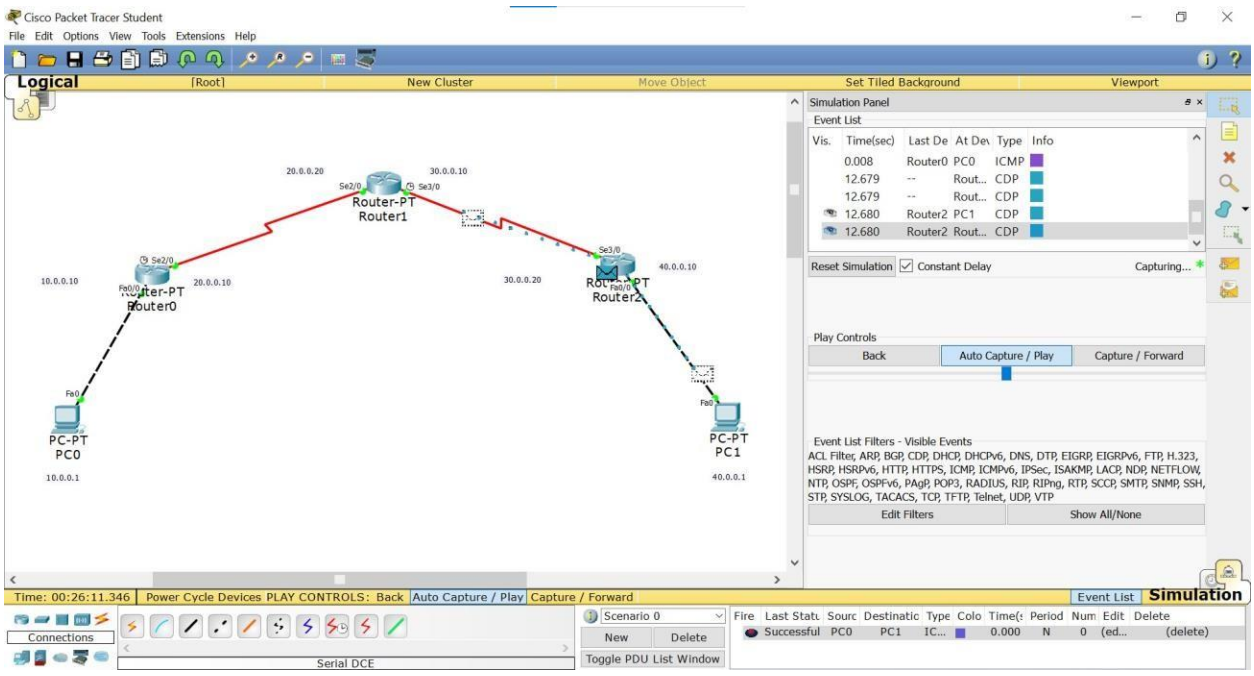
Subnet and intermediate, IP address of ~~the~~ particular devices but here in this experiment we use a default IP address ie. 0.0.0.0 and Subnet mask 0.0.0.0 so that it can create a pass through channel to all the packets that are sent will be transferred by the intermediate device. ~~It~~ This is generally used in large no of device connection

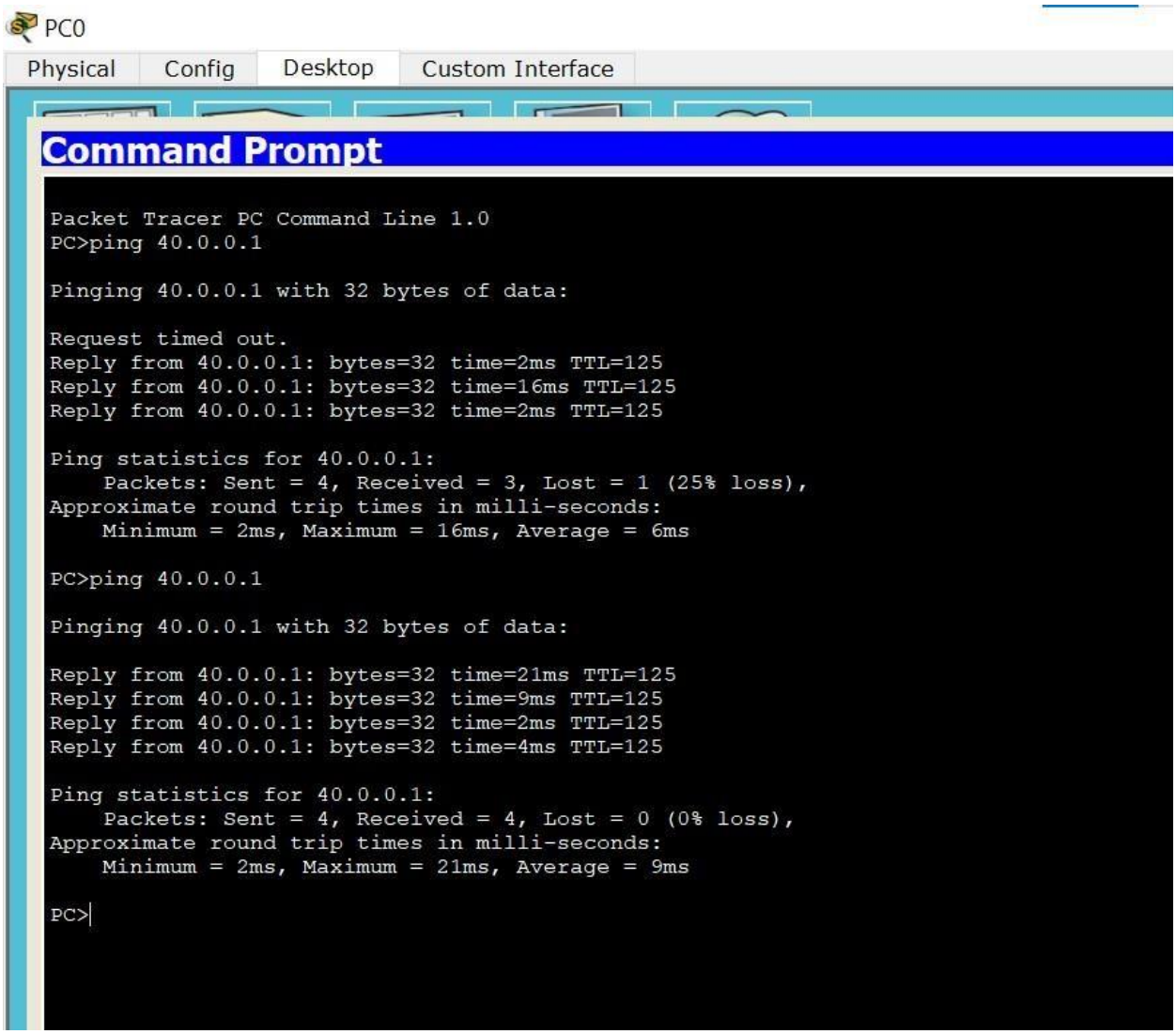


TOPOLOGY:



OUTPUT:

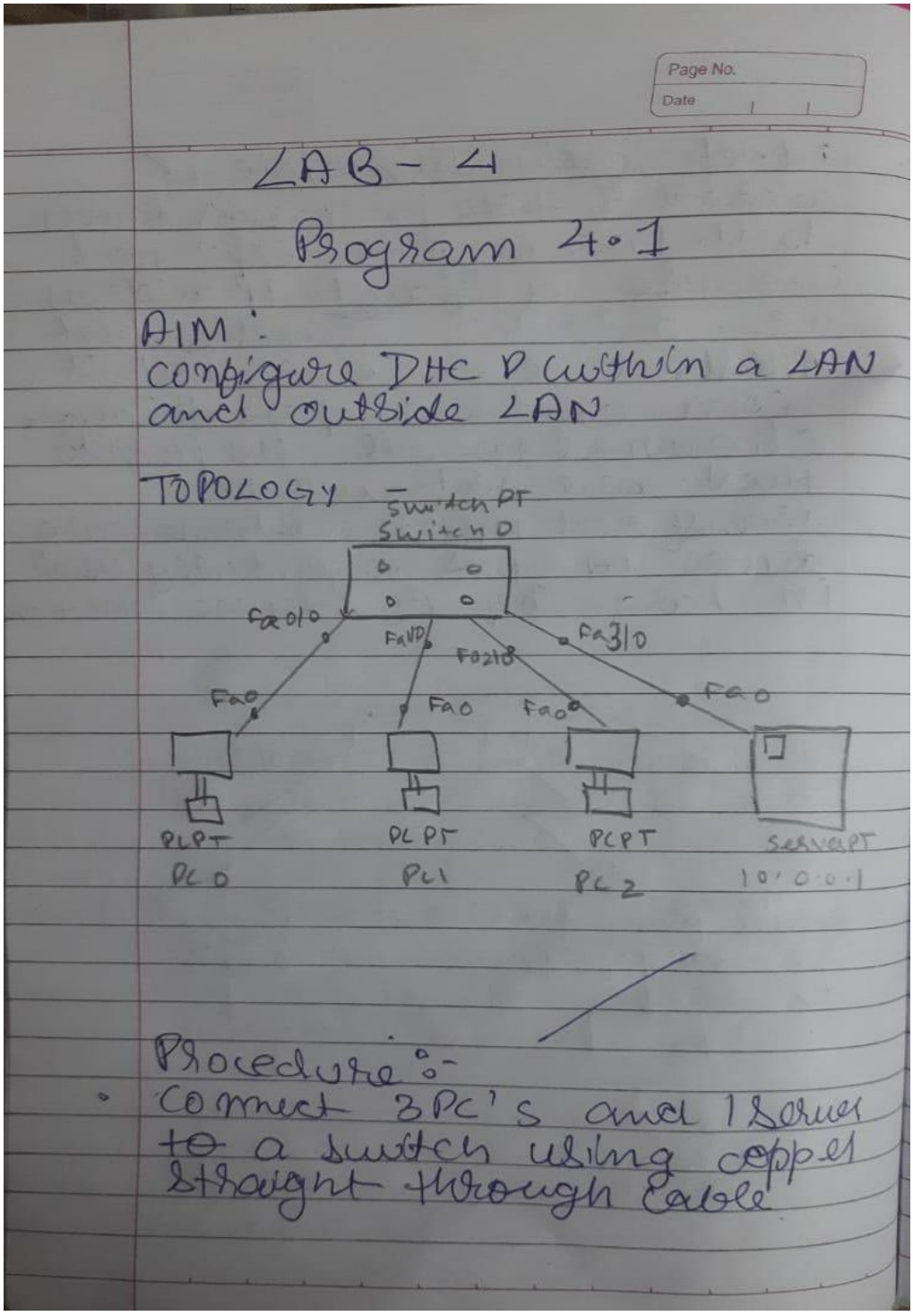



PC0
Physical
Config
Desktop
Custom Interface
Command Prompt
Packet Tracer PC Command Line 1.0
PC>ping 40.0.0.1
Pinging 40.0.0.1 with 32 bytes of data:
Request timed out.
Reply from 40.0.0.1: bytes=32 time=2ms TTL=125
Reply from 40.0.0.1: bytes=32 time=16ms TTL=125
Reply from 40.0.0.1: bytes=32 time=2ms 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>

LAB-4

Configure DHCP within a LAN and outside LAN.

OBSERVATION:



- Click on server and go to service tab select DHCP and turn on the DHCP service.
- Set the IP address as 10.0.0.2 and click on save button.
- Before this set the IP address of server in config Tab under fast Ethernet as 10.0.0.1
- Next click on PC 0 and go to desktop tab, click on IP config. select DHCP here it will request for an IP address and successfully get the DHCP request also set the IP address
- Repeat this steps for other 2 PC's
- To send a packet across PC's go to PC's command prompt and type ping destination IP address

PING OUTPUT:

Packet tracer PC command line 10.0

PC> Ping 10.0.0.3

Pinging 10.0.0.3 with 32 bytes of data

Page No.
Date

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 = 0 ms, Maximum = 0
Average = 0 ms

OBservation

- DHCP is used to dynamically assign an IP address to any device or node.
- It is a client-server protocol in which servers manage a pool of unique IP address & info about client config parameters
- DHCP-enabled clients send a request to DHCP server when they want to connect to a network.

Page No.
Date

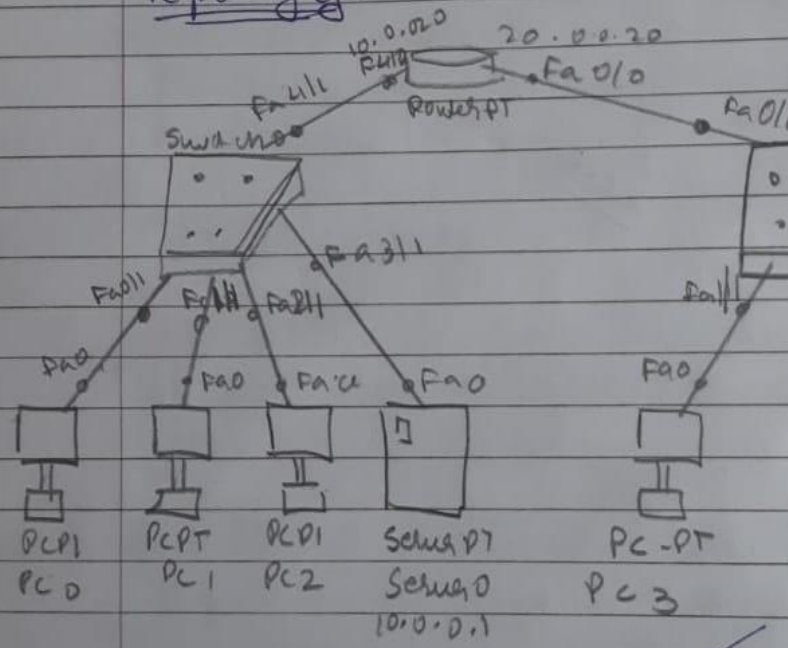
- DHCP server responds to the client request by providing IP config information from address pools previously specified by network administrator.

Page No.
Date

Program 4.02

Aim:-
configure DHCP within LAN and outside lan.

Topology:-



Procedure:-

Add a Router a switch and 2 PC to 4.1 program Network & connect the Router to both switches.

Page No.
Date

- Set the Server IP address of Server and with the help of Server Set the first 3 PC's IP address through DHCP
- Now Set the Router IP address with the following commands Statically.

Step1: No
Step2: Enable
Step 3: ConfigT
Step 4: Interface fastethernet 4/0
Step 5: IP address 10.0.0.20
Step 6: NO shut 255-0-0-0
Step 7: Exit
Step 8: Interface fast ethernet 0/0
Step 9: IP address 20.0.0.20 255.0.0.0
Step 10: No shut
Step 11: Exit
Step 12: Exit
Step 13: Show IP Route

- Go to Server and set the gateway as 10.0.0.20
- Again go to Router CLI and follow these commands.

Step 14: ConfigT
Step 15: Interface fast ethernet 0/0

Step 16: IP helper-address 20.0.0.1
Step 17: No Shut
Step 18: Exit.

- Now, go to Services and add one more pool name as Server Pool1 Start IP address as 20.0.0.2 and default gateway as 20.0.0.20 then click add & Save.

- Now set the other two PC's IP address by going to this Desktop → IP config and selecting DHCP which will automatically generate its IP address.

- Now the network is complete and can send packets from any PC to other by typing ping destination IP address in their respective command prompts

PING OUTPUT

Packet Tracer PC Command Line 1.0
PC> ping 20.0.0.2
Pinging 20.0.0.0.2 with 32 bytes of data
Request timed out.

Page No.
Date

~~Reply from 20.0.0.2 :~~
Reply from 20.0.0.2 : bytes = 32 time = 0ms TTL = 128
Reply from 20.0.0.2 : bytes = 32 time = 0ms TTL = 128
Reply from 20.0.0.2 : bytes = 32 time = 0ms TTL = 128

Ping statistics for 20.0.0.2

packets Sent = 4, Recieved = 3, lost = 1 (25% loss)
Approximate round trip times in milliseconds:
Minimum = 0ms, Maximum = 0ms, Average = 0ms.

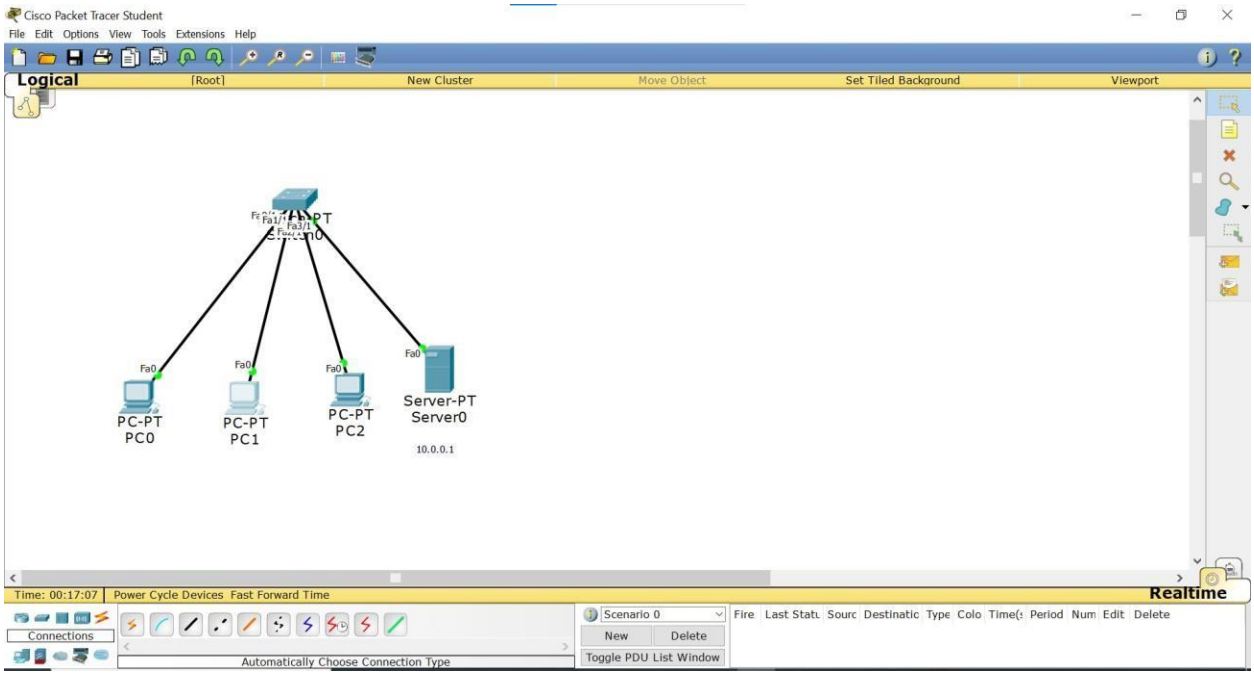
Observation :-

- DHCP is used to assign IP address dynamically to different devices.
- To assign continious IP address. we create a serverpool where we assign the starting IP address and a default gateway number. For PC's under different switches. We create a different serverpool again and start. This takes care of delivering the packets to correct destination IP address & also sends back the Acks to the initial device.

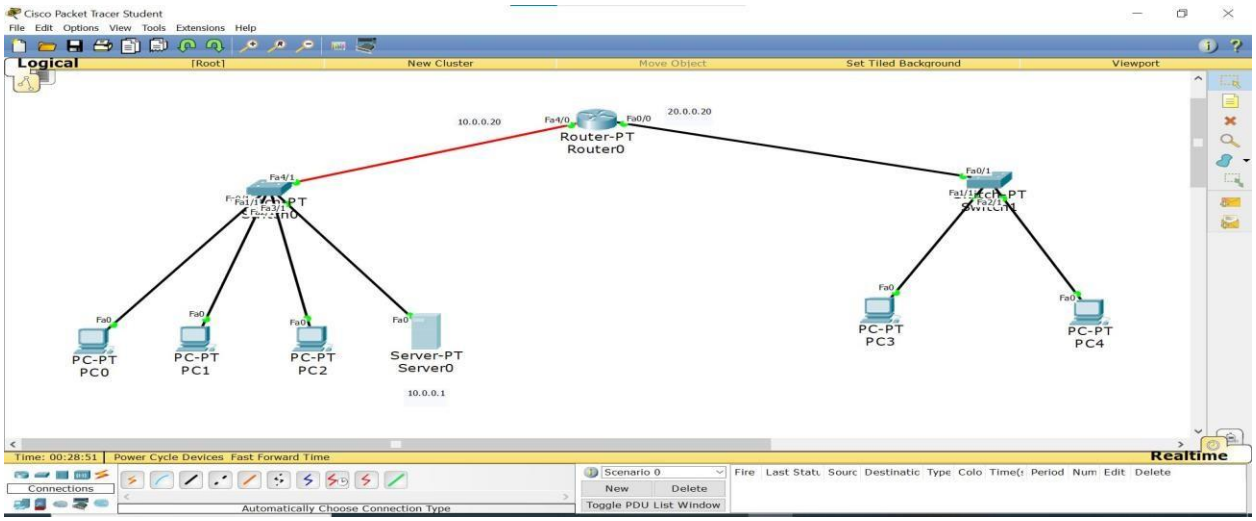
19/12

TOPOLOGY:

PROGRAM 4.1:

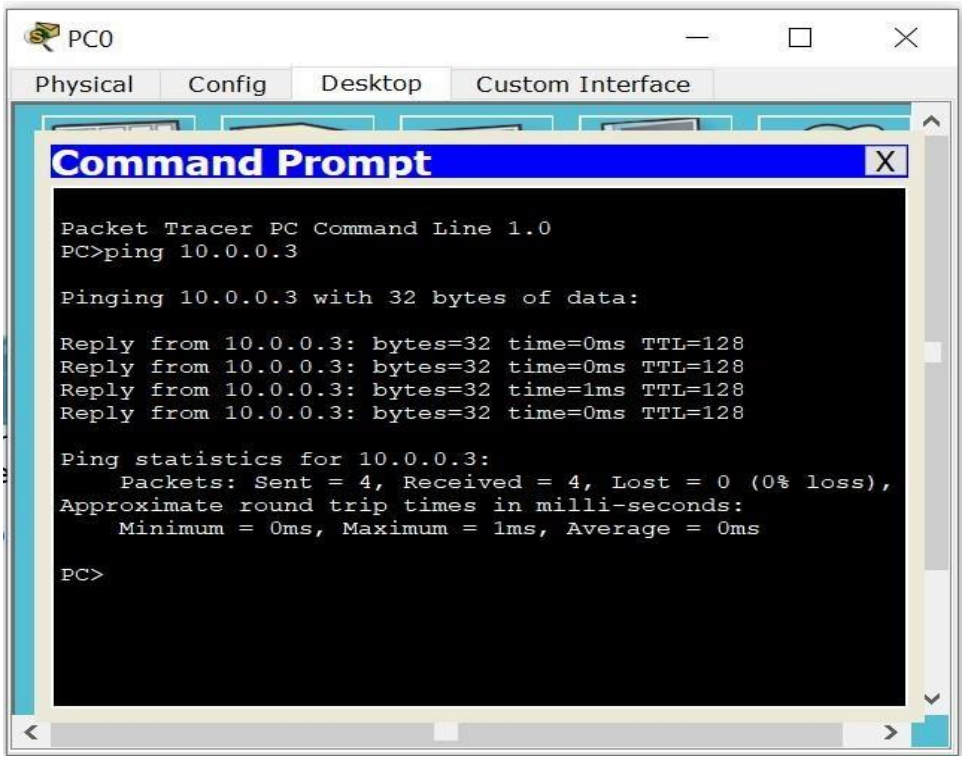


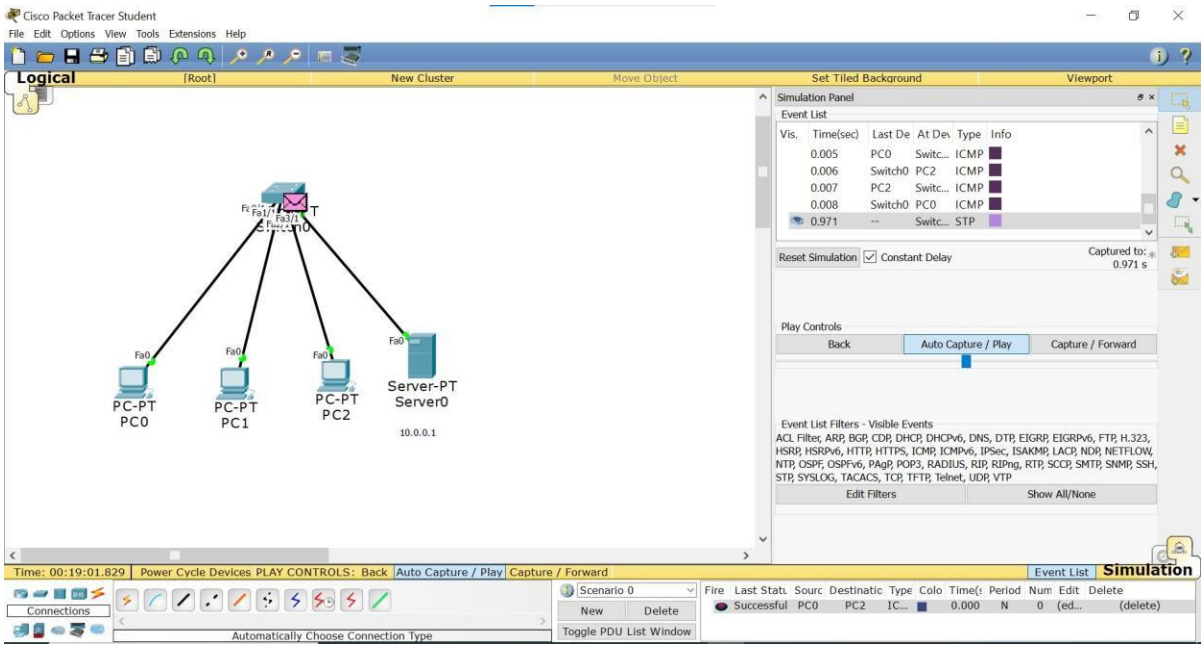
PROGRAM 4.2:



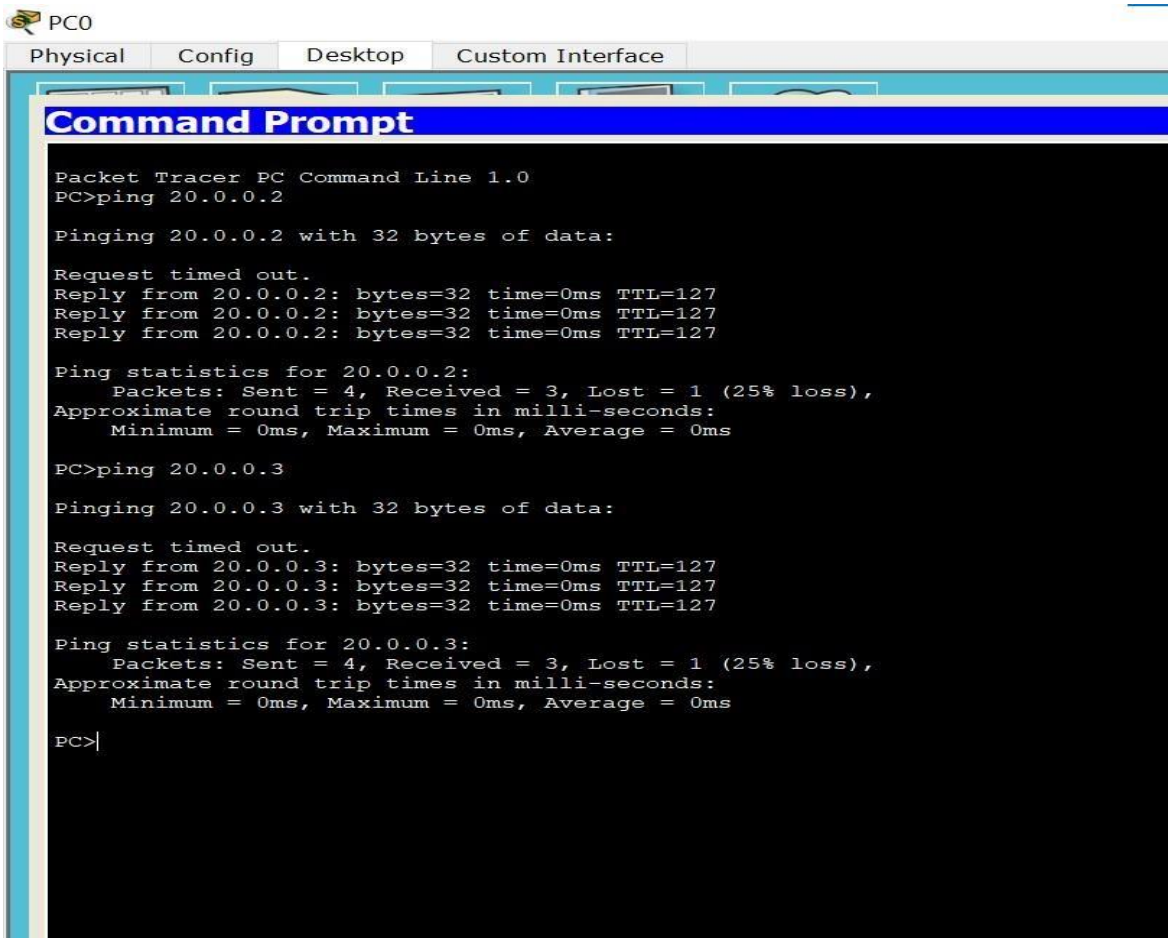
OUTPUT:

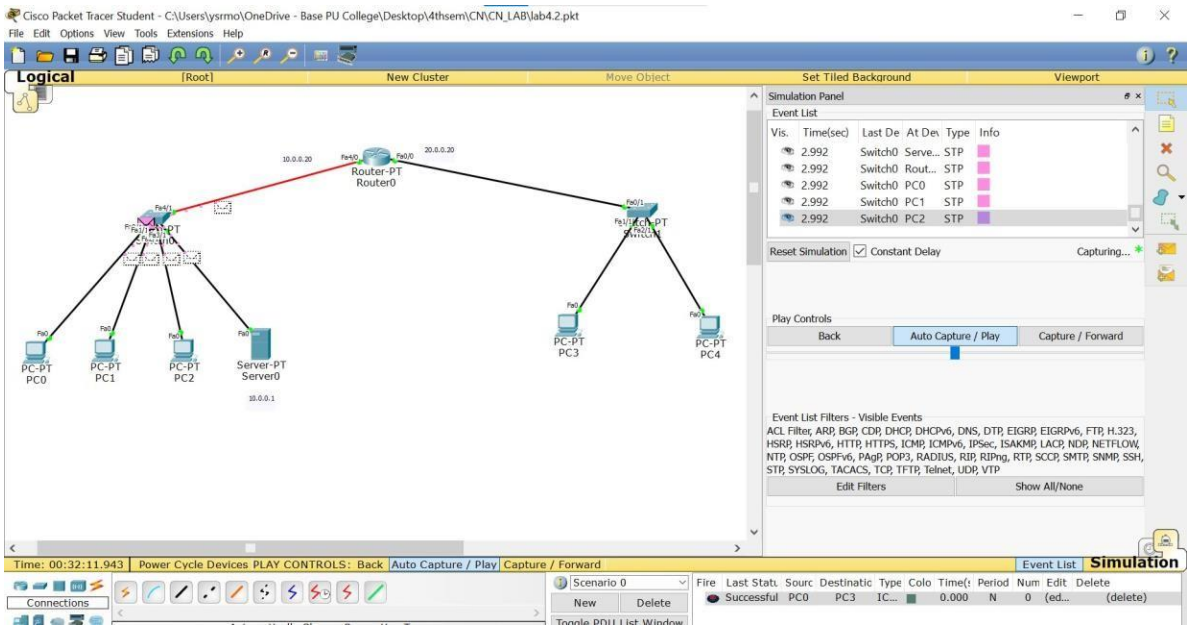
PROGRAM 4.1:





PROGRAM 4.2:

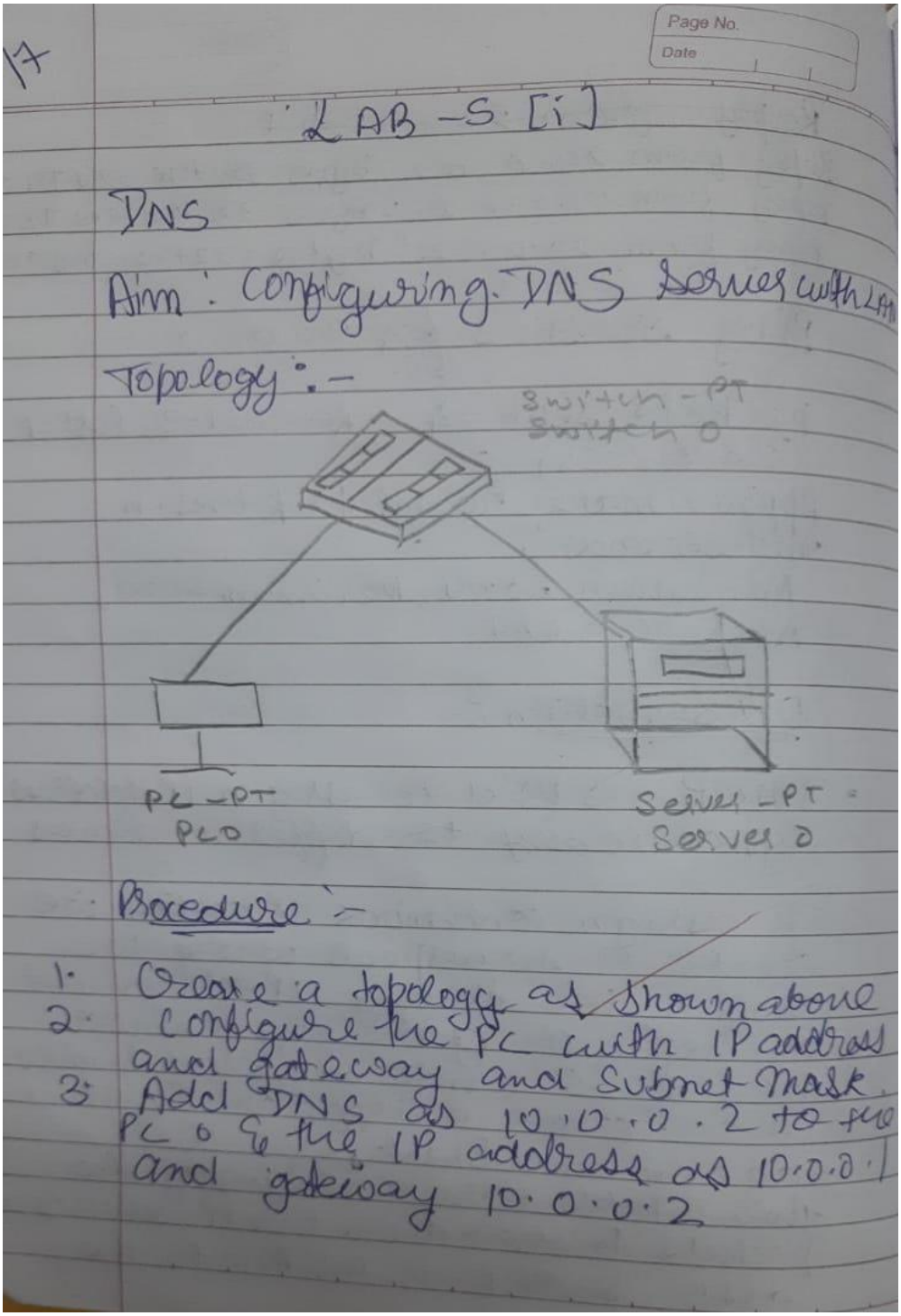

PC0
Physical
Config
Desktop
Custom Interface
Command Prompt
Packet Tracer PC Command Line 1.0
PC>ping 20.0.0.2
Pinging 20.0.0.2 with 32 bytes of data:
Request timed out.
Reply from 20.0.0.2: bytes=32 time=0ms TTL=127
Reply from 20.0.0.2: bytes=32 time=0ms TTL=127
Reply from 20.0.0.2: bytes=32 time=0ms 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>


Logical
Router-PT
Router0
PC-PT
PC0
PC-PT
PC1
PC-PT
PC2
Server-PT
Server0
PC-PT
PC3
PC-PT
PC4
Simulation Panel
Event List
Reset Simulation
Constant Delay
Play Controls
Back
Auto Capture / Play
Capture / Forward
Edit Filters
Show All/None
Simulation

LAB-5

Configure Web Server, DNS within a LAN.

OBSERVATION:



Page No.
Date

4. Click on Server and click on Services → HTTP & HTTPS click on 'ON' and add Name! test.com Address :- 10.0.0.2

5. Now go to Config → Fast Ethernet – IP address 10.0.0.2 Subnet Mask 255.0.0.0

6. Now go to Settings in the config gateway 10.0.0.2 & DNS Server 10.0.0.2

7 Now go to PC click on desktop → web browser and type the 'test.com' in URL and click on go

8. You will see the html.index that has written in the HTTPS button

Observation

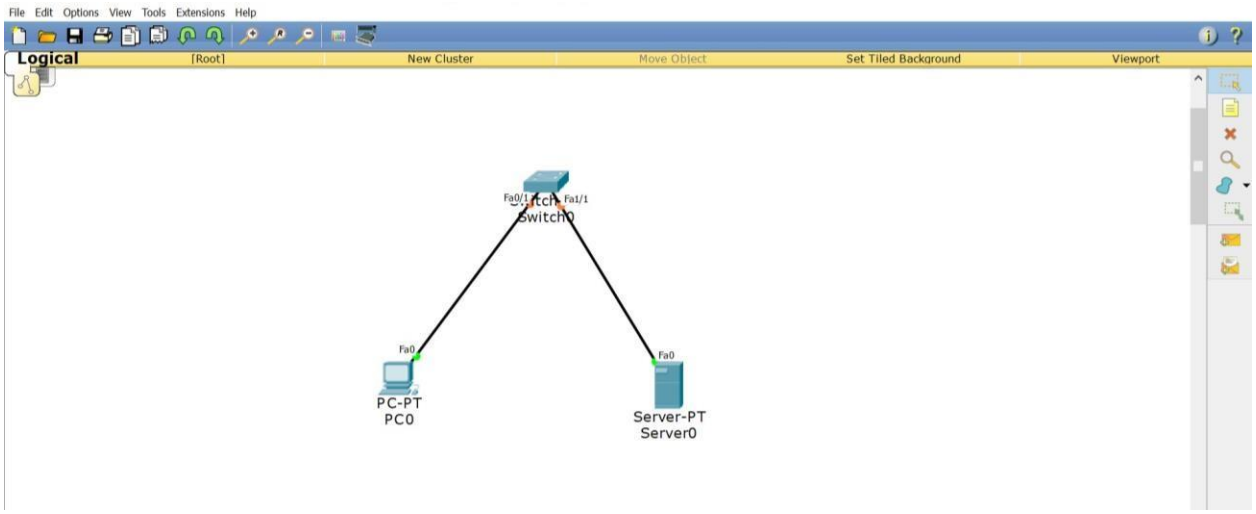
DNS is the Domain Name System. DNS is linked to the internet and focused on a system using Internet protocol (IP). DNS Servers are required for the

Page No.
Date

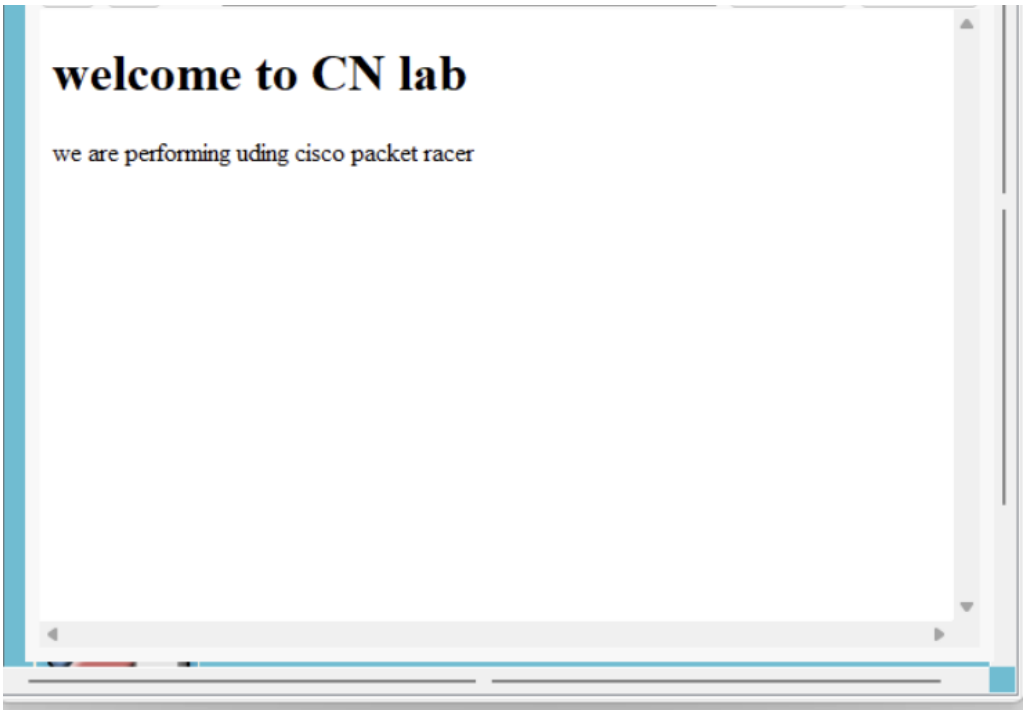
working of DNS. the IP address is calculated with the aid of a lookup table.

→ PC communicated with the server using the IP address as well as using the Domain Name.

TOPOLOGY:



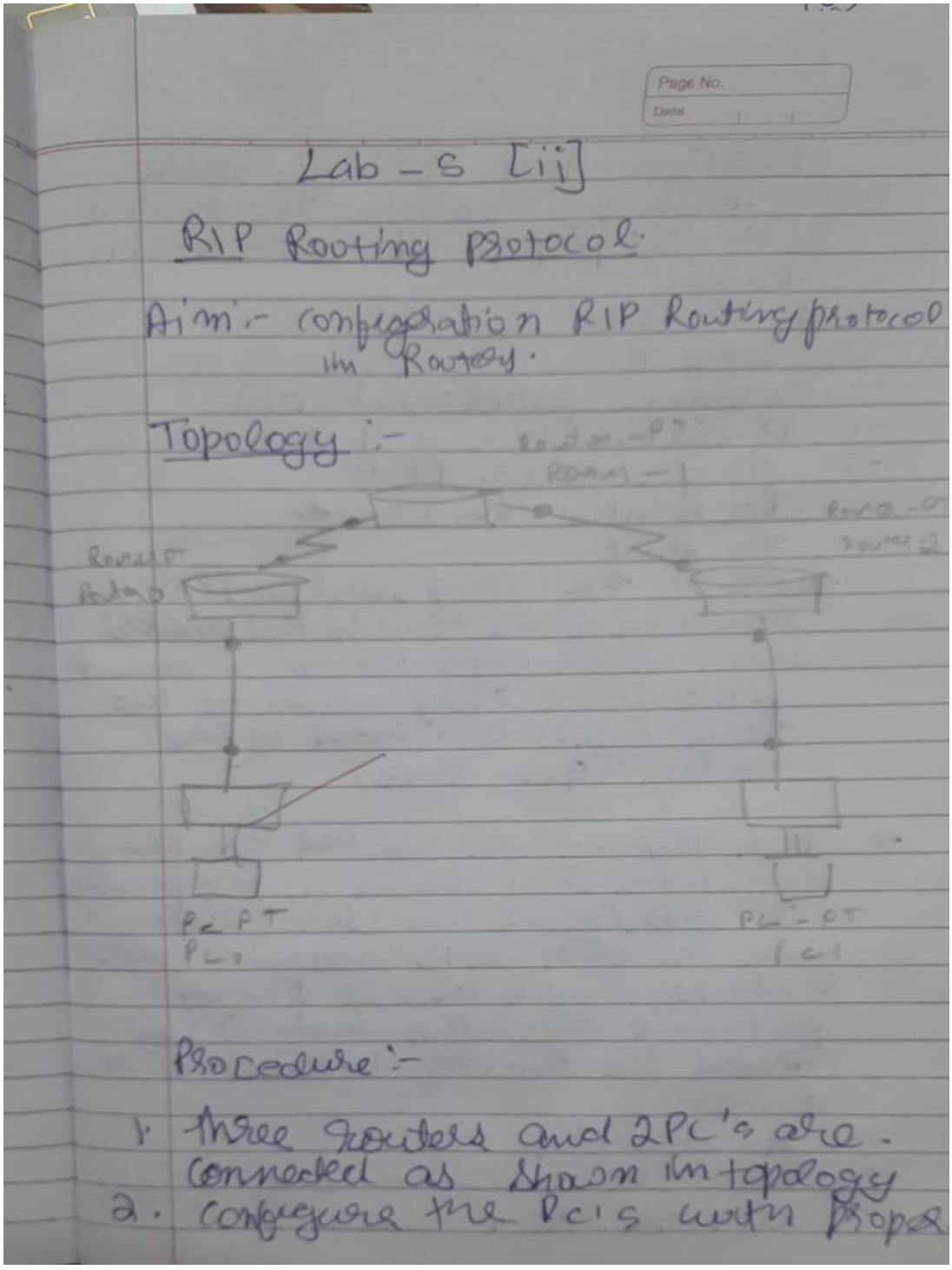
OUTPUT:



WEEK6

Configure RIP routing Protocol in Routers.

OBSERVATION:



IP address and gateway address

3. Similarly, configure the Routers with the proper IP address in the mode.

→ N. Enable
→ config T
→ Interface gastethernet 0/0
→ IP address 10.0.0.1 255.0.0.0
→ Encapsulation PPP
→ Clock Rate 64000
no shut

Note:- the encapsulation PPP should be given to all the routers and 'clock rate 64000' command should be only given to the clock symbolled sides of the routers ('i' open sides)

→ For making the routers to know about the other devices in the previous 2 experiments we used 1 static and the other with dynamic routing but here we use a Routing protocol algorithm that it self makes the router to know other devices

→ Router rip
→ Network 20.0.0.0 ↓ router2
→ network 30.0.0.0

Page No.
Date

→ Router RIP
→ network 30.0.0.0 } Router 3
network 40.0.0.0

→ Router RIP
→ network 10.0.0.0 } Router 1
→ network 20.0.0.0

ping output

P< > ping 40.0.0.1

Pinging 40.0.0.1 with 32 bytes of data:
Reply from 40.0.0.1 bytes=32 time=0ms TTL=2
" " " " " " " "
" " " " " " " "
" " " " " " " "
Ping statistics from 40.0.0.1
packets Sent=4 Received 4 Lost=0
(0% loss)
Approximate round trip times in ms
Minimum =0 ms, Maximum, 0ms Average
=0ms

P. T. O

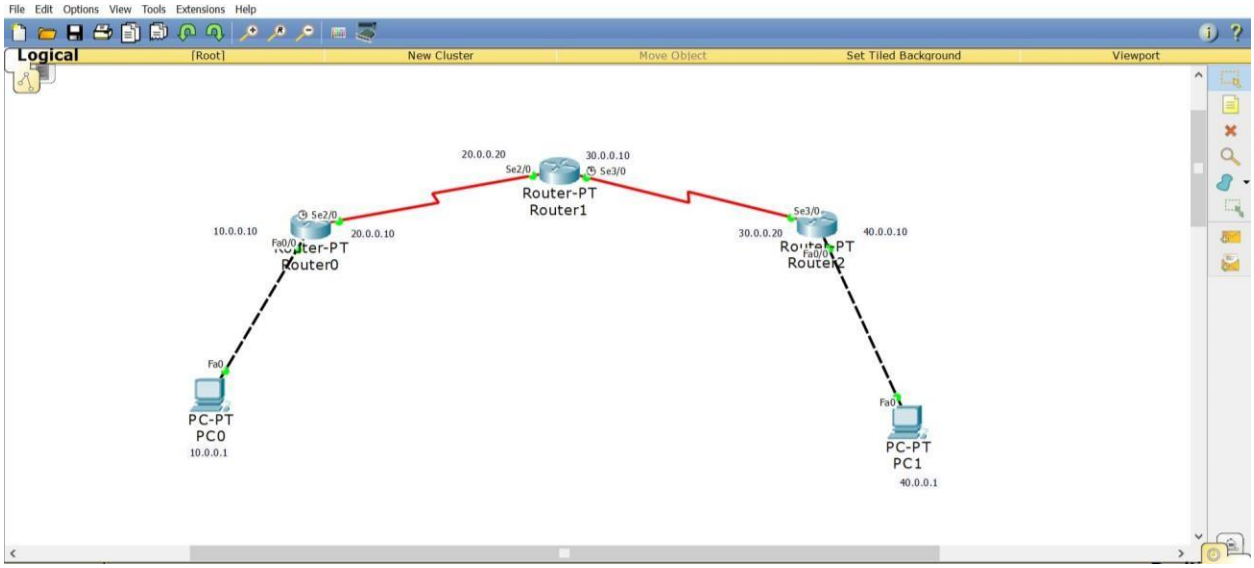
Page No.
Date

Observation:-

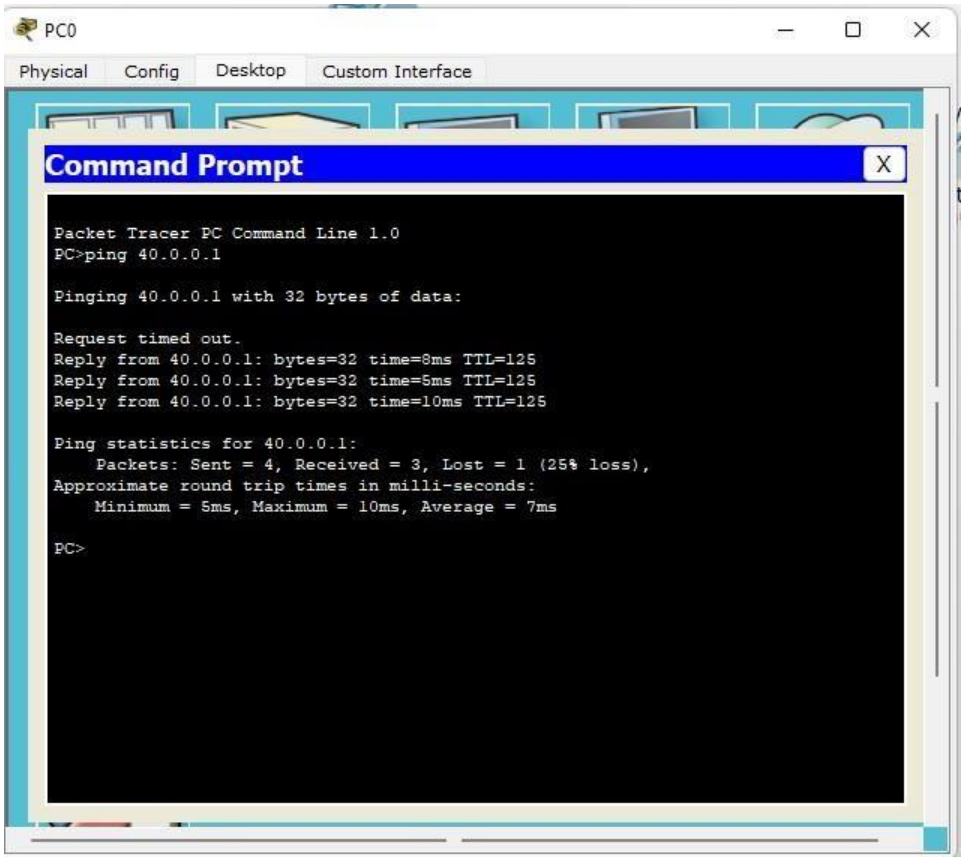
RIP is the Routing Information protocol is a distance vector protocol that uses hop count as its primary matric. RIP defines how routers should be share information when moving traffic among an interconnected group of local area networks

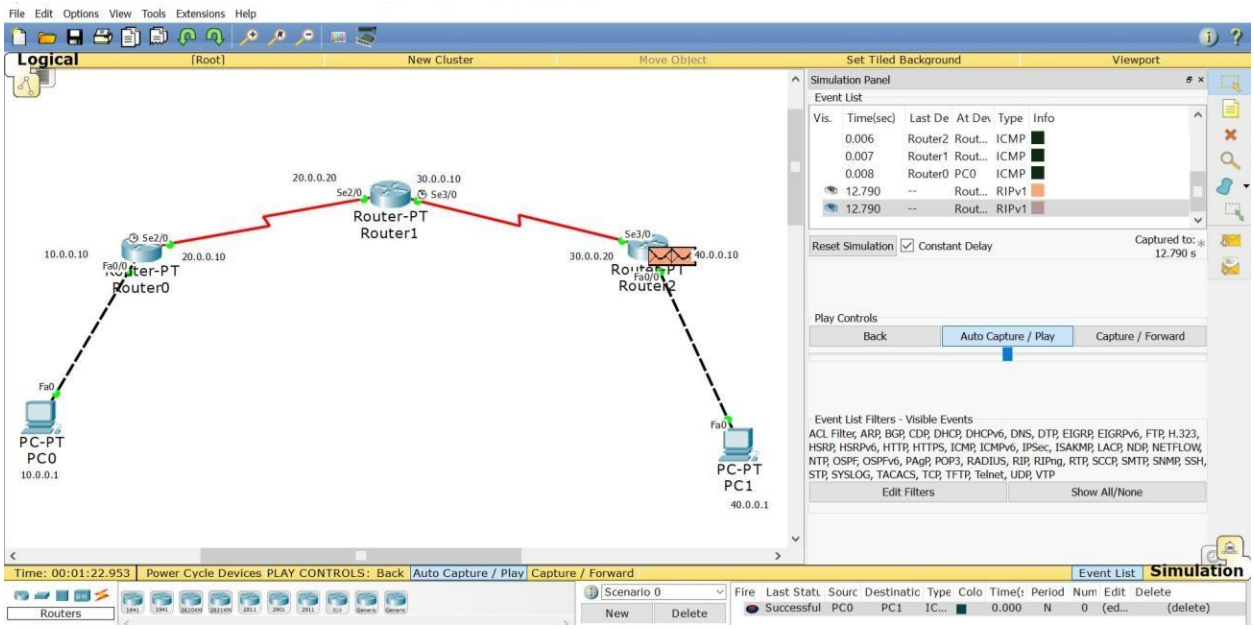
→ the RIP protocol here, used to connect the routers to one other and PC's using RIP protocol and message is pinged successfully

TOPOLOGY:



OUTPUT:

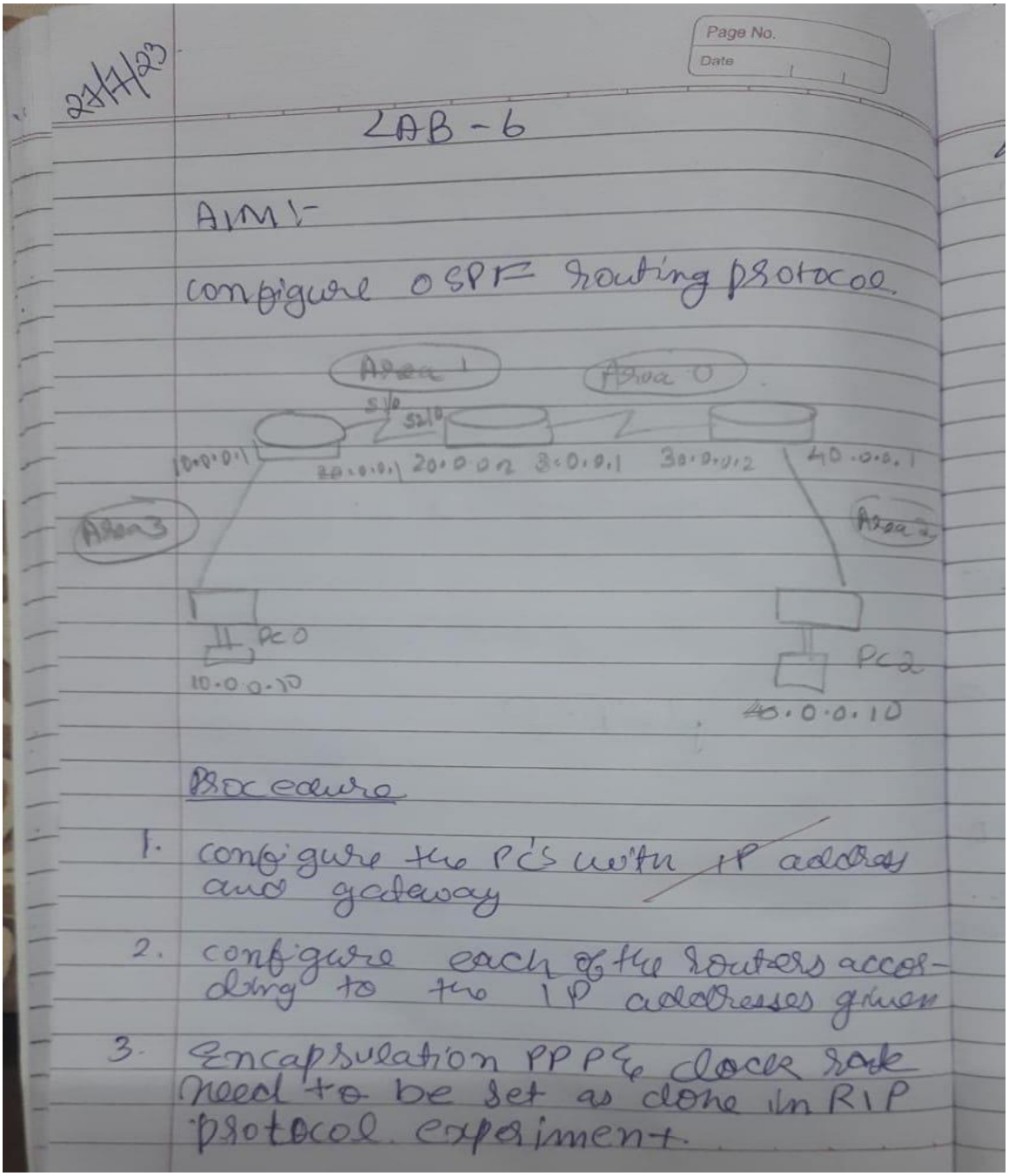



File Edit Options View Tools Extensions Help
Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
Router-PT
Router1
Router-PT
Router0
Router-PT
Router2
PC-PT
PC0
PC-PT
PC1
Simulation Panel
Event List
Reset Simulation
Constant Delay
Play Controls
Back
Auto Capture / Play
Capture / Forward
Event List Filters - Visible Events
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

WEEK7

Configure OSPF routing protocol.

OBSERVATION:



Page No.
Date

4. Now enable IP routing by configuring OSPF routing protocol in all routers.

In Router R_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
```

In Router R2

```
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
```

In Router R3

```
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
0.255.255.253 area
R3 (config-routes)# exit

4) Loopback:-

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

R2 (config-if)# interface loopback
R2(config-if)# ip address 172.16.1.
253 255.255.0

R2(config-if)# no shutdown

R3 (config-if)# interface loopback
R3(config-if)# ip address 172.16.
1.254 255.255.0
R3 (config-if)# no shut

(T) (L)

Page No.
Date

5] Virtual link.

→ In R1

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

→ In Router R2

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

→ Show IP routes.

O IA 10.0.0.0/8 [110/129] via 30.0.0.1 serial 3/0
O IA 20.0.0.0/8 [110/128] via 30.0.0.1 serial 3/0
30.0.0.0/8 is variably subnetted
2 is subnet, 2 masks
C 30.0.0.0/8 is directly connected, serial 3/0
C 30.0.0.1/32 is directly connected, serial 3/0
C 40.0.0.0/8 is directly connected fastethernet 0/0
C 172.16.0.0/16 is directly connected. Loop back 0

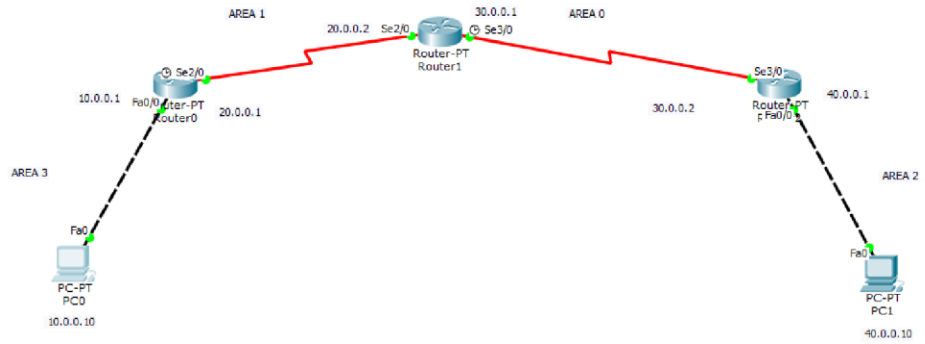
Page No.
Date

Ping Output

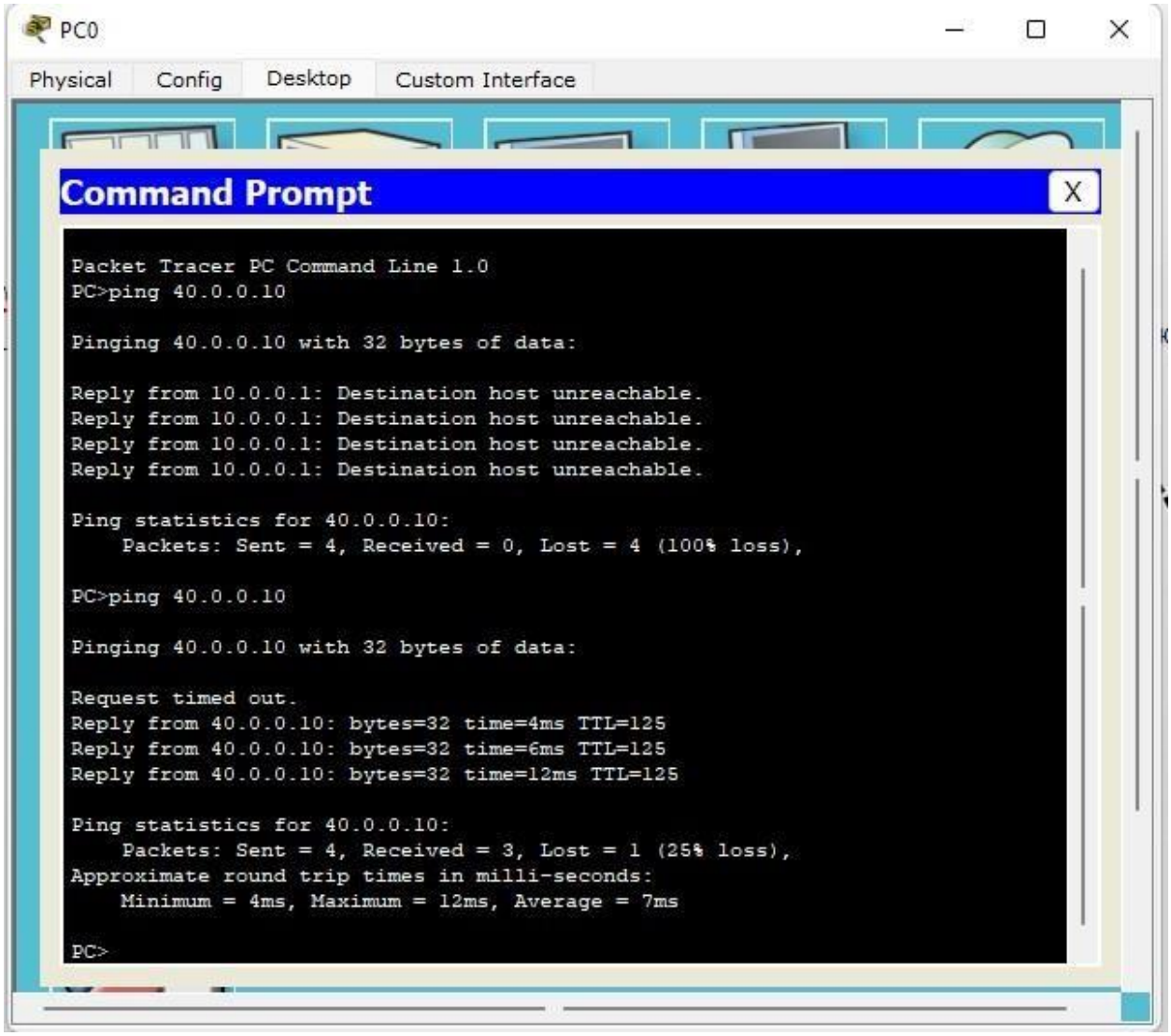
pinging 40.0.0.10 with 32 bytes of data
Request timimed out
Reply from 40.0.0.10 bytes = 32 time = 2ms TTL=128
Reply from 40.0.0.10 bytes = 32 time = 2ms TTL=128
Reply from 40.0.0.10 bytes = 32 time = 10ms TTL=128
Pinging statistics for 40.0.0.10:
Packets: Sent = 4, Recieved = 3, Lost = 1 (25% loss) Approx Round trip in Ms
Min = 2ms Max = 10ms Average = 7ms

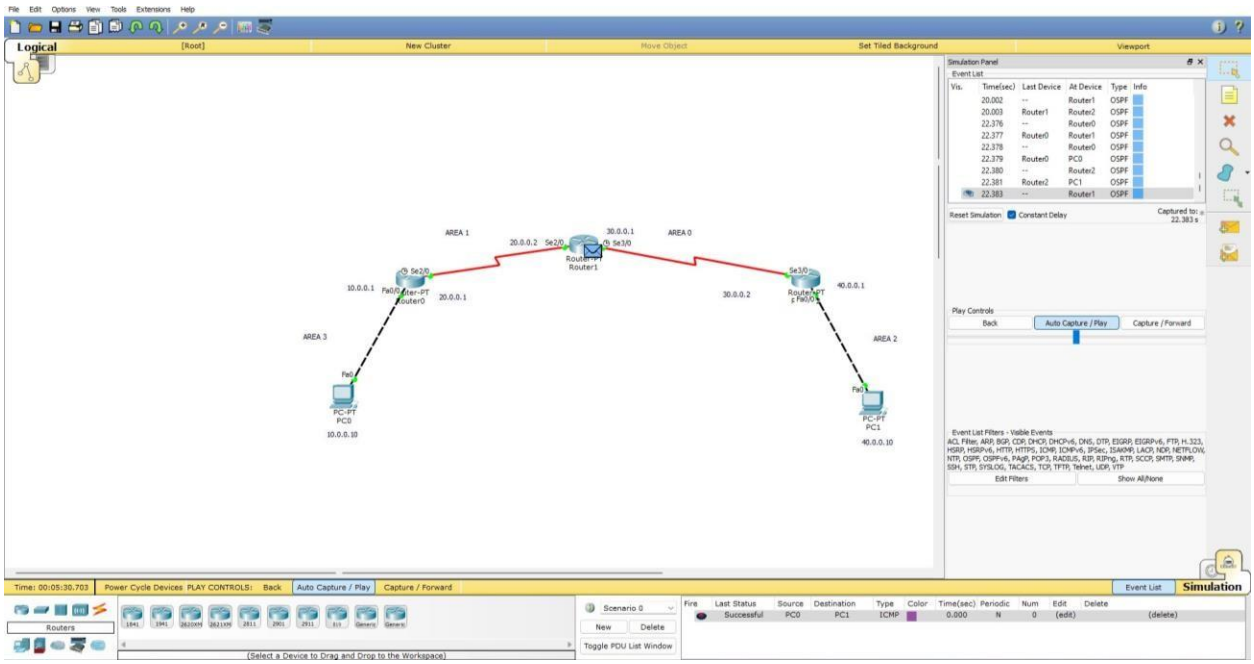
1/8/23

TOPOLOGY:



OUTPUT:

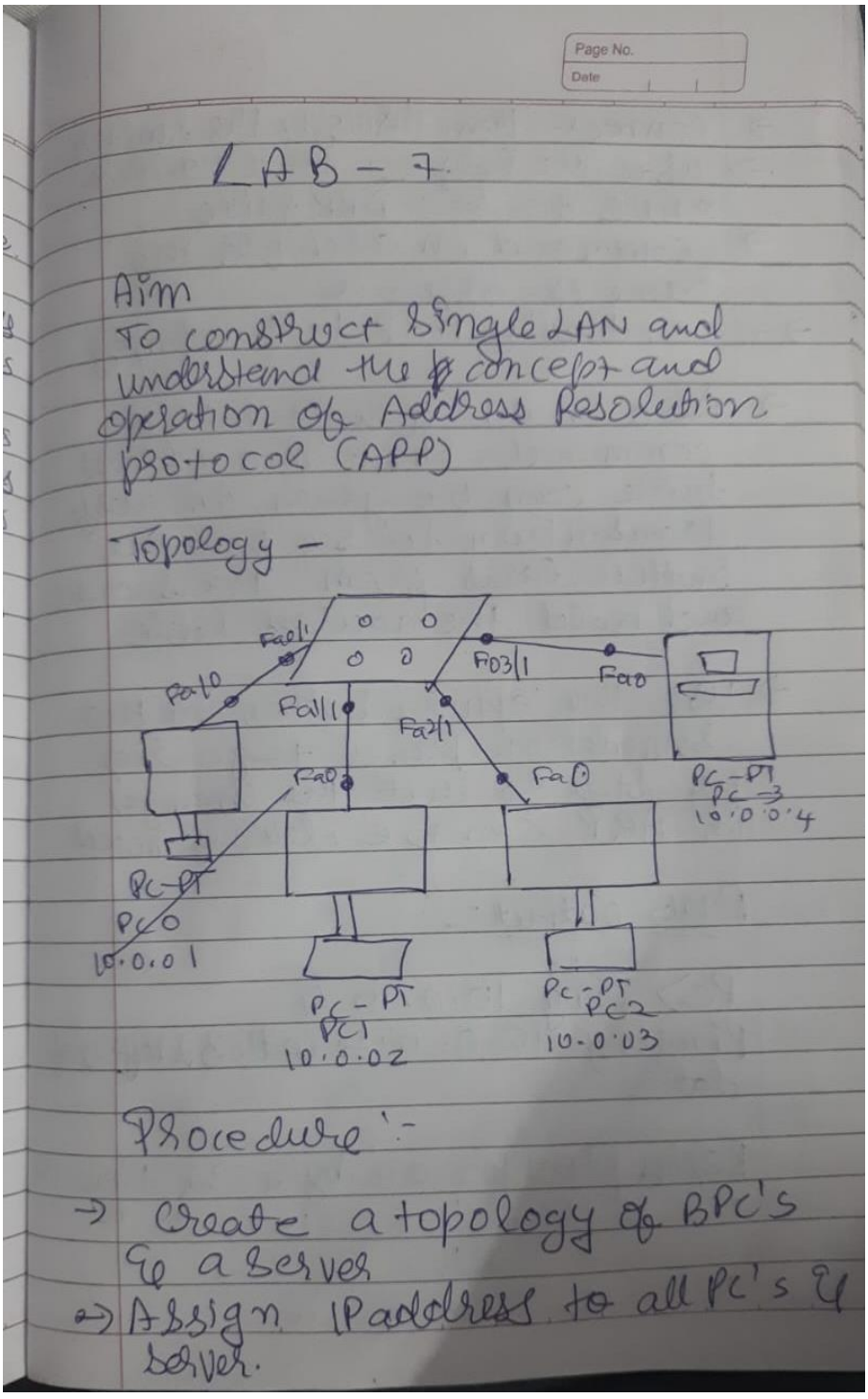


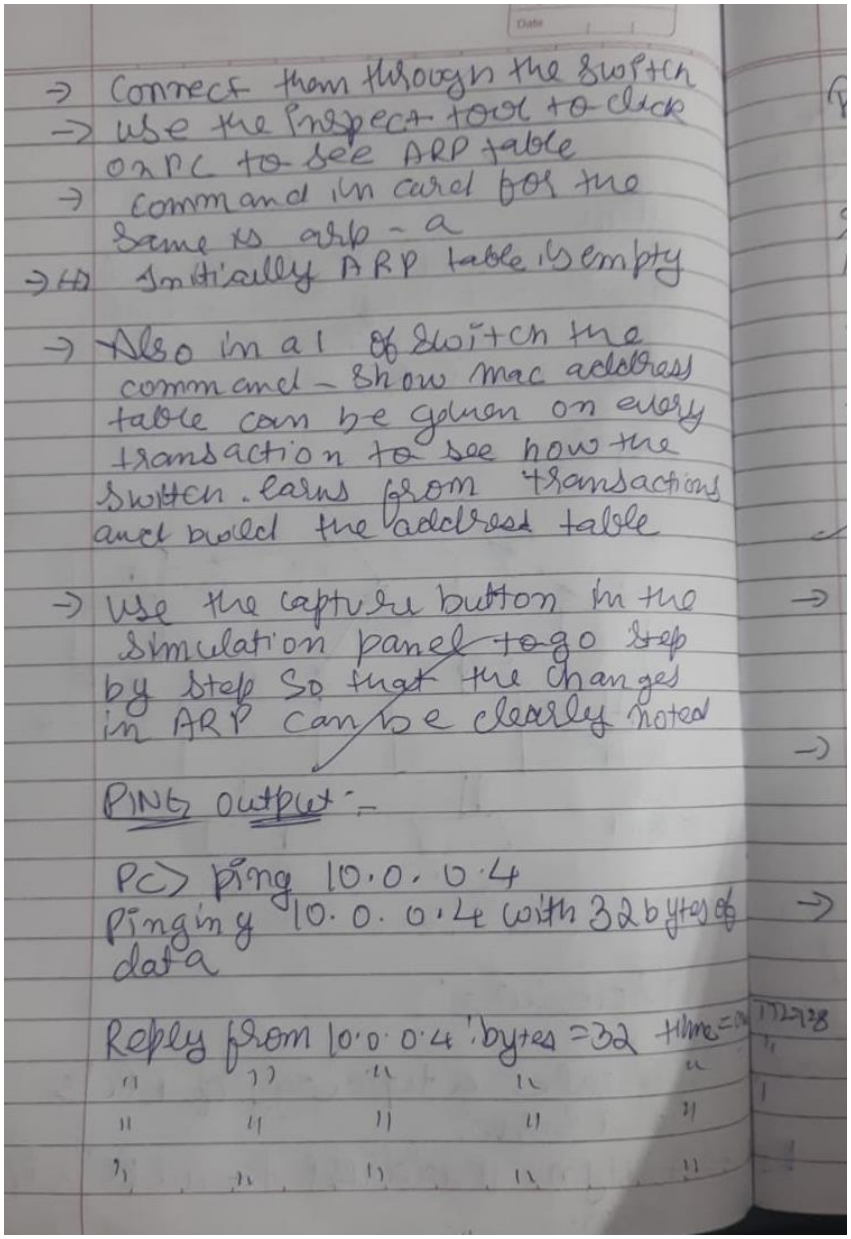

File Edit Options View Tools Extensions Help
Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
AREA 1
AREA 0
AREA 3
AREA 2
20.0.0.2
30.0.0.1
10.0.0.1
20.0.0.1
30.0.0.2
40.0.0.1
10.0.0.10
40.0.0.10
Router1
Router0
PC0
PC1
Simulation Panel
Event List
Play Controls
Back
Auto Capture / Play
Capture / Forward
Time: 00:05:30.703
Power Cycle Devices PLAY CONTROLS:
Routers
Scenario 0
New
Delete
Toggle PDU List Window
(Select a Device to Drag and Drop to the Workspace)
Event List
Simulation

LAB 8

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

OBSERVATION:





→ Connect them through the switch
→ Use the Inspect tool to click on PC to see ARP table
→ command in card for the same is arp - a
→ Initially ARP table is empty

→ Also in a switch the command - show mac address table can be given on every transaction to see how the switch learns from transactions and build the address table

→ Use the capture button in the simulation panel to go step by step so that the changes in ARP can be clearly noted

PING output :-

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 =

Page No.
Date

Ping statistics for 10.0.0.4
packets: Sent = 4, Received = 4,
Lost = 0 (0% loss), Approximate
round trip time in milliseconds:
Minimum = 0ms, Maximum = 0ms, Average = 0

PC > arp -a

Internet address Physical Address Type
10.0.0.4 0-060.2fa0.324d dynamic
19/8/2023

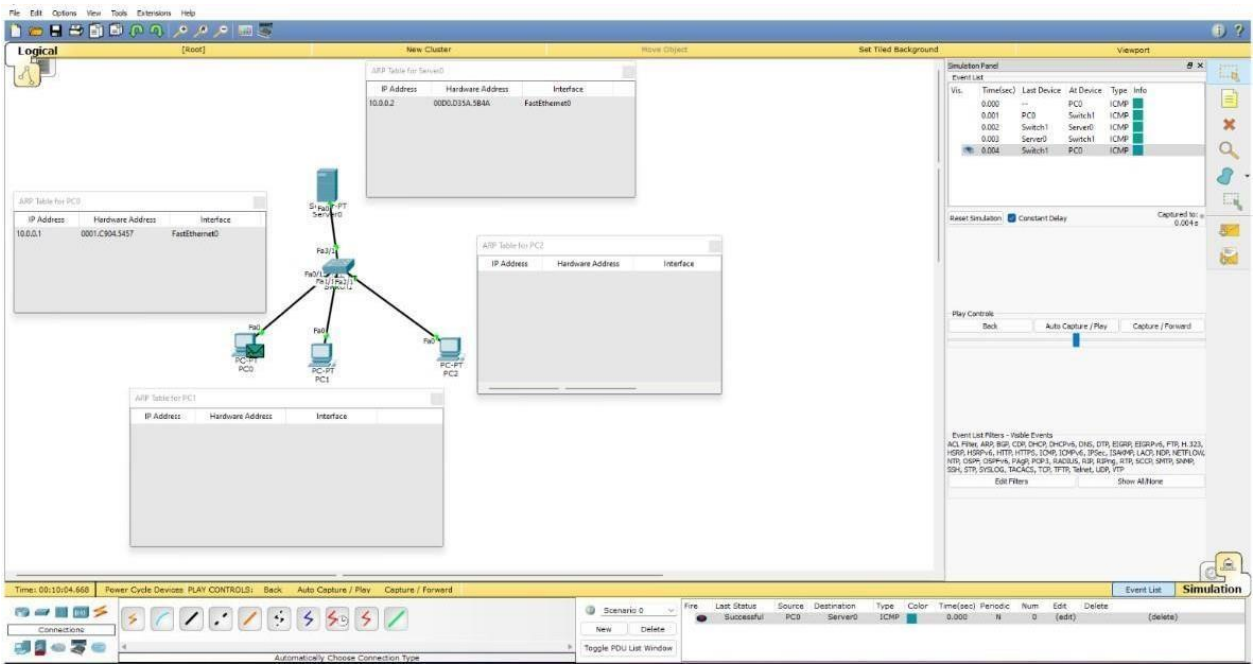
Observation :-

→ When we ping 1 PC and server the address of server is known to PC & vice versa

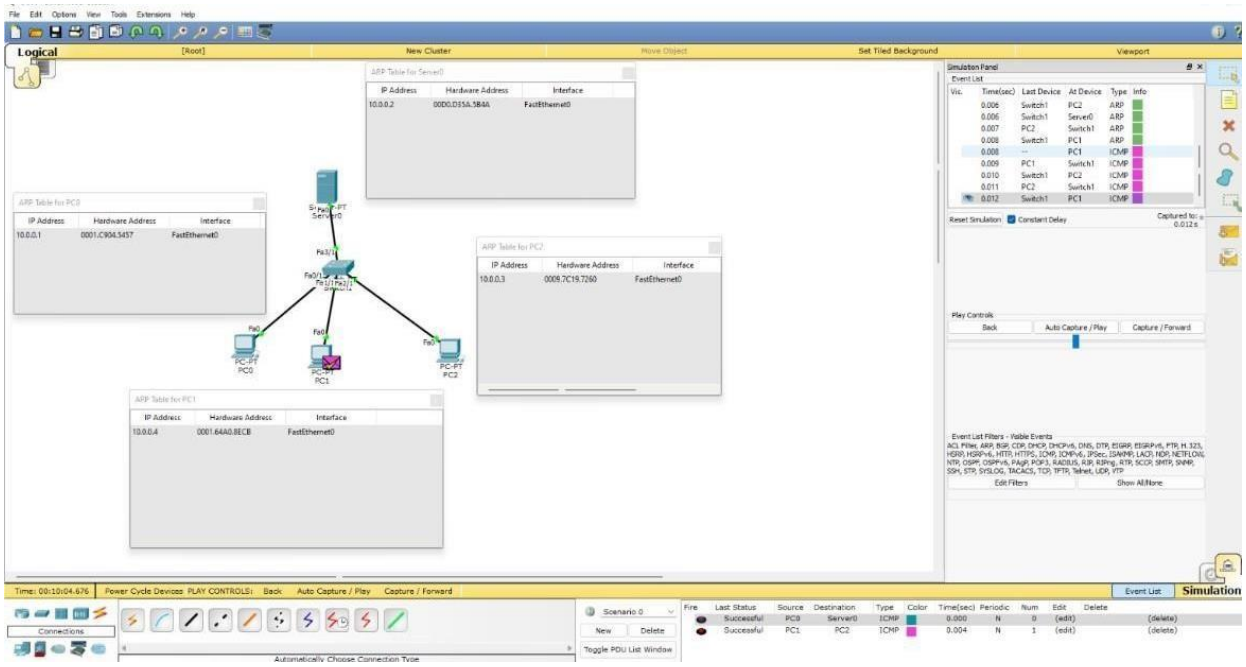
→ When we ping between other two PC's simultaneously the address of each other are known

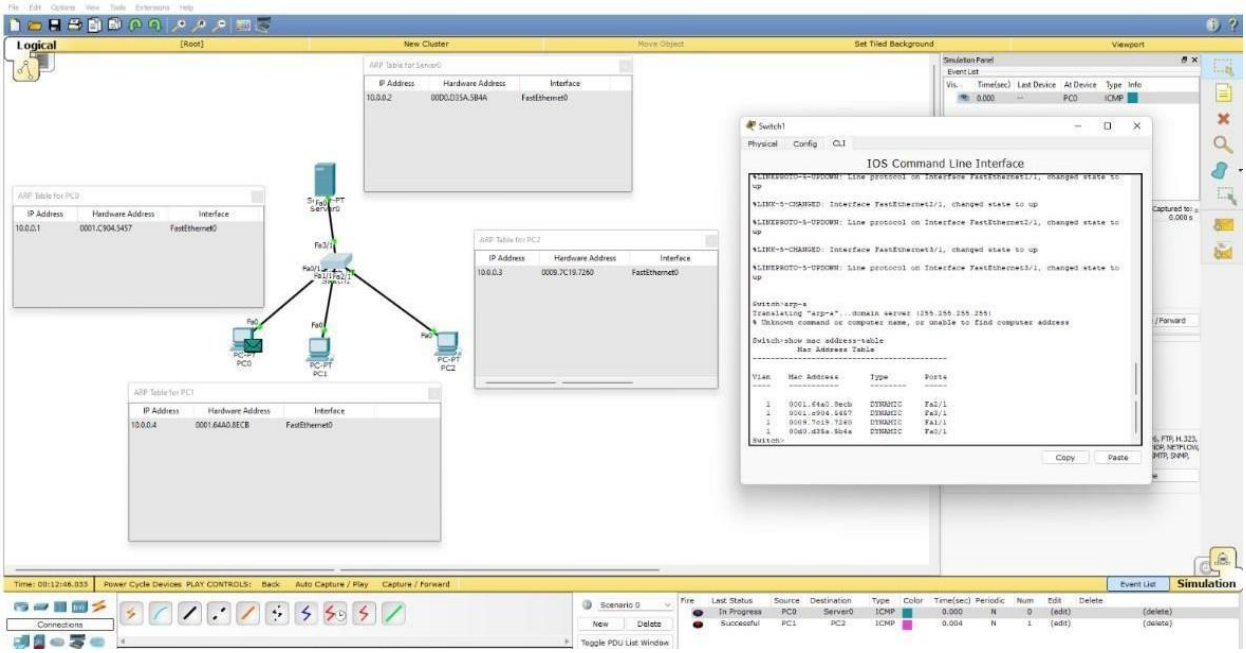
→ Every time a host requests a MAC address in order to send a packet to another host in the LAN, it checks its ARP cache to see if the IP to mac address translation address already exists if the translation doesn't exist it performs ARP.

TOPOLOGY:



OUTPUT:

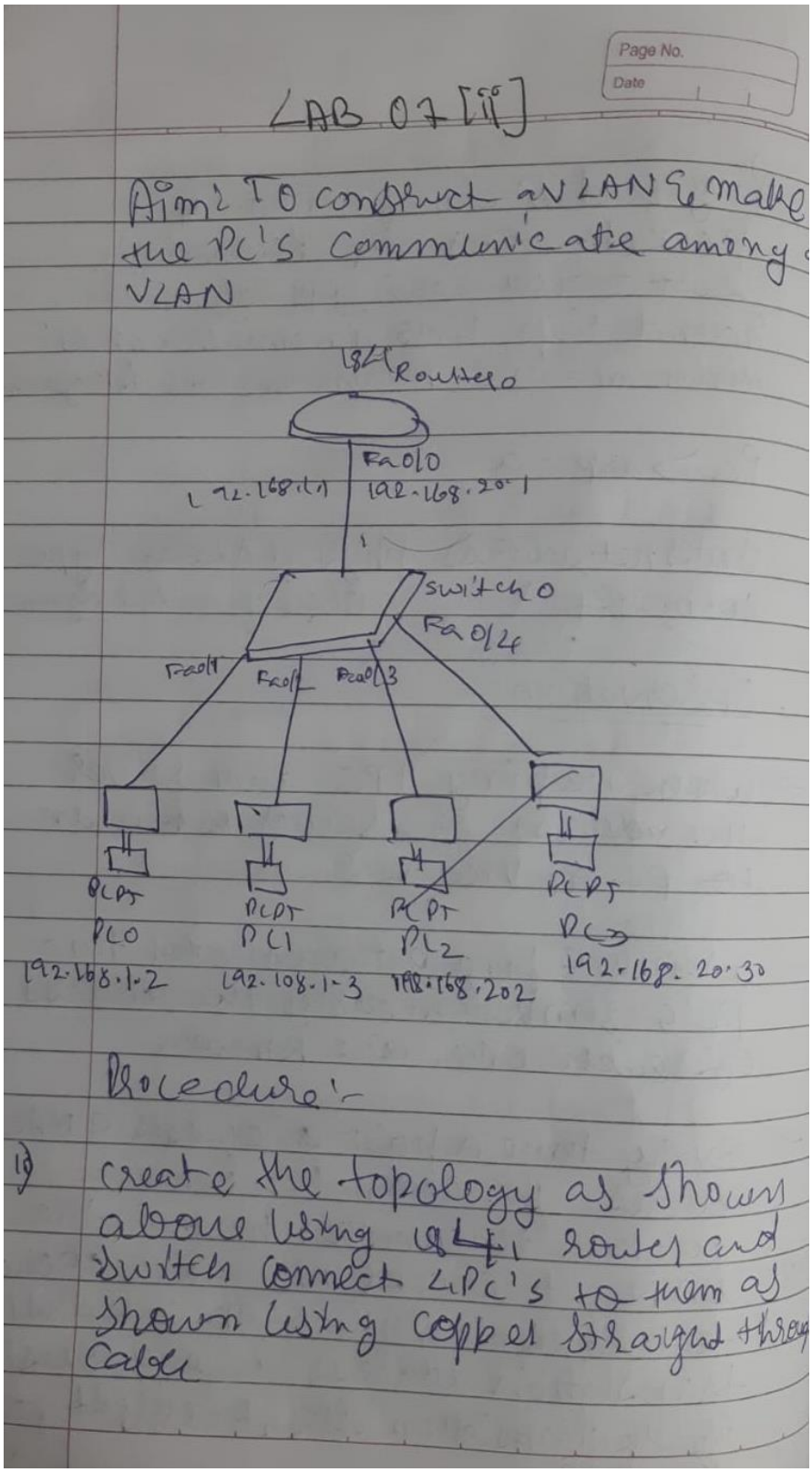



Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
ARP Table for Server0
IP Address
Hardware Address
Interface
10.0.0.2
00D0.D35A.3B4A
FastEthernet0
ARP Table for PC0
10.0.0.1
0001.C904.5457
ARP Table for PC2
10.0.0.3
0009.7C19.7260
ARP Table for PC1
10.0.0.4
0001.64A0.8ECB
Switch1
Physical
Config
CLI
IOS Command Line Interface
%LINK-5-CHANGED: Interface FastEthernet2/1, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet2/1, changed state to up
%LINK-5-CHANGED: Interface FastEthernet3/1, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet3/1, changed state to up
Switch>arp -a
Translating "arp-a"...domain server (255.255.255.255)
% Unknown command or computer name, or unable to find computer address
Switch>show mac address-table
Mac Address Table
Vlan Mac Address Type Ports
1 0001.64a0.8ecb DYNAMIC Fa2/1
1 0001.c904.5457 DYNAMIC Fa3/1
1 0009.7c19.7260 DYNAMIC Fa1/1
1 00d0.d35a.3b4a DYNAMIC Fa3/1
Switch>
Copy
Paste
Simulation Panel
Event List
Vis.
Time(sec)
Last Device
At Device
Type
Info
PC0
ICMP
Power Cycle Devices
PLAY CONTROLS:
Back
Auto Capture / Play
Capture / Forward
Event List
Simulation
Connections
Scenario 0
New
Delete
Toggle PDU List Window
Fire
Last Status
Source
Destination
Type
Color
Time(sec)
Periodic
Num
Edit
Delete
In Progress
PC0
Server0
ICMP
0.000
N
0
(edit)
(delete)
Successful
PC1
PC2
ICMP
0.004
N
1
(edit)
(delete)

LAB 9

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

OBSERVATION:



Page No.
Date

2 We use class C addressing here

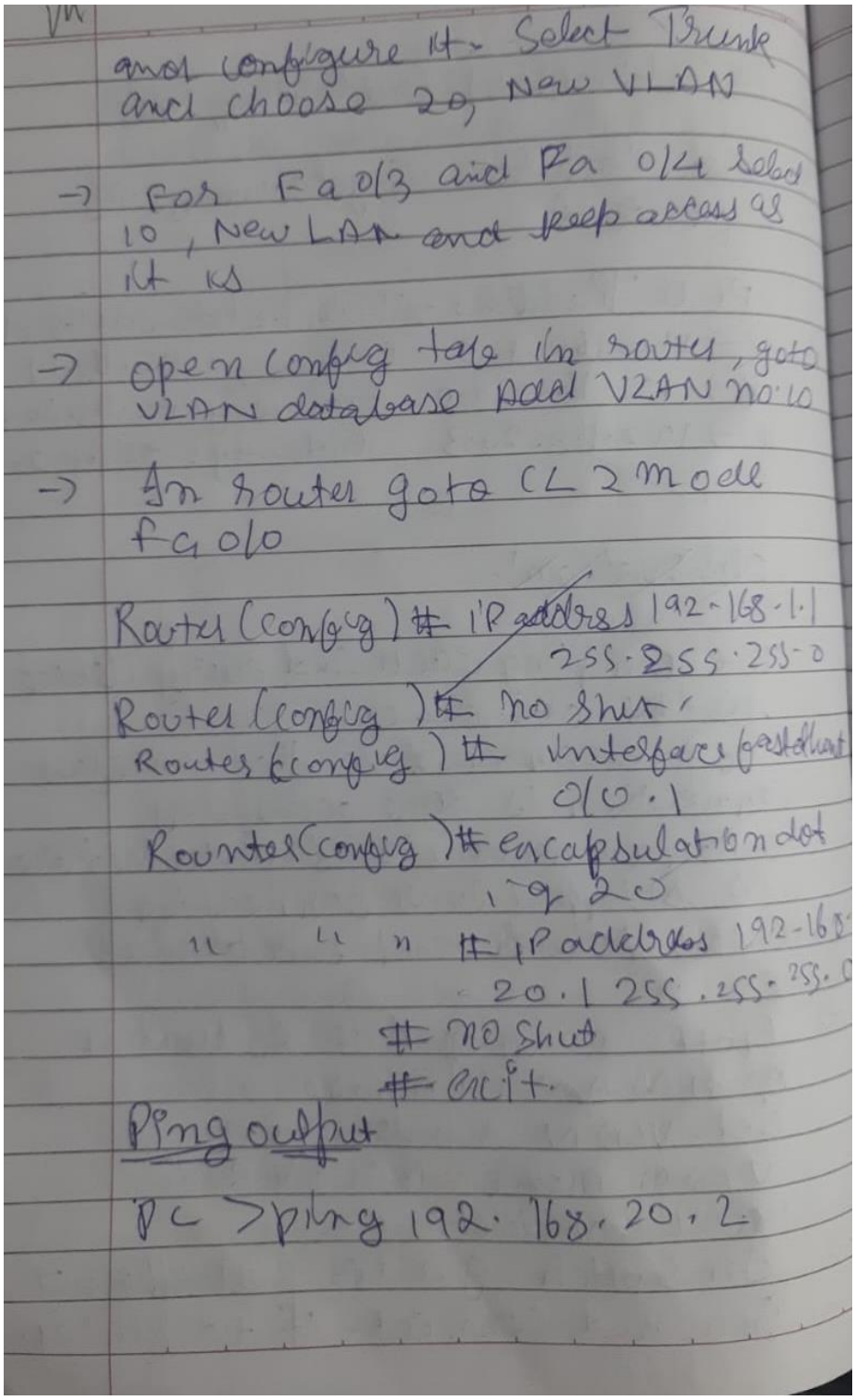
3. Set IP address & gateways as follows

PC0 IP address = 192.168.1.2 Gateway: 192.168.1.1
PC1 IP → 192.168.1.3 Gateway: 192.168.1.1
IP → 192.168.20.2 Gateway: 192.168.20.1
IP → 192.168.20.3 Gateway: 192.168.20.1

Observation:

We can ping each and every device to the other device
So we can observe that wireless connection is done accessfully
When connection is established there is stripped lines connecting access points and end devices.

→ Go to config tab of switch
open VLAN database
Set VLAN number = 20
VLAN name = New VLAN
click on add
→ In switch go to fast ethernet 5/0 and connect it to router.



and configure it - Select Trunk and choose 20, New VLAN

→ For Fa 0/3 and Fa 0/4 Select 10, New LAN and keep access as it is

→ Open config tab in Router, goto VLAN database Add VLAN no: 10

→ In Router goto CLI mode fa 0/0

Router(config)# IP address 192.168.1.1 255.255.255.0

Router(config)# no shut

Router(config)# interface fastEthernet 0/0.1

Router(config)# encapsulation dot 1q 20

" " " # IP address 192.168.20.1 255.255.255.0

no shut

exit

Ping output

PC > ping 192.168.20.2

Page No.
Date

Pinging 192.168.20.2 with 32 bytes of data

Request timed out.

Reply from 192.168.20.2 bytes=32 time=1ms TTL=

" " " " " " " " "

" " "

Ping Statistics for 192.168.20.2

Packets Sent = 4 Recieved = 3 lost = 1 (25% loss)

Approximate round trip in ms!

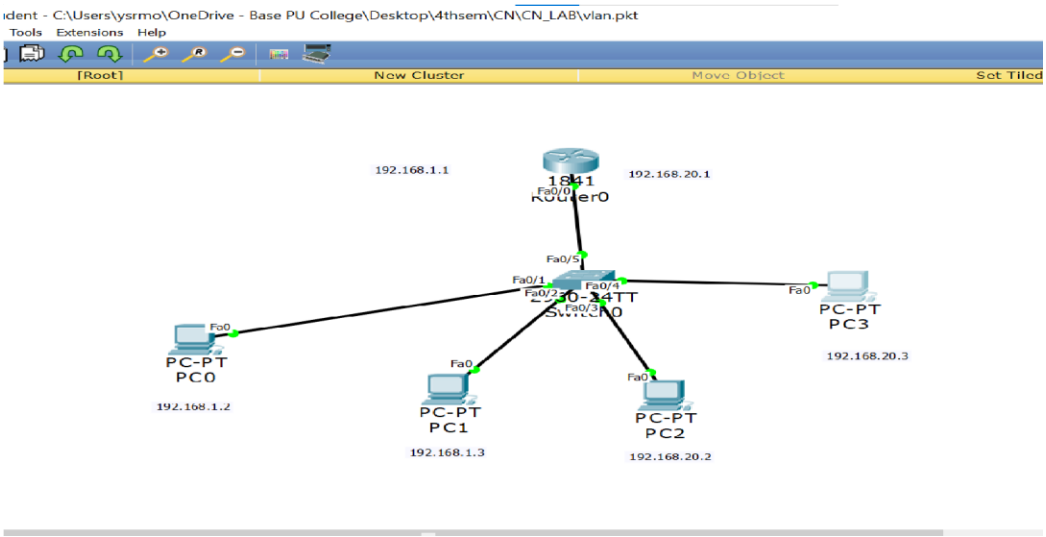
Min = 0 ms Max = 2ms Average = 0ms

Observation 19/8/2022

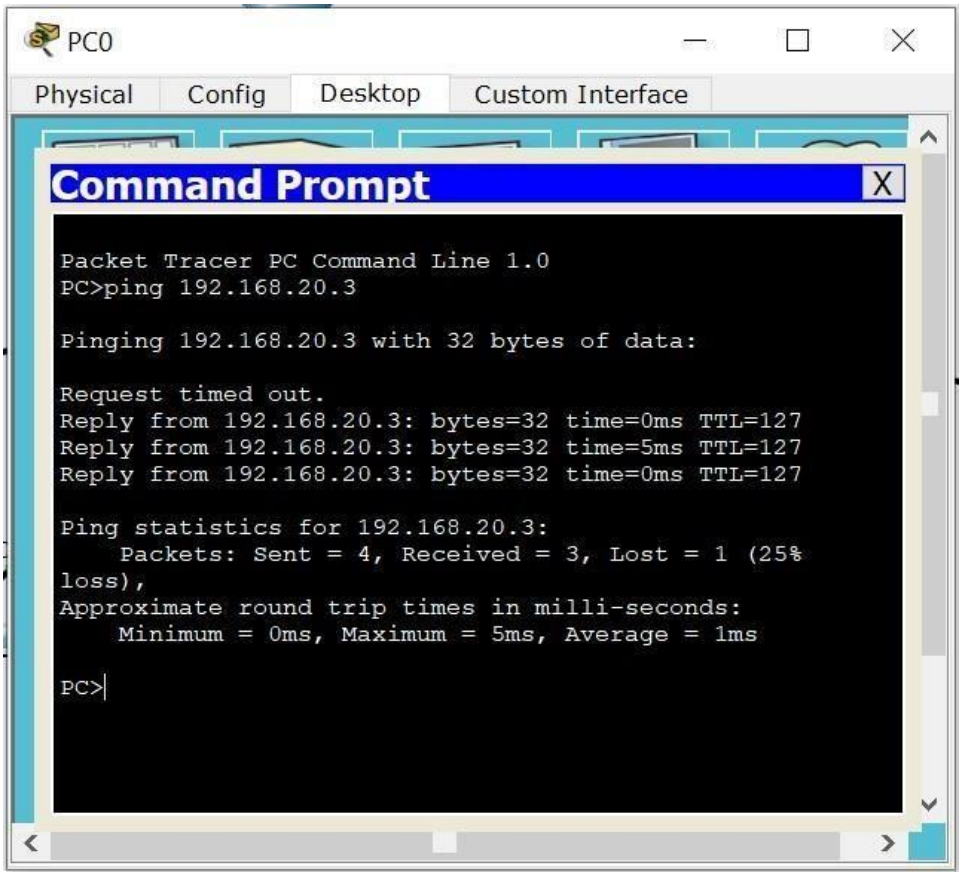
⇒ We can observe that after VLAN is configured we can successfully ping PC2 (192.168.20.2) from PC0 (192.168.1.2)

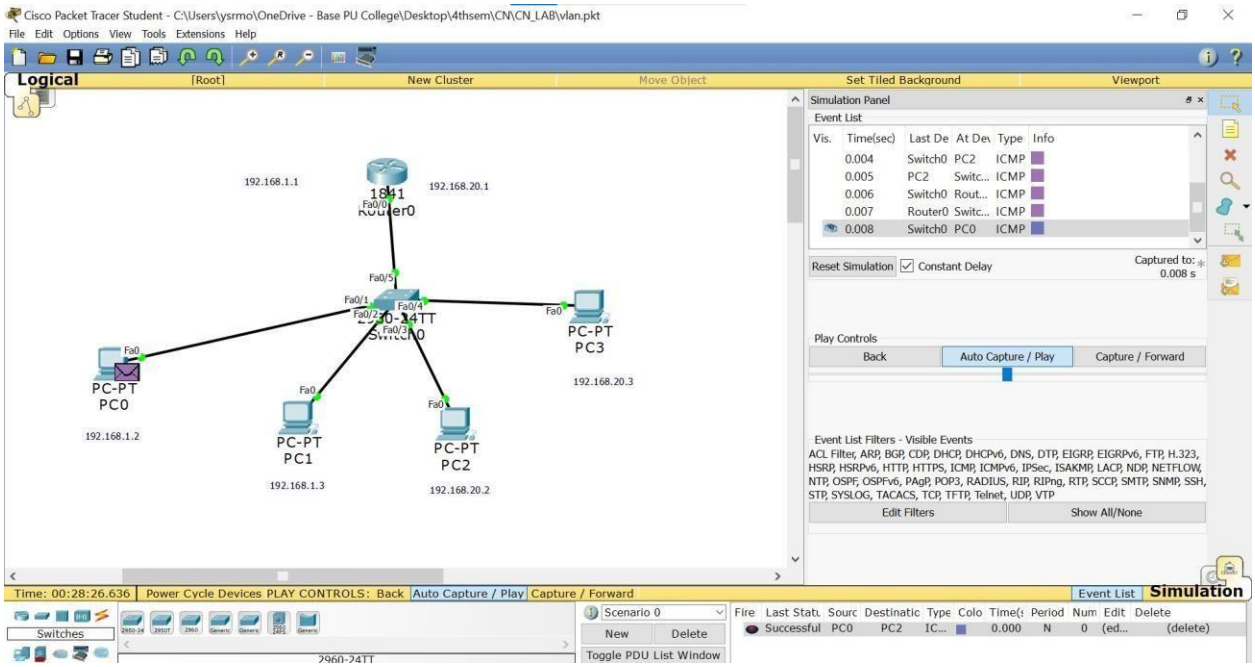
⇒ PC2 and PC3 are grouped together and communication among them is done via VLAN 192.168.20.1 is a sub interface 0/0.1 router.

TOPOLOGY:



OUTPUT:

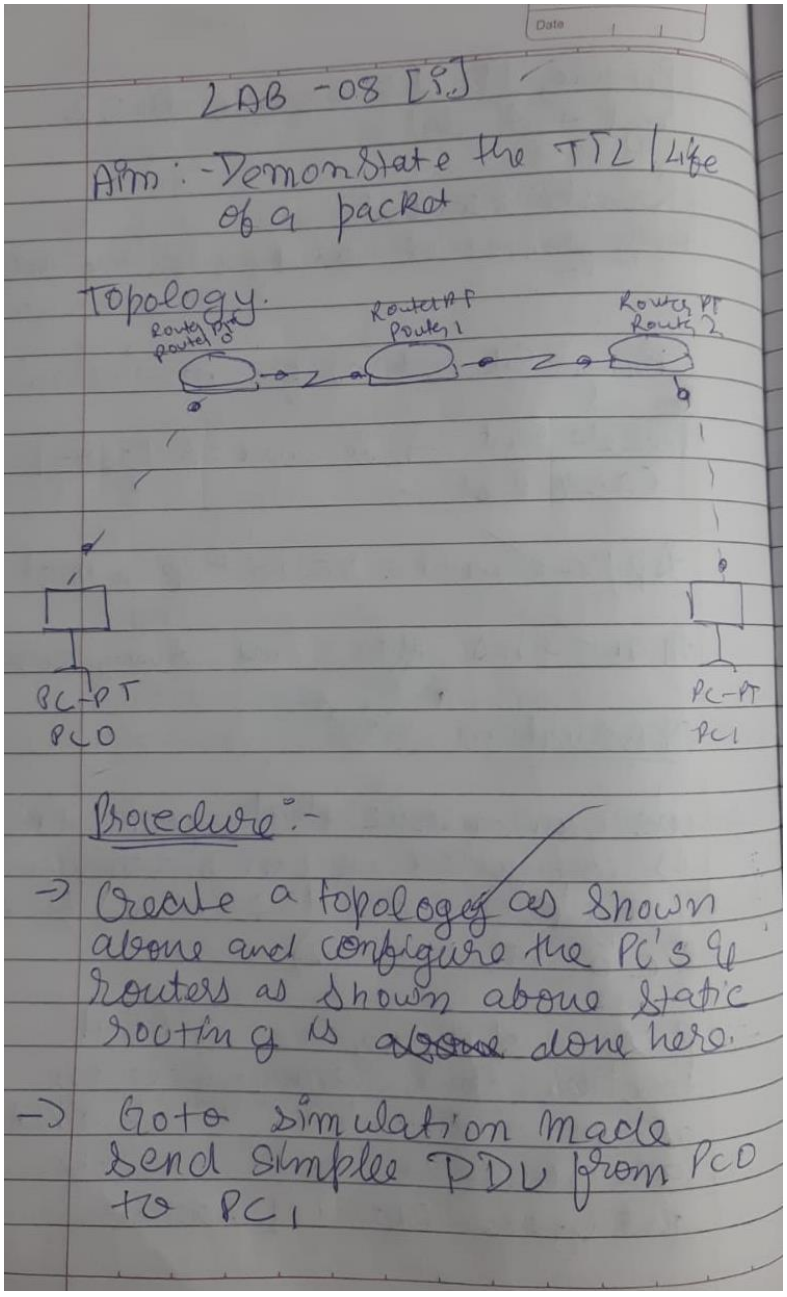


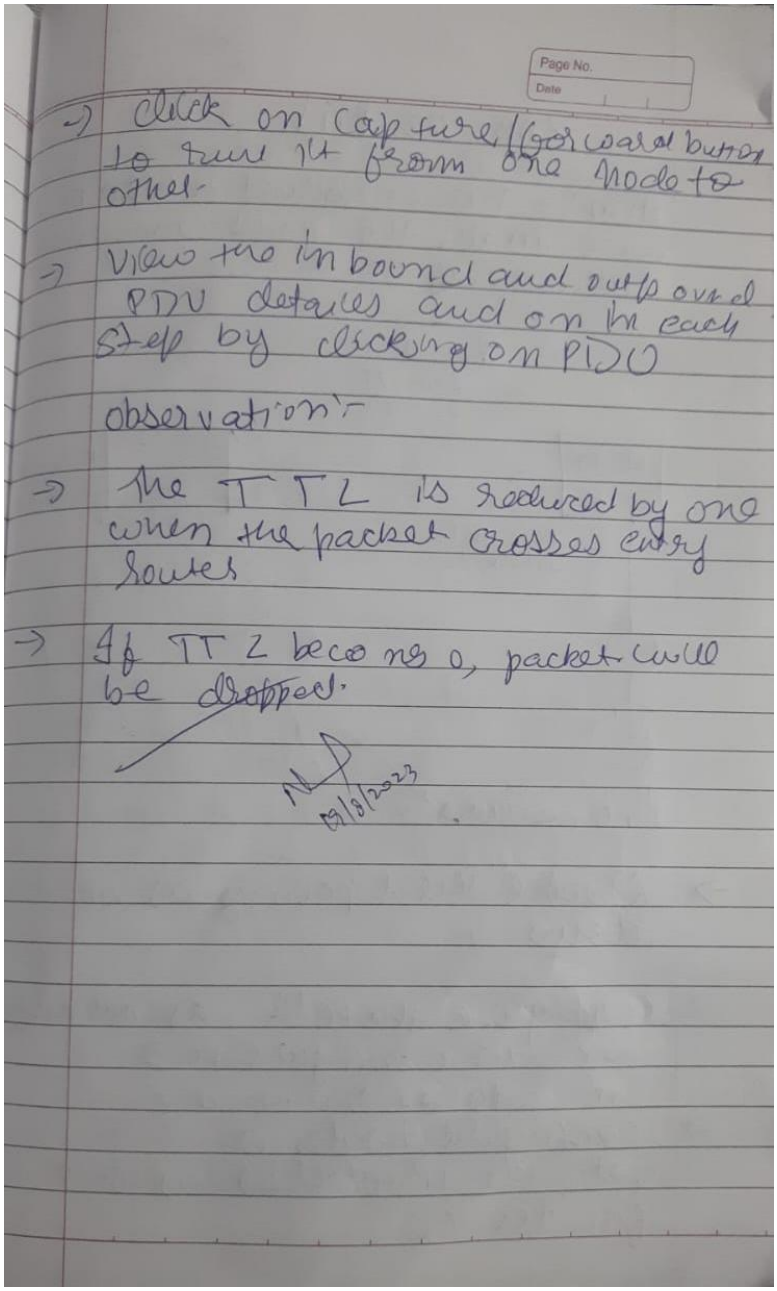

Cisco Packet Tracer Student - C:\Users\ysrmo\OneDrive - Base PU College\Desktop\4thsem\CN\CN_LAB\vlan.pkt
File Edit Options View Tools Extensions Help
Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
192.168.1.1
192.168.20.1
Router0
Fa0/5
Fa0/1
Fa0/2
Fa0/4
Fa0/3
Switch0
Fa0
PC-PT 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
Simulation Panel
Event List
Vis. Time(sec) Last De At De 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
Scenario 0
New
Delete
Toggle PDU List Window
Fire Last Stat Sourc Destinatic Type Colo Time(s Period Num Edit Delete
Successful PC0 PC2 IC... 0.000 N 0 (ed... (delete)

LAB 10

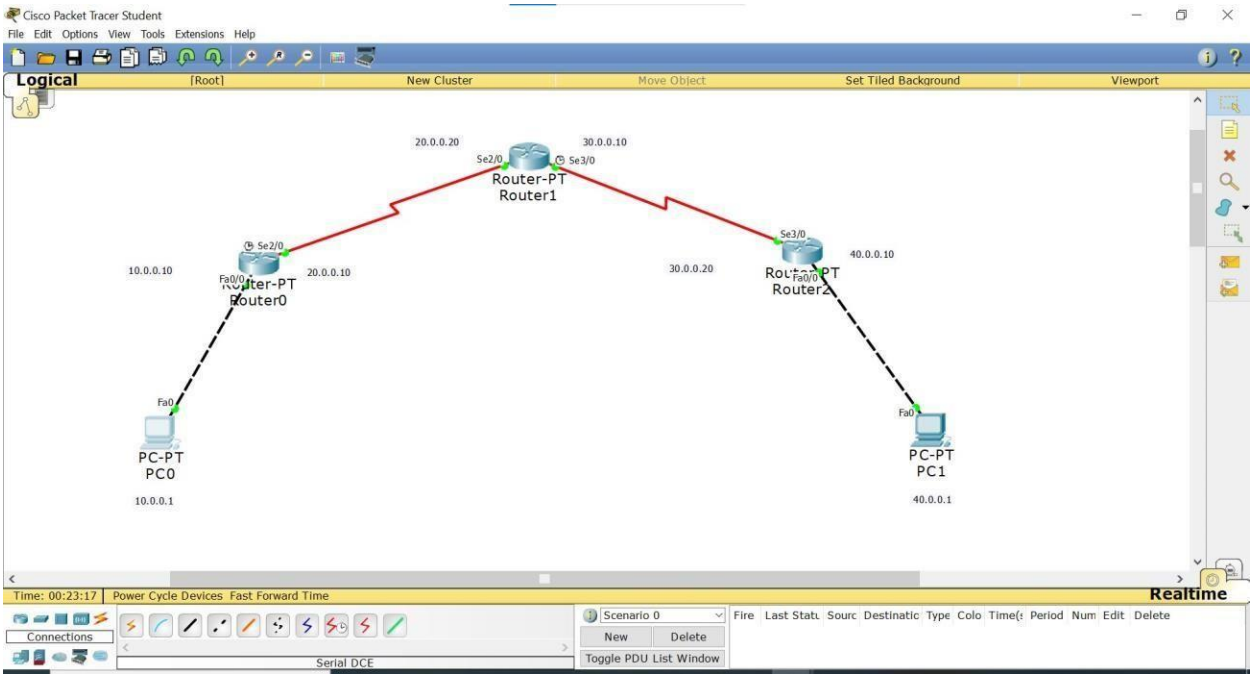
Demonstrate the TTL/ Life of a Packet.

OBSERVATION:

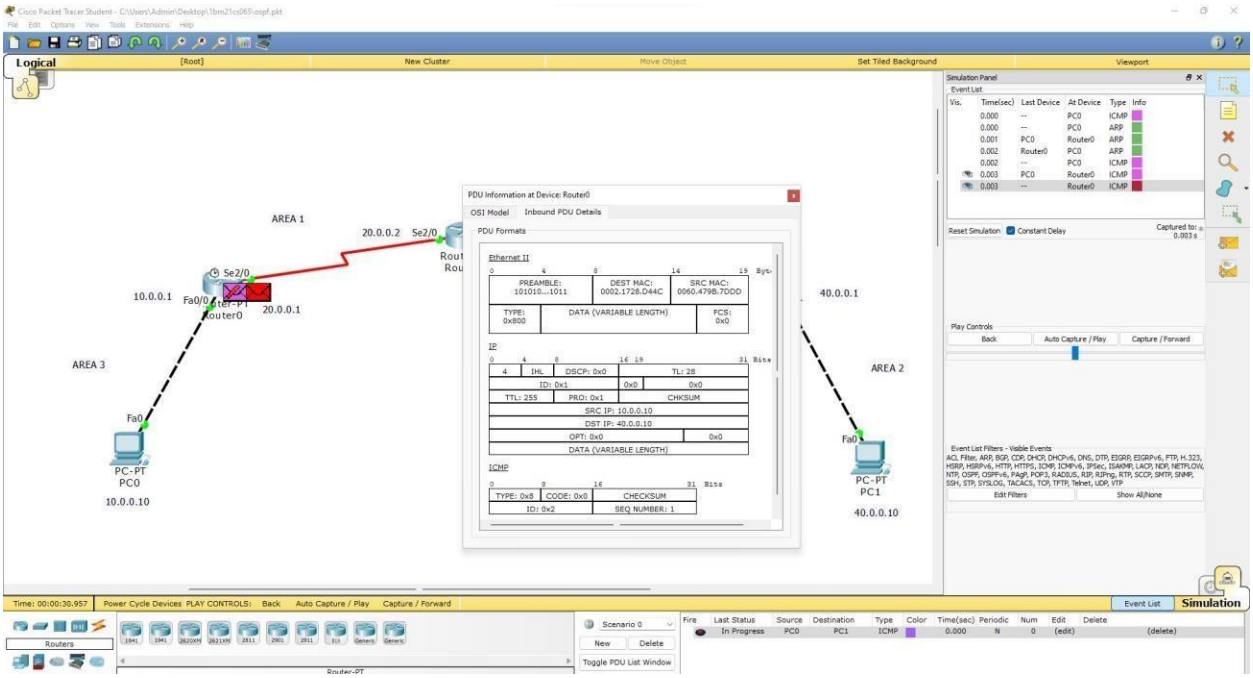


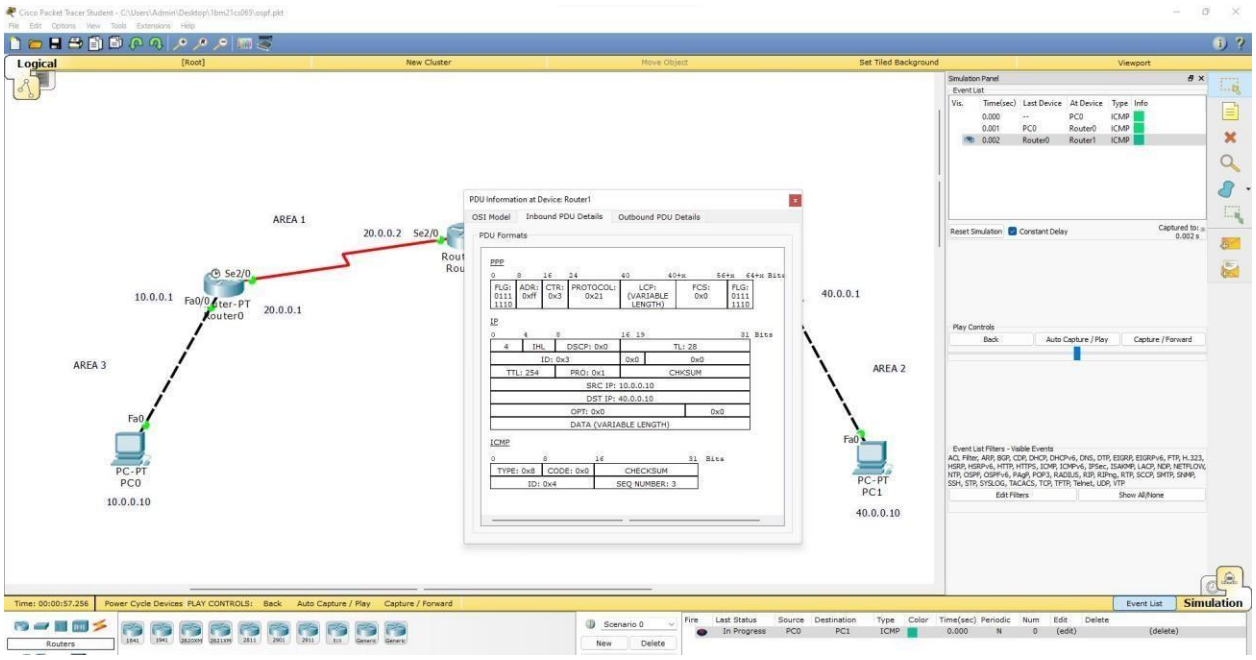

Page No.
Date
→ Click on capture/forward button to turn it from one node to other.
→ View the inbound and outbound PDU details and on in each step by clicking on PDU
observation:-
→ The TTL is reduced by one when the packet crosses entry routes
→ If TTL becomes 0, packet will be dropped.
09/8/2023

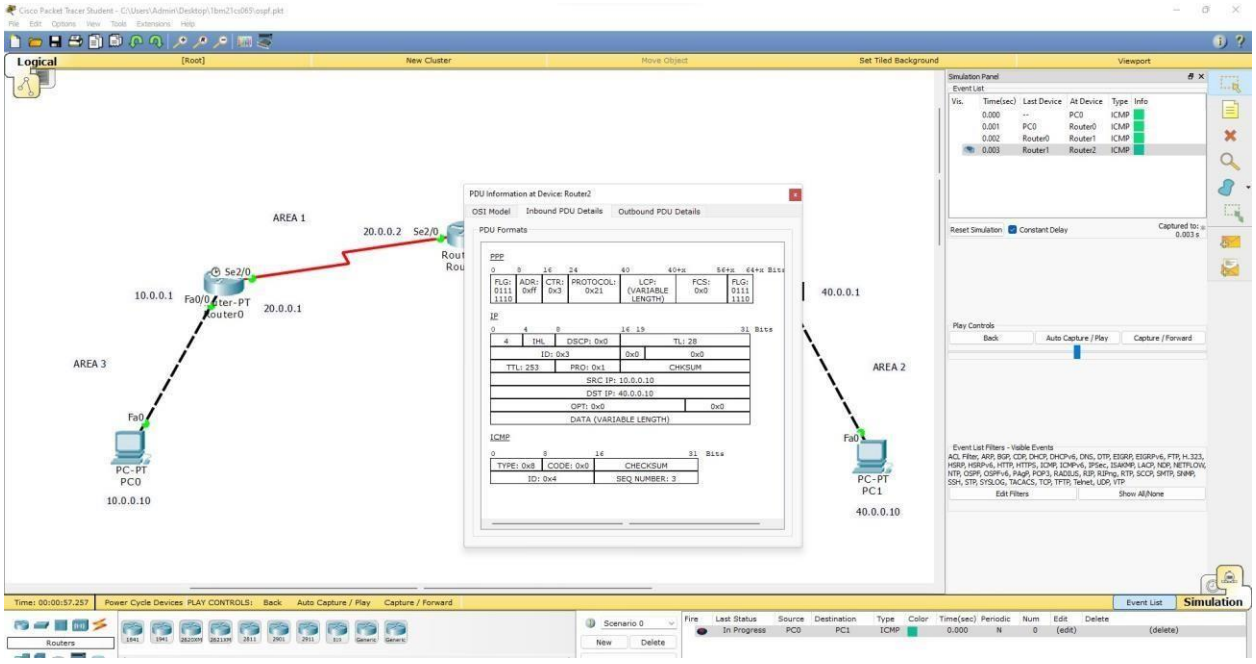
TOPOLOGY:

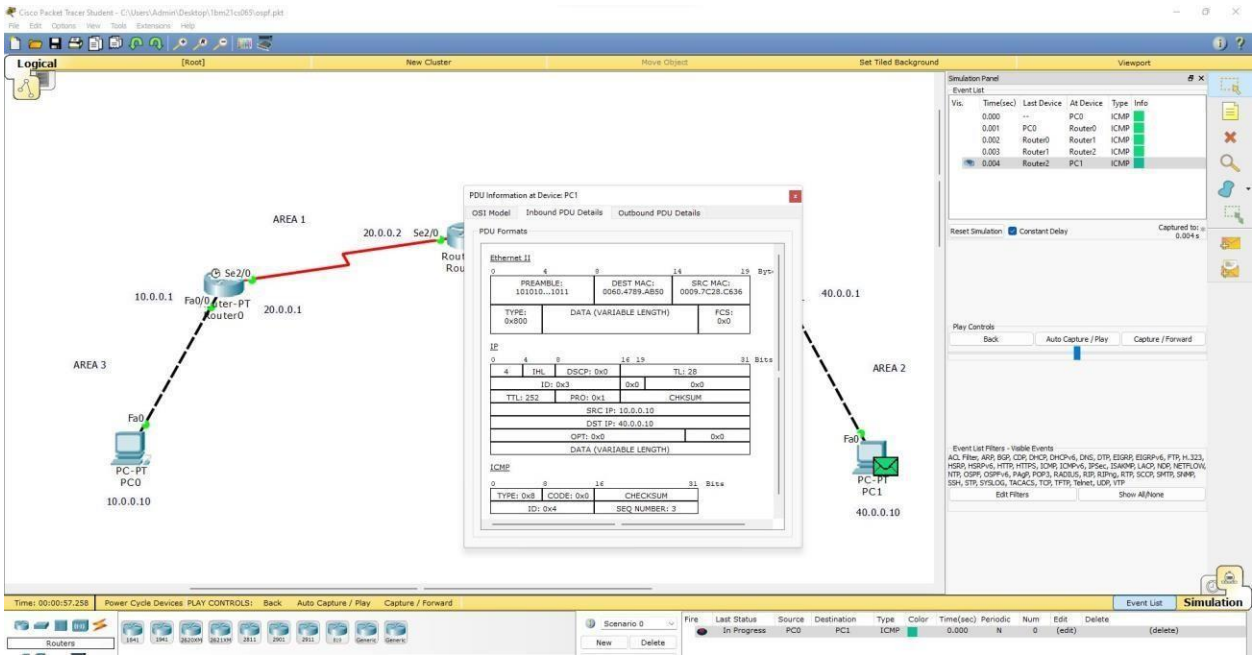


OUTPUT:




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: Router1
OSI Model
Inbound PDU Details
Outbound PDU Details
PDU Formats
PPP
IP
ICMP
TTL: 254
SRC IP: 10.0.0.10
DST IP: 40.0.0.10
TYPE: 0x8
CODE: 0x0
CHECKSUM
ID: 0x4
SEQ NUMBER: 3

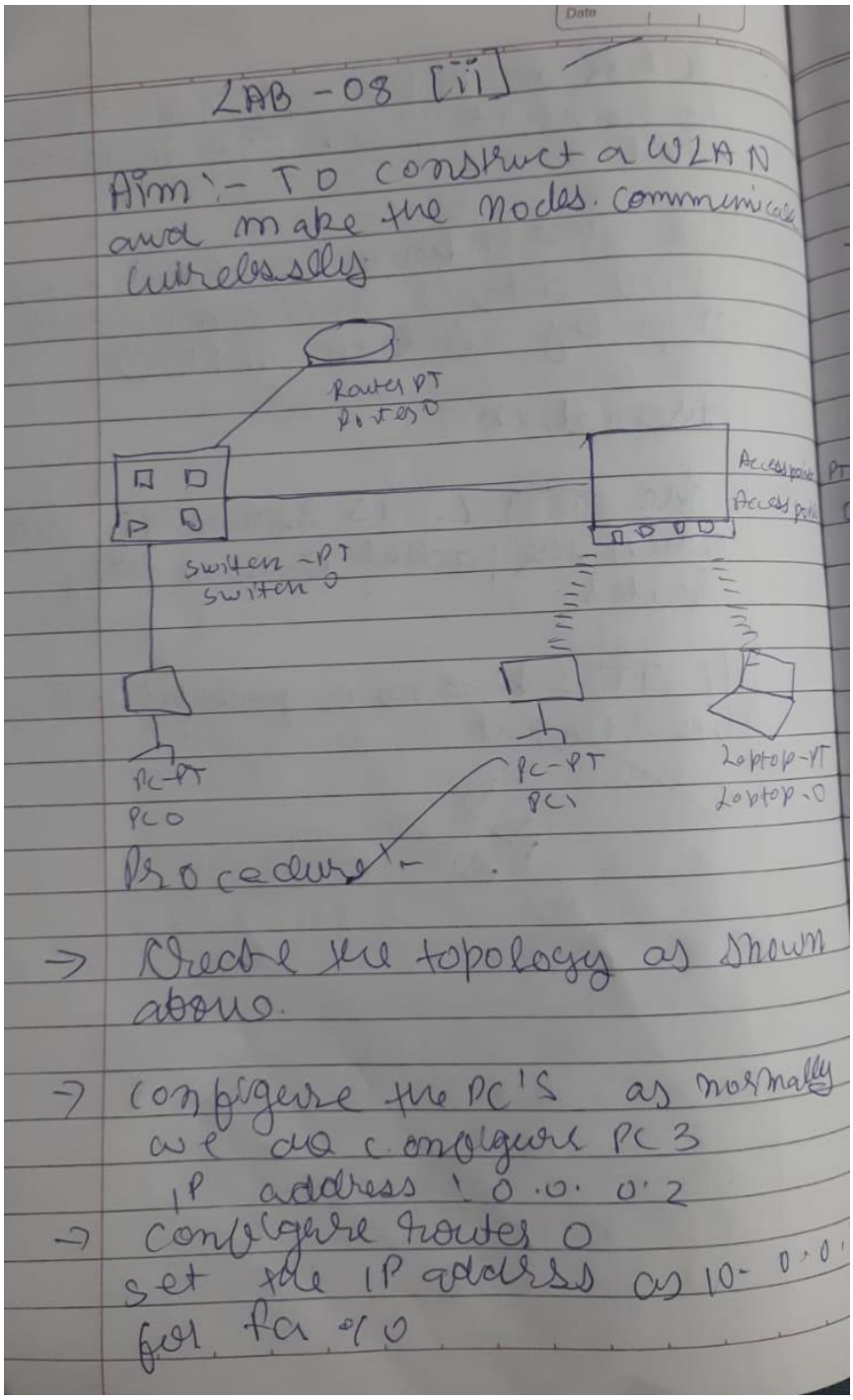

AREA 1
20.0.0.2
Se2/0
10.0.0.1
Fa0/0
20.0.0.1
AREA 3
AREA 2
40.0.0.1
Fa0
PC-PT
PC0
10.0.0.10
PC-PT
PC1
40.0.0.10
PDU Information at Device: Router2
OSI Model
Inbound PDU Details
Outbound PDU Details
PDU Formats
PPP
IP
ICMP
TTL: 253
SRC IP: 10.0.0.10
DST IP: 40.0.0.10
TYPE: 0x8
CODE: 0x0
CHECKSUM
ID: 0x4
SEQ NUMBER: 3

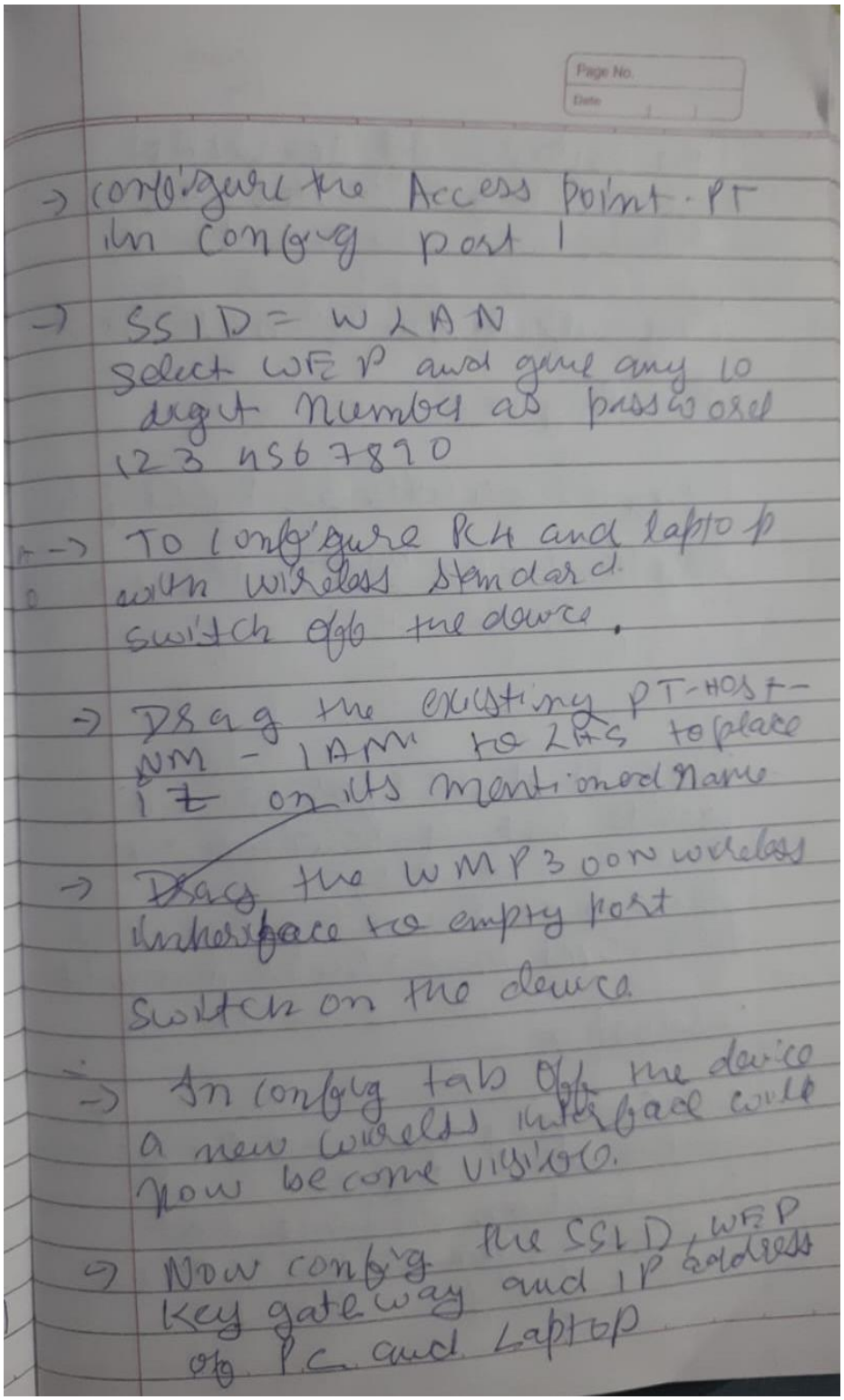

Logical
AREA 1
AREA 2
AREA 3
20.0.0.2
Se2/0
10.0.0.1
Fa0/0
20.0.0.1
40.0.0.1
Fa0
PC-PT
PC0
10.0.0.10
PC-PT
PC1
40.0.0.10
PDU Information at Device: PC1
OSI Model
Inbound PDU Details
Outbound PDU Details
PDU Formats
Ethernet II
PREAMBLE: 101010...1011
DEST MAC: 0060.4789.AB50
SRC MAC: 0009.7C28.C636
TYPE: 0x800
DATA (VARIABLE LENGTH)
FCS: 0x0
IP
SRC IP: 10.0.0.10
DST IP: 40.0.0.10
ICMP
TYPE: 0x8
CODE: 0x0
CHECKSUM
SEQ NUMBER: 3
Simulation Panel
Event List
Play Controls
Back
Auto Capture / Play
Capture / Forward
Time: 00:00:57.258
Power Cycle Devices
PLAY CONTROLS:
Routers
Scenario 0
New
Delete
Simulation

LAB 11

To construct a WLAN and make the nodes communicate wirelessly

OBSERVATION:





→ configure the Access point PT in config port 1

→ SSID = WLAN
Select WEP and give any 10 digit Number as password 1234567890

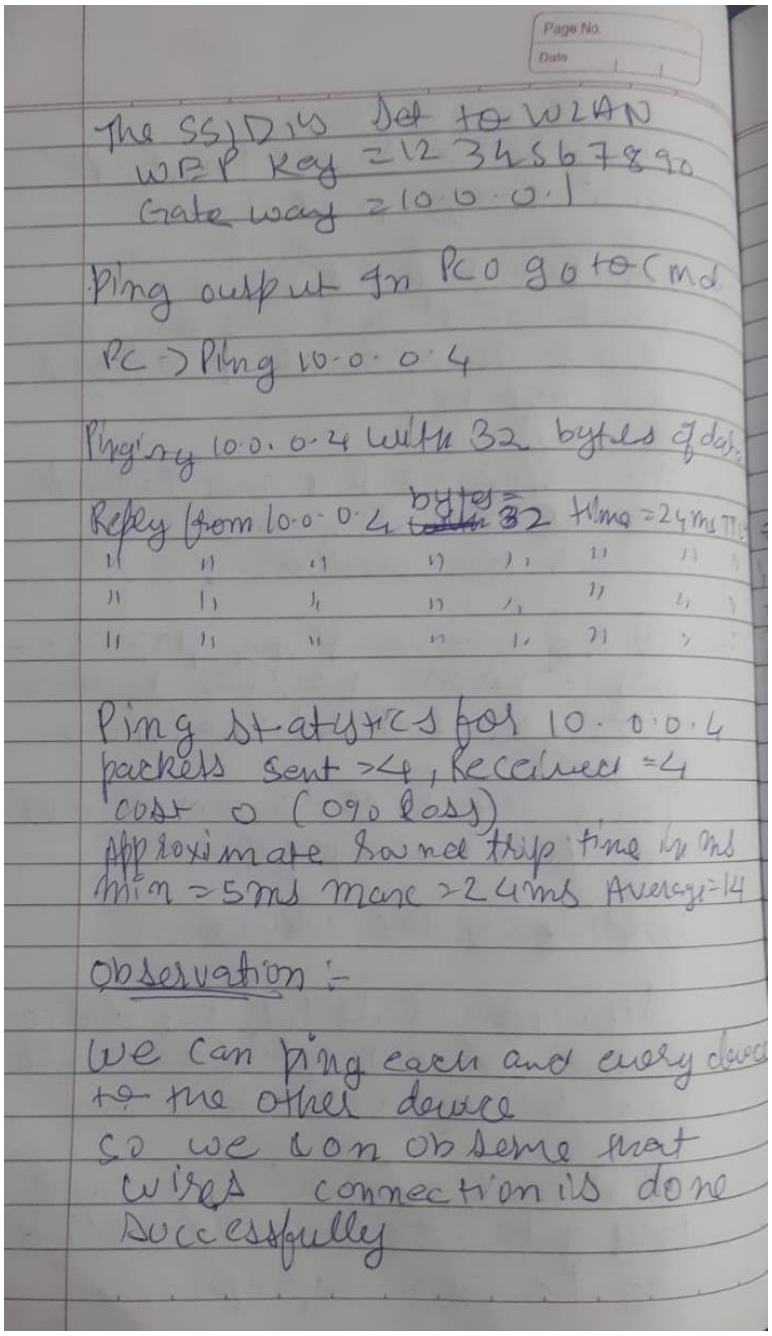
→ To configure PC4 and laptop with wireless standard switch off the device.

→ Drag the existing PT-HOST-NM-1AM to LHS to place it on its mentioned name

→ Drag the WMP300N wireless interface to empty port
Switch on the device.

→ In config tab off the device a new wireless interface would now become visible.

→ Now config the SSID, WEP key gateway and IP address of PC and Laptop



The SSID is set to WLAN
WEP Key = 12345678 90
Gate way = 10.0.0.1

Ping output for PC0 go to Cmd

PC -> Ping 10.0.0.4

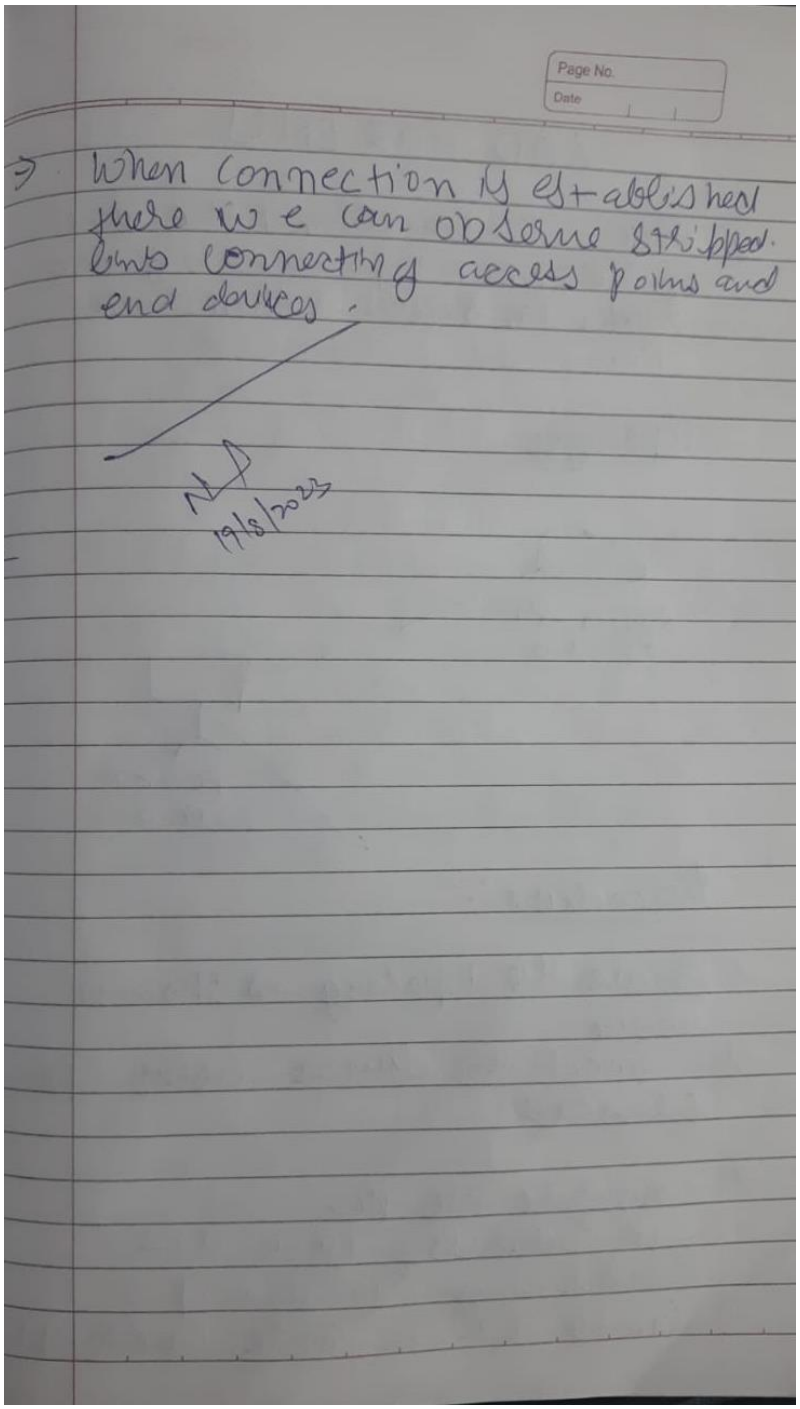
Pinging 10.0.0.4 with 32 bytes of data:

Reply from 10.0.0.4 ~~bytes~~ 32 time = 24ms TTL =

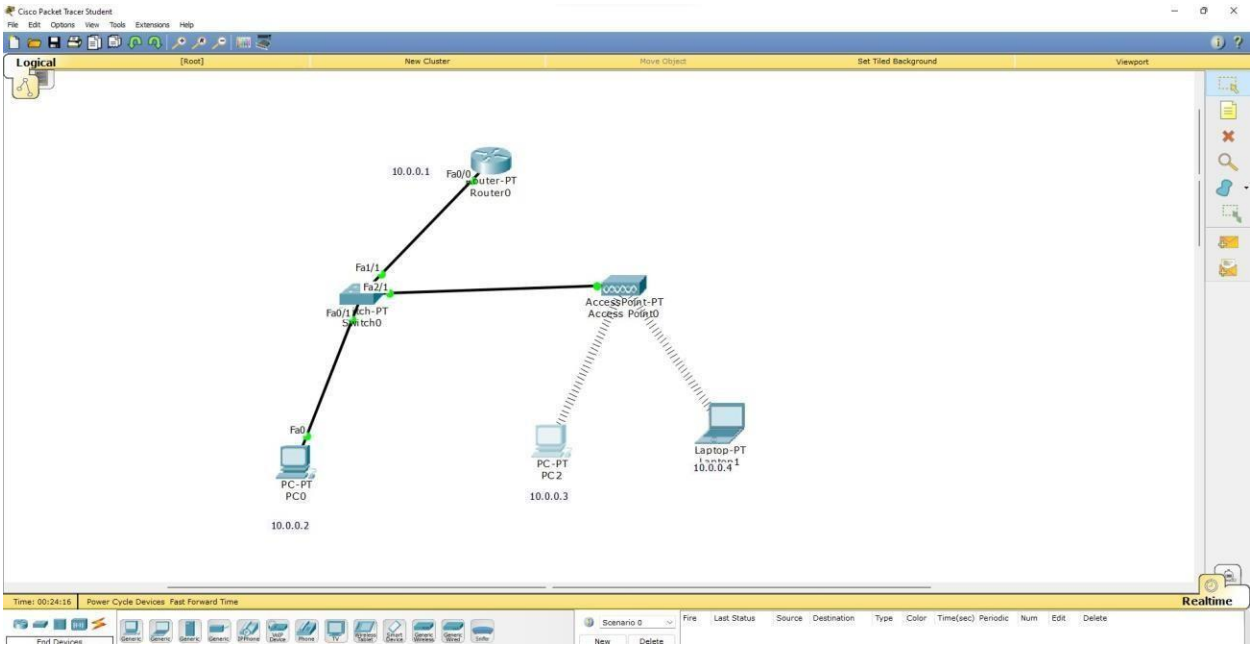
Ping statistics for 10.0.0.4
packets sent = 4, Received = 4
cost 0 (0% loss)
Approximate round trip time in ms
min = 5ms max = 24ms Average = 14

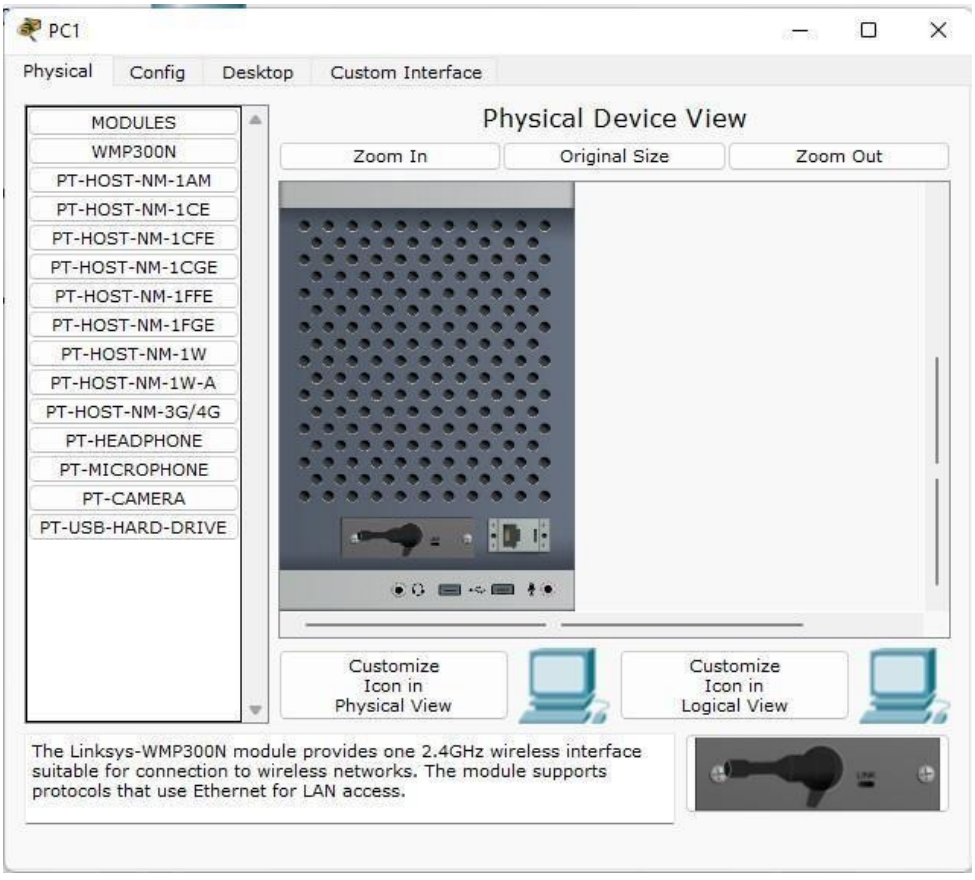
Observation :-

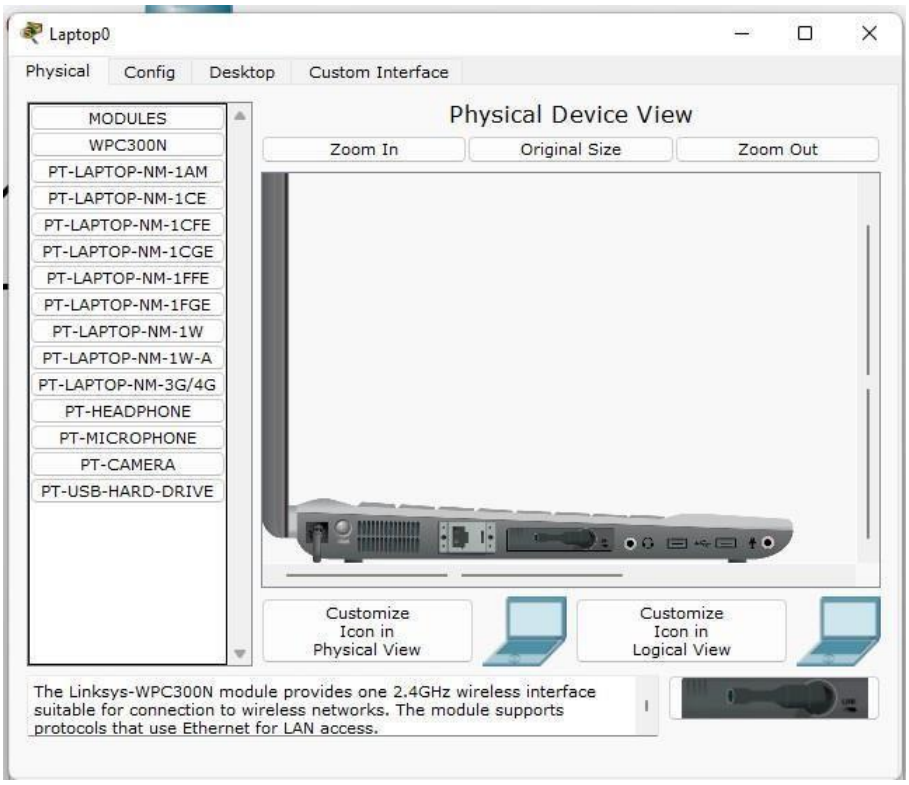
We can ping each and every device to the other device
So we can observe that wired connection is done successfully


Page No.
Date
⇒ When connection is established there we can observe stripped ends connecting access points and end devices.
19/8/2023

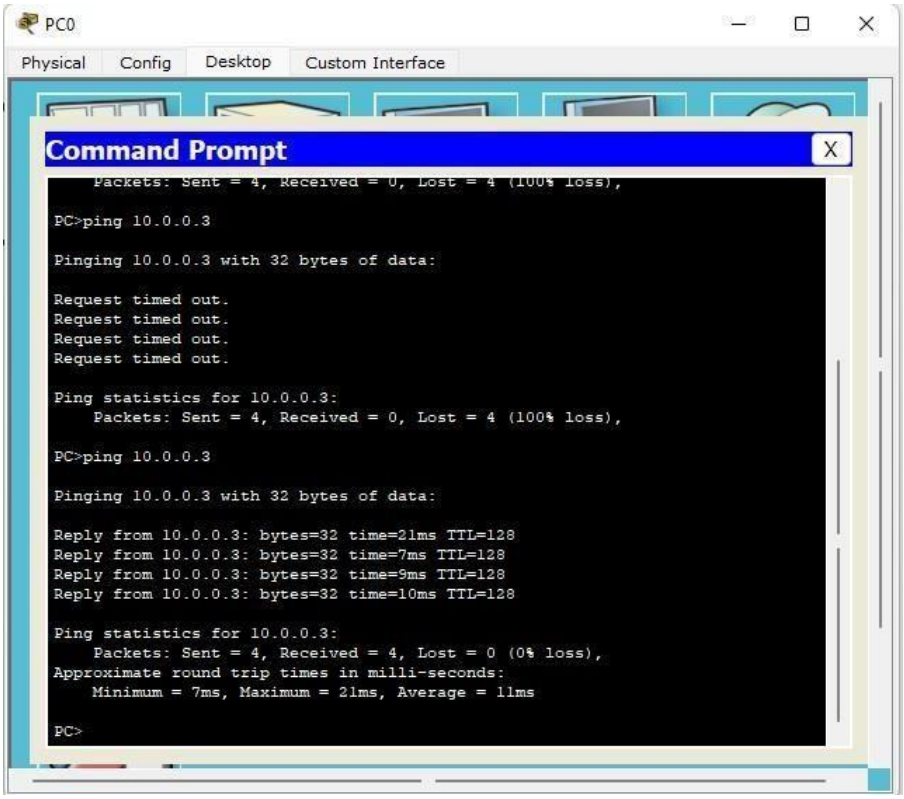
TOPOLOGY:


Logical
10.0.0.1
Router-PT
Router0
Fa1/1
Fa2/1
Switch-PT
Switch0
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.

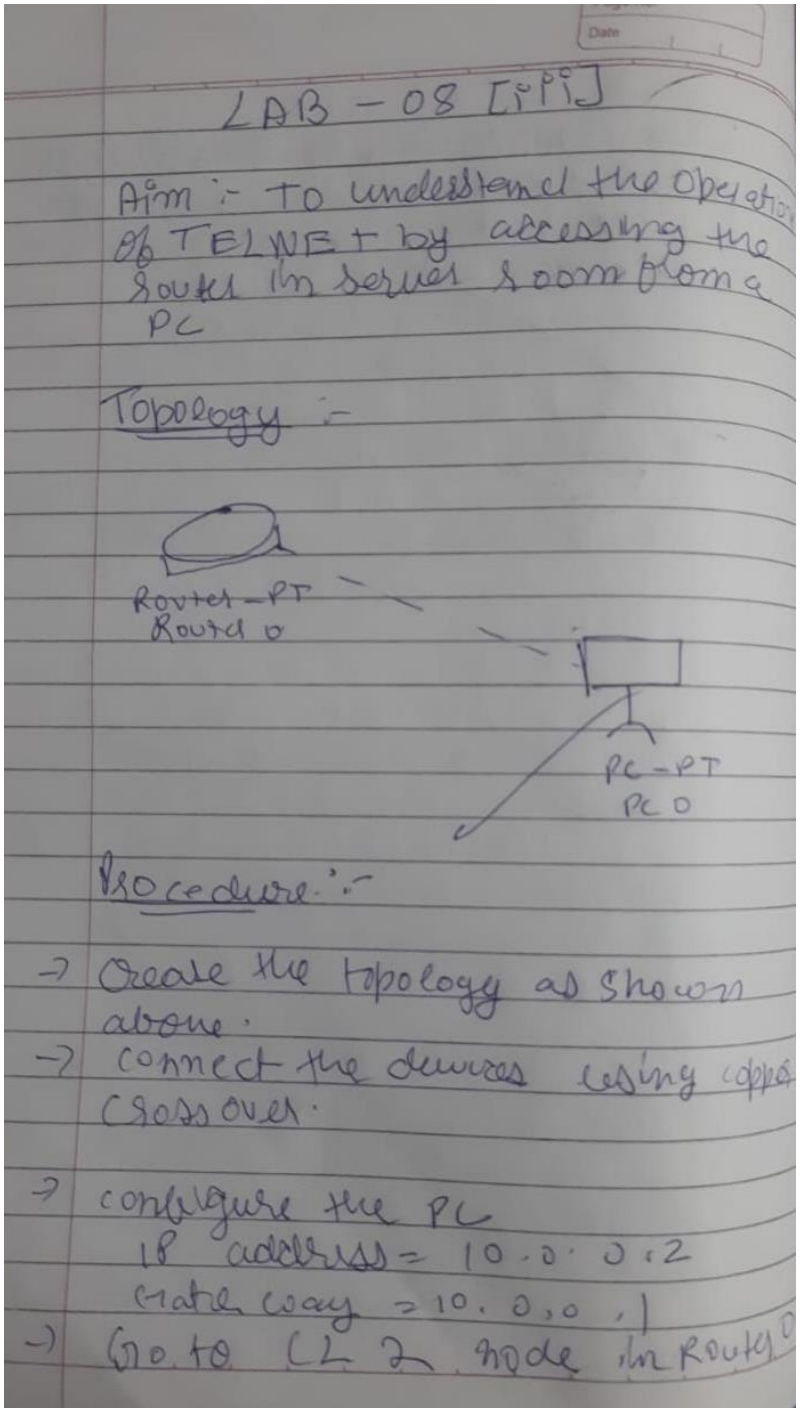
OUTPUT:

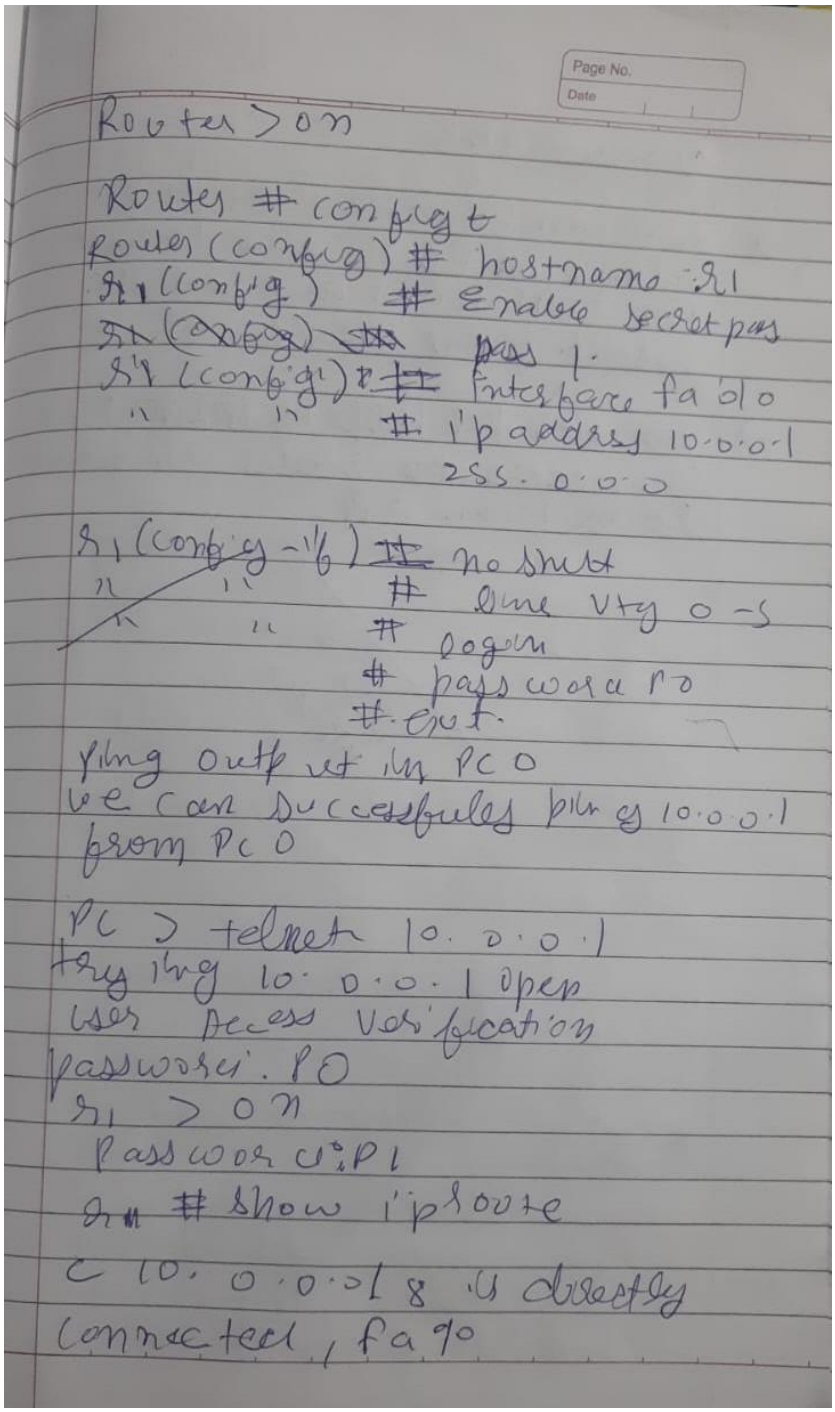

PC0
Physical
Config
Desktop
Custom Interface
Command Prompt
X
Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
PC>ping 10.0.0.3
Pinging 10.0.0.3 with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.
Ping statistics for 10.0.0.3:
Packets: Sent = 4, Received = 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>

WEEK12

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

OBSERVATION:



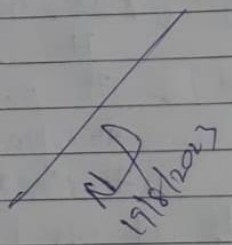

Page No.
Date
Router > en
Router # config t
Router (config) # hostname R1
R1 (config) # Enable secret pass
R1 (config) # int fa 0/0
" " # ip address 10.0.0.1
255.0.0.0
R1 (config-if) # no shut
" " # line vty 0 5
" " # login
password pp
exit
ping output in PC 0
we can successfully ping 10.0.0.1
from PC 0
PC > telnet 10.0.0.1
trying 10.0.0.1 open
User Access Verification
password: PP
R1 > en
Password: PP
R1 # show ip route
C 10.0.0.0/8 is directly
connected, fa 0/0

Date

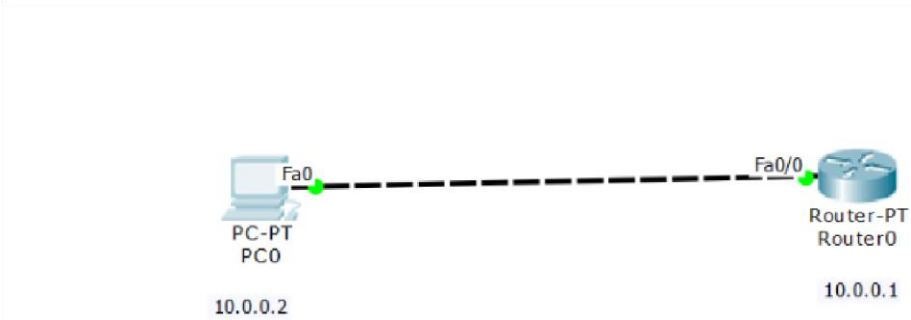
Observations:-

→ We can observe that the admin in PC is able to run in router CLI and see the result from the PC

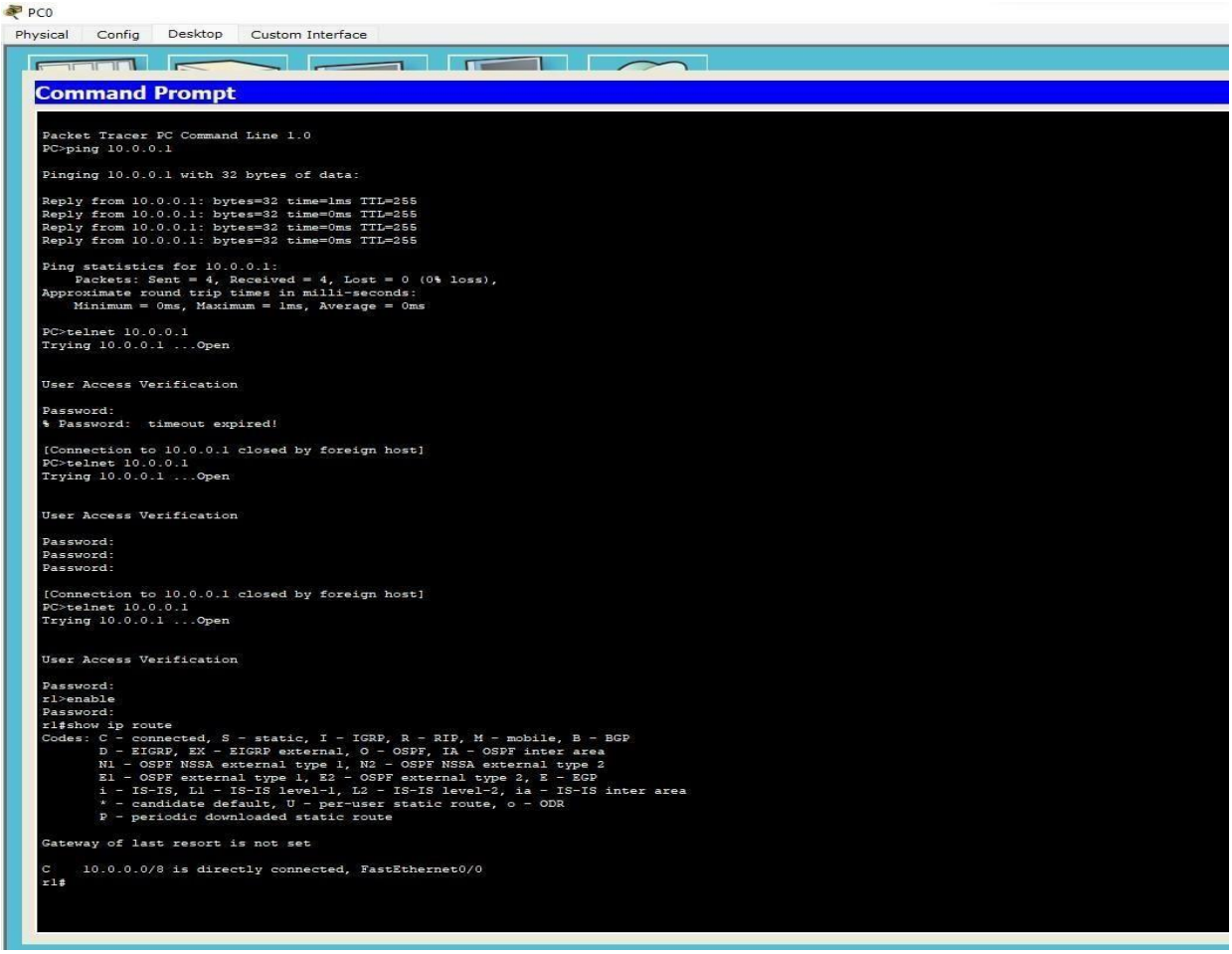
→ So with the help of TELNET we can access the router in Server room from a PC



TOPOLOGY:



OUTPUT:



LAB13

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

CODE:

```
#include<stdio.h>
int arr[17];

void xor(int x[], int y[])
{
    int k=0;
    for(int i=1;i<16;i++)
    { if(x[i]==y[i])
            arr[k++]=0;
        else
            arr[i]=1;
    }
}

void main()
{ int dd[17],div[33],ze[17],i,k;

    printf("Enter the dataword \n");
    for(i=0;i<17;i++)
    scanf("%d",&div[i]);

    for(i=i;i<33;i++)
        div[i]=0;

    for(i=0;i<17;i++) ze[i]=0;
    printf("Enter dividend \n");
    for(i=0;i<17;i++)
        scanf("%d",&dd[i]);
```

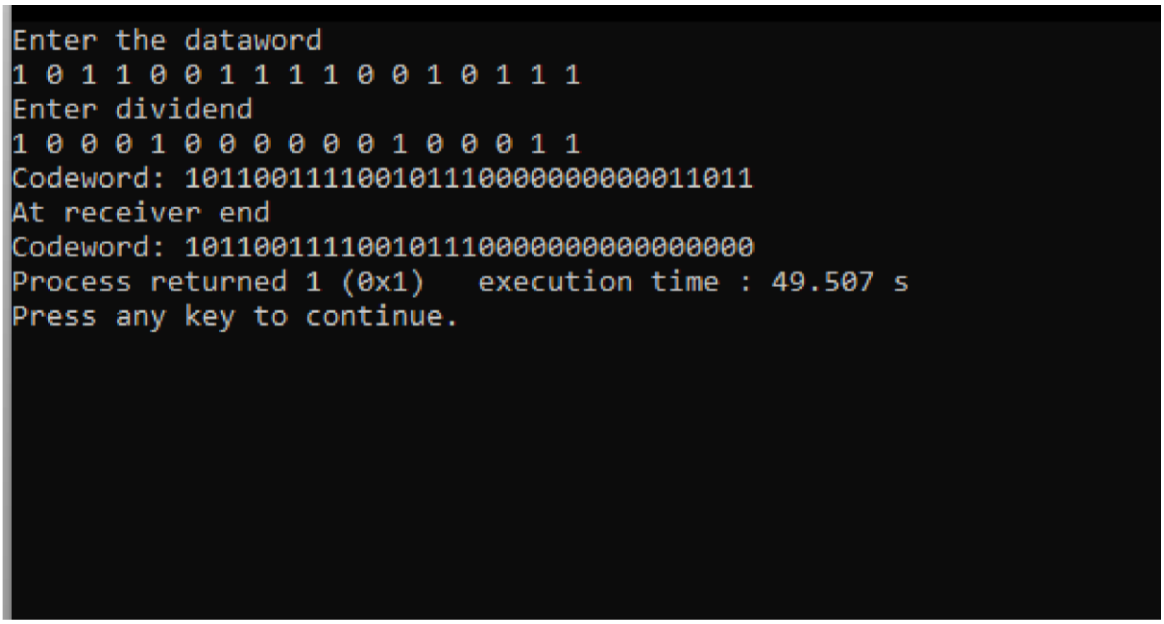
```
i=0;
k=0;
    for(i=i;i<17;i++)
```

```
        arr[k++]=div[i];
while(i<33)
{ if(arr[0]==0)
        xor(arr,ze);
    else
        xor(arr,dd);

    arr[16]=div[i++];

}
k=0;
for(i=17;i<33;i++)
div[i]=arr[k++];
printf("Codeword: ");
    for(i=0;i<33;i++)
    printf("%d",div[i]);

for(i=0;i<17;i++)
    arr[i]=0; printf("\nAt

receiver end \n");

 k=0;
    for(i=i;i<17;i++)
        arr[k++]=div[i];
while(i<33)
{ if(arr[0]==0)
    xor(arr,ze);
    else
        xor(arr,dd);

    arr[16]=div[i++];

}
k=0;
```

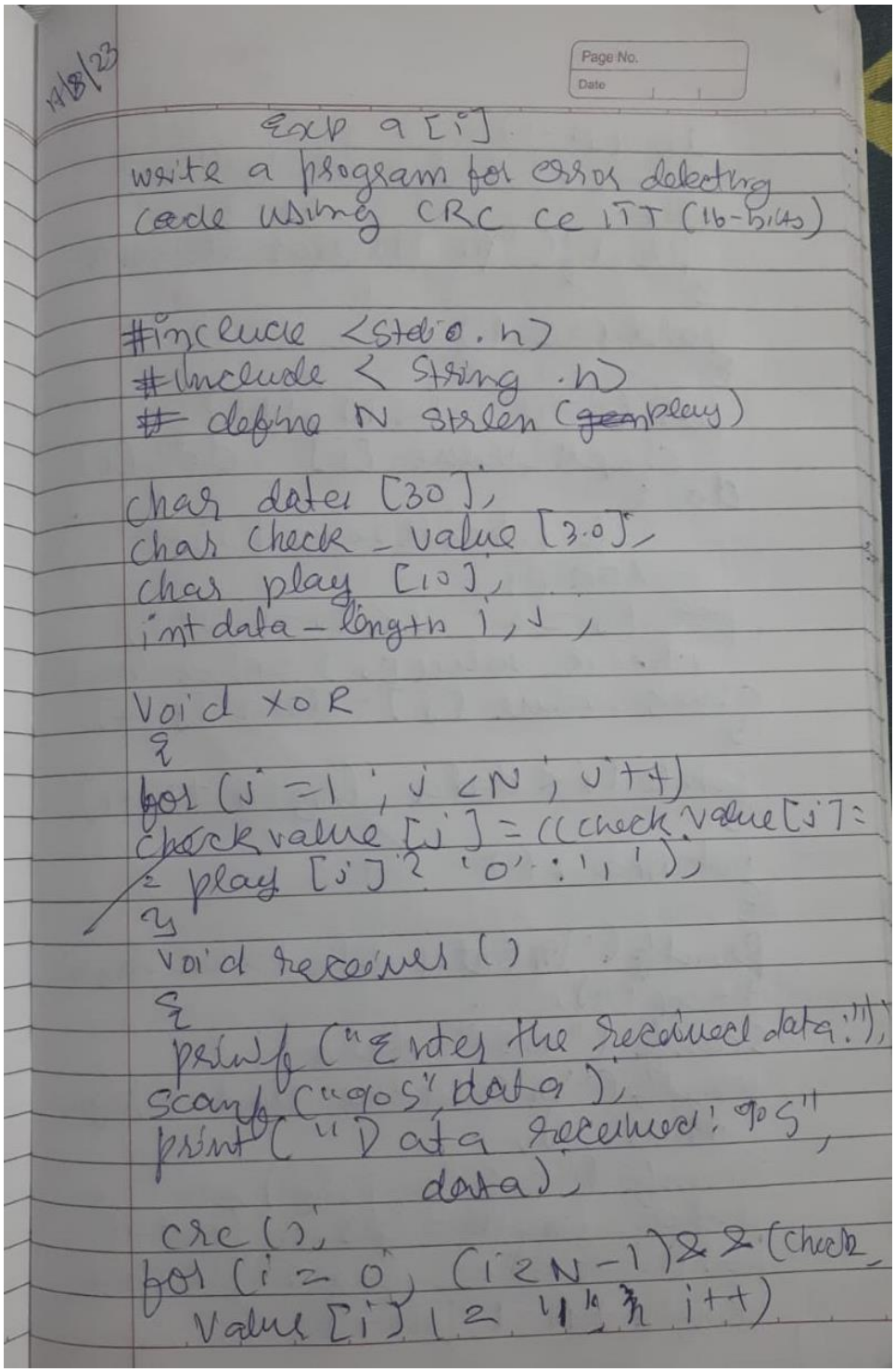
```
for(i=17;i<33;i++)
div[i]=arr[k++];
```

```
    printf("Codeword: ");
        for(i=0;i<33;i++)
            printf("%d",div[i]);
}
```

OUTPUT:



OBSERVATION:



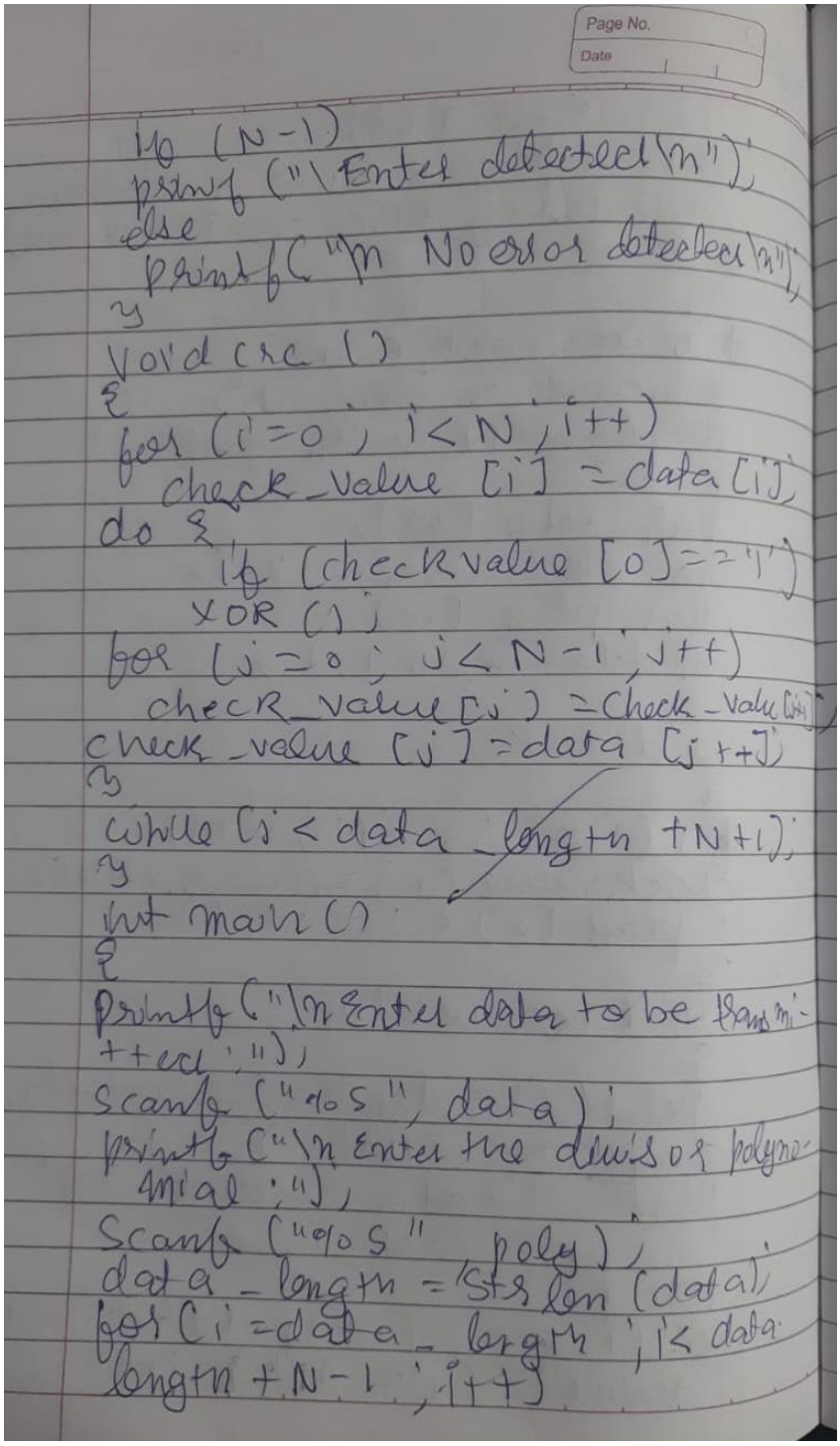
17/8/23

Page No.
Date

Exp 9 [i]

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

```
#include <stdio.h>
#include <string.h>
# define N strlen (~~gen~~poly)

char data1 [30];
char check_value [30];
char poly [10];
int data_length i, j;

Void XOR
{
for (j = 1; j < N; j++)
check value [j] = ((check value [j] ==
= poly [j])? '0' : '1');
}
Void receiver ()
{
printf ("Enter the received data:");
scanf ("%s", data);
printf ("Data received: %s",
        data);

crc ();
for (i = 0; (i < N-1) && (check_
  value [i] != '1'); i++)
```



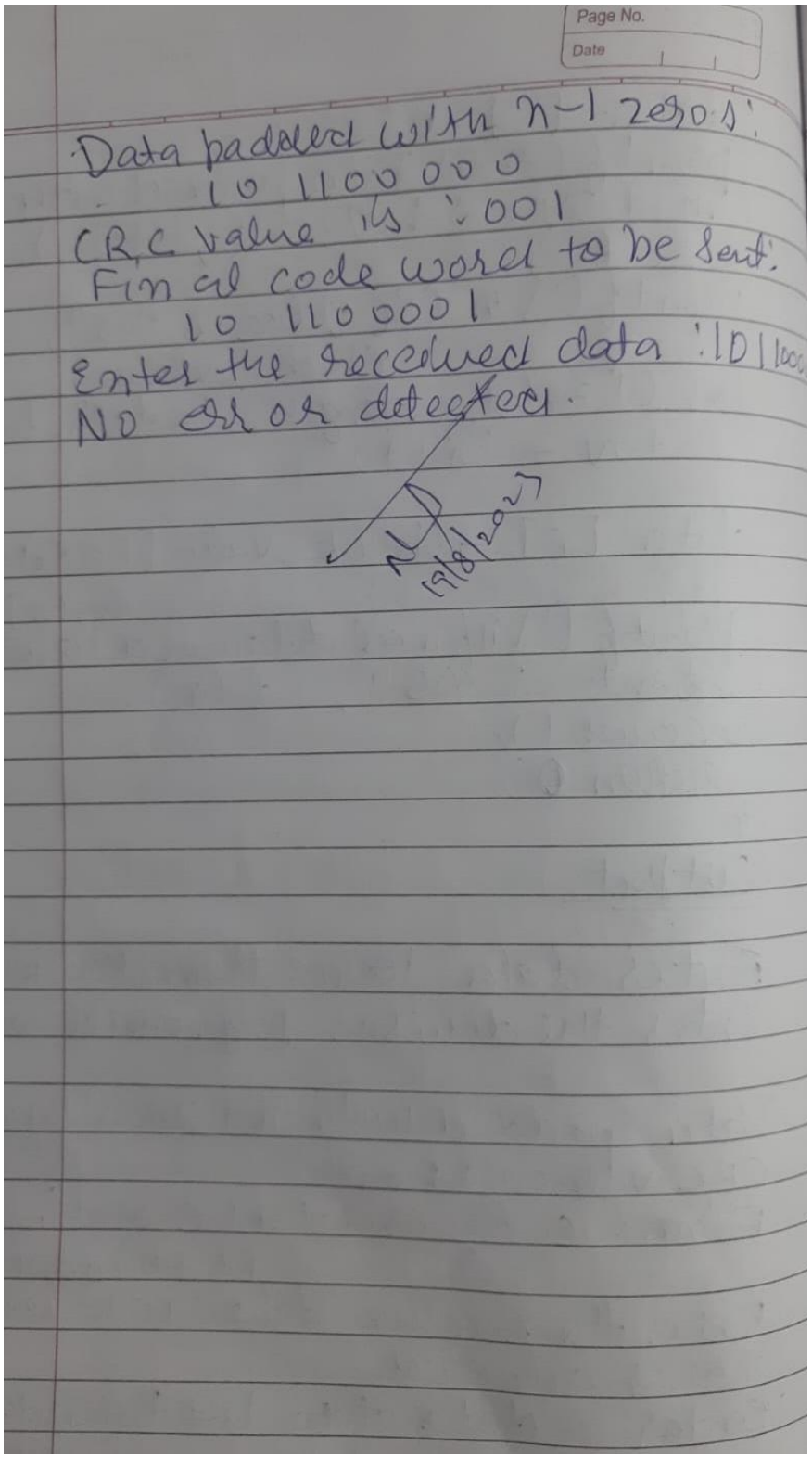
```
if (N-1)
printf ("\Enter detected\n");
else
printf("\n No error detected\n");
}
void crc ()
{
for (i=0 ; i<N ; i++)
  check_value [i] = data [i];
do {
    if (checkvalue [0]=='1')
    XOR ();
for (j=0 ; j<N-1 ; j++)
  check_value [j] = check_value [j+1];
check_value [j] = data [j+N];
}
while (i < data_length +N+1);
}
int main ()
{
printf ("\n Enter data to be transmi-
++ted : ");
scanf ("%s", data);
printf ("\n Enter the divisor polyno-
mial : ");
scanf ("%s", poly);
data_length = strlen (data);
for (i=data_length ; i< data
length +N-1 ; i++)
```

```
data [i] = 0;
printf("\n Data padded with
n-1 zeros : %s", data);
crc();
printf("\n CRC value is %s",
                    check_value);
for (i = data_length;
    i < data_length
    + N - 1; i++)

data [i] - check_value [i - data
                        _length];
printf ["\n final data word to be
    sent : %s", data);
receiver();
return 0;
}
```

Output

Enter data to be transmitted: 101010
Enter the divisor polynomial: 1011

Data padded with n-1 zeros: 101010
000
CRC value is: 001
Final codeword to be sent:
101010001
Enter the received data: 1000 1002
Error detected
Enter data to be transmitted:
Enter the divisor polynomial: 1001 1100


Page No.
Date
Data padded with n-1 zeros:
10 1100000
CRC value is : 001
Final code word to be sent:
10 1100001
Enter the received data : 10 1100
NO error detected.
19/8/2022

LAB 14

Write a program for congestion control using Leaky bucket algorithm.

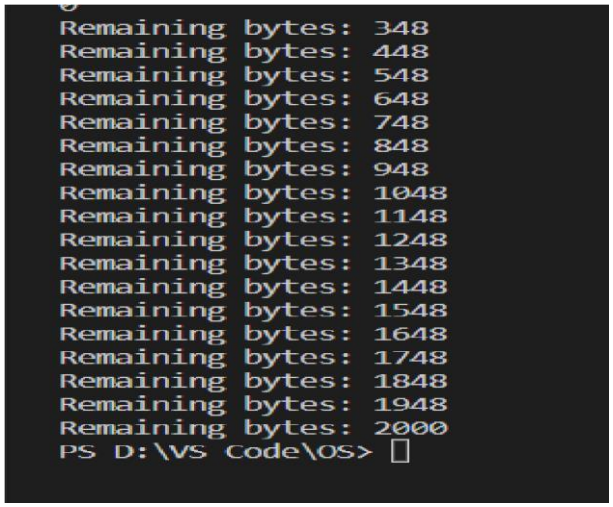
CODE:

```
#include <stdio.h>
#include <stdlib.h> // Include this for the rand() function
int main()
{
   int buckets, outlets, k = 1, num, remaining;
   printf("Enter Bucket size and outstream
   size\n"); scanf("%d %d", &buckets, &outlets);
   remaining = buckets; while (k)
   {
      num = rand() % 1000; // Generate a random number between 0 and
999 if (num < remaining)
      {
         remaining = remaining - num; printf("Packet of %d bytes
         accepted\n", num); // Added missing
variable
      }
      else
      {
         printf("Packet of %d bytes is discarded\n", num);
      }
      if (buckets - remaining > outlets)
      {
         remaining += outlets; // Fixed the calculation
      }
      else
         remaining = buckets; printf("Remaining
      bytes: %d \n", remaining); printf("If you
```

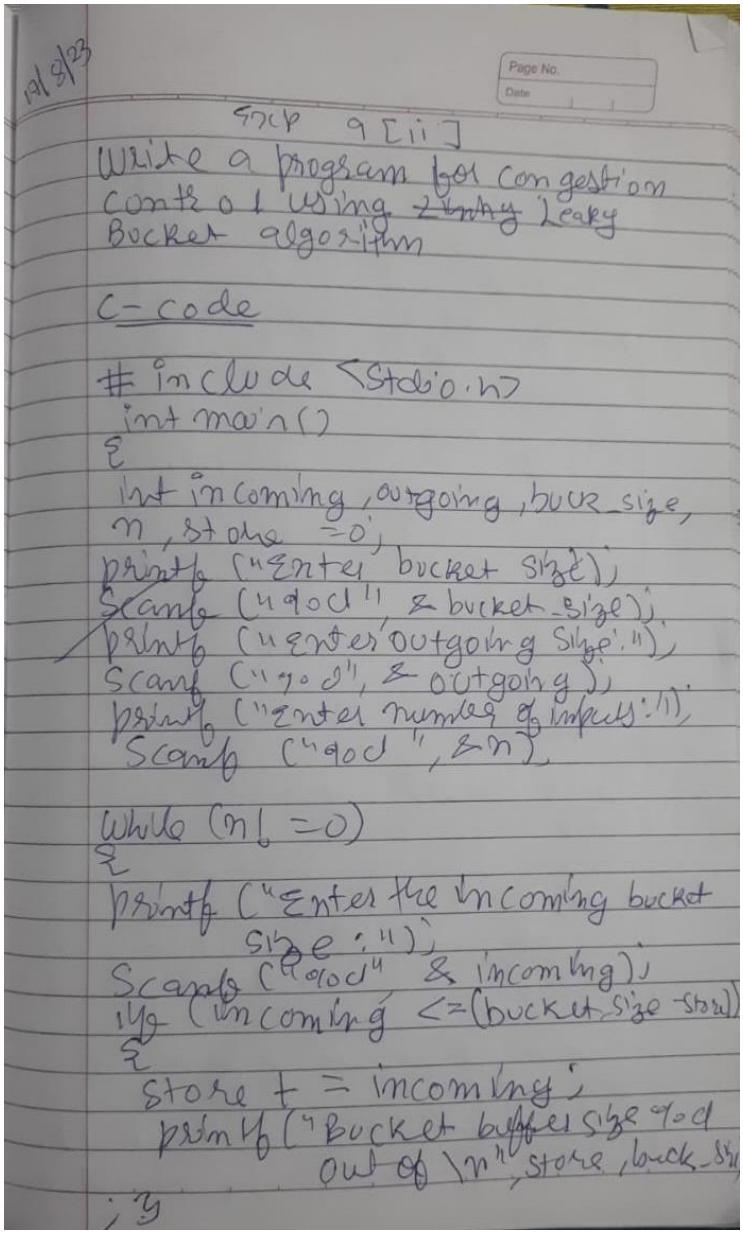
```
        want to stop input, press 0, otherwise,
        press 1\n"); scanf("%d", &k);
    }
    while (remaining < buckets) // Fixed the condition
    {
        if (buckets - remaining > outlets)
        {
            remaining += outlets; // Fixed the calculation
        }
        else
            remaining = buckets;
        printf("Remaining bytes: %d \n", remaining);
    }
    return 0; // Added a return statement to indicate successful completion
}
```

OUTPUT:

```
Enter Bucket size and outstream size
2000
100
Packet of 41 bytes accepted
Remaining bytes: 2000
If you want to stop input, press 0, otherwise, press 1
1
Packet of 467 bytes accepted
Remaining bytes: 1633
If you want to stop input, press 0, otherwise, press 1
1
Packet of 334 bytes accepted
Remaining bytes: 1399
If you want to stop input, press 0, otherwise, press 1
1
Packet of 500 bytes accepted
Remaining bytes: 999
If you want to stop input, press 0, otherwise, press 1
1
Packet of 169 bytes accepted
Remaining bytes: 930
If you want to stop input, press 0, otherwise, press 1
1
Packet of 724 bytes accepted
Remaining bytes: 306
If you want to stop input, press 0, otherwise, press 1
1
Packet of 478 bytes is discarded
Remaining bytes: 406
If you want to stop input, press 0, otherwise, press 1
1
Packet of 358 bytes accepted
Remaining bytes: 148
If you want to stop input, press 0, otherwise, press 1
1
Packet of 962 bytes is discarded
Remaining bytes: 248
If you want to stop input, press 0, otherwise, press 1
0
Remaining bytes: 348
Remaining bytes: 448
Remaining bytes: 548
Remaining bytes: 648
Remaining bytes: 748
```


Remaining bytes: 348
Remaining bytes: 448
Remaining bytes: 548
Remaining bytes: 648
Remaining bytes: 748
Remaining bytes: 848
Remaining bytes: 948
Remaining bytes: 1048
Remaining bytes: 1148
Remaining bytes: 1248
Remaining bytes: 1348
Remaining bytes: 1448
Remaining bytes: 1548
Remaining bytes: 1648
Remaining bytes: 1748
Remaining bytes: 1848
Remaining bytes: 1948
Remaining bytes: 2000
PS D:\VS Code\OS>

OBSERVATION:



Exp 9[ii]

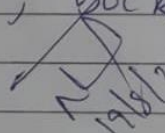
Write a program for congestion control using Leaky Bucket algorithm

C-code

```
#include <stdio.h>
int main()
{
int incoming, outgoing, buck_size,
n, store =0;
printf("Enter bucket size");
scanf("%d", &bucket_size);
printf("enter outgoing size:");
scanf("%d", &outgoing);
printf("enter number of inputs:");
scanf("%d", &n);

while(n!=0)
{
printf("Enter the incoming bucket
        size:");
scanf("%d" &incoming);
if(incoming <=(bucket_size-store))
{
store += incoming;
printf("Bucket buffer size %d
        out of %d", store, buck_si
;}
```

else
{
printf("Dropped %d no of packets\n"
incoming - (buck_size - store));

printf("Bucket buffer size %d out of
%d \n", store, buck_size);
store = buck_size;
}
store = store - outgoing;
printf("After outgoing %d packets
left out of %d in buffer\n",
store, buck_size);
n--;
}
}



Output :-

Enter bucket size: 5000
enter outgoing rate: 2000
enter number of inputs: 2
Enter the incoming packet size: 3000
Bucket buffer size 3000 out of 5000
After outgoing 1000 packets left out of
5000 in buffer
Enter the incoming packet size: 1000
Bucket buffer size 2000 out of 5000
After Outgoing 0 packets left
out of 5000 in buffer

LAB 15

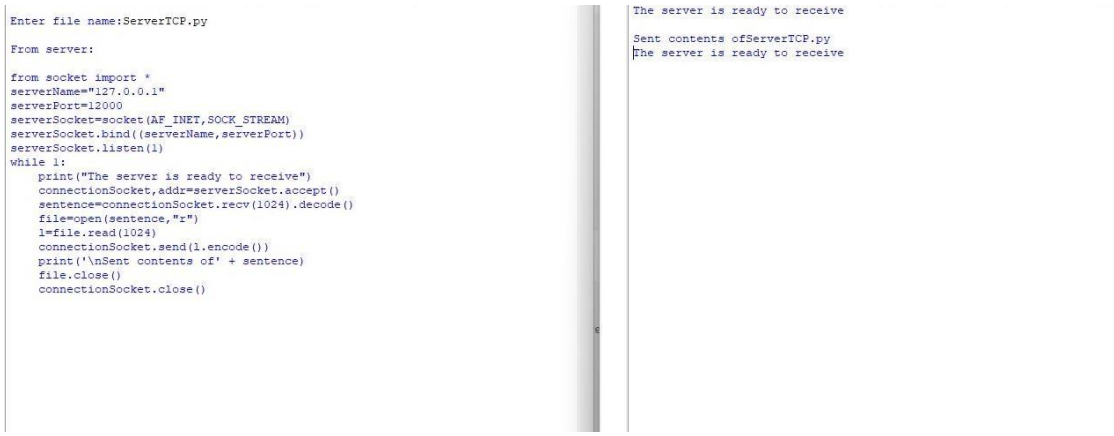
Using TCP/IP sockets, write a client-server program to make client sending the file name and the server to send back the contents of the requested file if present.

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:

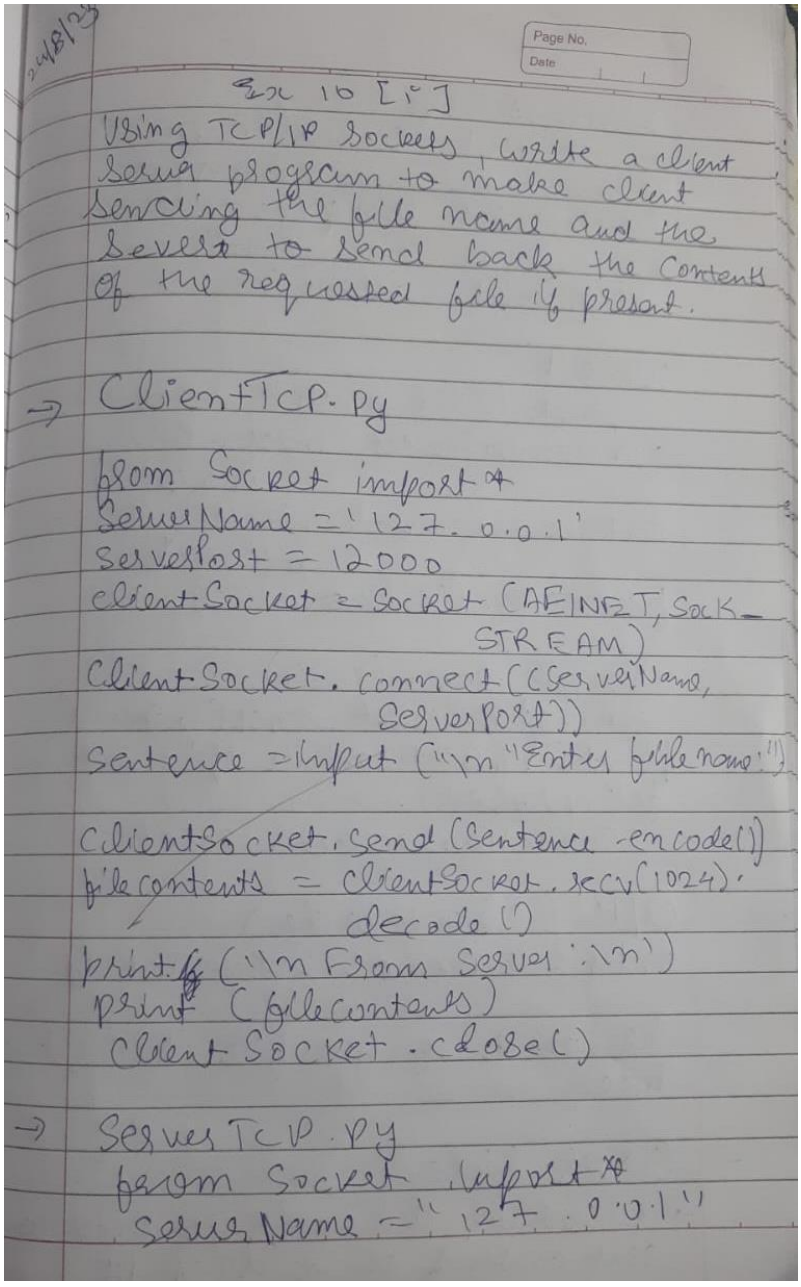


```
Enter file name:ServerTCP.py

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 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()
```

```
The server is ready to receive

Sent contents ofServerTCP.py
The server is ready to receive
```

‘

OBSERVATION:



Page No.
Date

24/8/23

Ex 10 [i]

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.

→ ClientTCP.py

```
from Socket import *
ServerName = '127.0.0.1'
ServerPort = 12000
clientSocket = socket(AF_INET, SOCK_
                      STREAM)
ClientSocket.connect((ServerName,
                      ServerPort))
Sentence = input("\n Enter filename:")

ClientSocket.send(Sentence.encode())
filecontents = clientSocket.recv(1024).
               decode()
print('\n From Server :\n')
print(filecontents)
clientSocket.close()
```

→ ServerTCP.py

```
from Socket import *
ServerName = '127.0.0.1'
```

Page No.
Date

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('\n Sent contents of' + sentence)
file.close()
connectionSocket.close()

O/P (client side)
Enter file name: serverTCP.py

The contents of the file is displayed here.

-> Server side
The server is ready to receive
sent contents of serverTCP.py

LAB 16

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

CODE:

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

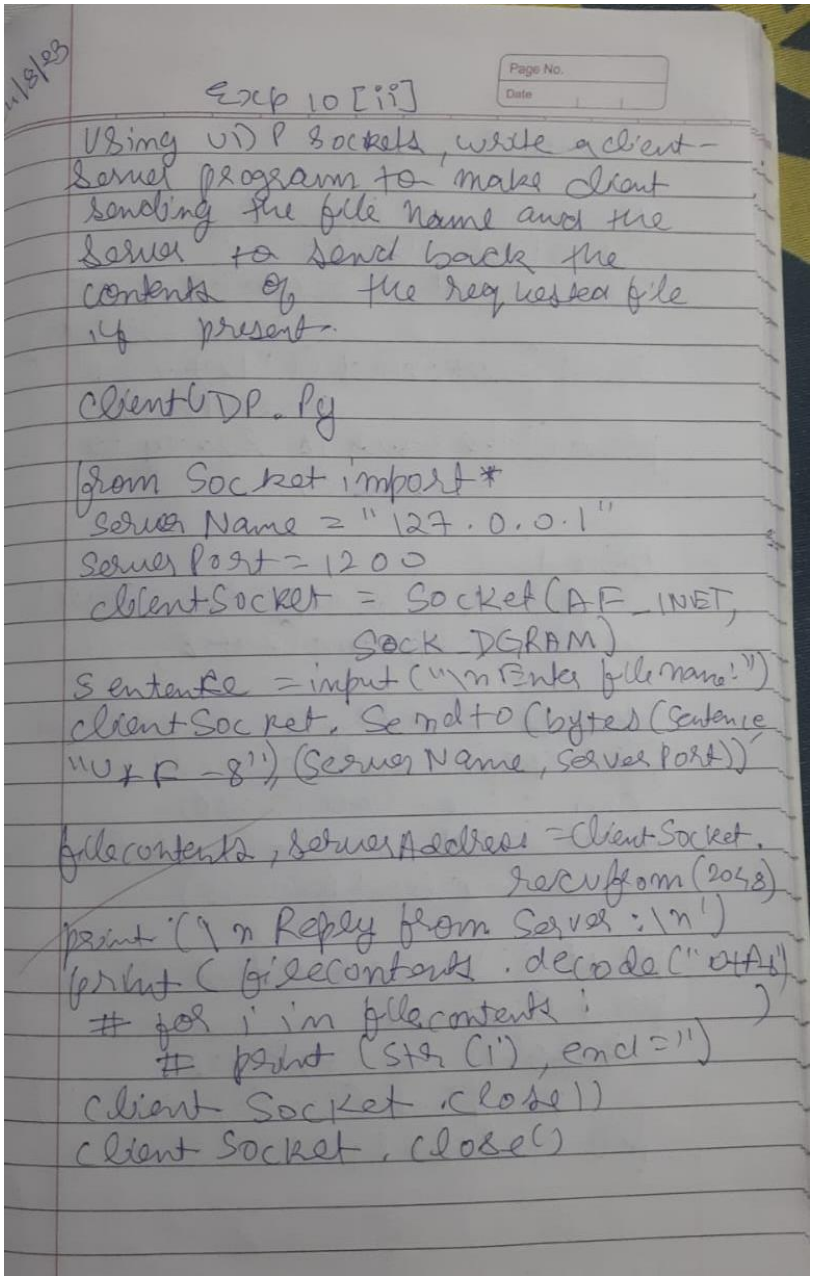
from socket import * serverPort = 12000
serverSocket = socket(AF_INET, SOCK_DGRAM)
serverSocket.bind((“127.0.0.1”, serverPort))
print (“The server is ready to receive”) while
1:
sentence, clientAddress = serverSocket.recvfrom(2048)
sentence = sentence.decode(“utf-8”) file=open(sentence,”r”)
con=file.read(2048) 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:

```
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()
```

```
The server is ready to receive

Sent contents of  ServerUDP.py
```

OBSERVATION:



4/9/23

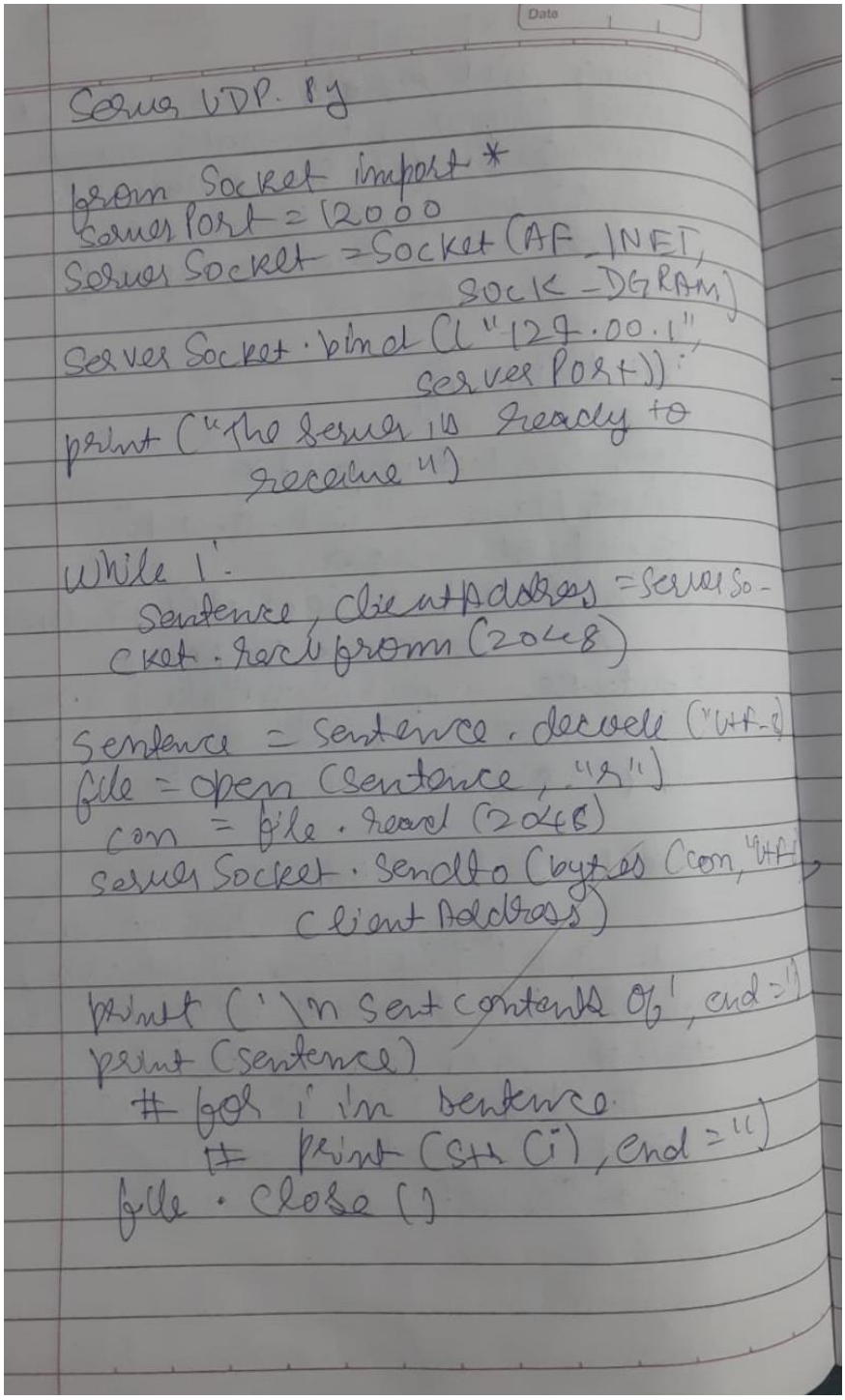
Page No.
Date

Exp 10 [ii]

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.

ClientUDP.Py

```
from Socket import *
serverName = "127.0.0.1"
serverPort = 1200
clientSocket = socket(AF_INET,
                      SOCK_DGRAM)
Sentence = input("\nEnter filename:")
clientSocket.sendto(bytes(Sentence,
"UTF-8"), (serverName, serverPort))

fileContents, serverAddress = clientSocket.
                              recvfrom(2048)
print("\n Reply from Server :\n")
print(fileContents.decode("UTF8")
# for i in fileContents:
    # print(str(i), end="")
clientSocket.close()
clientSocket.close()
```



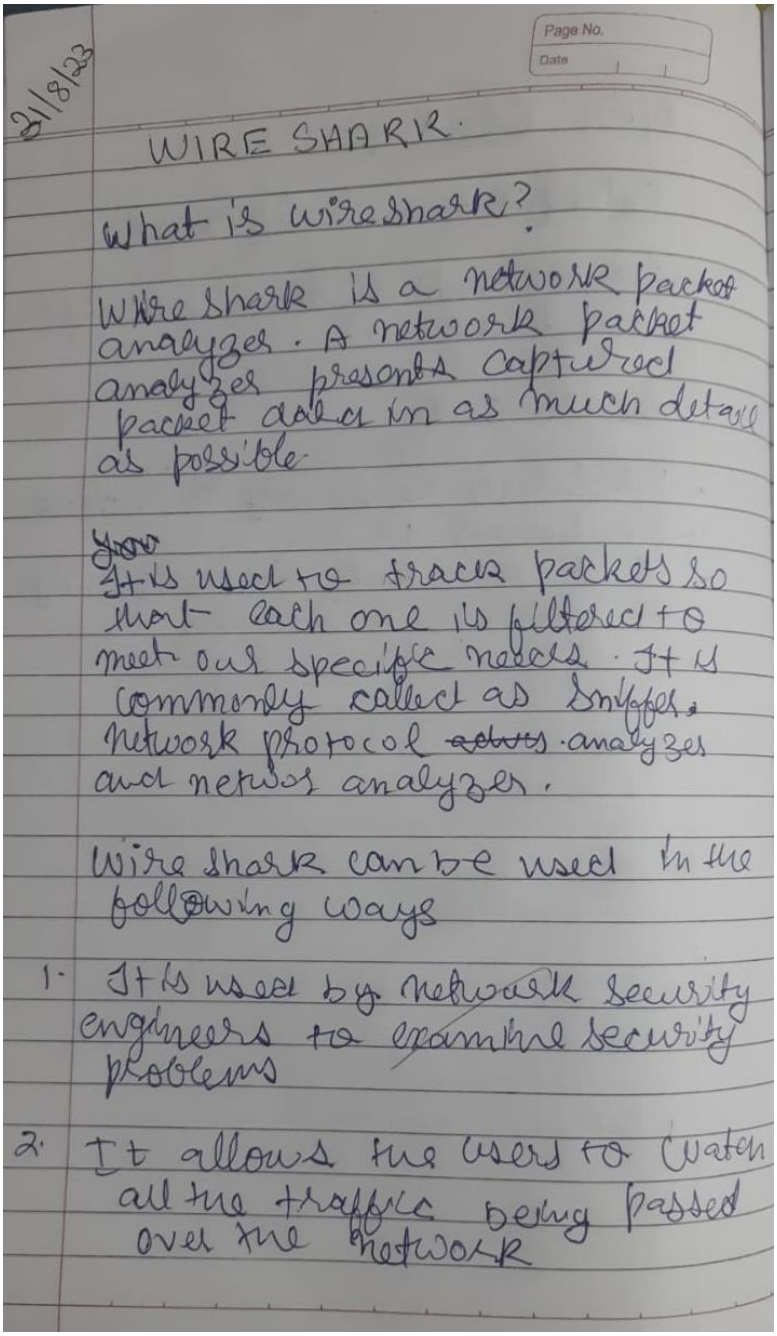
Server UDP.py

```
from Socket import *
serverPort = 12000
serverSocket = Socket(AF_INET,
                      SOCK_DGRAM)
serverSocket.bind(("127.00.1",
                   serverPort))
print("The Server is ready to
      receive")

while 1:
    sentence, clientAddress = serverSo-
    cket.recvfrom(2048)

sentence = sentence.decode("utf-8")
file = open(sentence, "r")
con = file.read(2048)
serverSocket.sendto(bytes(con, "utf-
                    clientAddress)

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

WEEK17

Tool Exploration -Wireshark

OBSERVATION:



21/8/23

Page No.
Date

WIRE SHARK.

What is wireshark?

Wire shark is a network packet analyzer. A network packet analyzer presents captured packet data in as much detail as possible.

It is used to trace packets so that each one is filtered to meet our specific needs. It is commonly called as Sniffer, Network protocol ~~analyzer~~ analyzer and network analyzer.

Wire shark can be used in the following ways

1. It is used by network security engineers to examine security problems

2. It allows the users to watch all the traffic being passed over the network

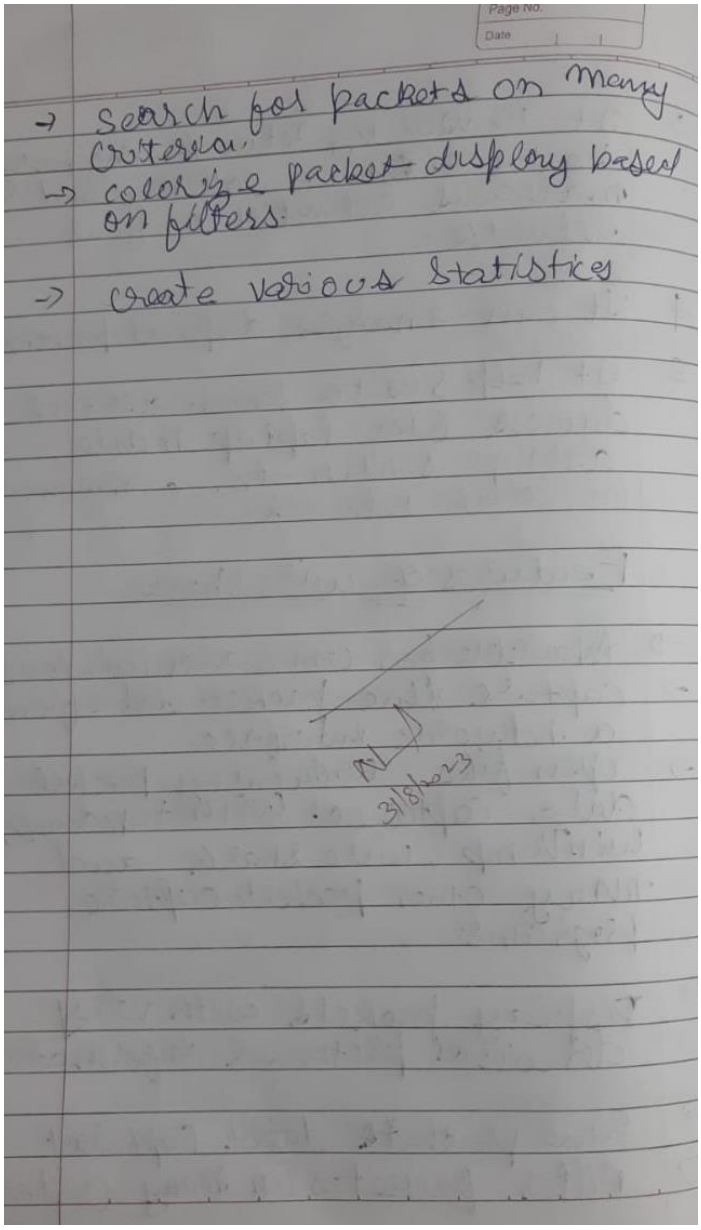
3. It is used by network engineering to troubleshoot latency and malicious activities on your network.

4. It can analyse dropped packets

5. It helps us to know all the devices like laptop, mobile, desktop, switch etc. communicate in local network.

Features of wireshark

→ Available for unix and windows
→ Capture live packet data from a network interface.
→ Open files containing packet data captured with tcpdump/ WinDump, wireshark and many other packet capture programs.
→ Display packets with very detailed protocol information
→ Save packet data captured
→ Filter packets on many criteria


Page No.
Date
→ Search for packets on many criteria.
→ colorize packet display based on filters.
→ create various statistics
NL
31/8/2023