VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"JnanaSangama", Belgaum -590014, Karnataka.

LAB REPORT
on

COMPUTER NETWORKS

Submitted by

ROHAN PAWAR V (1BM22CS416)

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" carried out by **ROHAN PAWAR V (1BM21CS416),** who is a 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.

SOWMYA T
Assistant Professor
Department of CSE
BMSCE, Bengaluru

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

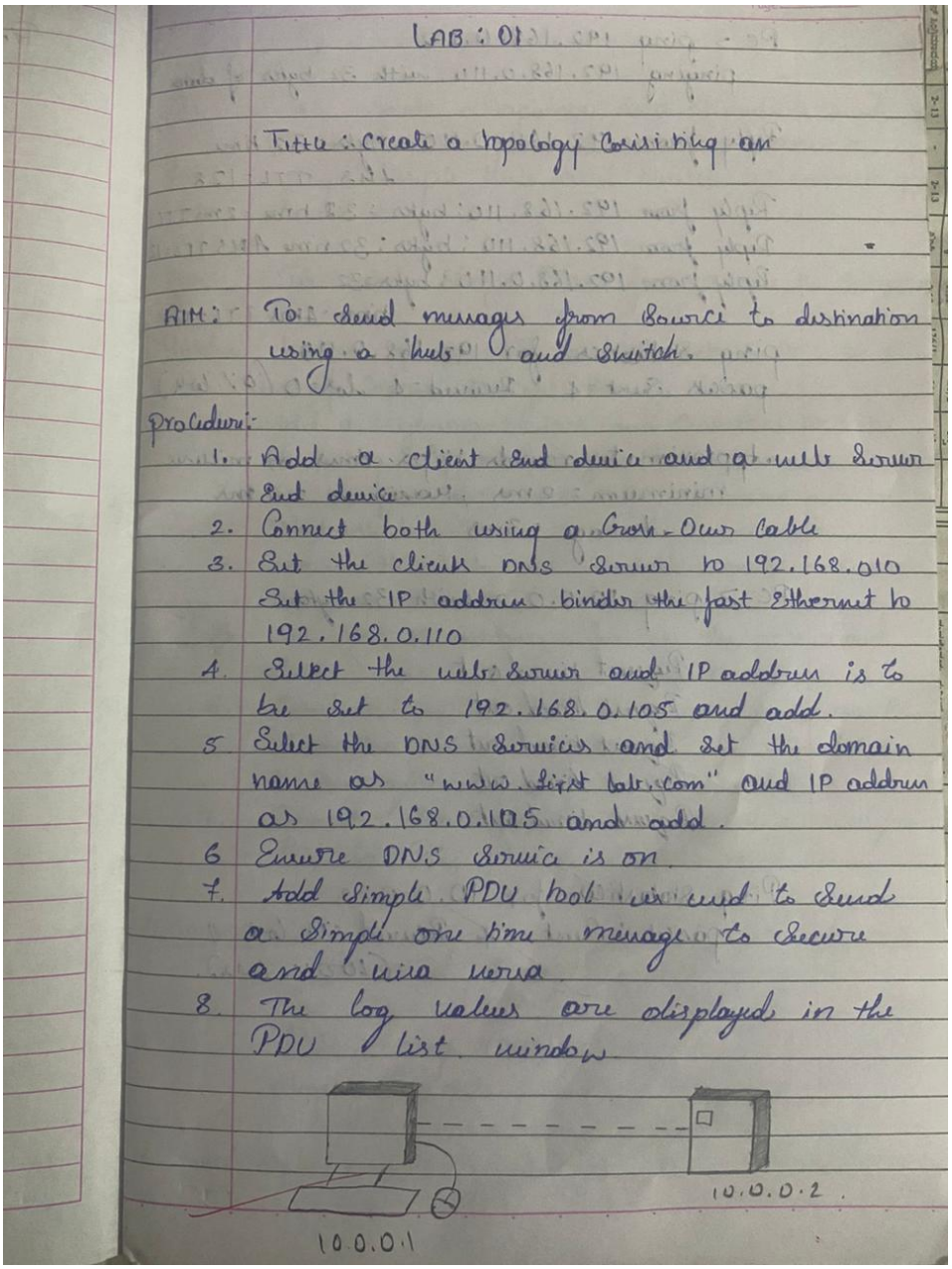
`

Index

WEEK 1

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

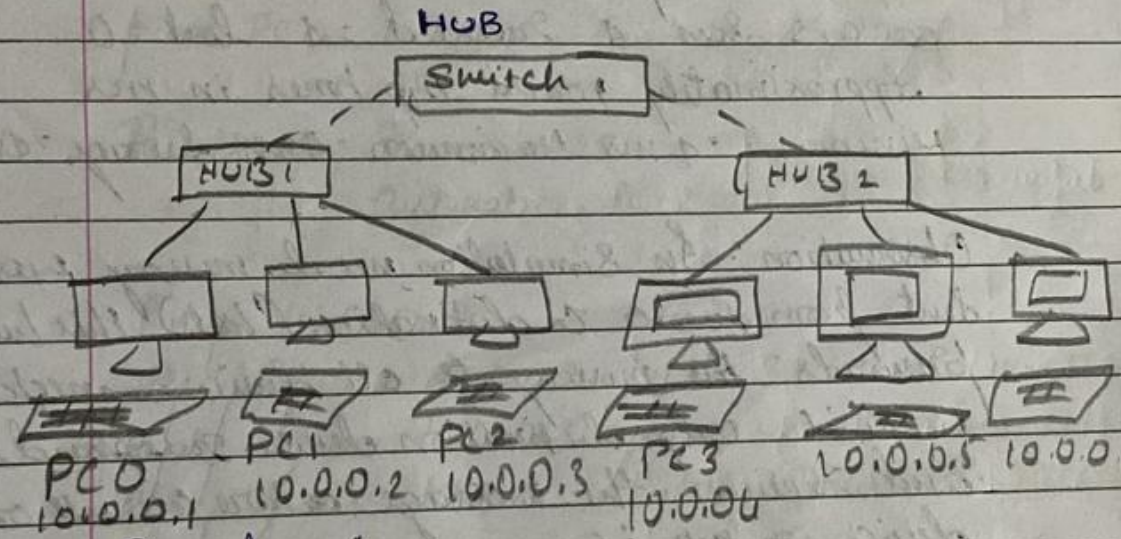


Date __/__/__
Page

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

AIM:- To make and understand topology using hub switch and hybrid topology using both.

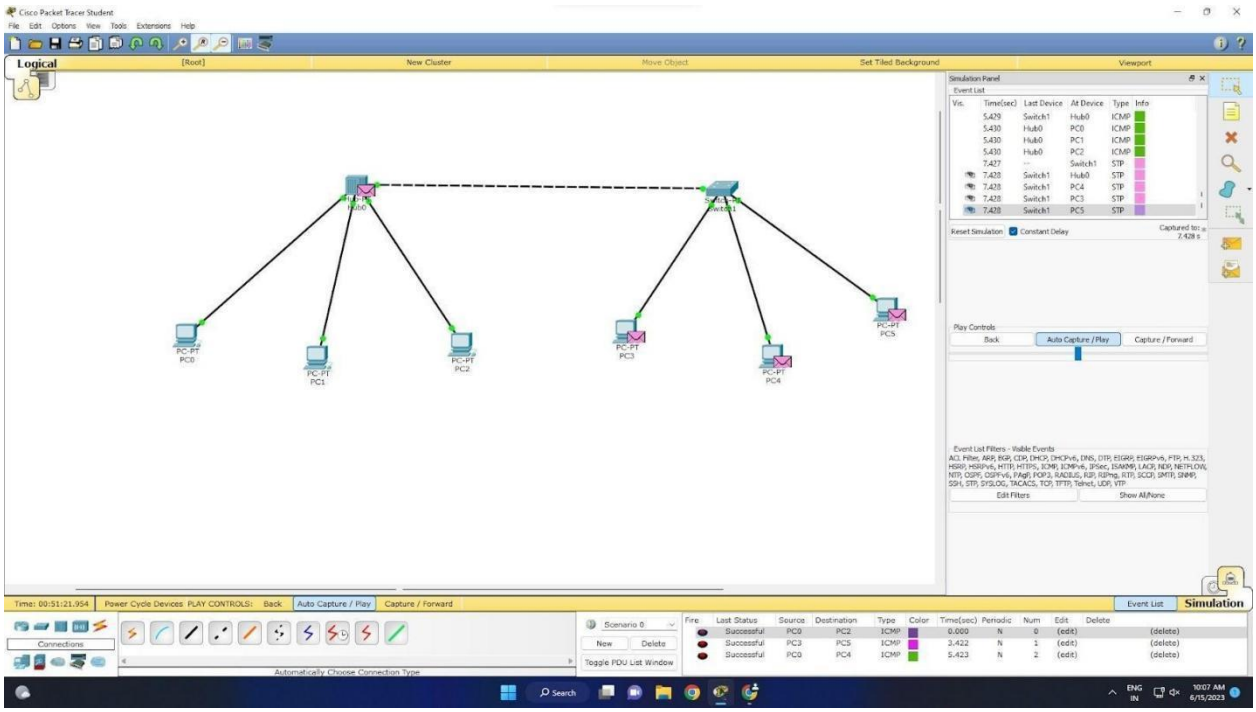
(*) Network with HUB TOPology.



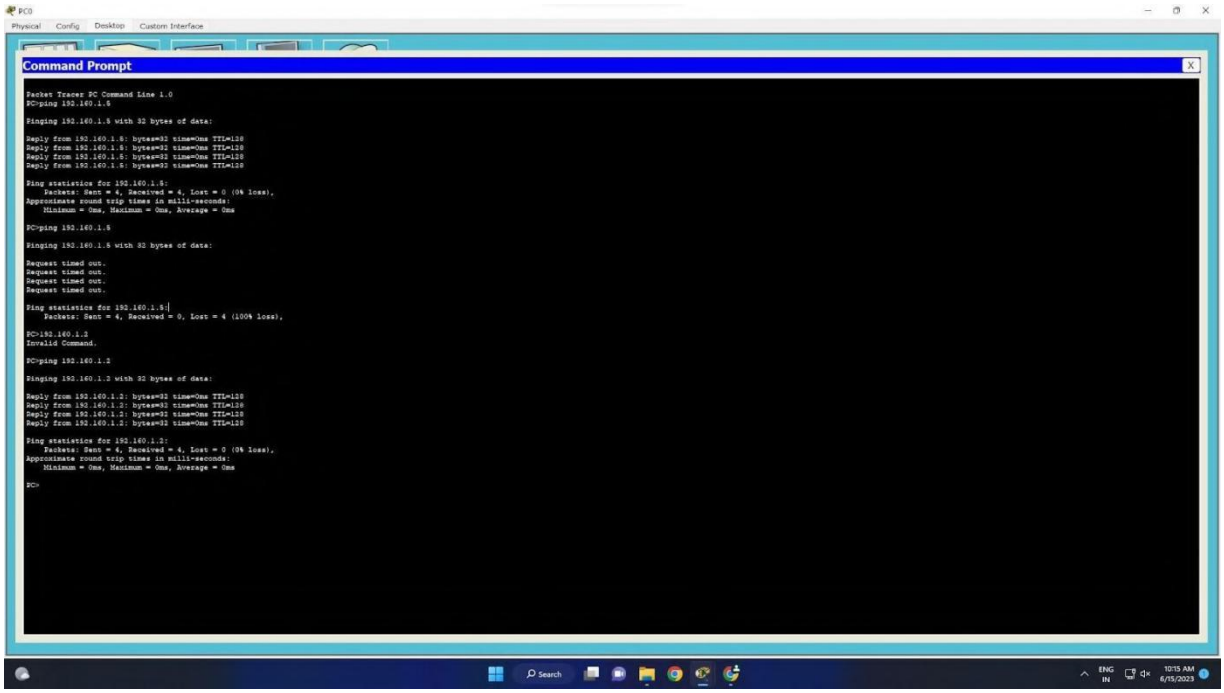
Procedure:

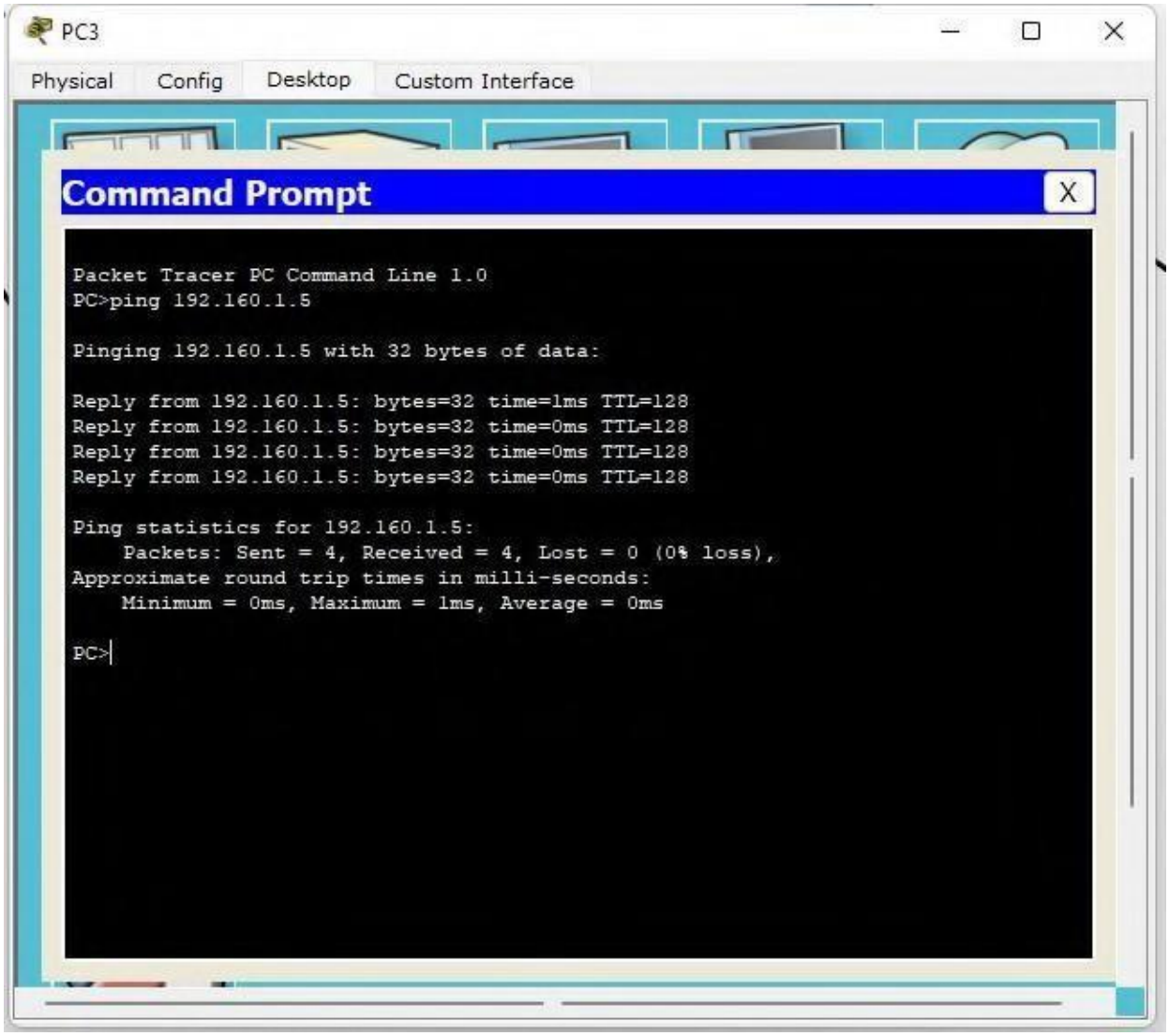
1. Select End user here as PC and a generic hub
2. click on End user → Config → fastethernet → assign IP address
3. Send a simple PDU message from PC1 to PC3 in the Simulation mode
4. In real time mode click on PC1 and under desktop Command prompt ping another PC

TOPOLOGY:



OUTPUT:

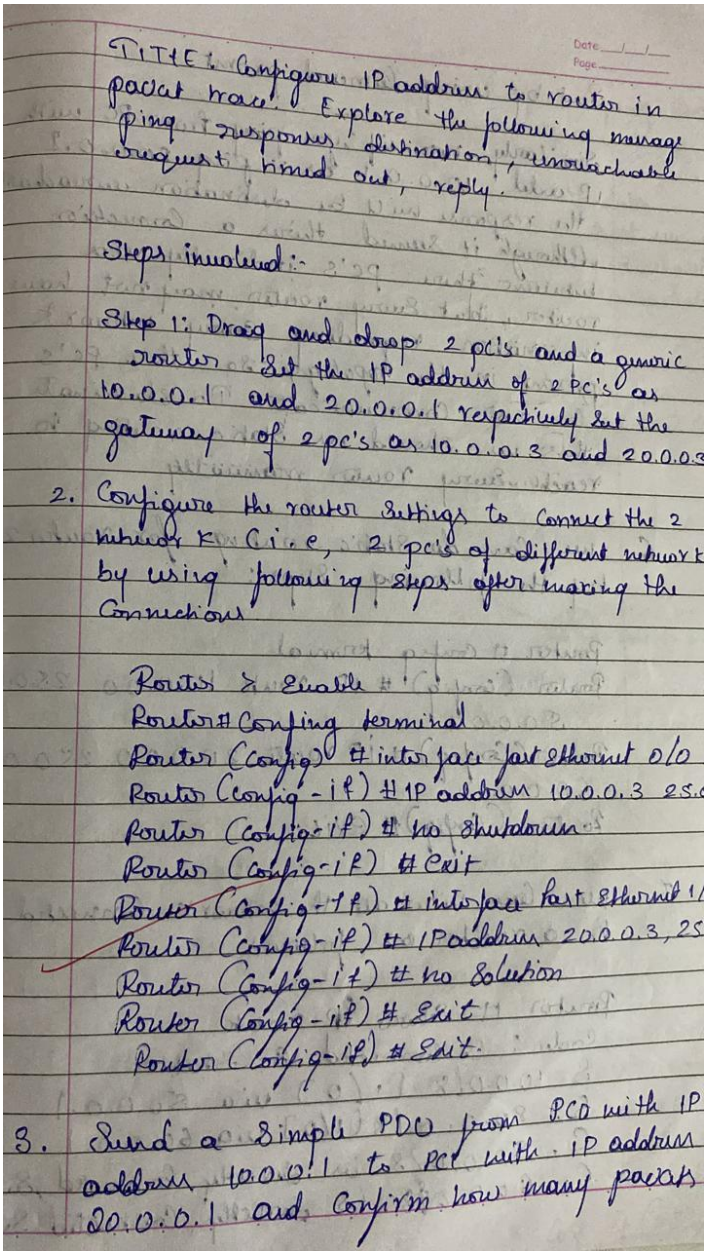



PC3
Physical
Config
Desktop
Custom Interface
Command Prompt
X
Packet Tracer PC Command Line 1.0
PC>ping 192.160.1.5
Pinging 192.160.1.5 with 32 bytes of data:
Reply from 192.160.1.5: bytes=32 time=1ms TTL=128
Reply from 192.160.1.5: bytes=32 time=0ms TTL=128
Reply from 192.160.1.5: bytes=32 time=0ms 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>

WEEK 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:



TITLE: Configure IP address to router in packet trace. Explore the following messages ping responses, destination, unreachable request timed out, reply.

Steps involved:-

Step 1: Drag and drop 2 pc's and a generic router. Set the IP address of 2 PC's as 10.0.0.1 and 20.0.0.1 respectively Set the gateway of 2 pc's as 10.0.0.3 and 20.0.0.3

2. Configure the router settings to connect the 2 network i.e, 2 pc's of different network by using following steps after making the connections

Router > Enable #
Router# Config terminal
Router (Config) # interface fast Ethernet 0/0
Router (Config-if) # IP address 10.0.0.3 25.
Router (Config-if) # no shutdown
Router (Config-if) # Exit
Router (Config-if) # interface fast Ethernet 1/
Router (Config-if) # IP address 20.0.0.3, 25
Router (Config-if) # no Shutdown
Router (Config-if) # Exit
Router (Config-if) # Exit

3. Send a simple PDU from PC0 with IP address 10.0.0.1 to PC with IP address 20.0.0.1 and Confirm how many packets

Sent by using Comand promt

Similarly, now if you ping from PC with IP add 10.0.0.1 as > ping 20.0.0.2 the response will be dubination unreachable Although it seemed there a connection between these PC's indirectly via router, but Every router may not have information regarding Every network present in the topology so these PC's can not communicate To Eliminate this we should use Static routing to reach Every router manually

6. we can do Static routing for router 2 by the following steps:

Router # Config termial
Router (Config) # IP route 10.0.0.0 25.0.0.0 50.0.0.1
Router (Config) # IP route 20.0.0.0 25.0.0.0 60.0.0.1
Router (Config) # exit

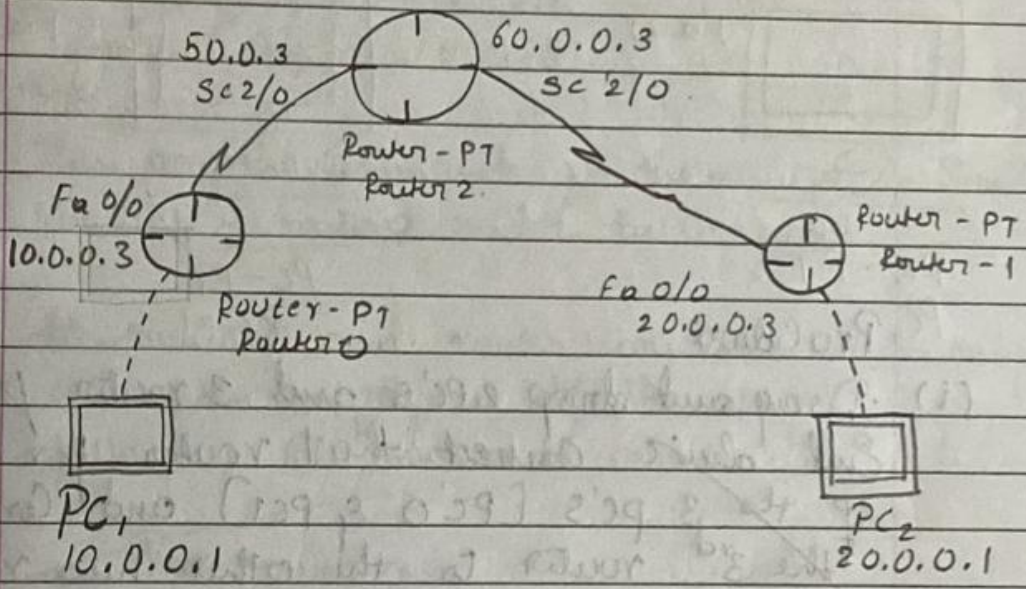
7. we can view all the network connected to a router as follows

Router # Show ip route

Before making static route from ping 10.0.0.1 Command prompt: pc> ping 10.0.0.1

Pinging 10.0.0.1 with 32 bytes of data
Reply from 20.0.0.1 destination host unreach
Reply from 20.0.0.1 destination host unreachable
Reply from 20.0.0.1 destination host unreachable

Request timed out
ping statics for 10.0.0.1:
packets: Sent 4, Received = 0 Lost 4 (100% loss)

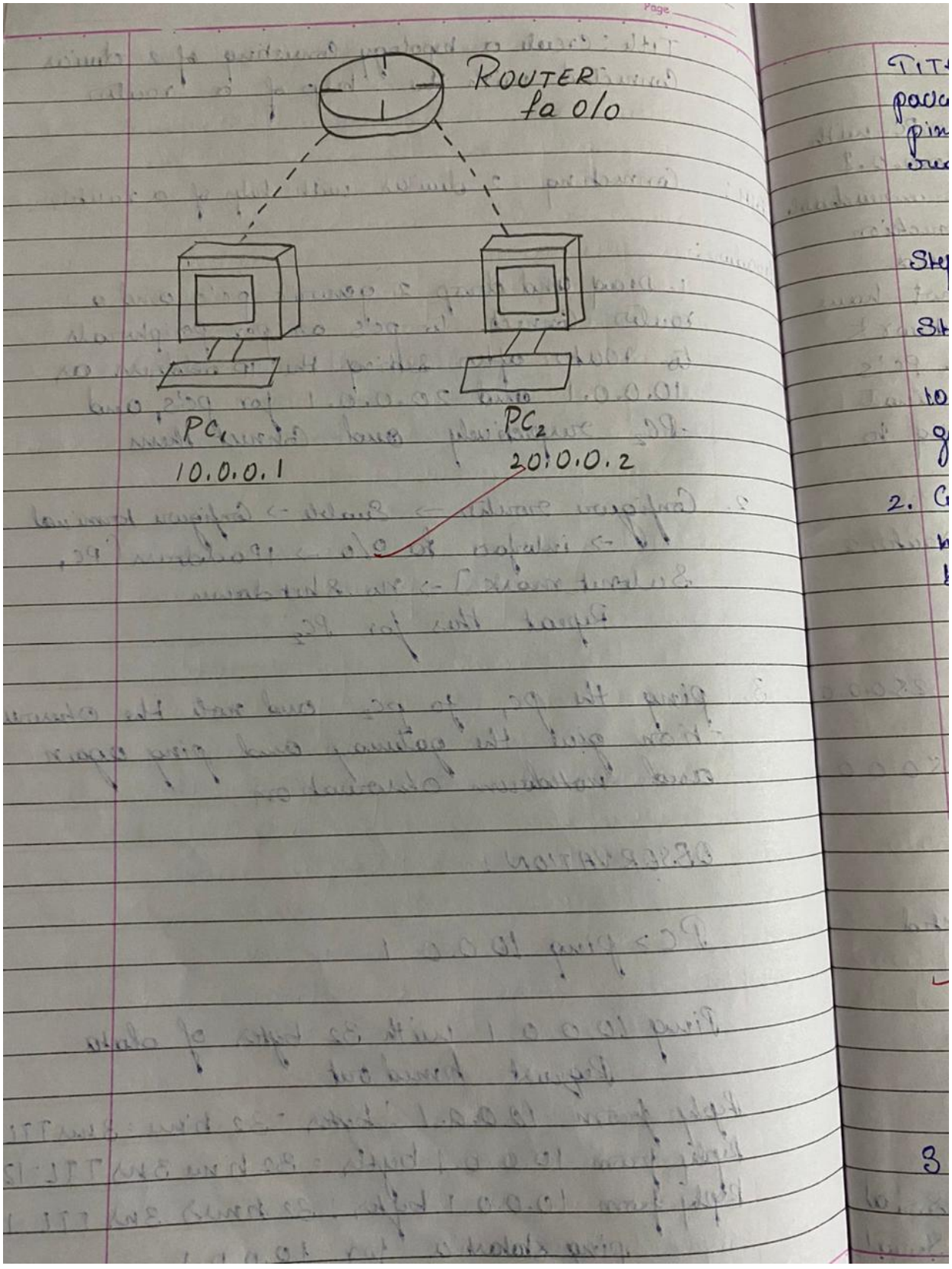


After static route, from PC1 to PC2.
Command prompt: PC> ping 20.0.0.1
ping 20.0.0.1 with 32 bytes of data

23/6/23
Sp: 7

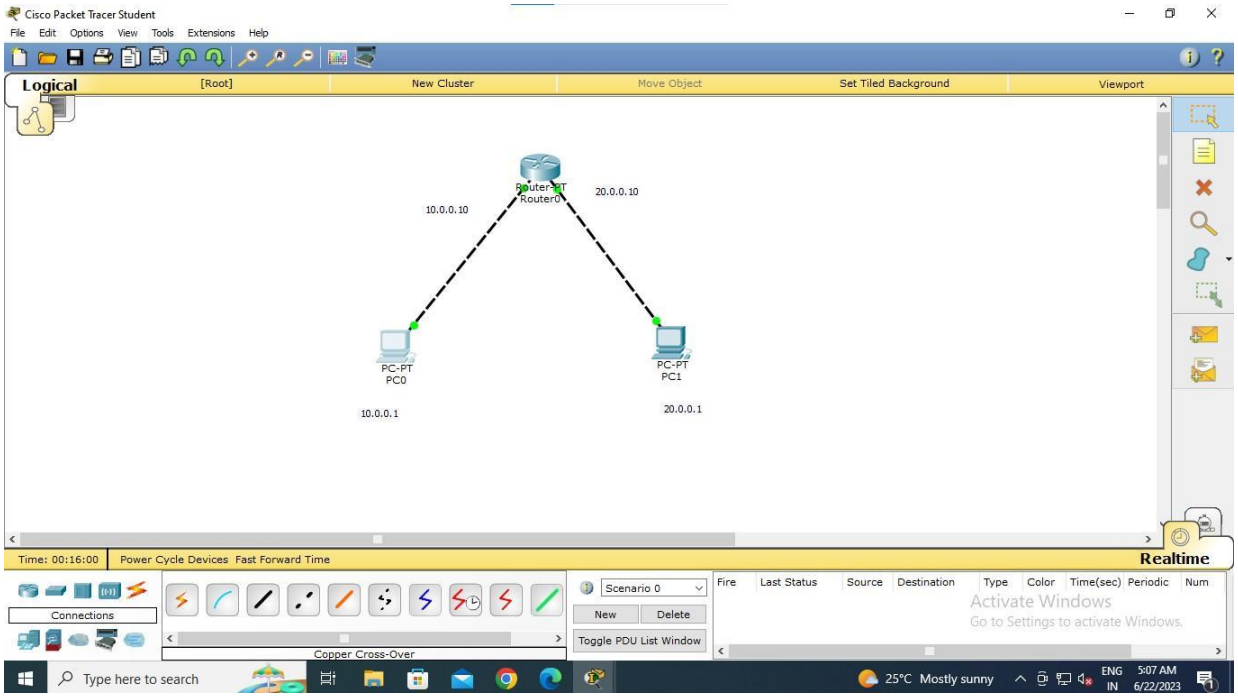
~~Request timed out~~

Reply from 20.0.0.1 bytes 32 time = 3ms TTL: 125
Reply from 20.0.0.1 bytes 32 time = 3 ms TTL = 125
Reply from 20.0.0.1: bytes = 32 time = ms TTL = 125

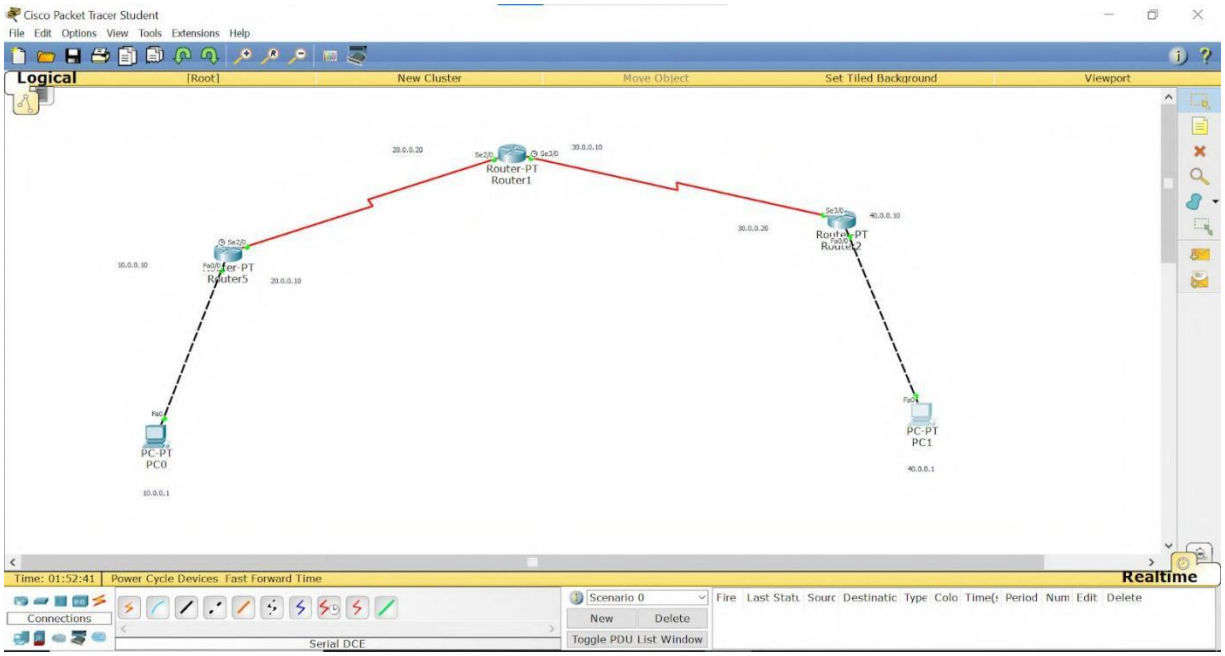

ROUTER
fa 0/0
PC1
10.0.0.1
PC2
20:0.0.2

TOPOLOGY:

PROGRAM 2.1

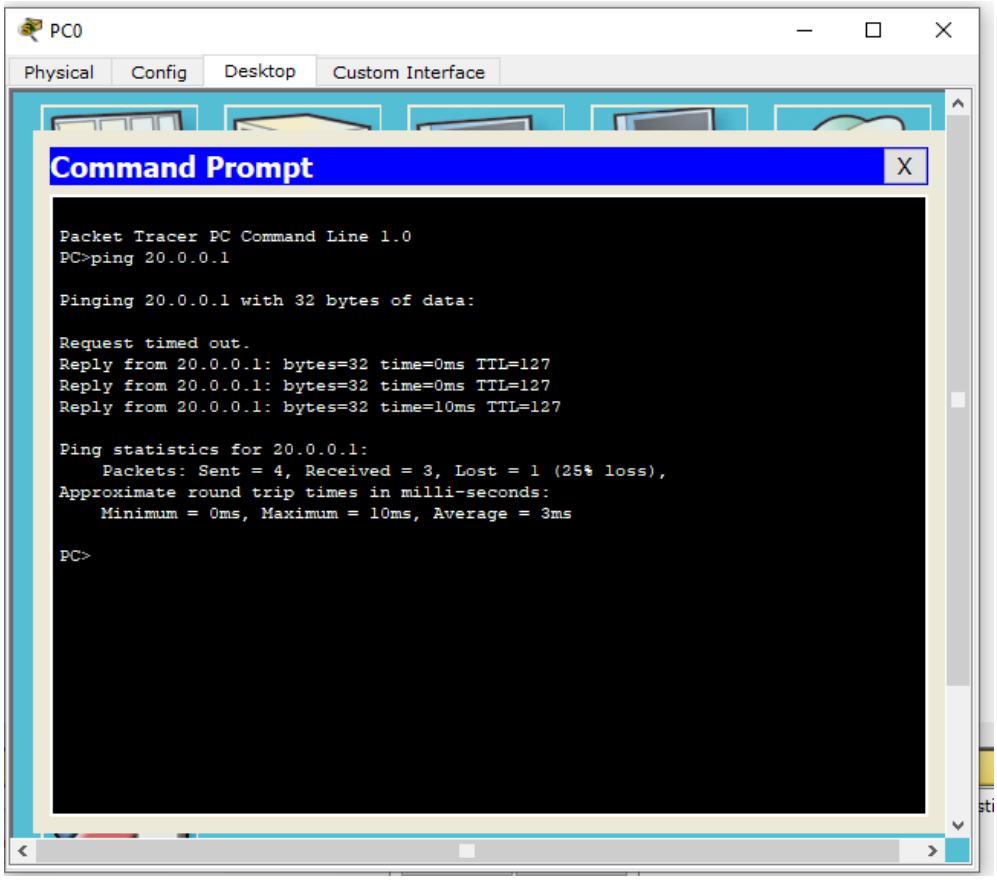


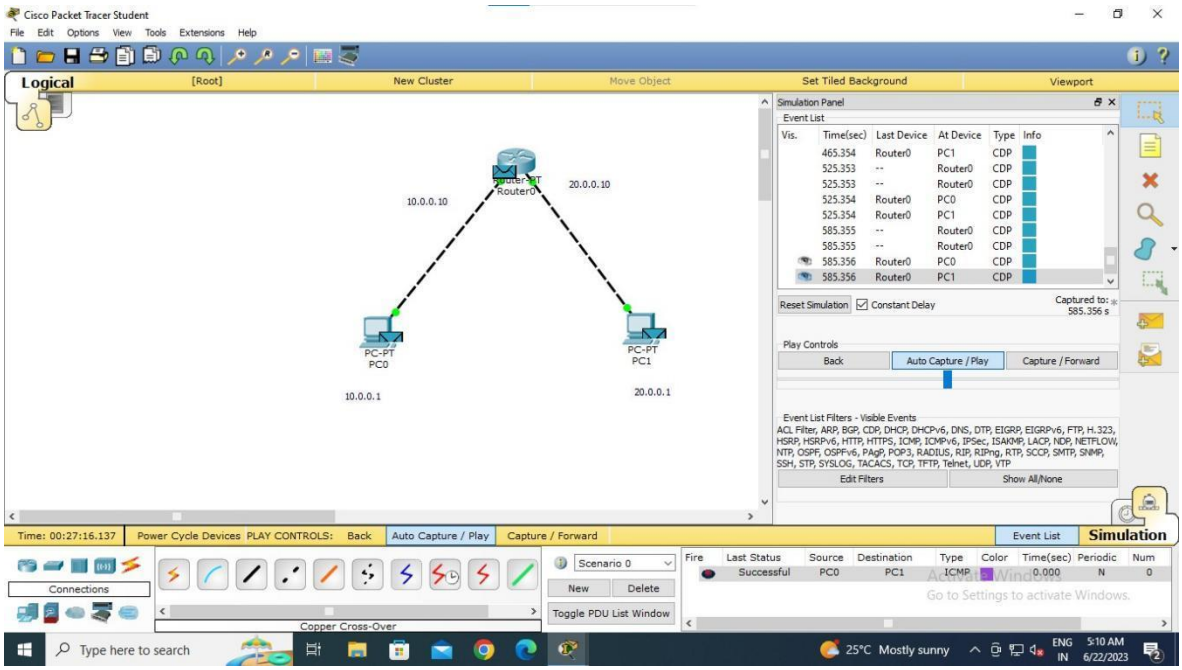
PROGRAM 2.2



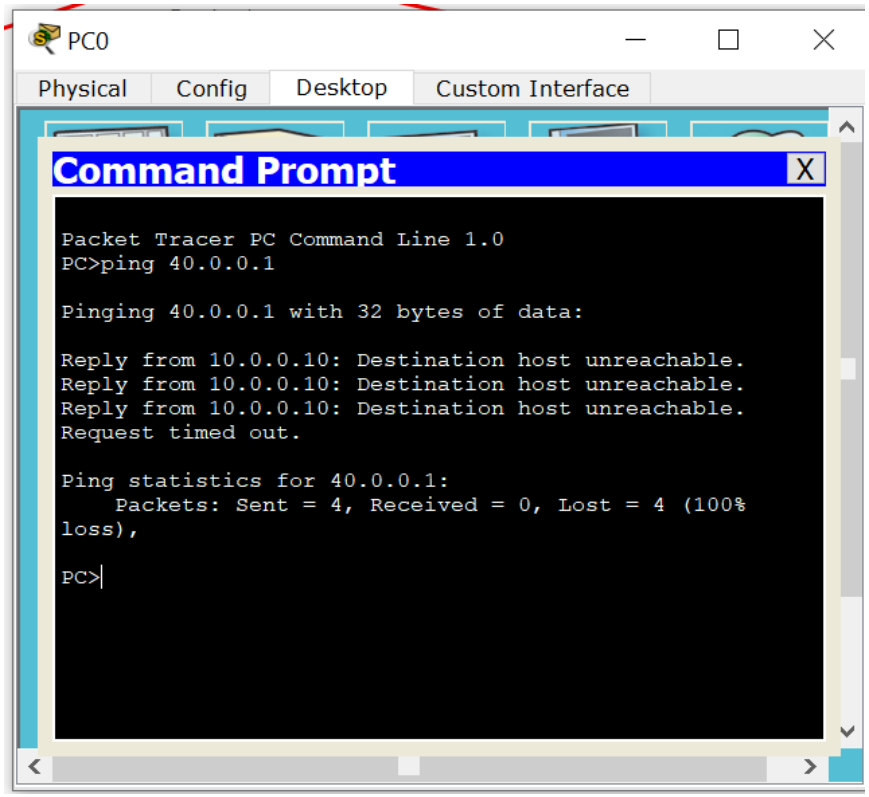
OUTPUT:

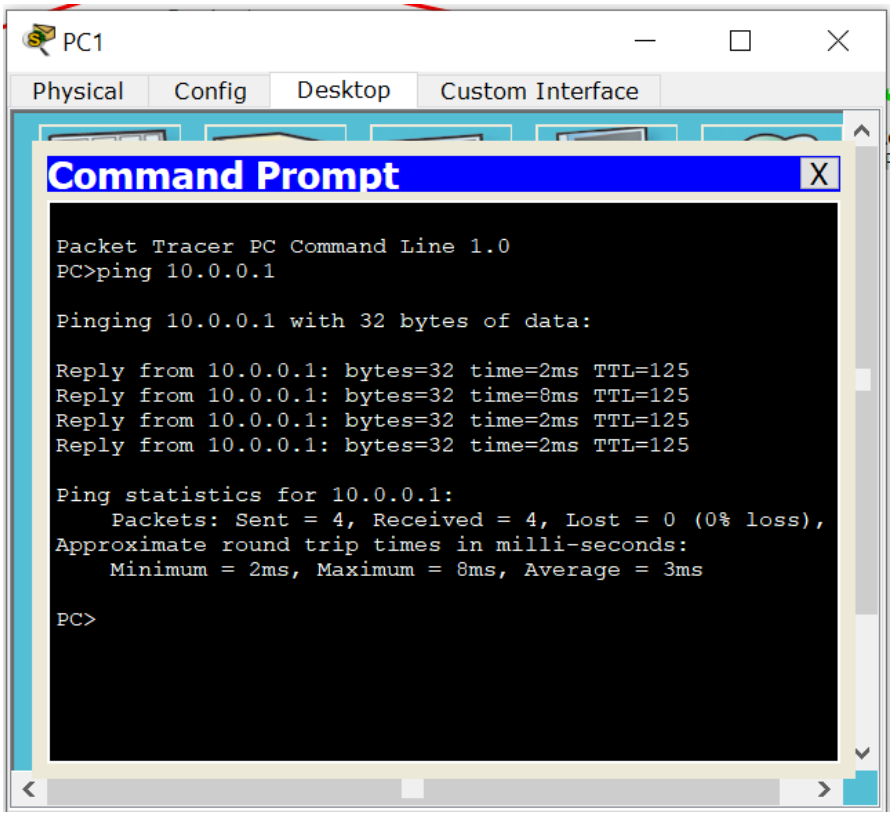
PROGRAM 2.1

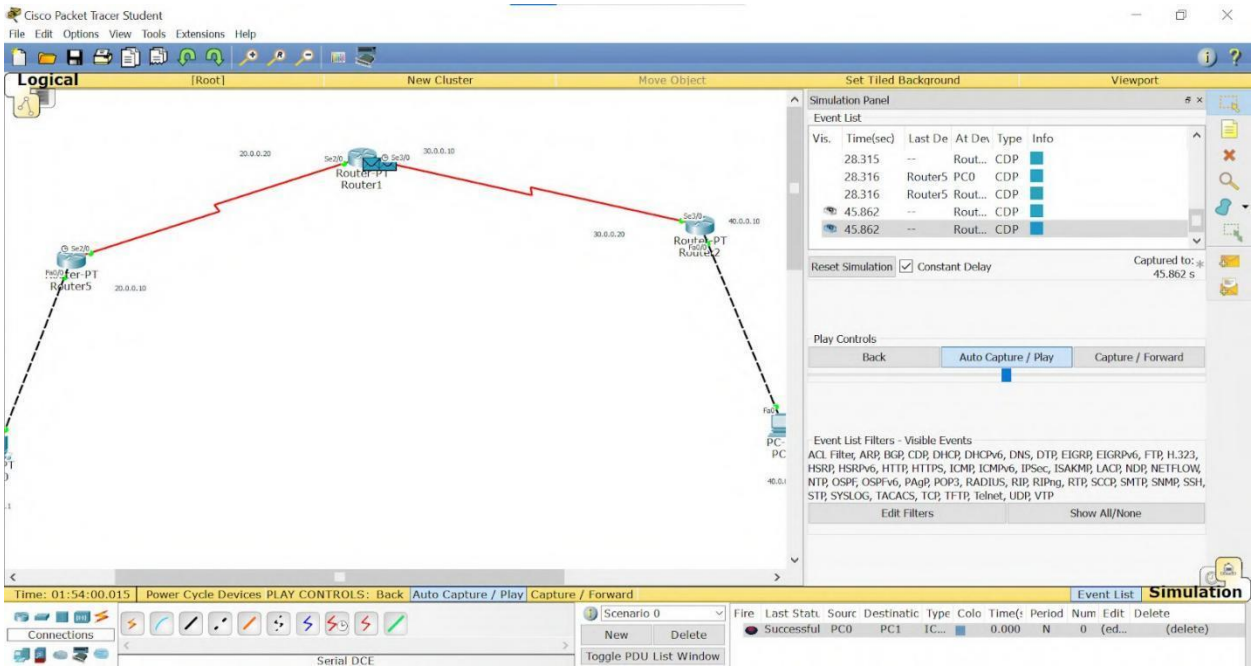




PROGRAM 2.2


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:
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
X
Packet Tracer PC Command Line 1.0
PC>ping 10.0.0.1
Pinging 10.0.0.1 with 32 bytes of data:
Reply from 10.0.0.1: bytes=32 time=2ms TTL=125
Reply from 10.0.0.1: bytes=32 time=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
Router2
Router-PT
Router5
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)

WEEK 3

Configure default route, static route to the Router.

OBSERVATION:

Router # Config t
router (Config) # ip route 10.0.0.0 255.0.0.0
30.0.0.2
router (Config) # ip route 20.0.0.0 255.0.0.0 40.0.0.2

router (Config) # Exit
router #

5. Now check the routing information for router 0
Router # Show ip route

C. Connected S-Static * Candidate default
Gateway of last resort is 30.0.0.2 to network 0.0.0.0
C 10.0.0.0/8 is directly connected, fast Ethernet 0/0

C 30.0.0.0/8 is directly connected, fast Ethernet 0/0 Serial 2/0
S* 0.0.0.0/0 [1/0] via 30.0.0.1

Router 2

Router # Show ip route
C - Connected S-Static

S 10.0.0.0/8 [1/0] via 30.0.0.1
S 20.0.0.0/8 [1/0] via 40.0.0.1
C 30.0.0.0/8 is directly connected, Serial 2/0
C 40.0.0.0/8 is directly Connected Serial 3/0

Ping Commands (o/p)

Pinging 20.0.0.1 with 32 byte of data

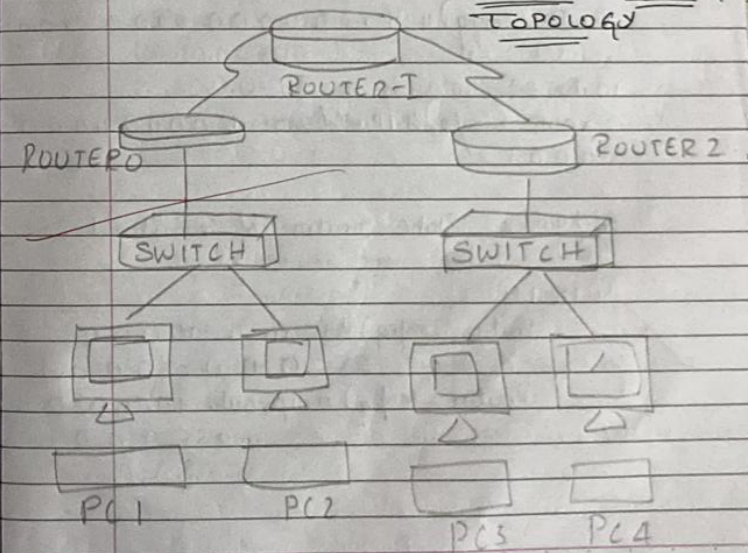
Reply from 20.0.0.1 bytes=32 time 4ms TTL=125.

Reply from 20.0.0.1 bytes 32 time=1ms TTL=125
Reply from 20.0.0.1 = bytes=32 time=17ms TTL=125
Reply from 20.0.0.1 = bytes=32 time=25ms TTL=125

Ping Statistics for 20.0.0.1
packet sent=4 Received=4 Lost=0 (0% Lost)

Approximate round trip times in ms
min=9ms max=25ms Average=16ms

DEFAULT ROUTING TOPOLOGY



Router # Config t
router (Config) # ip route 10.0.0.0 255.0.0.0
30.0.0.2
router (Config) # ip route 20.0.0.0 255.0.0.0 40.0.0.2

router (Config) # Exit
router #

5. Now check the routing information for router 0
Router # Show ip route

C. Connected S-Static * Candidate default
Gateway of last resort is 30.0.0.2 to network is 0.0.0.0
C 10.0.0.0/8 is directly connected, fast Ethernet 0/0

C 30.0.0.0/8 is directly connected, fast Ethernet 0/0 Serial 2/0
S* 0.0.0.0/0 [1/0] via 30.0.0.1

Router 2

Router # Show ip route
C - Connected S-Static

S 10.0.0.0/8 [1/0] via 30.0.0.1
S 20.0.0.0/8 [1/0] via 40.0.0.1
C 30.0.0.0/8 is directly connected, Serial 2/0
C 40.0.0.0/8 is directly connected Serial 3/0

Ping Commands (o/p)

Pinging 20.0.0.1 with 32 byte of data

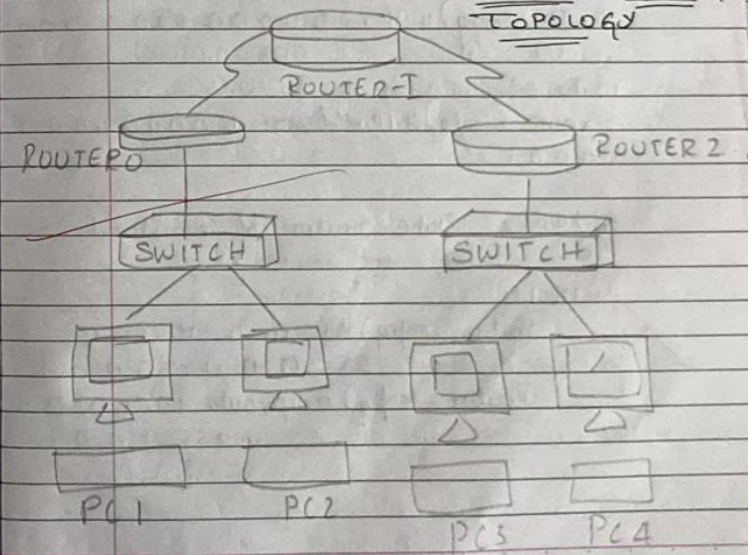
Reply from 20.0.0.1 bytes=32 time 4ms TTL=125.

Reply from 20.0.0.1 bytes 32 time=1ms TTL=125
Reply from 20.0.0.1 = bytes=32 time=17ms TTL=125
Reply from 20.0.0.1 = bytes=32 time=25ms TTL=125

Ping Statistics for 20.0.0.1
packets sent=4 Received=4 Lost=0 (0% Lost)

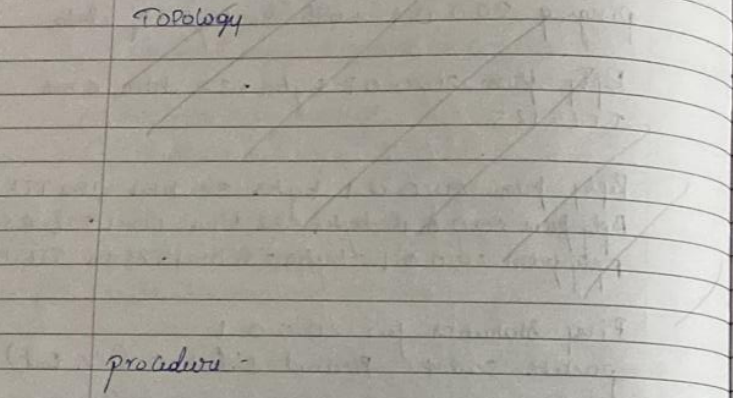
Approximate round trip times in ms
min=9ms max=25ms Average=16ms

DEFAULT ROUTING TOPOLOGY



DEFAULT ROUTING

TOPOLOGY



procedure -

Config using default route to router 0 and router2 router 0

Router (Config) # IP route 0.0.0.0 0.0.0.0 20.0.0.1

router 2:

router (Config) # IP address route 0.0.0.0 0.0.0.0 30.0.0.1

Configure 2 Static router of router 1

Router 1

Router (Config) # IP route 10.0.0.0 255.0.0.0 20.0.0.1

Router (Config) # ip route 10.0.0.0 255.0.0.0 30.

OBSERVATION

Static router to router 1 have been added and default router to router 0 & router 2

C 10.0.0.0/8 is directly connected.

S* 0.0.0.0/0 [1/0] via 20.0.0.2

router 2

Show IP route

C 30.0.0.0/8 is directly connected 8e2/0
C 10.0.0.0/8 is directly connected, Fe 0/0
S* 0.0.0.0/0 [1/0] via 30.0.0.1

router

Show IP route

S 10.0.0.0/8 [1/0] via 20.0.0.1
C 200.0.0/8 is directly connected 8e2/0
C 30.0.0.0/8 directly connected 8e3/0
S 10.0.0.0/8 [1/0] via 30.0.0.2.

Output

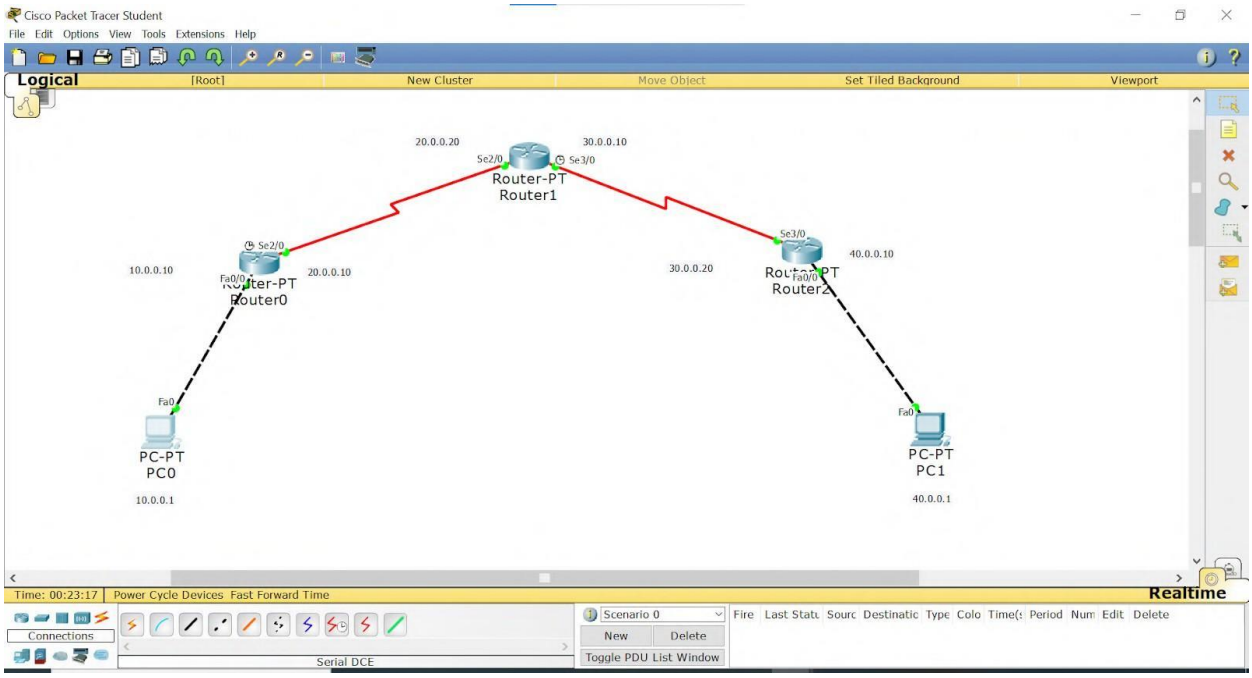
ping request from PC1

→ Ping 10.0.0.10

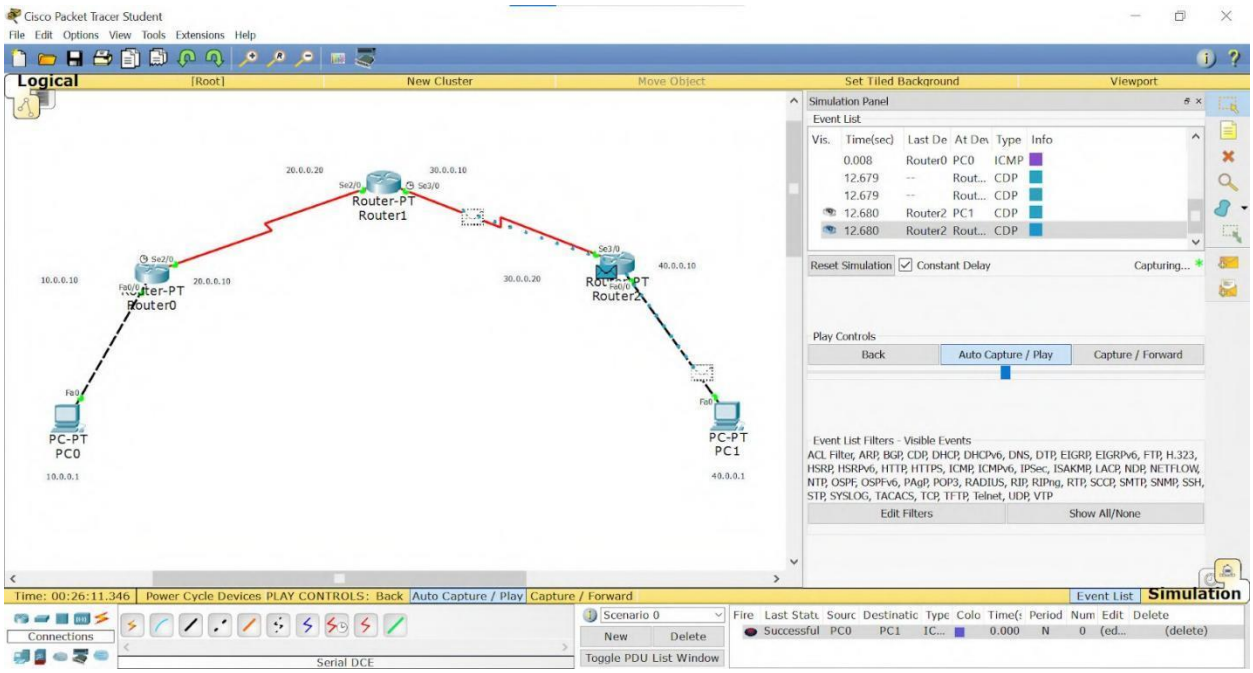
pinging 10.0.0.10 with 32 bytes of data

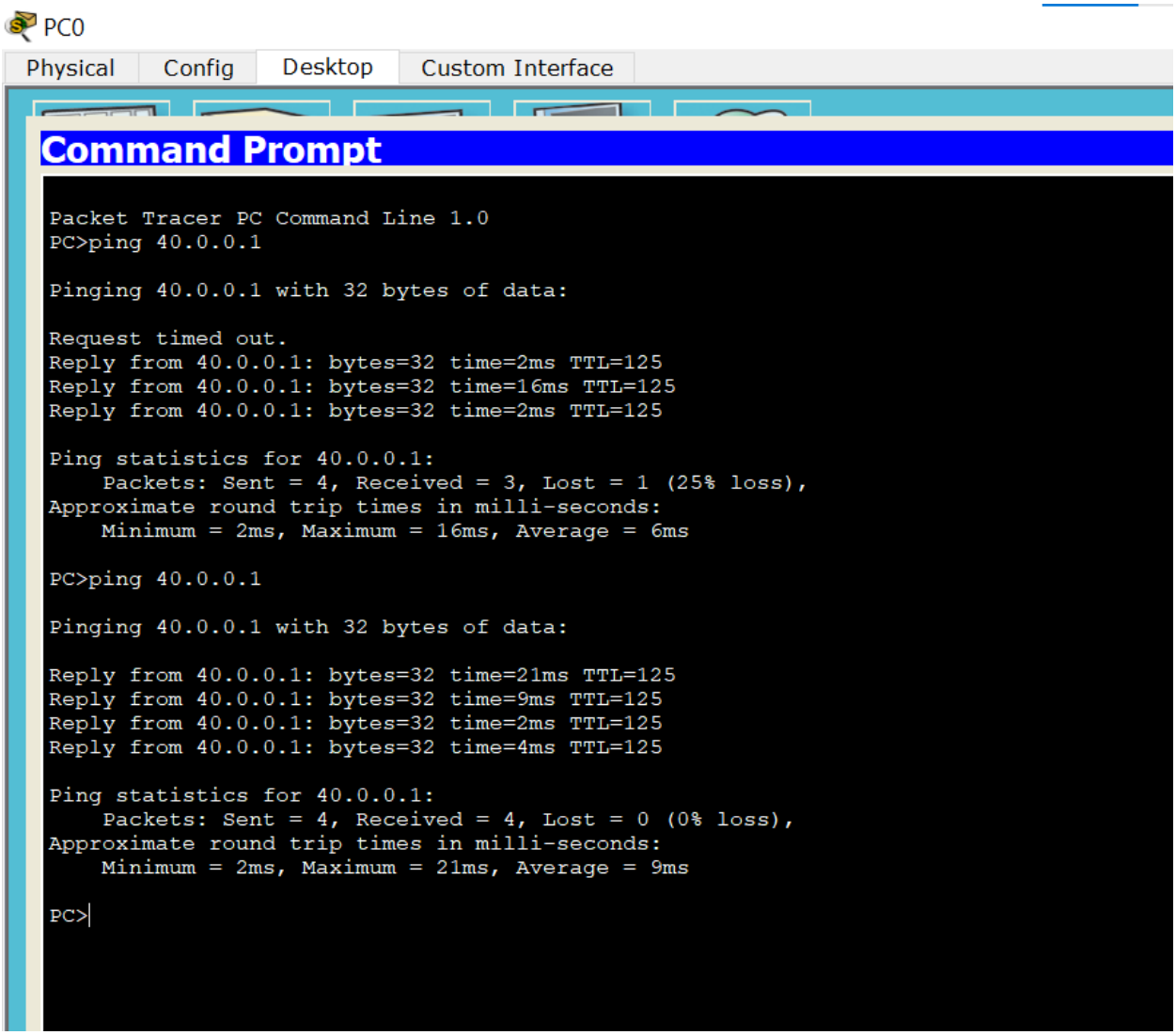
reply from 10.0.0.10 bytes 32 time=9ms TTL=125

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>

WEEK 4

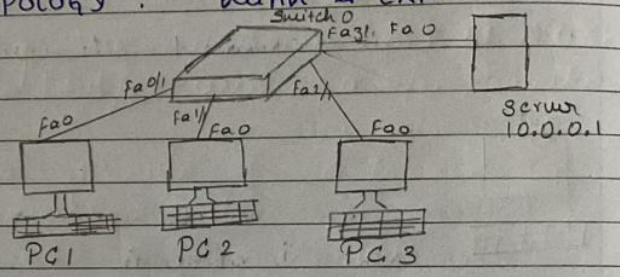
Configure DHCP within a LAN and outside LAN.

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 Starting 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.

Date __/__/__
Page _____

OBSERVATION:

→ All pc's connected to the Switch and Server gets automatically assigned IP address starting from 10.0.0.2

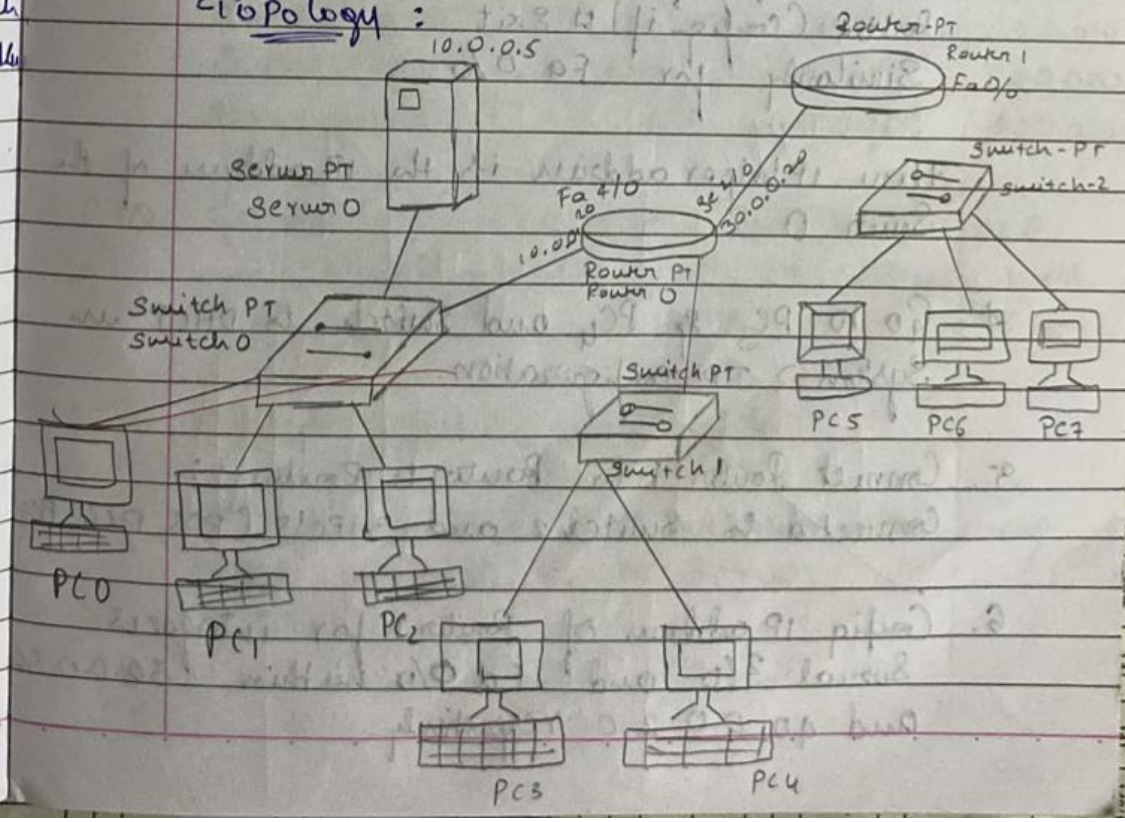
→ PC1 is assigned 10.0.0.2
→ PC2 is assigned 10.0.0.3
→ PC3 is assigned 10.0.0.4

Result

All the PC's are assigned as IP address automatically Dynamically.

AIM: To Configure DHCP outside a LAN

Topology:



Procedure:

1. Repeat the procedure we did for the LAN (DHCP within a LAN).
2. Now add 2 router another set of PC's and switch 1 (LAN 2)
3. Config the IP address set of PORTS FO4/0 and Fa 0/0 Router 0.

Router > Enable
Router # Config terminal
Router Config # interface Fa 4/0
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 PC_3 & PC_4 and switch to DHCP in System → IP configuration.
5. Connect Router 1 to Router 0 Router 1: Connected to Switch 2 and 3 PC's (PC5, PC6, PC7)
6. Config IP address of Router 1 for interfaces Serial 2/0 and Fa 0/0 within 30.0.0.30 and 40.0.0.20 respectively.

Date
Page

Procedure:

1. Repeat the procedure we did for the LAN (DHCP within a LAN).
2. Now add 2 router another set of PC's and Switch 1 (LAN2)
3. Config the IP address set of PORTS FO4/0 and Fa 0/0 Router 0.

Router > Enable
Router # Config terminal
Router Config # interface Fa 4/0
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 PC_3 & PC_4 and switch to DHCP in System → IP configuration.
5. Connect Router 1 to Router 0 Router 1: Connected to Switch 2 and 3 PC's (PC5, PC6, PC7)
6. Config IP address of Router 1 for interfaces Serial 2/0 and Fa 0/0 within 30.0.0.30 and 40.0.0.20 respectively.

Date / /
Page

7. Config IP address of the Router 0 for serial 2/0 as 30.0.0.20

8. perform Static routing for router 0 ->

Router (Config) # IP route
40.0.0.0 255.0.0.0
30.0.0.30.

For Router 1 =>
Router (Config) # IP route 10.0.0.0
255.0.0.0 30.0.0.30.

9. Go to Server and create 2 more Server Pools with different names

~~Server Pool :~~

	Default Gateway	DNS Server	Start IP
Server Pool 1	10.0.0.20	10.0.0.5	10.0.010
Server Pool 2	10.0.0.20	10.0.0.5	200.020
Server pool 3	10.0.0.20	10.0.0.5	40.0.010

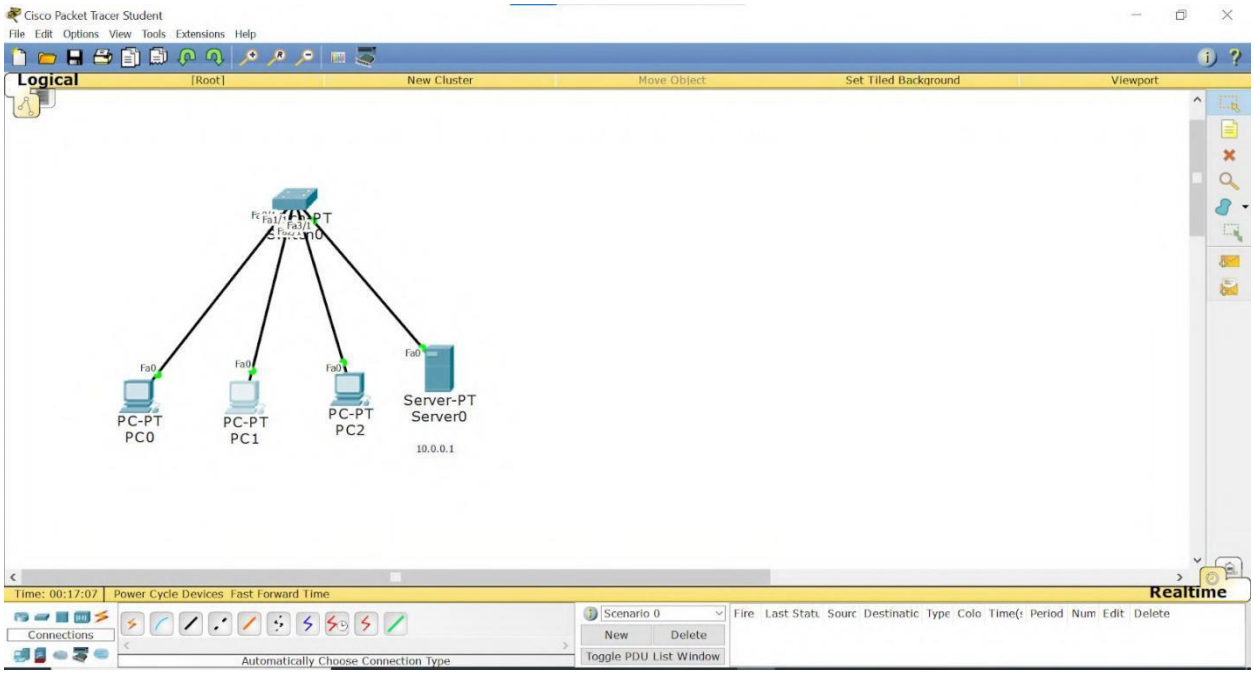
10. Go to PC5, PC6, PC7, Switch to DHCP. in IP Configuration. the IP address will be given to Each PC5.

OBSERVATION / Result:

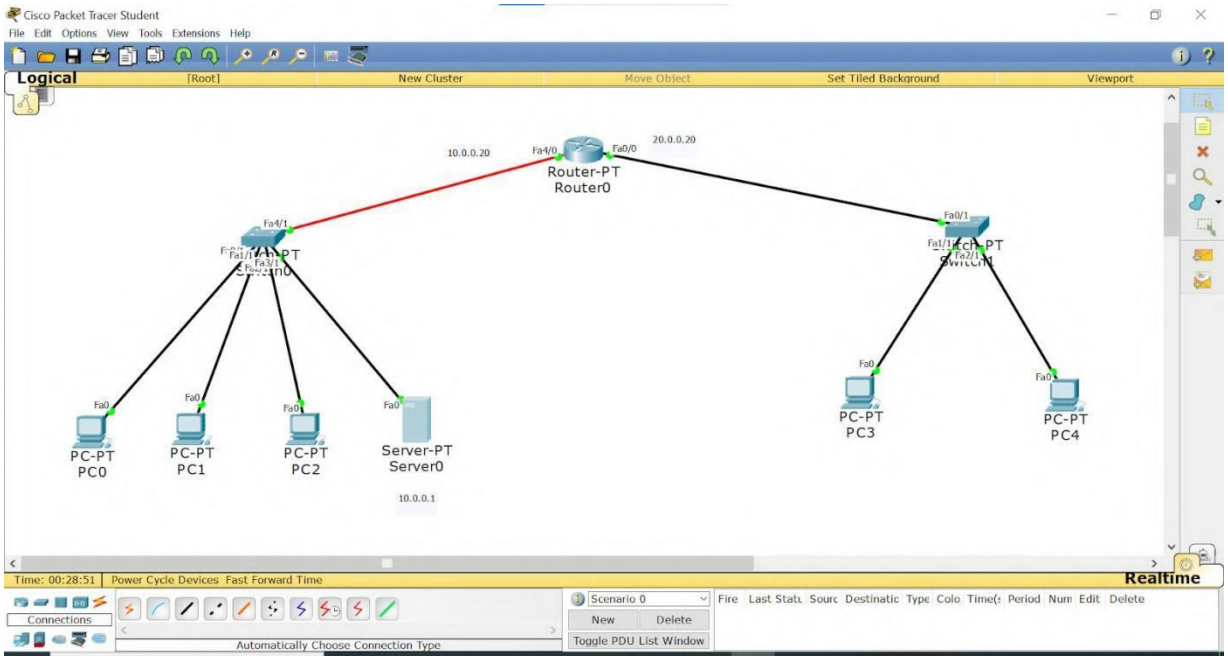
IP address are set automatically using DHCP protocol by the Server.

TOPOLOGY:

PROGRAM 4.1:

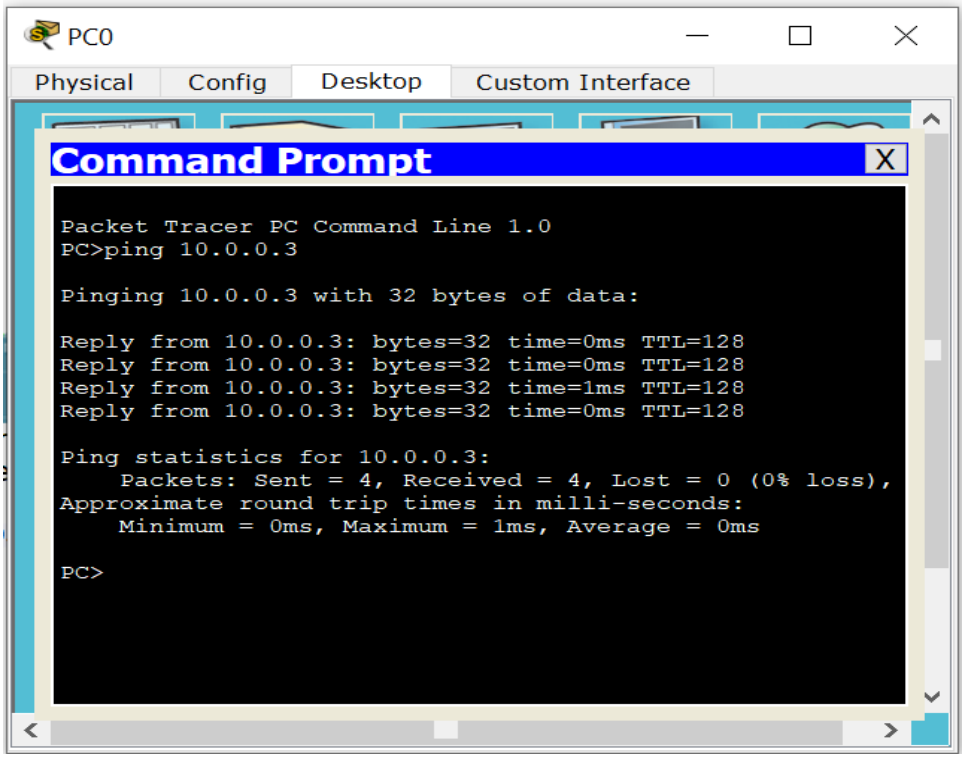


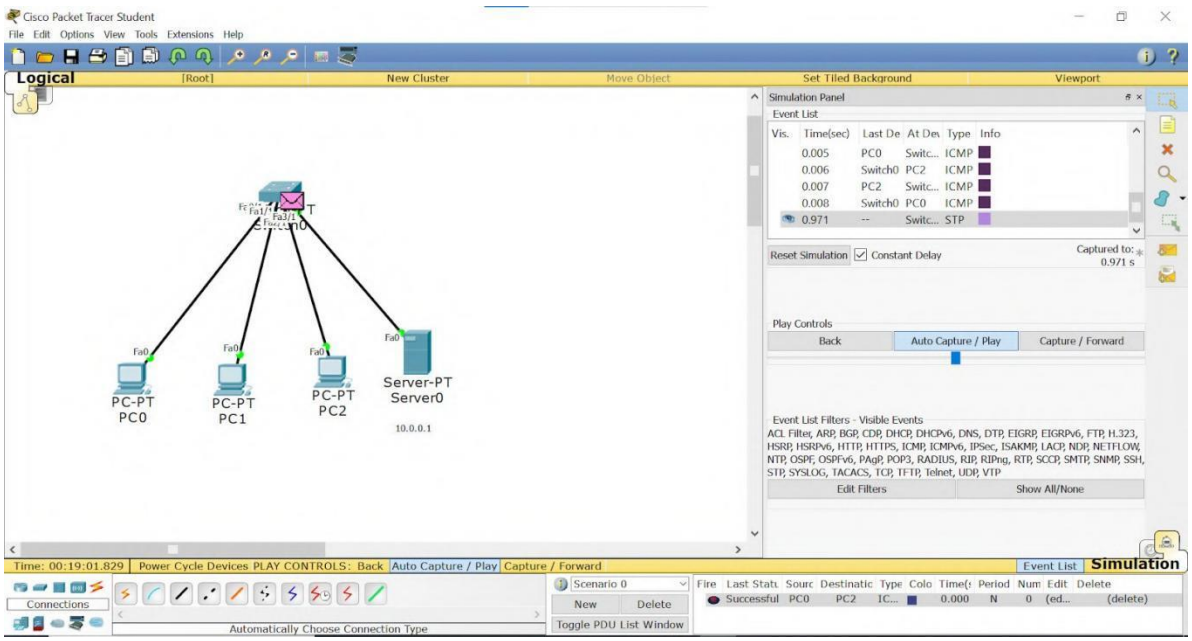
PROGRAM 4.2:



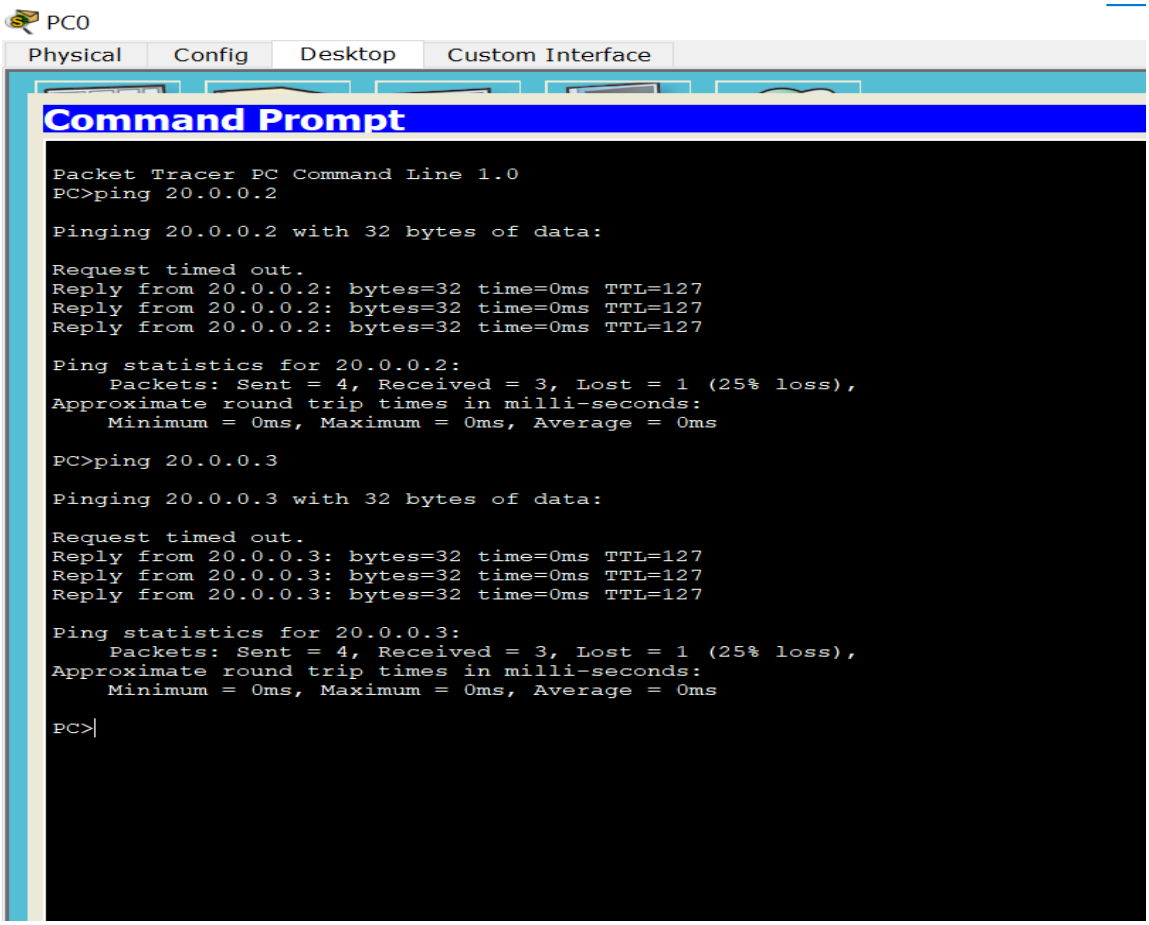
OUTPUT:

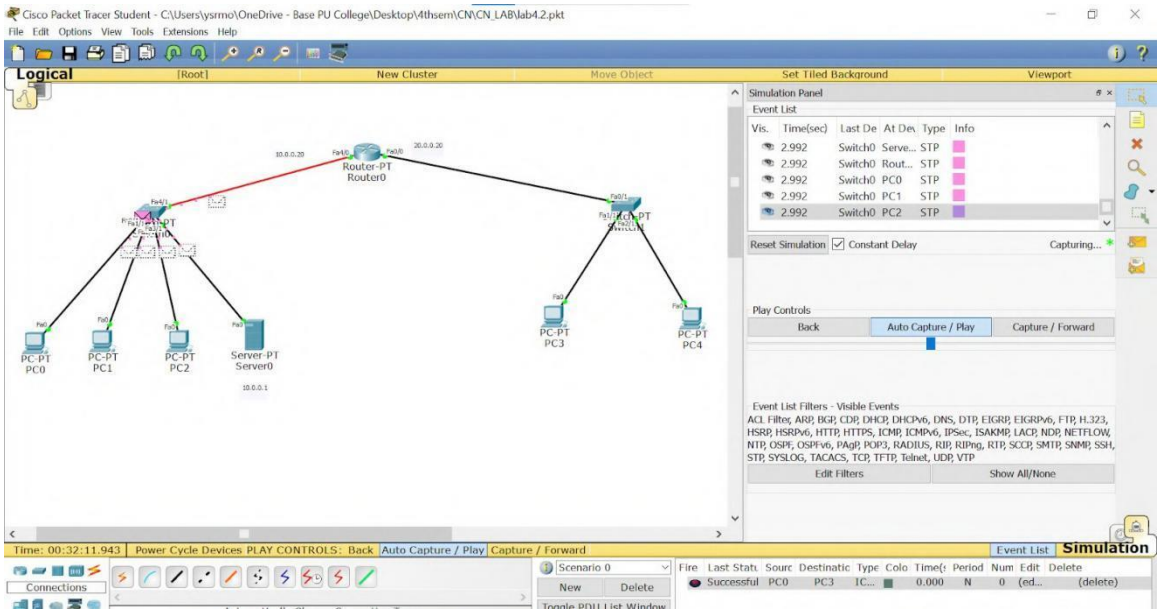
PROGRAM 4.1:





PROGRAM 4.2:

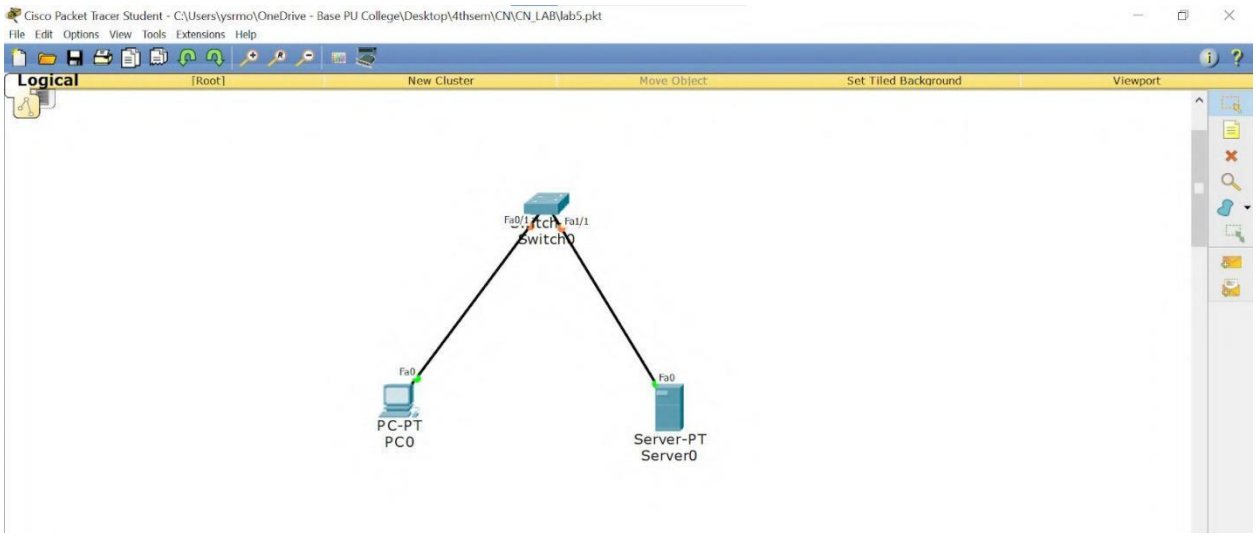

PC0
Physical
Config
Desktop
Custom Interface
Command Prompt
Packet Tracer PC Command Line 1.0
PC>ping 20.0.0.2
Pinging 20.0.0.2 with 32 bytes of data:
Request timed out.
Reply from 20.0.0.2: bytes=32 time=0ms TTL=127
Reply from 20.0.0.2: bytes=32 time=0ms TTL=127
Reply from 20.0.0.2: bytes=32 time=0ms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


Cisco Packet Tracer Student - C:\Users\ysrmo\OneDrive - Base PU College\Desktop\4thsem\CN\CN_LAB\lab4.2.pkt
File Edit Options View Tools Extensions Help
Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
Router-PT
Router0
Server-PT
Server0
PC-PT
PC0
PC-PT
PC1
PC-PT
PC2
PC-PT
PC3
PC-PT
PC4
Simulation Panel
Event List
Vis. Time(sec) Last De At De Type Info
2.992 Switch0 Serve... STP
2.992 Switch0 Rout... STP
2.992 Switch0 PC0 STP
2.992 Switch0 PC1 STP
2.992 Switch0 PC2 STP
Reset Simulation
Constant Delay
Capturing...
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: 00:32:11.943
Power Cycle Devices PLAY CONTROLS: Back Auto Capture / Play Capture / Forward
Event List
Simulation
Connections
Scenario 0
New
Delete
Fire Last Statu Sourc Destinatic Type Colo Time(s Period Num Edit Delete
Successful PC0 PC3 IC... 0.000 N 0 (ed... (delete)

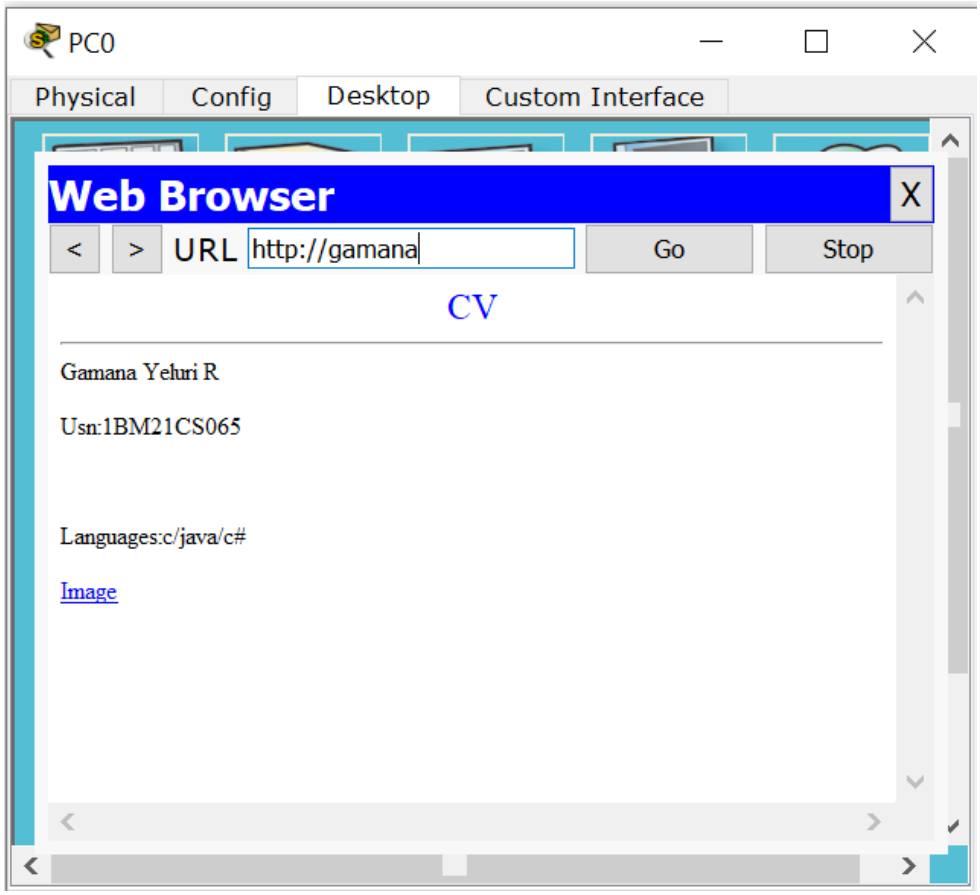
WEEK 5

Configure Web Server, DNS within a LAN.
OBSERVATION:

TOPOLOGY:

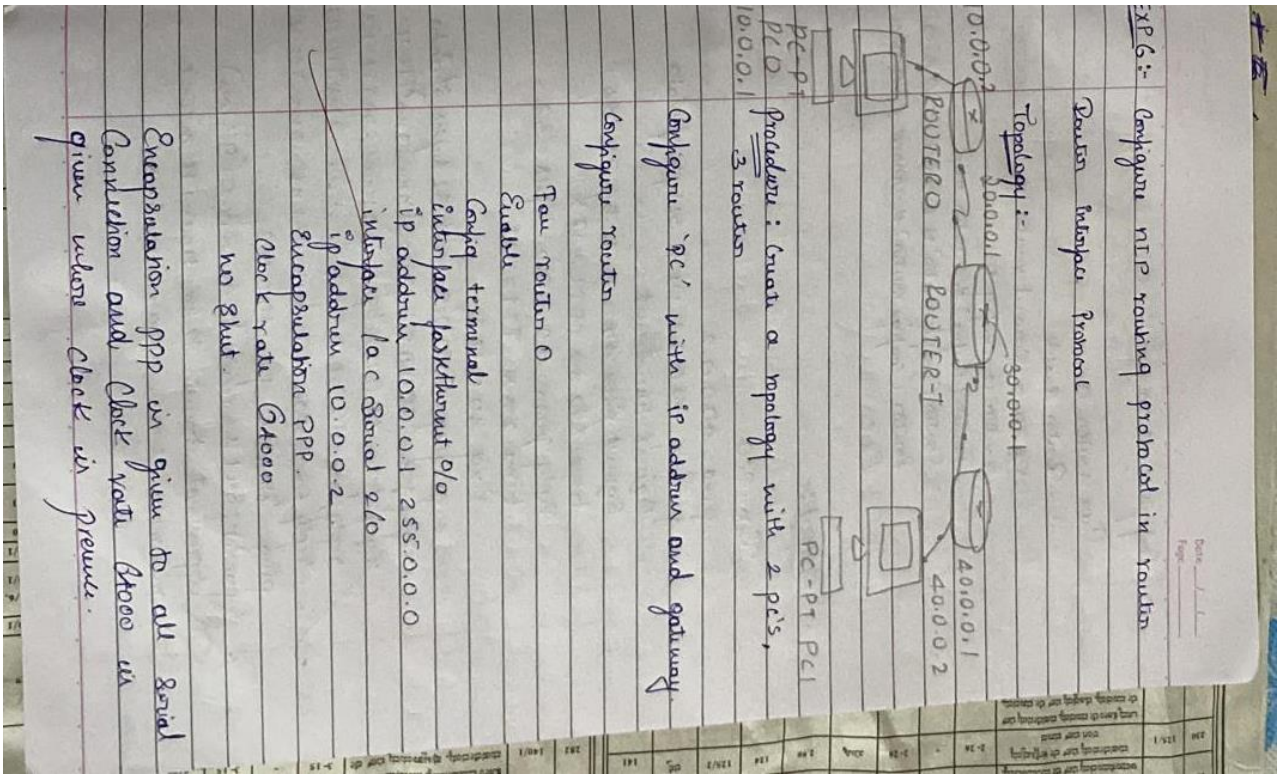


OUTPUT:



WEEK 6

Configure RIP routing Protocol in Routers.
OBSERVATION:



-> Configure all router

For router 0

Router Enable

Router # show id router

Router # Config_t

Router (config) # router rip

Router (config-router) # network 10.0.0.0

Router (config-router) # network 20.0.0.0

Exit

Output :-

In PC0

ping 40.0.0.2

pinging 40.0.0.2 with 32 byte of data

Request timed out

Request from 40.0.0.2 by tes = 32

time = 2 ms TTL = 125

Reply from 40.0.0.2 bytes = 32.

time 2 ms TTL = 125

Ping 40.0.0.2

pinging 40.0.0.2 with 32 bytes of data

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

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

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

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

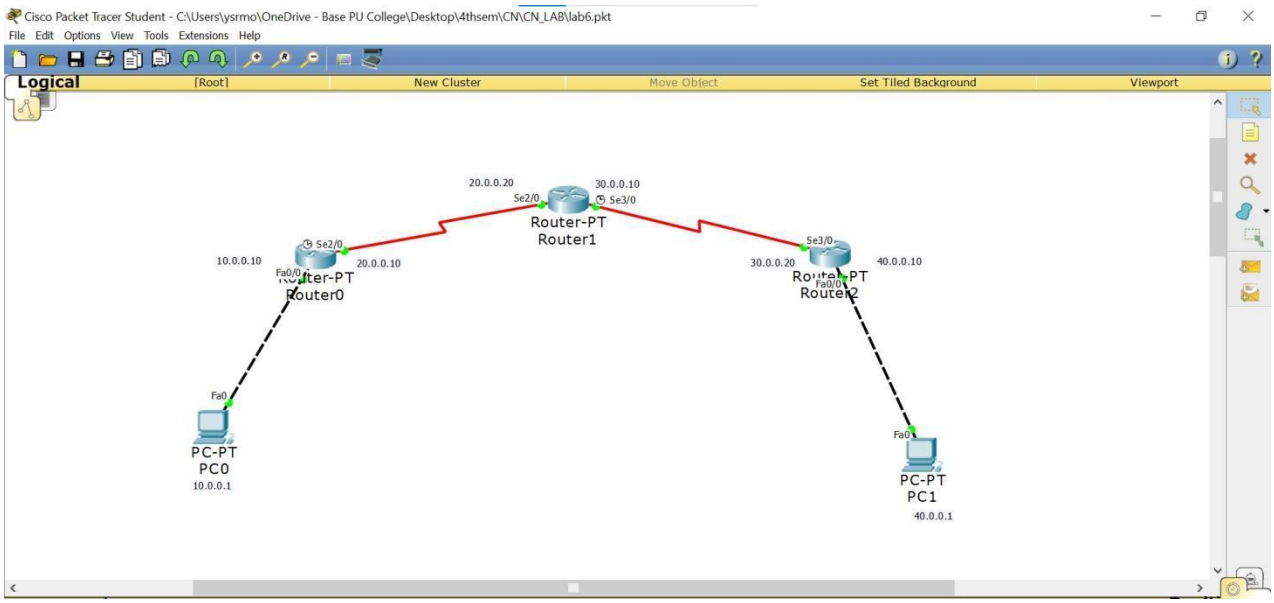
ping statistic for 40.0.0.2

packet: Sent 4 received = 4 lost = 0 (0% loss)

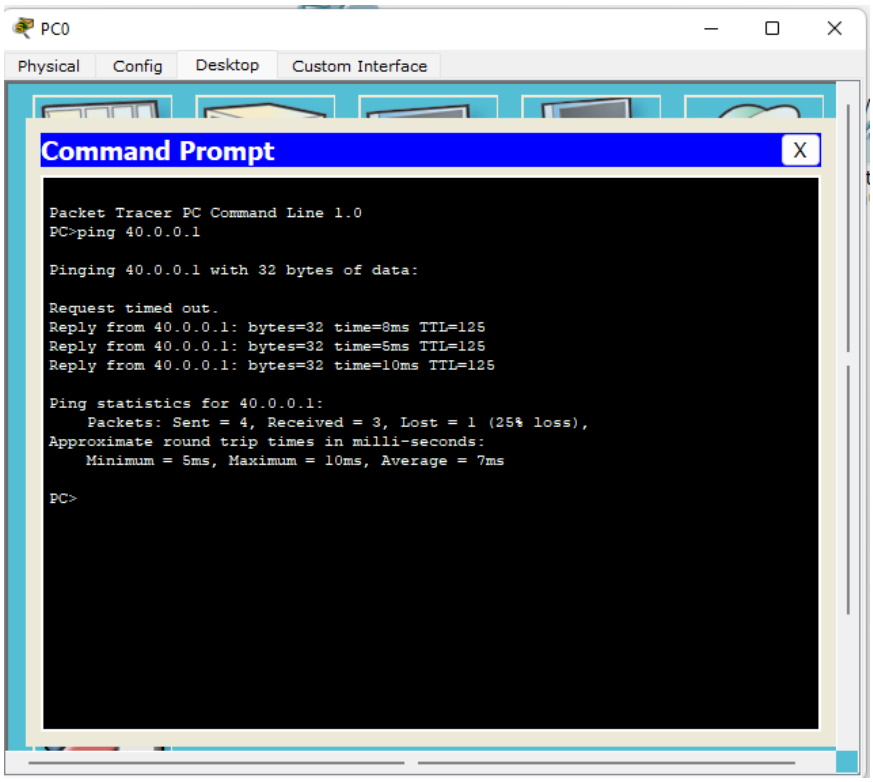
approximate round time times in millisec

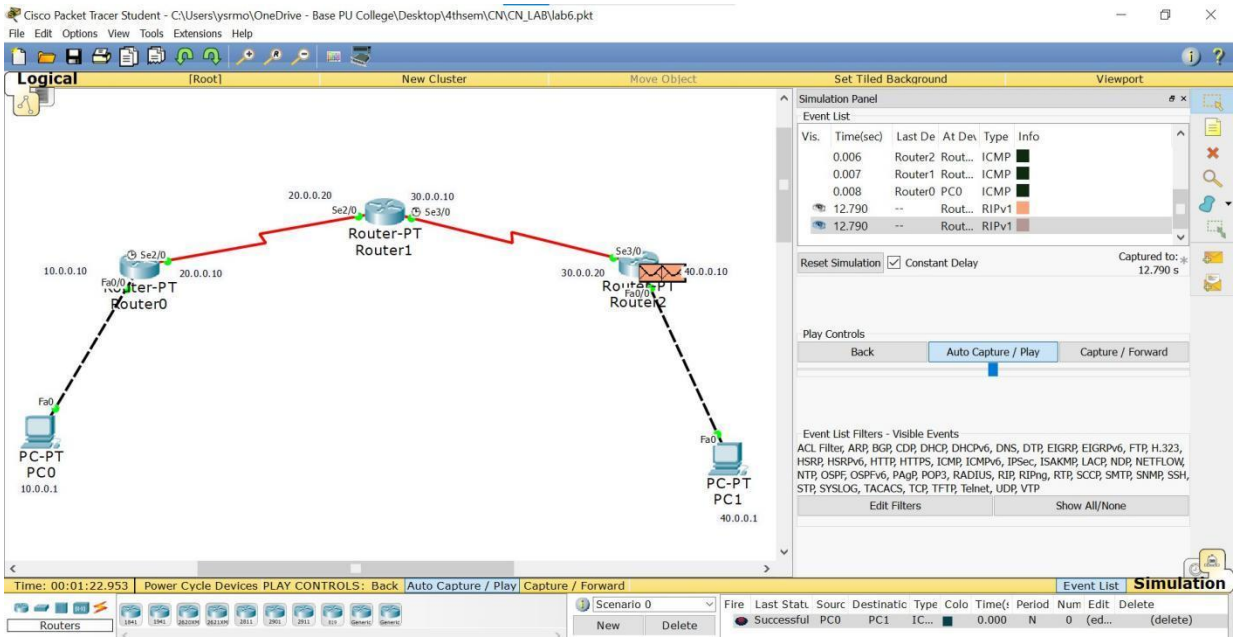
minimum = 2 ms, maximum = 13 ms

TOPOLOGY:



OUTPUT:

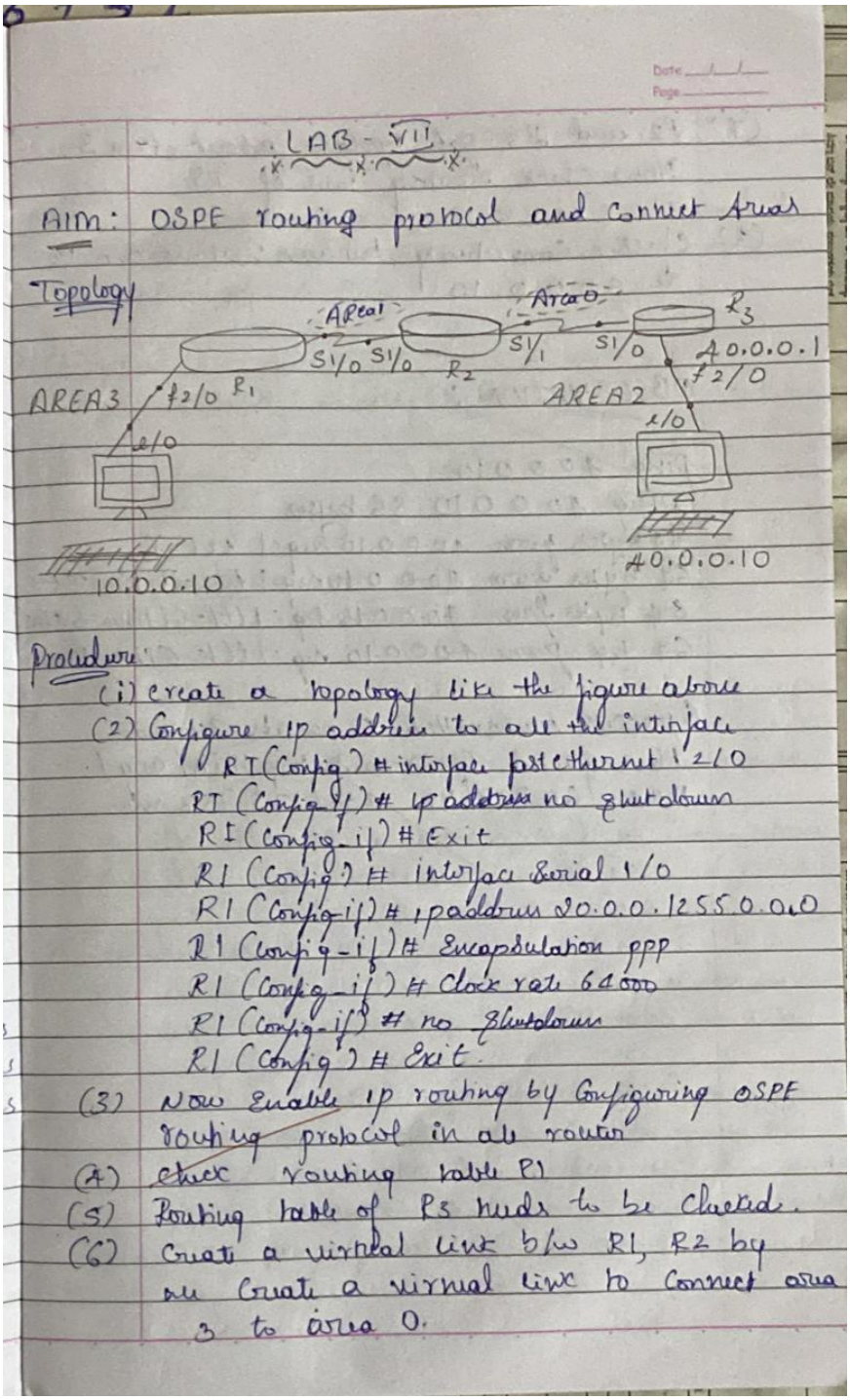



Cisco Packet Tracer Student - C:\Users\ysrmo\OneDrive - Base PU College\Desktop\4thsem\CN\CN_LAB\lab6.pkt
File Edit Options View Tools Extensions Help
Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
Router-PT
Router1
Router-PT
Router0
Router-PT
Router2
PC-PT
PC0
10.0.0.1
PC-PT
PC1
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 De 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)

WEEK 7

Configure OSPF routing protocol.

OBSERVATION:



(7) R2 and R3 get updates about Area 3, Now check routing table of R3

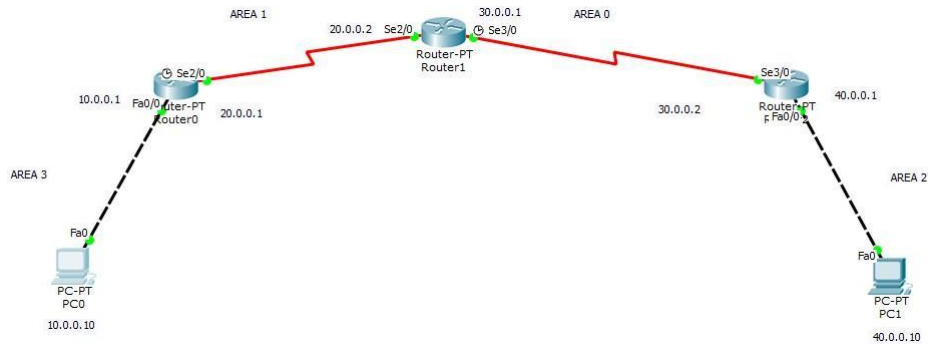
(8) Check Connectivity between host 10.0.0.10 to 40.0.0.10

OBSERVATION :-

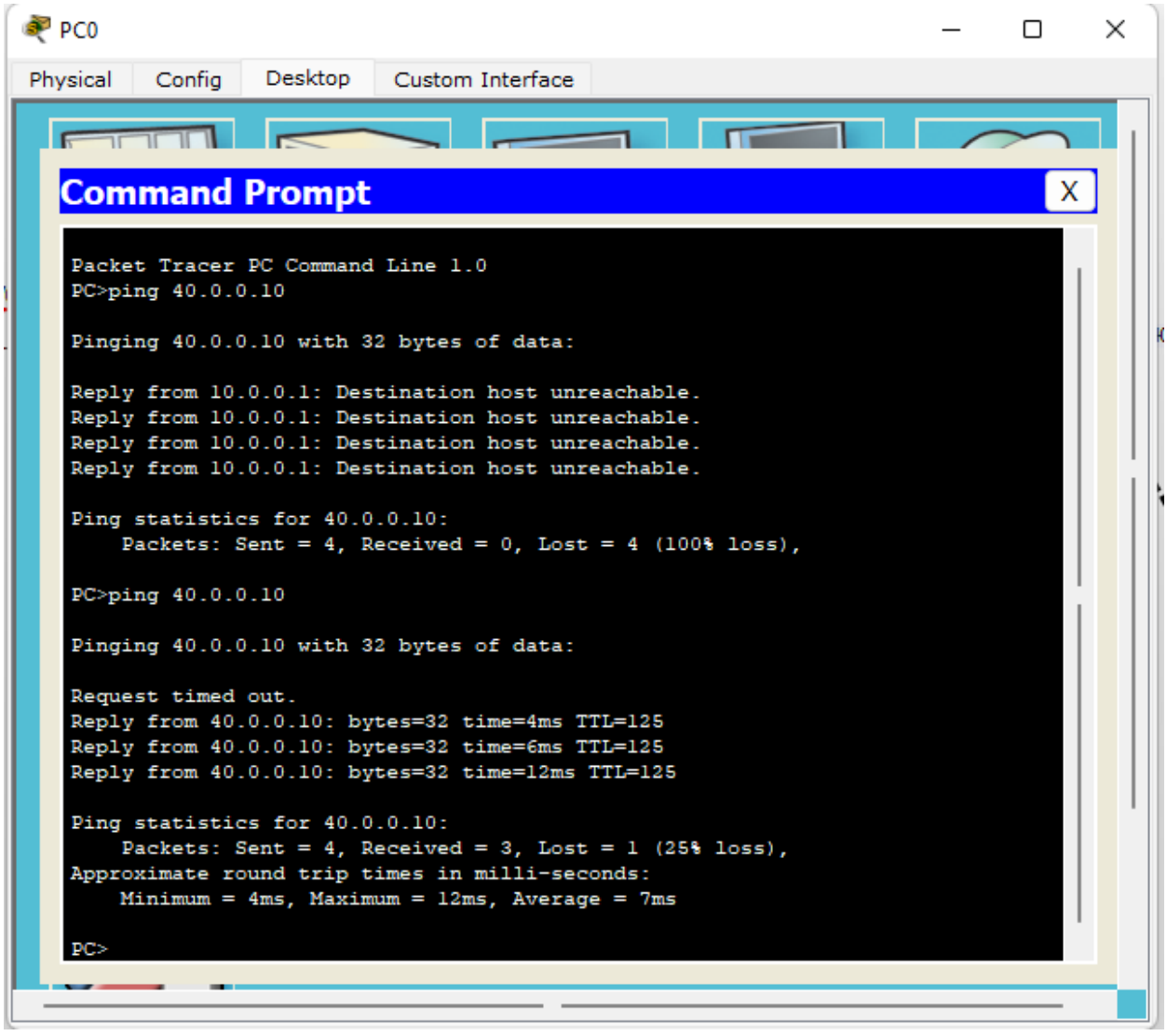
ping 40.0.0.10
ping 40.0.0.10 : 56 bytes
64 bytes from 40.0.0.10 seq=1 ttl=6
64 bytes from 40.0.0.10 seq=1 ttl=61 time 60ms
64 bytes from 40.0.0.10 seq=1 ttl=61 time=64ms
64 by from 40.0.0.10 seq=1 ttl=61 time time=6

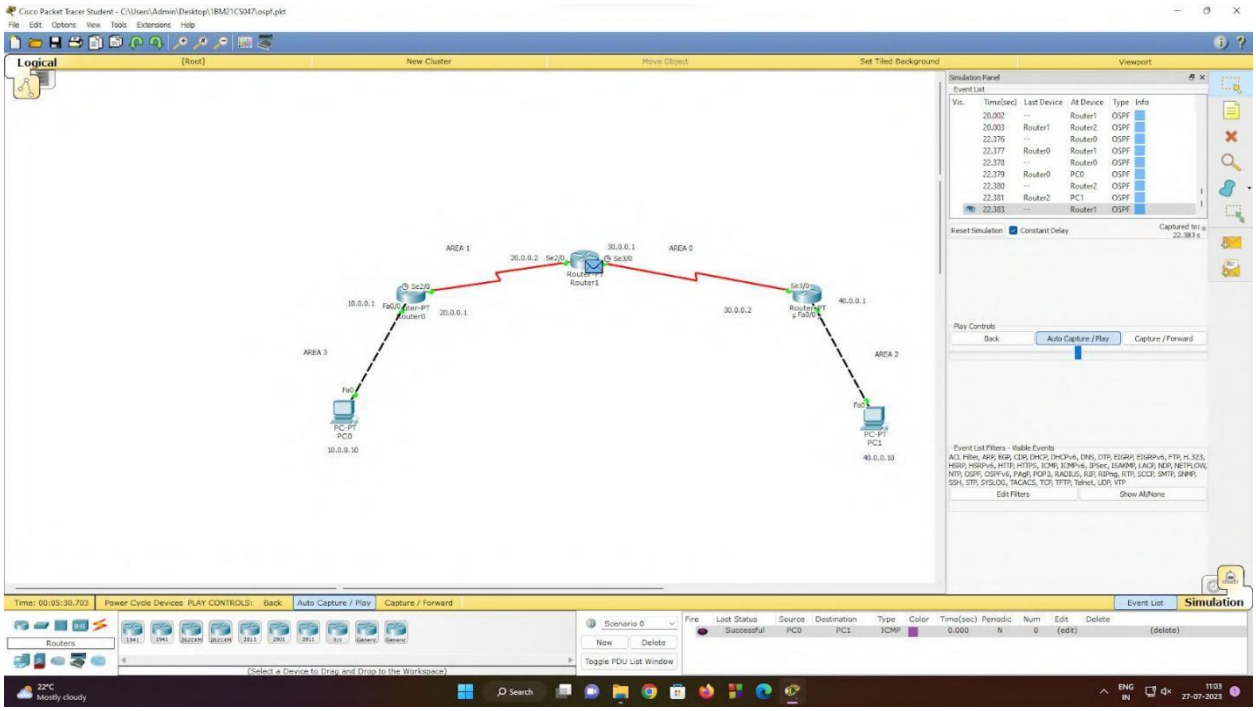
6 packets transmitted 5 packets received
16% packet loss round trip min/avg/
max 60.829/92.438/173.758 ms

4TOPOLOGY:



OUTPUT:

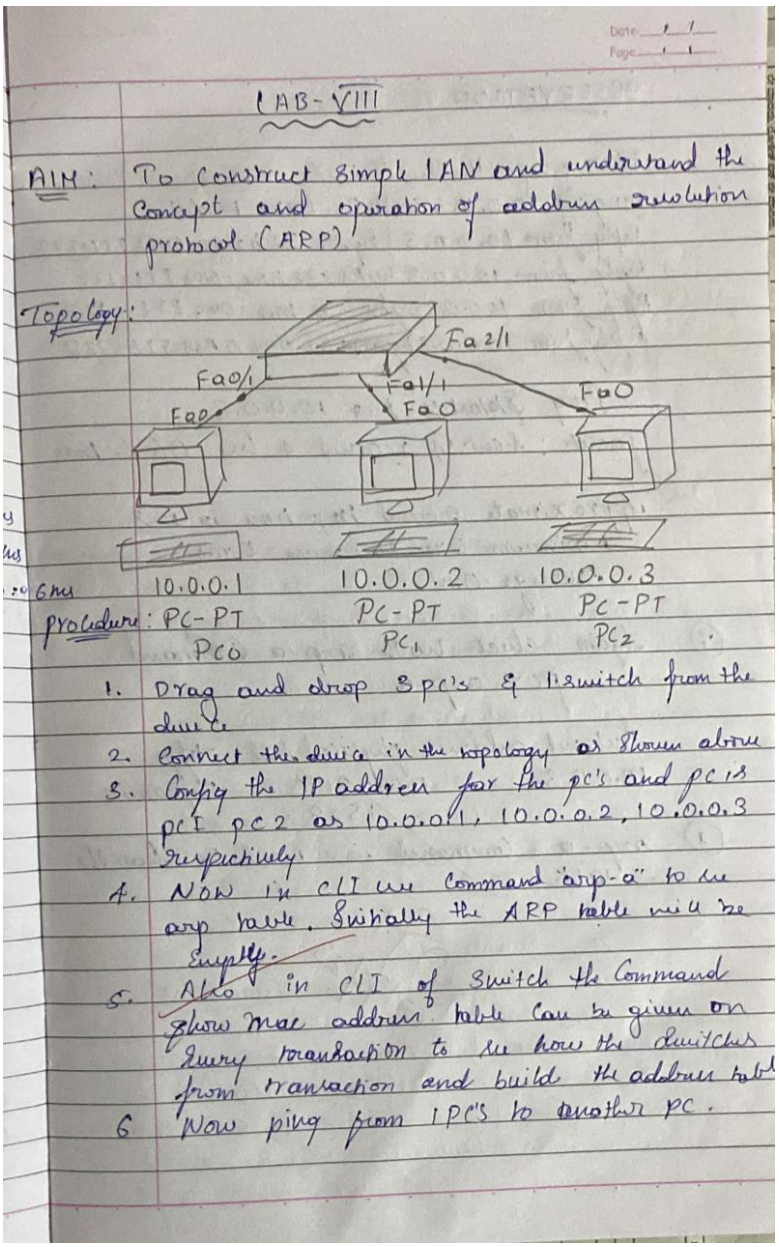



Logical
AREA 1
AREA 0
AREA 3
AREA 2
20.0.0.2
30.0.0.1
20.0.0.1
30.0.0.2
10.0.0.1
40.0.0.1
PC-PT PC0
10.0.0.10
PC-PT PC1
40.0.0.10
Router1
Simulation Panel
Event List
Play Controls
Back
Auto Capture / Play
Capture / Forward
Simulation
Power Cycle Devices
PLAY CONTROLS:
Routers
Scenario 0
New
Delete
Toggle PDU List Window
Successful
PC0
PC1
ICMP
(Select a Device to Drag and Drop to the Workspace)

WEEK 8

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

OBSERVATION:



LAB - VIII

AIM: To construct simple LAN and understand the concept and operation of address resolution protocol (ARP)

Topology:

Procedure:

1. Drag and drop 3 PC's & 1 switch from the devices
2. Connect the devices in the topology as shown above
3. Config the IP address for the pc's and pc is pc1 pc2 as 10.0.0.1, 10.0.0.2, 10.0.0.3 respectively
4. Now in CLI use command "arp -a" to see arp table. Initially the ARP table will be empty.
5. Also in CLI of switch the command show mac address table can be given on query transaction to see how the switches from transaction and build the address table
6. Now ping from PC's to another PC.

OBSERVATION

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 0 ms TTL = 128

Ping Statistics for 10.0.0.3
packets: Sent = 4 received = 4 lost = 0 (0% loss)

Approximate round trip time in ms
Minimum = 0ms Maximum = 0 ms
Average = 0ms

(7) Again check with arp-a Command
pc > arp-a

Internet address physical Add Type
10.0.0.3 0-0d0-217c Dynamic
1589

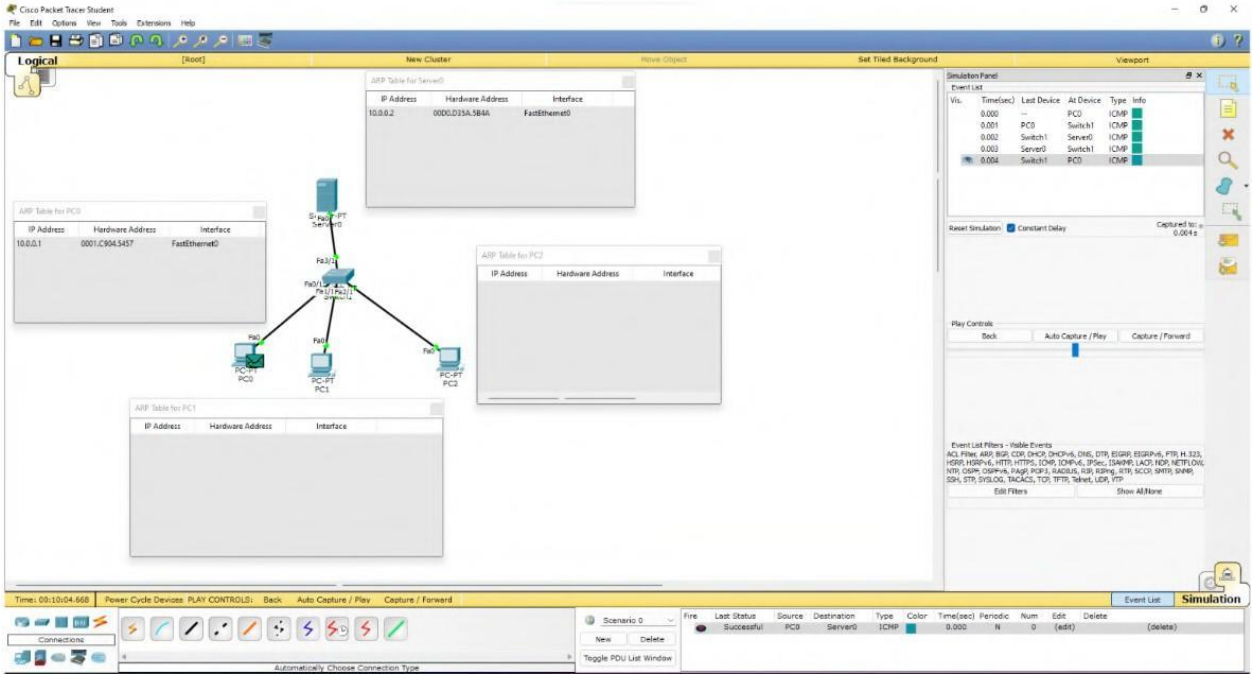
(8) arp-a . Commands is used to Clear the table

AIM

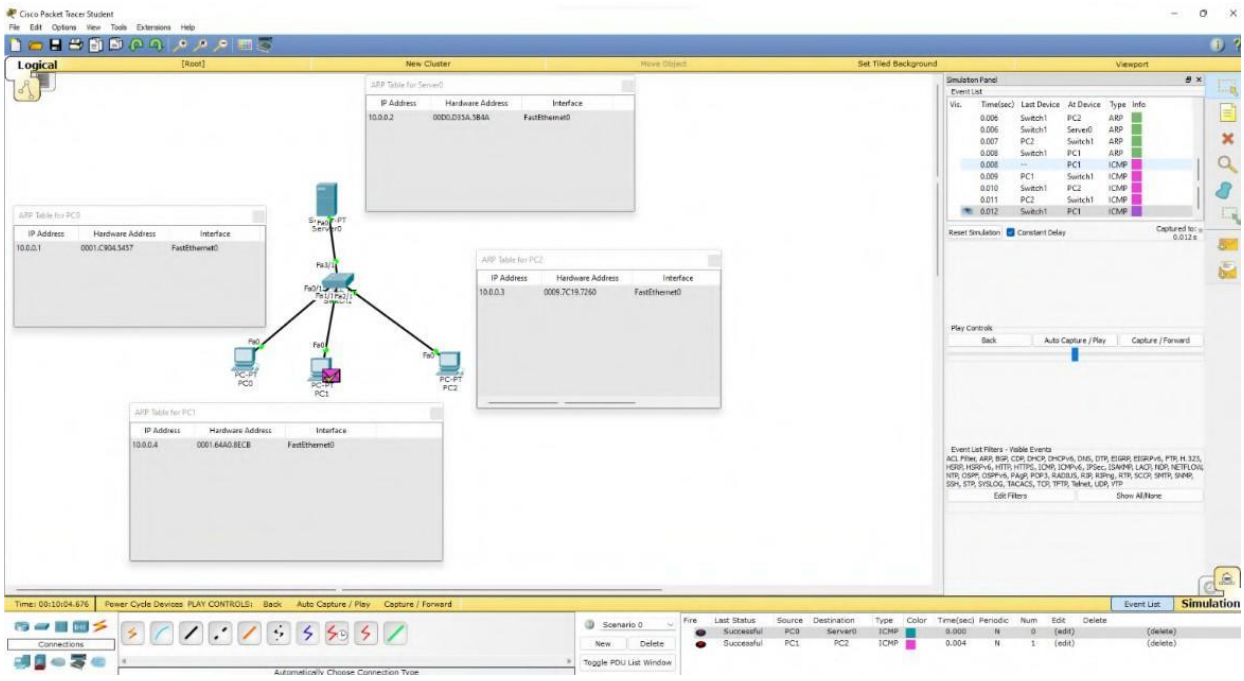
Top

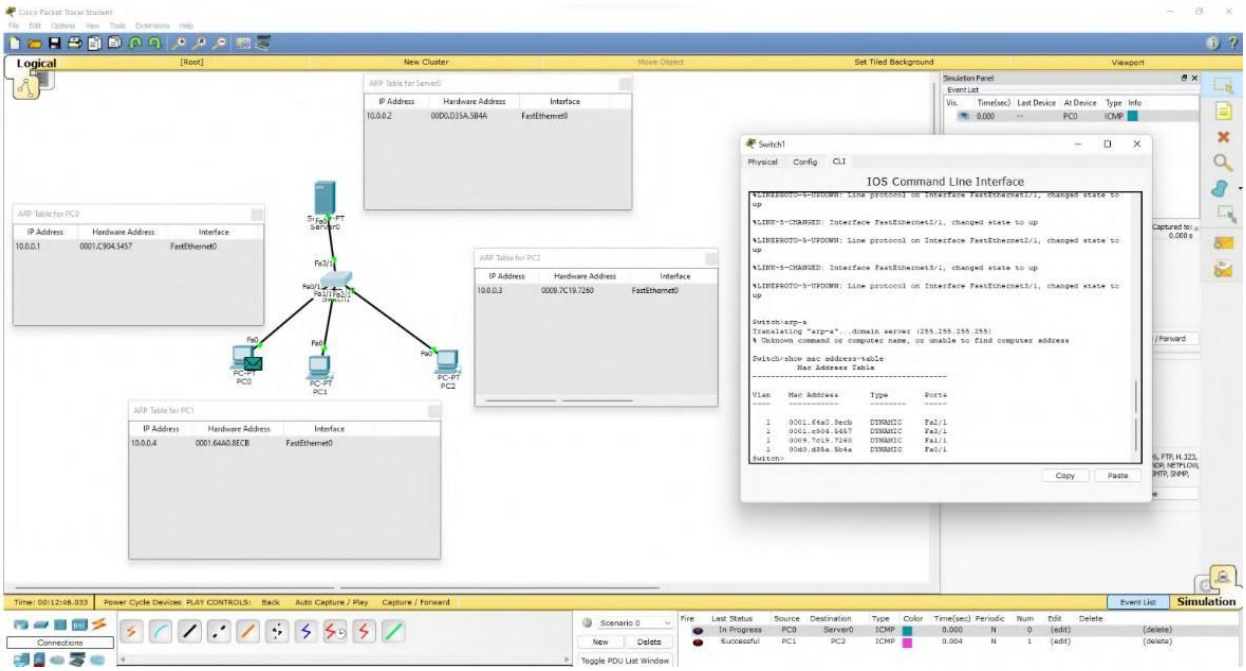
pro

TOPOLOGY:



OUTPUT:

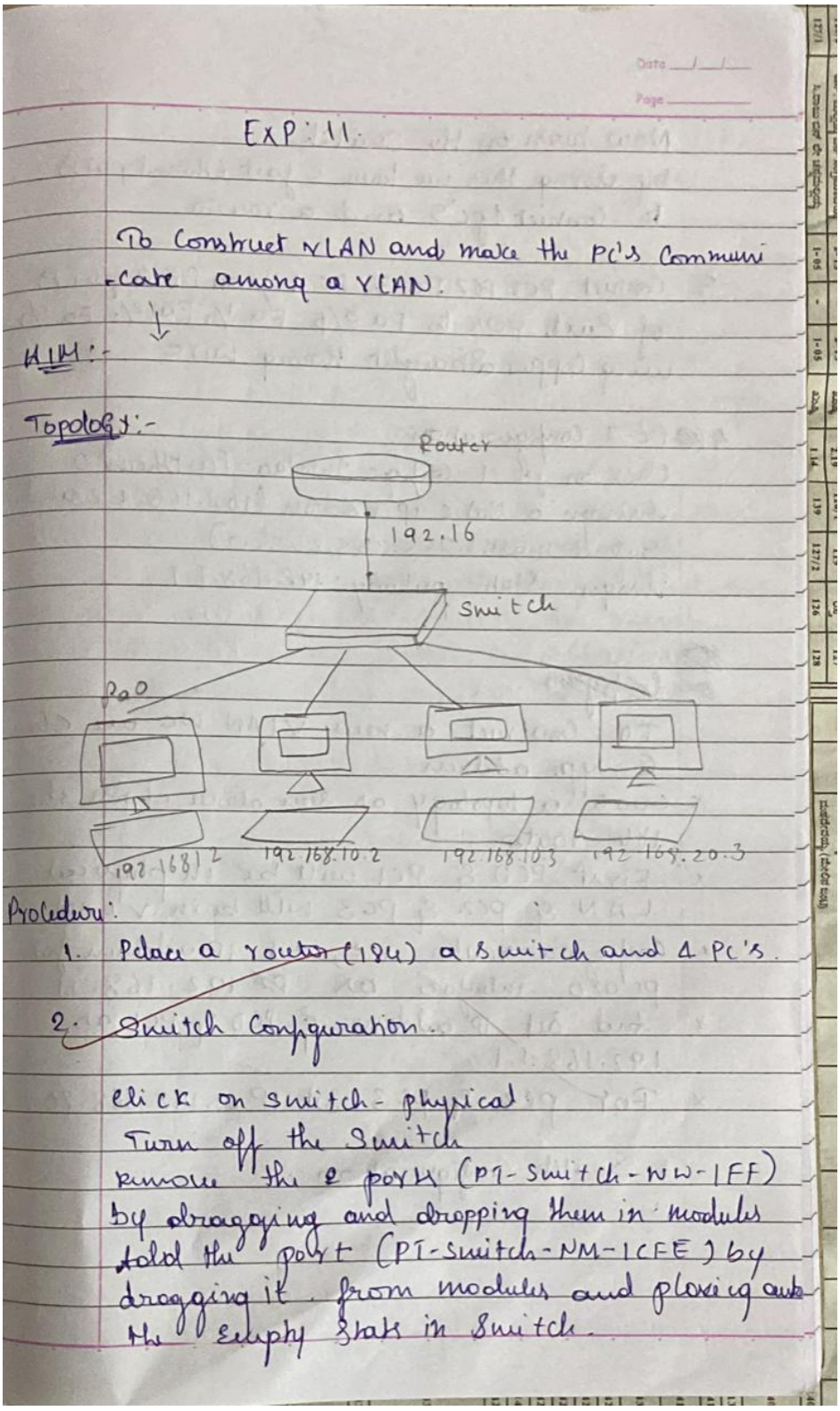



Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
ARP Table for Server0
IP Address
Hardware Address
Interface
10.0.0.2
00D0.D35A.5B4A
FastEthernet0
ARP Table for PC0
10.0.0.1
0001.C904.5457
FastEthernet0
ARP Table for PC2
10.0.0.3
0009.7C19.7260
FastEthernet0
ARP Table for PC1
10.0.0.4
0001.64A0.8ECB
FastEthernet0
Switch1
Physical
Config
CLI
IOS Command Line Interface
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet1/1, changed state to up
%LINK-5-CHANGED: Interface FastEthernet1/1, 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 Fa1/1
1 0009.7c19.7260 DYNAMIC Fa3/1
1 00d0.d35a.5b4a DYNAMIC Fa0/1
Switch>
Copy
Paste
Simulation Panel
Event List
Vis.
Time(sec)
Last Device
At Device
Type
Info
0.000
PC0
ICMP
Time: 00:12:48.033
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)

WEEK 9

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

OBSERVATION:



Now turn on the Switch
by doing this we have 5 fast Ethernet ports
to connect 4pc's and a router.

3. Connect PC1 PC2 PC-3, PC-4 from FE 0 port
of Each pc's to Fa 0/1 Fa 1/1 Fa 2/1 Fa 3/1
using Copper Straight through wire.

4. Pc-1 Configuration:
Click on pc-1 - Config - Interface - Fast Ethernet 0
Assign a Static IP address (102.168.1.2 and
Subnet mask: 255.255.255.0)
Assign Static gateway: 192.168.1.1

* ~~Configure~~
TO Construct a new VLAN we use Class
C type address
× Create a topology as seen above choose the
184 router
× First PC0 & PC1 will be in physical
LAN & PC2 & PC3 will be in VLAN
× Configure router i.e, set IP address for
pc0/0 interface as ~~PE~~ 192.168.1.1
* And Set IP address of PC0 & PC1 as
192.168.1.1
× For PC2 & PC3 Set IP as 192.168.20.2

* Switch Configuration:-

In Switch go to Config & Select VLAN database. Put VLAN no and name.
Ex:- VLAN NUMBER 20
x click on add
x Select the interface i.e, fa 0/1 (near the switch from router & make it trunk
x VLAN ~~tr~~. allows switch to forward frames from different VLANS over a single link. called trunk.
This is done by adding an additional header information called tag to the Ethernet is called VLAN tagging.
and make (Select) the interface that are Connecting VLAN PC'S to the switch
x Here it is fa 2/1 & 3/1 & Select & make VLAN as 20: NEW VLAN.

Router Configuration.

* Open Config Select VLAN database Enter the no & name of VLAN Created
Go to CLI

Router (VLAN) # exit
Apply Completed
Exiting
Routing #Config t
Routing (Config) # interface fa 0/0
Router (Config-Subif) # Encapsulation
Router (Config-subif) # ip address 192.168.20.1
255.255.255.0
Router (Config-Subif) # no shutdown.

-> Now Ping

from PC0 to PC3

x Ping 192.168.20.3

you will get a succesful transmission from PC0 to PC3

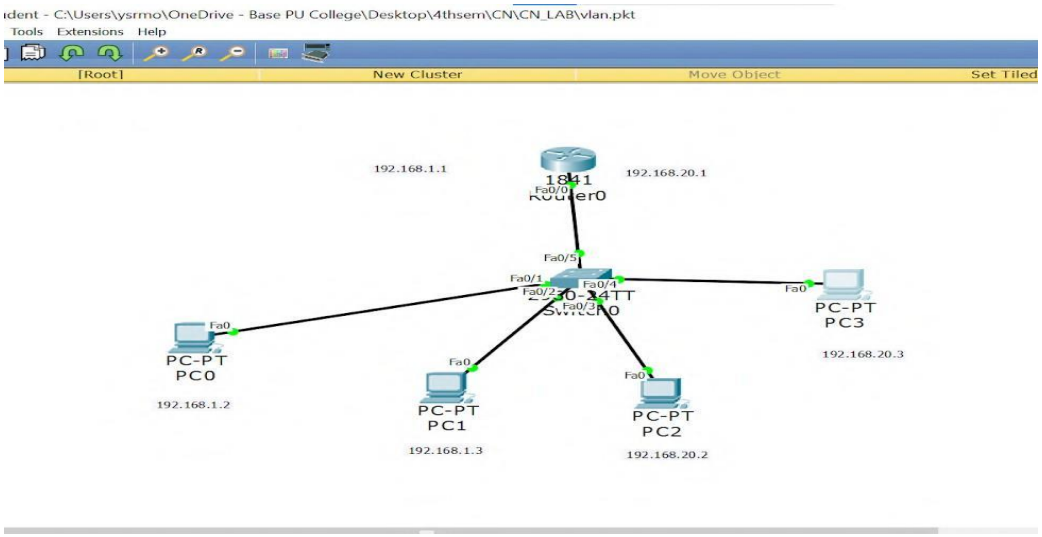
Observation :-

Even though we are using a single router we can multiple different network and those those networks will works as virtual networks and we can communicate from physical LAN to VLAN & vice versa.

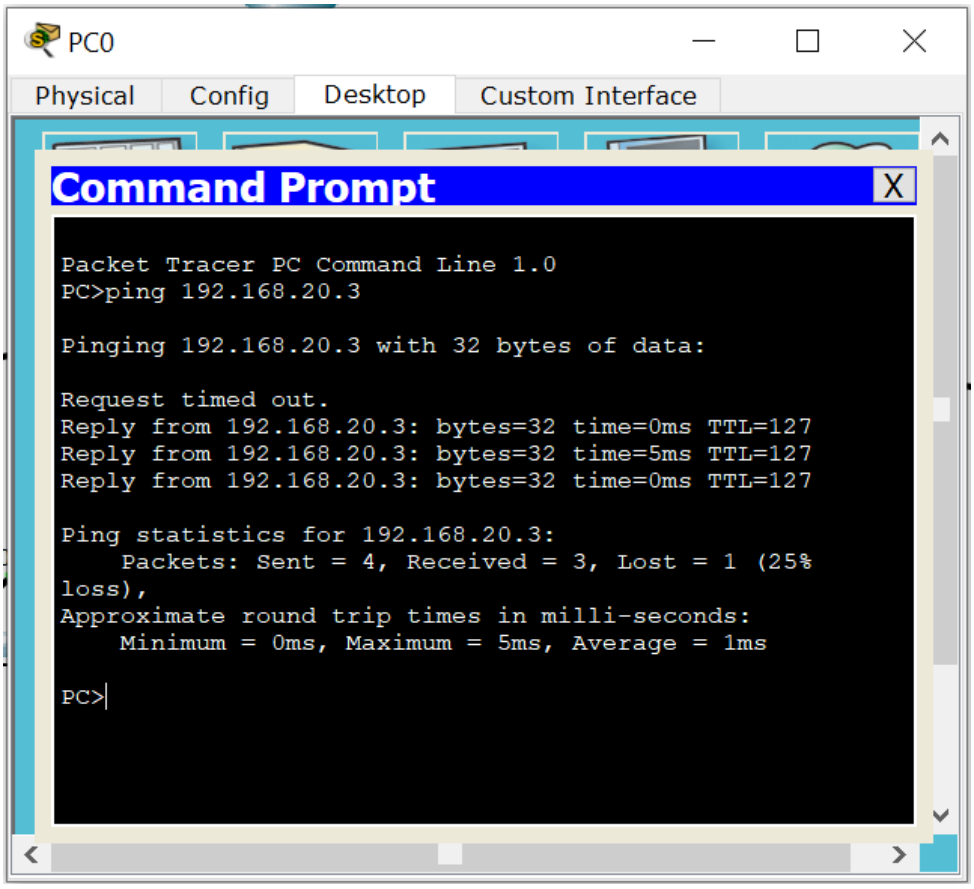
Sp.1

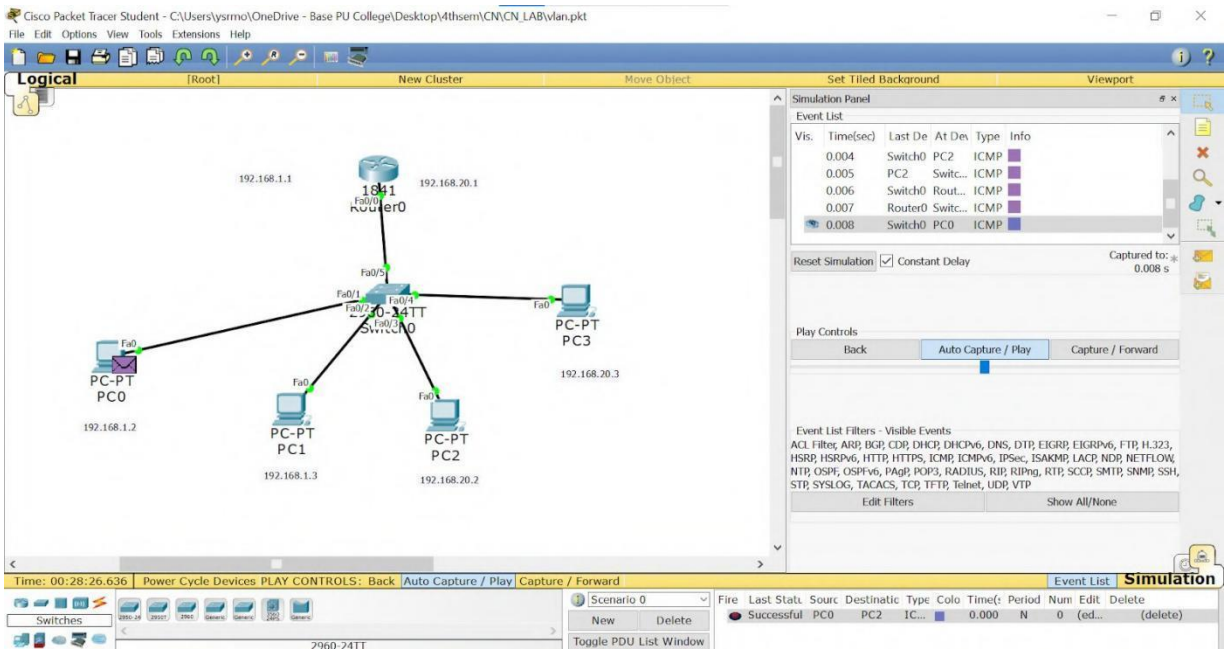
1/9/23

TOPOLOGY:


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
192.168.20.1
Fa0/0
Fa0/5
Fa0/1
Fa0/4
Fa0/2
Fa0/3
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

OUTPUT:

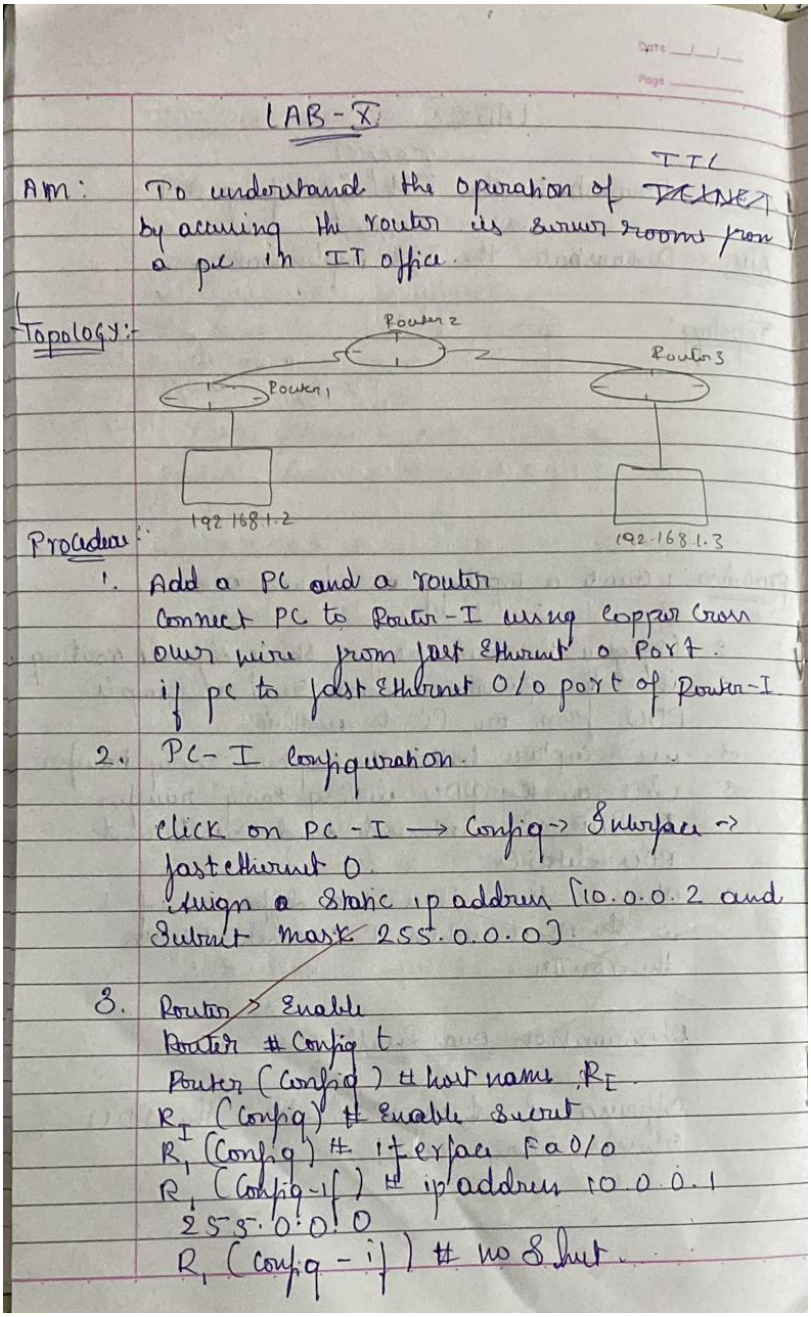

PC0
Physical
Config
Desktop
Custom Interface
Command Prompt
X
Packet Tracer PC Command Line 1.0
PC>ping 192.168.20.3
Pinging 192.168.20.3 with 32 bytes of data:
Request timed out.
Reply from 192.168.20.3: bytes=32 time=0ms TTL=127
Reply from 192.168.20.3: bytes=32 time=5ms TTL=127
Reply from 192.168.20.3: 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>


Logical
PC-PT PC0
PC-PT PC1
PC-PT PC2
PC-PT PC3
192.168.1.1
192.168.20.1
192.168.1.2
192.168.1.3
192.168.20.2
192.168.20.3
Simulation Panel
Event List
Reset Simulation
Constant Delay
Play Controls
Back
Auto Capture / Play
Capture / Forward
Edit Filters
Show All/None
Simulation

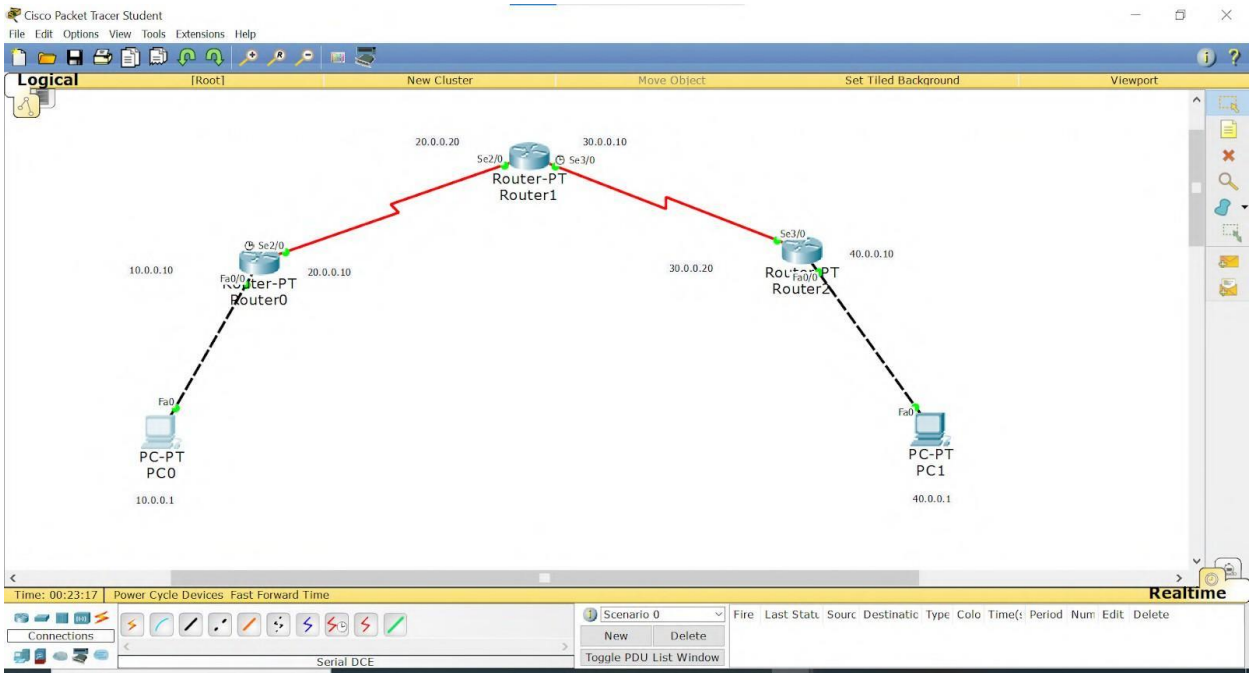
WEEK 10

Demonstrate the TTL/ Life of a Packet.

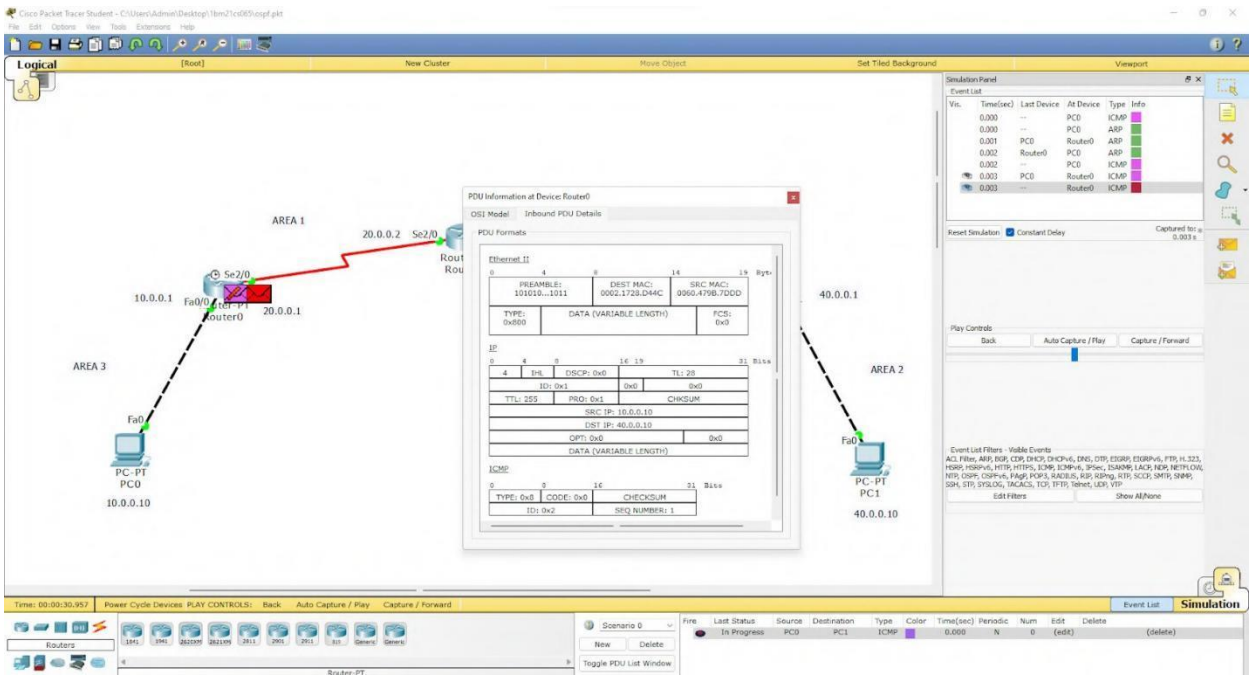
OBSERVATION:

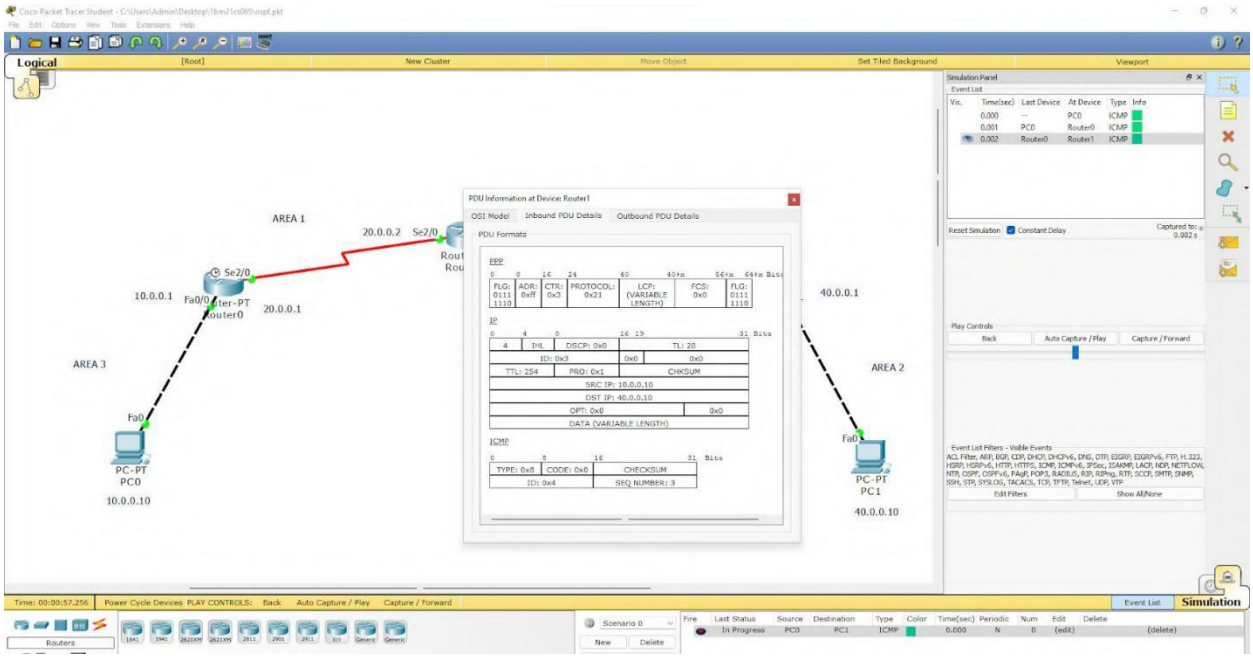


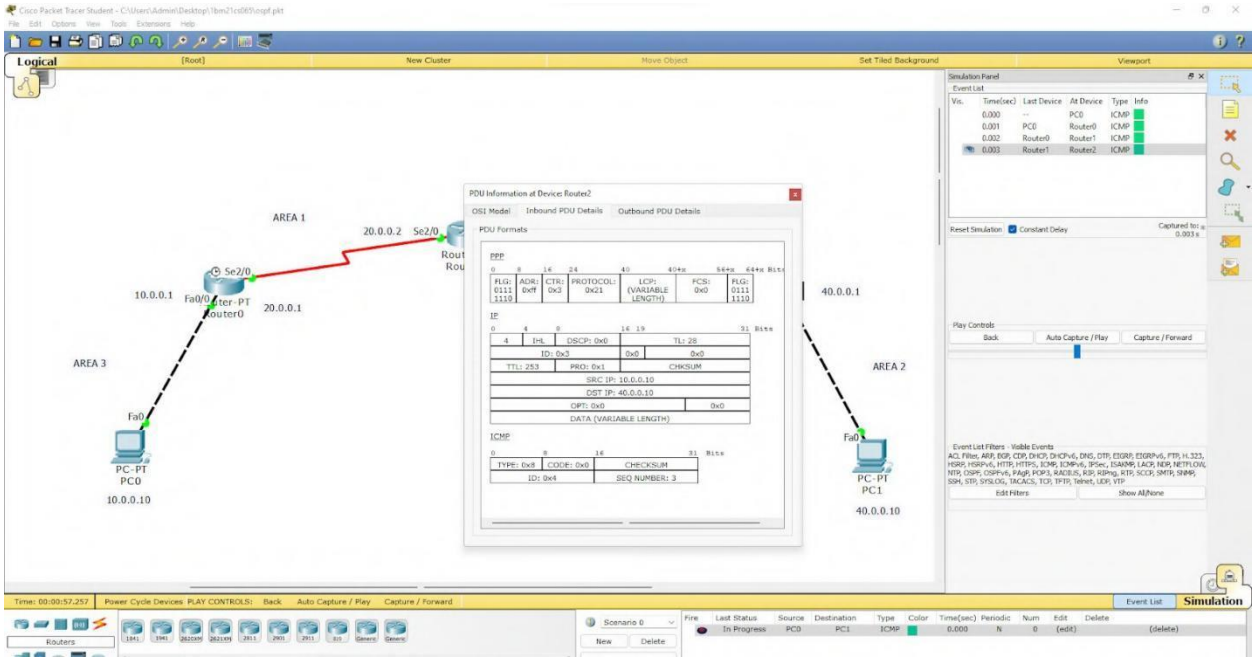
TOPOLOGY:

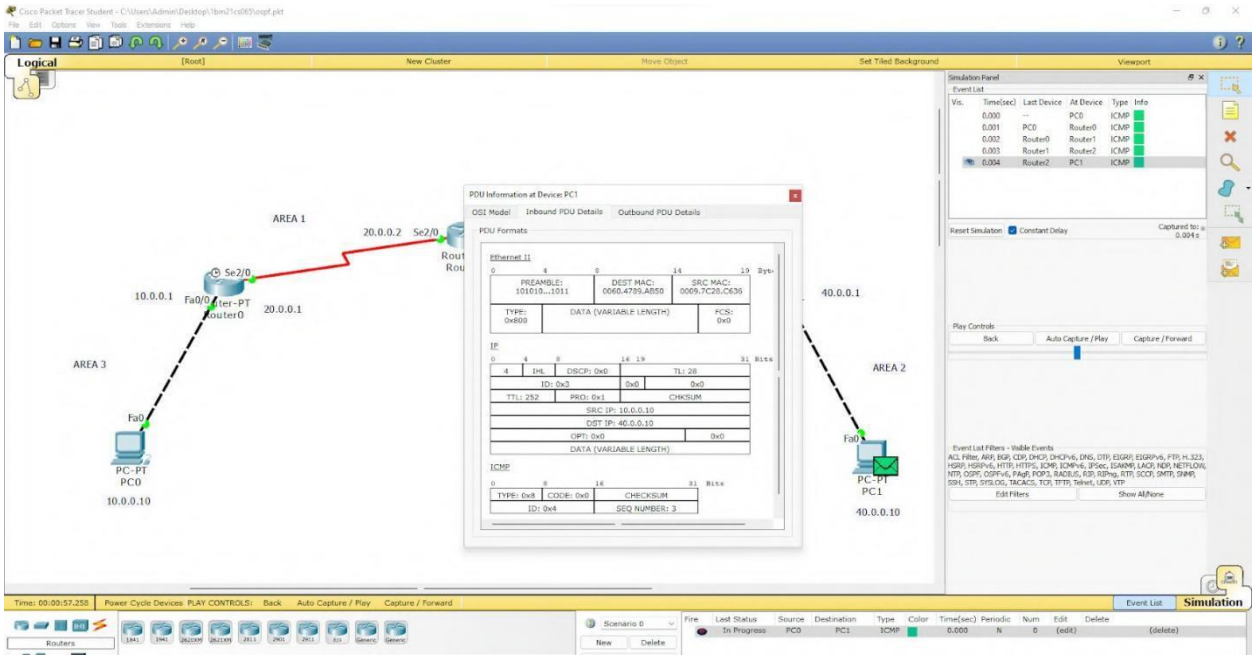


OUTPUT:




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
SRC IP: 10.0.0.10
DST IP: 40.0.0.10
DATA (VARIABLE LENGTH)
TYPE: 0x8
CODE: 0x0
CHECKSUM
SEQ NUMBER: 3
Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
Simulation Panel
Event List
Simulation

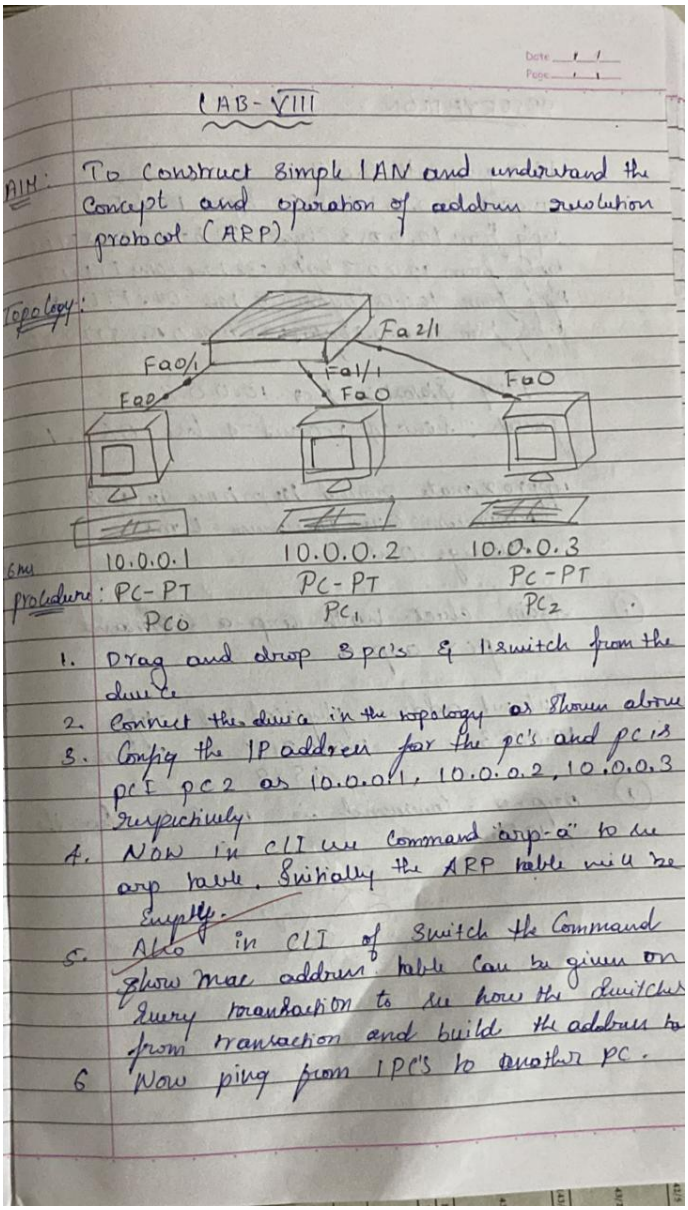

AREA 1
20.0.0.2
Se2/0
10.0.0.1
Fa0/0
20.0.0.1
AREA 3
AREA 2
40.0.0.1
Fa0
PC-PT
PC0
10.0.0.10
PC-PT
PC1
40.0.0.10
PDU Information at Device Router2
OSI Model
Inbound PDU Details
Outbound PDU Details
PDU Formats
PPP
IP
ICMP
SRC IP: 10.0.0.10
DST IP: 40.0.0.10
DATA (VARIABLE LENGTH)
TYPE: 0x8
CODE: 0x0
CHECKSUM
SEQ NUMBER: 3
Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
Simulation Panel
Event List
Simulation


Logical
AREA 1
20.0.0.2
Se2/0
10.0.0.1
Fa0/0
Router-PT
Router0
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
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
ID: 0x4
SEQ NUMBER: 3
Simulation Panel
Event List
Play Controls
Back
Auto Capture / Play
Capture / Forward
Simulation

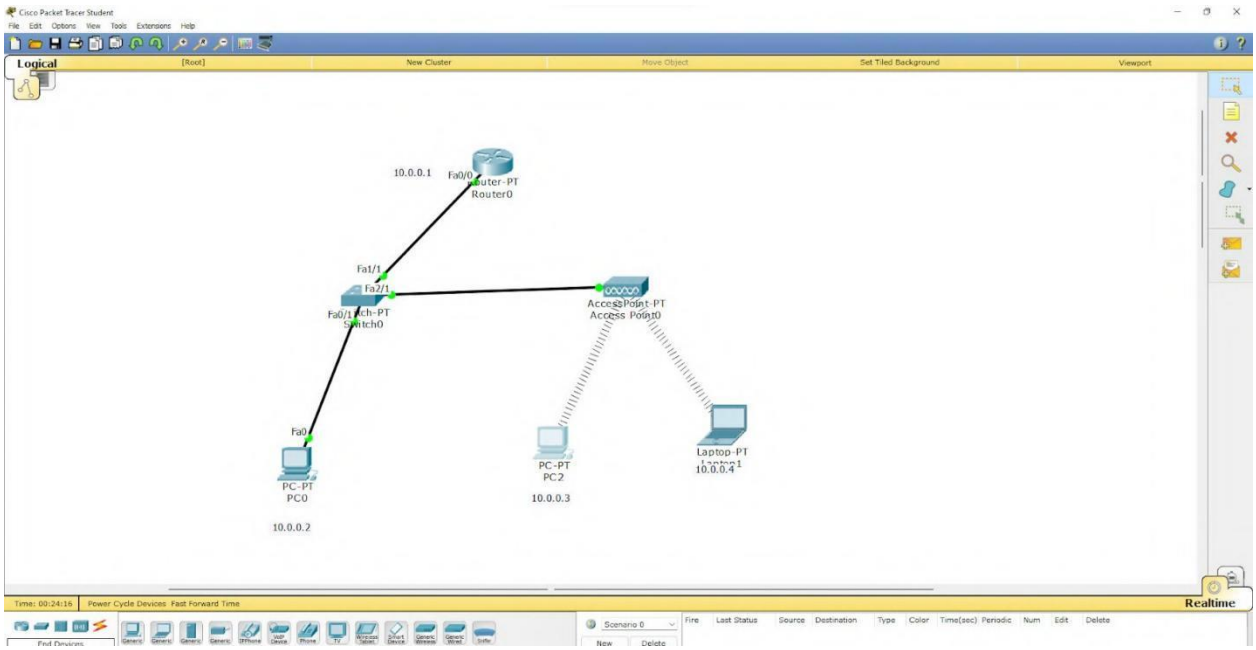
WEEK 11

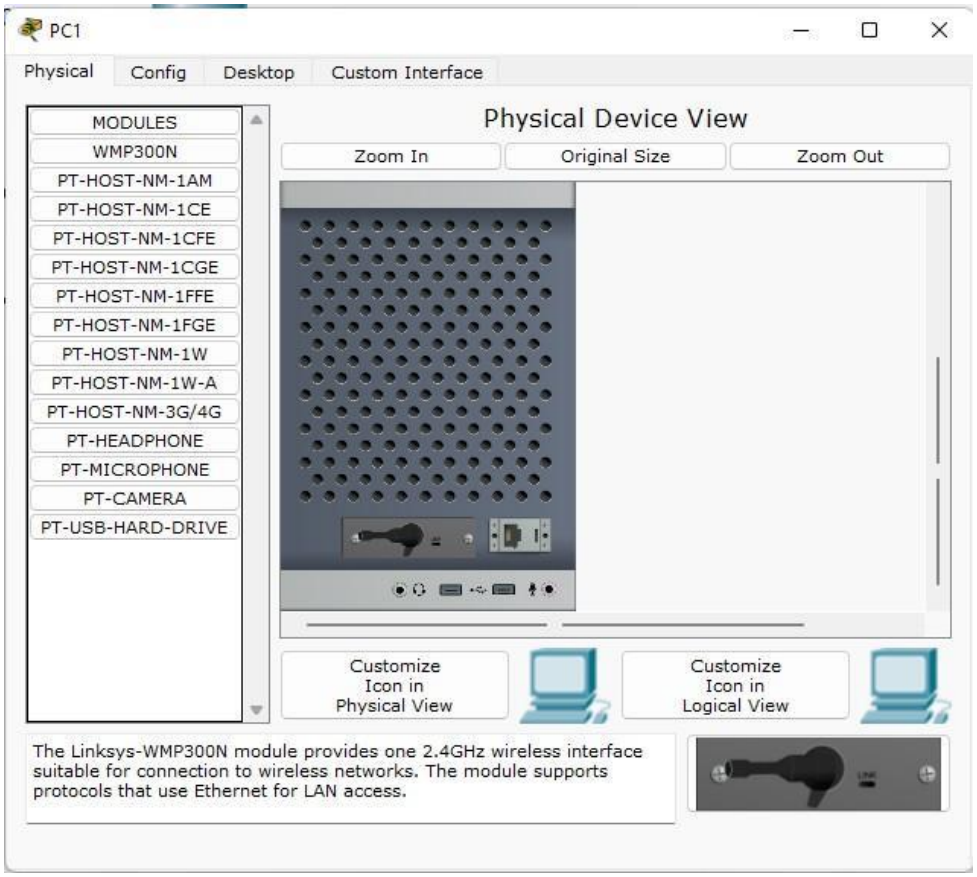
To construct a WLAN and make the nodes communicate wirelessly

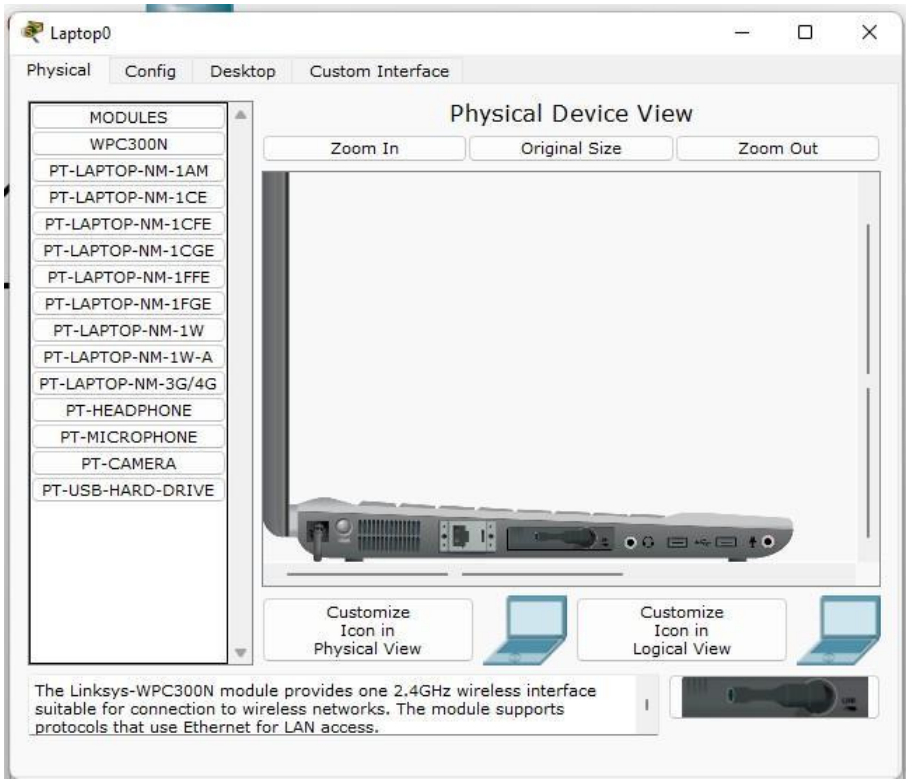
OBSERVATION:



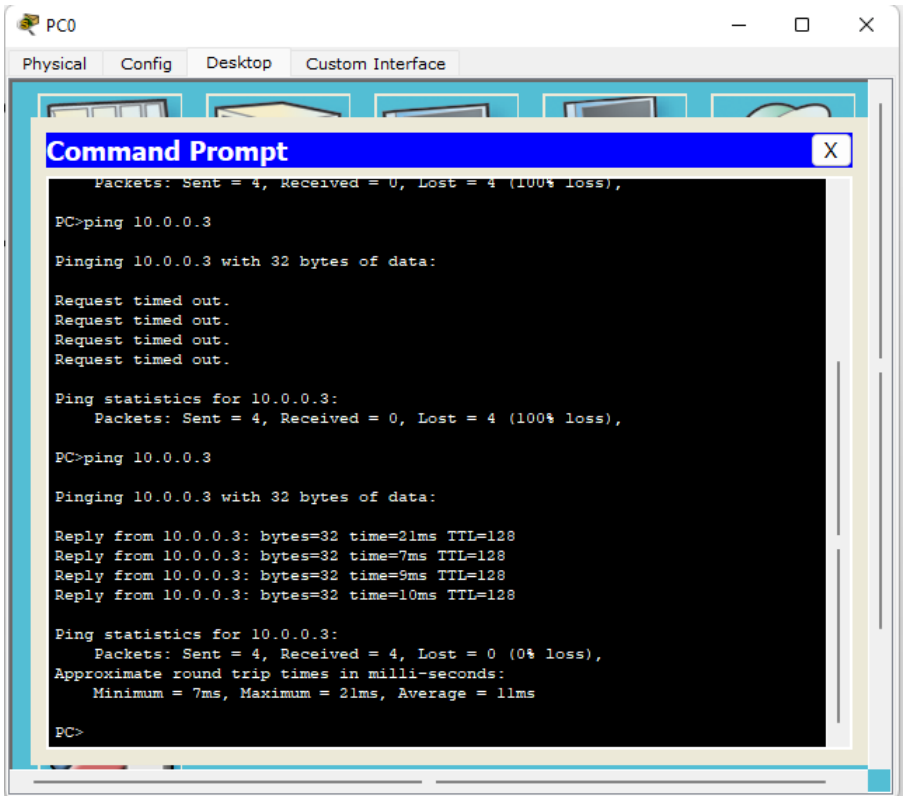
TOPOLOGY:


Logical
10.0.0.1
Fa0/0
Router-PT
Router0
Fa1/1
Fa2/1
Fa0/1
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.

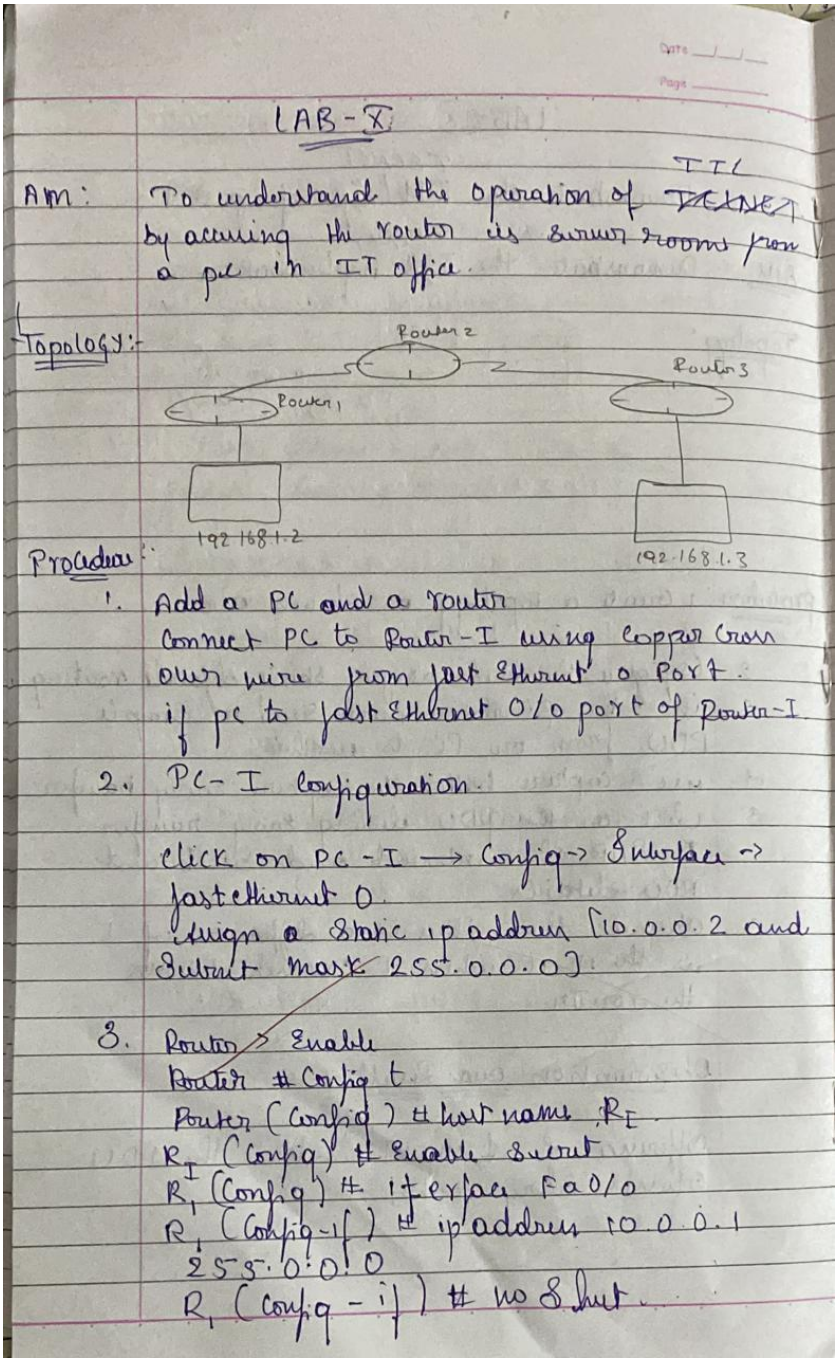
OUTPUT:


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>

WEEK 12

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

OBSERVATION:



-> Now Ping

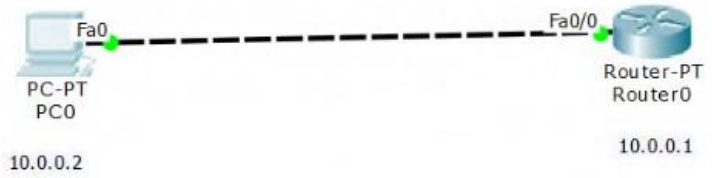
from PC0 to PC3
x Ping 192.168.20.3
You will get a succesful transmission from PC0 to PC3

Observation :-

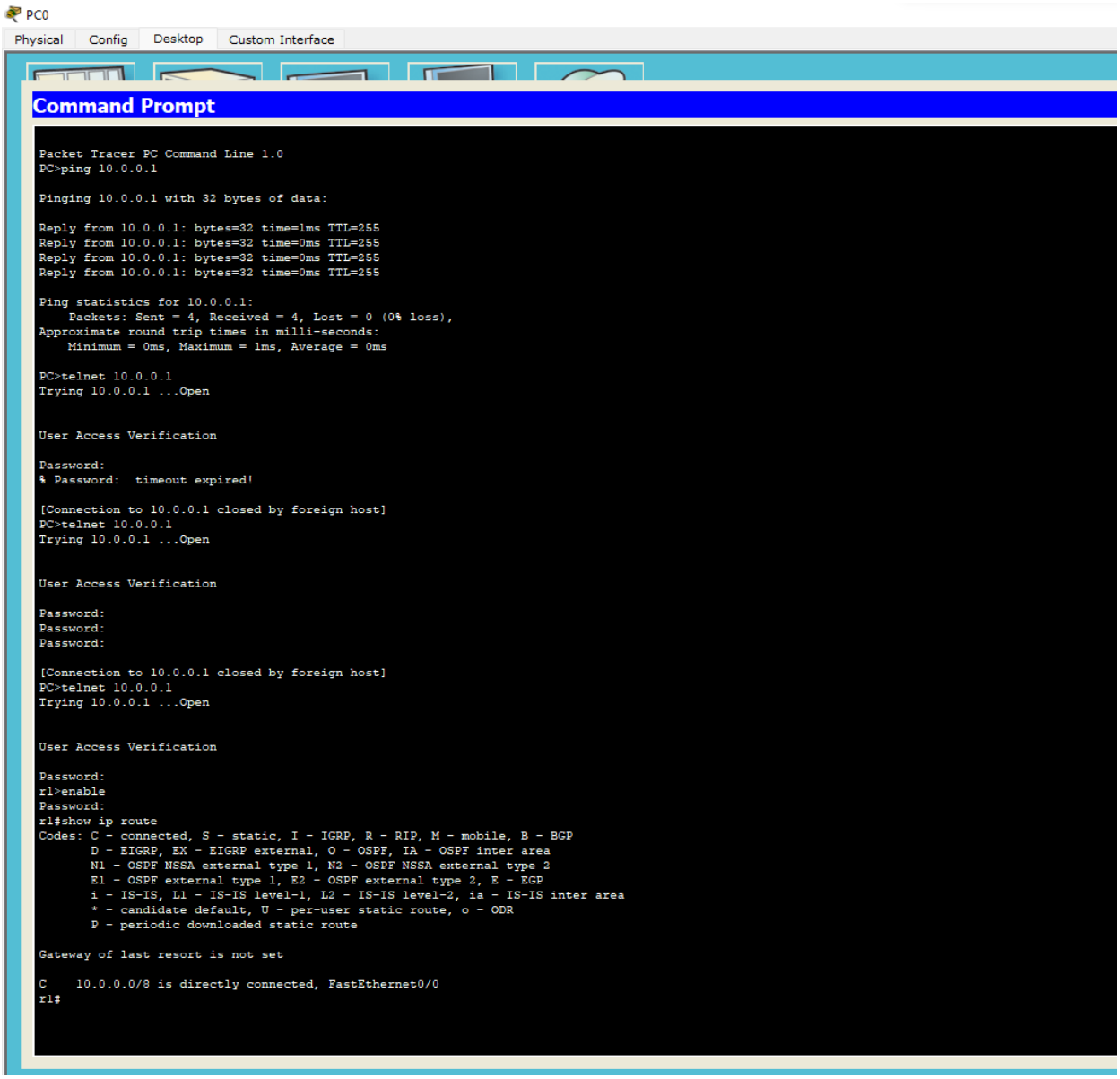
Even though we are using a single router we can multiple different network and those those networks will works as virtual networks and we can communicate from physical LAN to VLAN & vice versa.

Sd.
1/9/23

TOPOLOGY:



OUTPUT:


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:
rl>enable
Password:
rl#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
rl#

WEEK 13

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

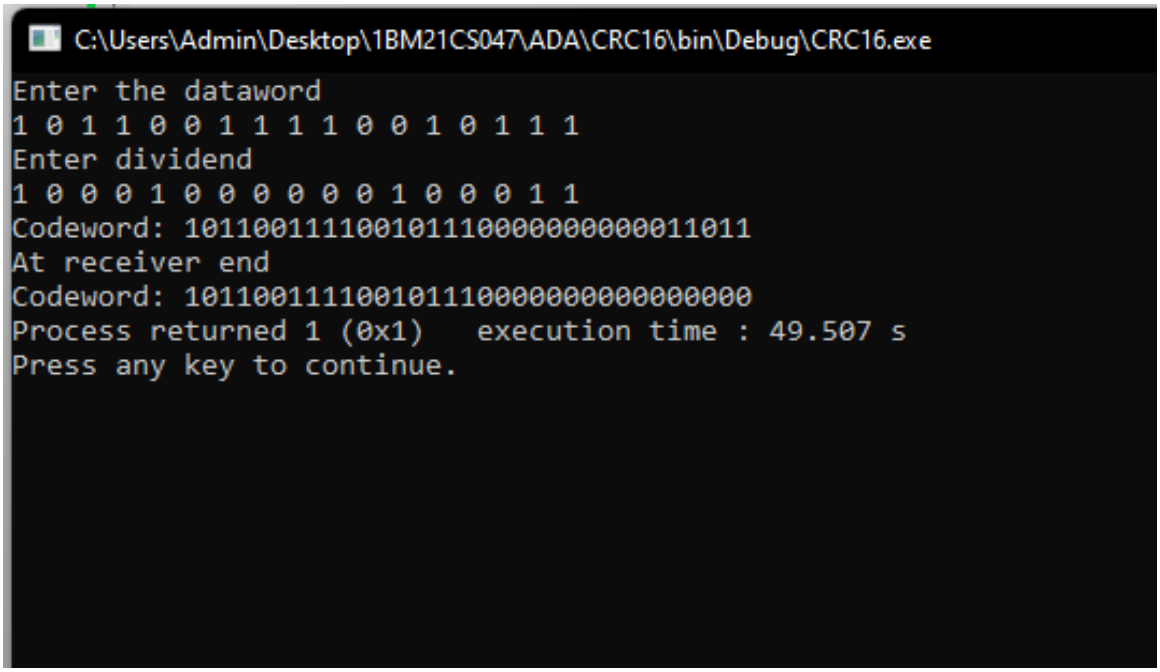
CODE:

```
#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");
```

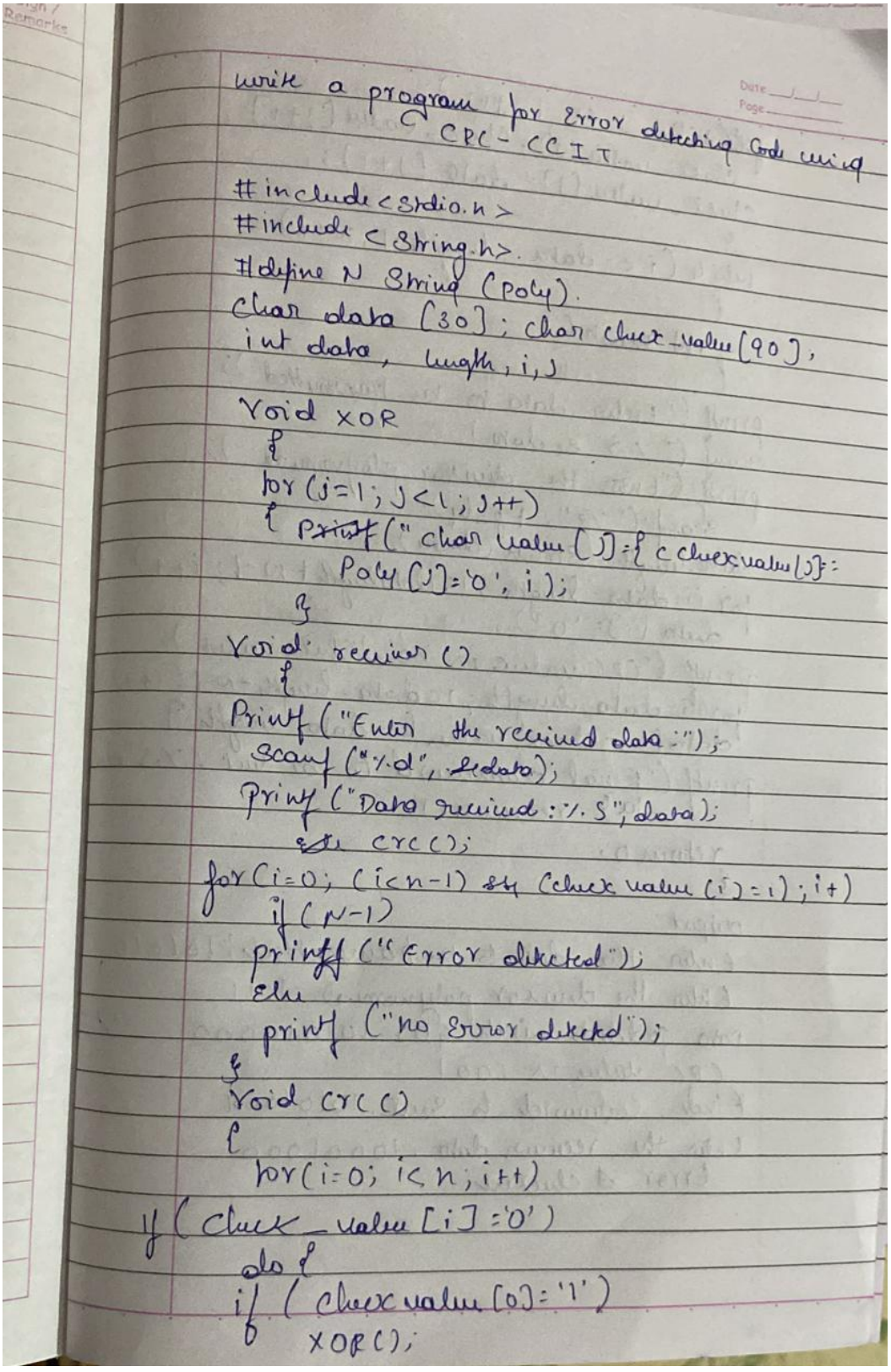
```
for(i=0;i<17;i++)
   scanf("%d",&dd[i]);

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:



write a program for Error detecting Code using CRC - CCITT

```
#include<stdio.h>
#include<string.h>
#define N strlen(poly).
char data[30]; char check_value[90],
int data, length, i,j

Void XOR
  {
  for(j=1;j<l;j++)
  { Printf(" char value[j]={ c checkvalue[j]:
      Poly[j]='0', i);
    }
Void receiver()
  {
Printf("Enter the received data:");
 scanf("%d", &data);
 Printf("Data received:%.s", data);
    crc();
for(i=0; (i<n-1) && (check value(i)=1); i++)
   if(N-1)
   printf("Error detected");
   Else
   printf("no Error detected");
}
Void CRC()
{
  for(i=0; i<n; i++)
if(check_value[i]='0')
   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[j+1];
}
while(i<=data_length-1; N+1)
{
  int main()
{
printf("Enter data to be transmitted");
scanf("%s", &data);
printf("Enter the divisor ploynomial");
  scanf("%s", &poly);
data_length = strlen(data);
for(i=data_length; i<data_length+N-1; i++)
  data[i] = '0');
printf("CRC value is %s", check_value);
for(i=data_length; i<data_length+N-1; i++)
  data[i] = check_value[i-data_length]
printf("Final dataword to be sent: %s", da
  -ta); receiver();

  return 0;
}
```

output:-

Enter the data to be transmitted: 10101O

Enter the divisor polynomial: 1011

Data padded with 0s: 101010000

CRC value is: 001

Final Codeword to Sent: 101010001

Enter the receiver data: 10001000

Error detected

WEEK 14

Write a program for congestion control using Leaky bucket algorithm.

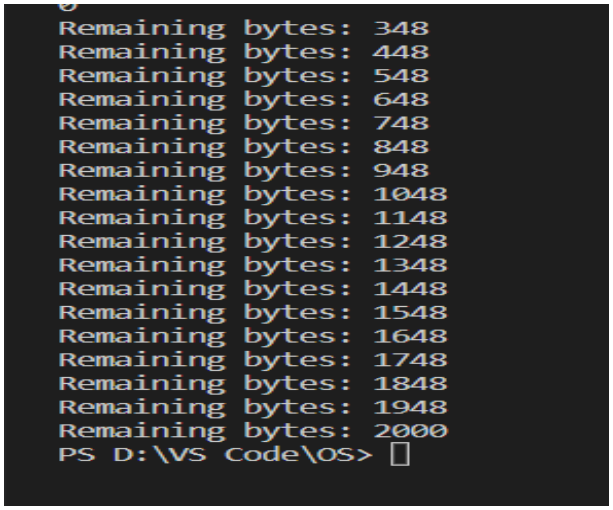
CODE:

```
#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);
```

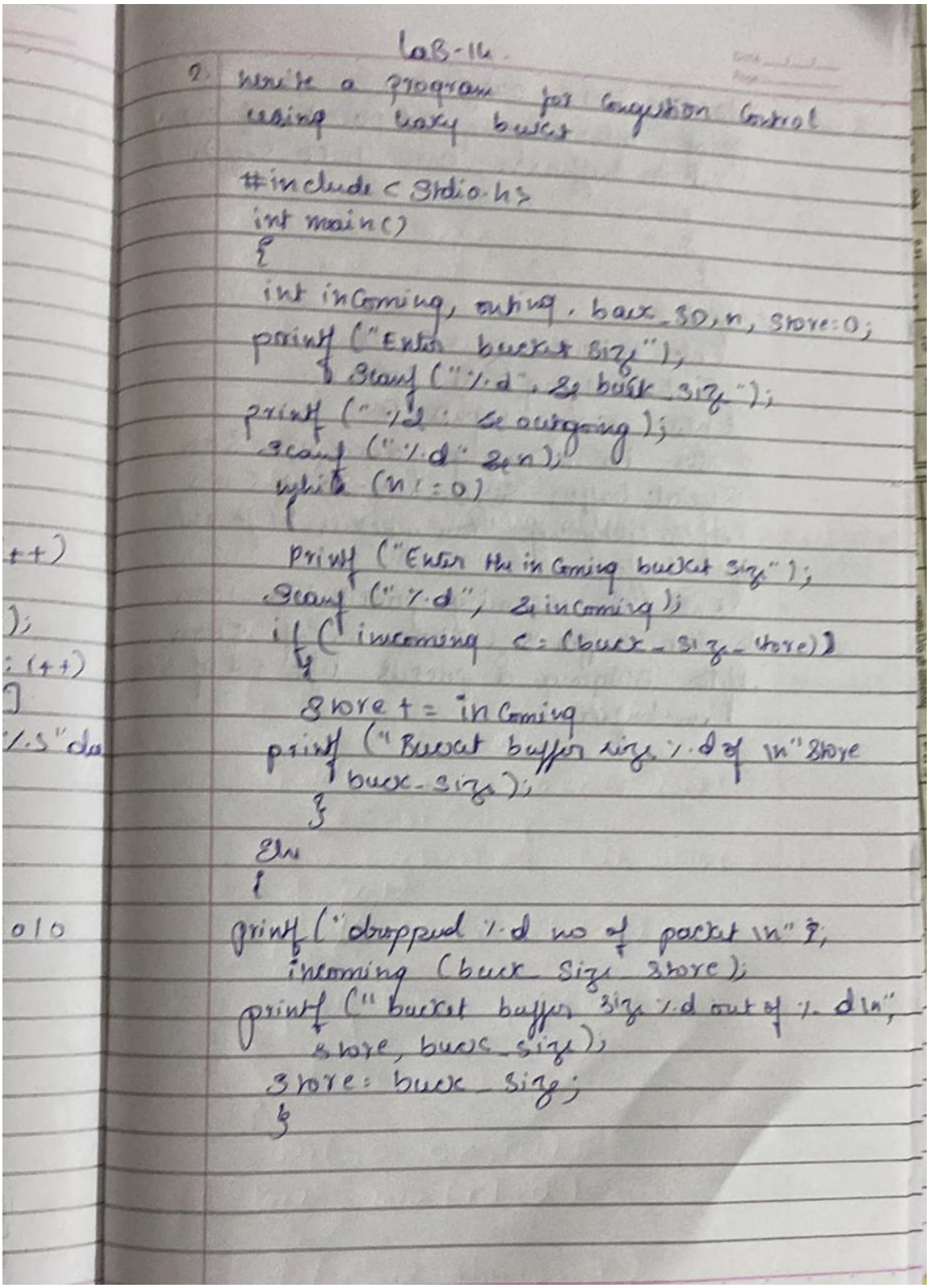
```
    }
    while (remaining < buckets) // Fixed the condition
    {
        if (buckets - remaining > outlets)
        {
            remaining += outlets; // Fixed the calculation
        }
        else
            remaining = buckets;
        printf("Remaining bytes: %d \n", remaining);
    }
  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

OUTPUT:

```
PS D:\VS Code> cd "d:\VS Code\OS\" ; if ($?) { gcc bucket.c -o bucket } ; if ($?) { .\bucket }
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:



Lab-14

2. Write a program for Congestion Control using Leaky bucket

```
#include <stdio.h>
int main()
{
int incoming, outgoing, buck_size, n, store=0;
printf("Enter bucket size");
scanf("%d", &buck_size");
printf("%d: outgoing");
scanf("%d", &n);
while (n!=0)
{
printf("Enter the incoming bucket size");
scanf("%d", &incoming);
if(incoming <= (buck_size - store))
{
store += incoming
printf("Bucket buffer size %d of in" store
buck_size);
}
else
{
printf("dropped %d no of packet in", incoming (buck_size store);
printf("bucket buffer size %d out of %d in",
store, buck_size);
store = buck_size;
}
```

```
store = store - outgoing
printf("after outgoing %d packets left out of
%d in buffer:\n", store, buff_size);
    n--;
  }
}
```

Output

Enter bucket Size = 5000
Enter outgoing rate = 2000
Enter no of inputs = 9
Enter the incoming packet size = 3000
Bucket buffer size 3000 out of 5000
After outgoing 1000 packets left out of buffer
Enter the incoming packet size : 1000
Bucket buffer size 2000 out of 5000
After outgoing 2 packets left out of 5000 in buffer.

```
store = store - outgoing
printf("after outgoing %d packets left out of
%d in buffer\n", store, buff_size);
    n--;
  }
}
```

Output

Enter bucket Size = 5000
Enter outgoing rate = 2000
Enter no of inputs = 9
Enter the incoming packet size = 3000
Bucket buffer size 3000 out of 5000
after outgoing 1000 packets left out of buffer
Enter the incoming packet size : 1000
Bucket buffer size 2000 out of 5000
after outgoing 2 packet left out of 5000 in buffer.

WEEK 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

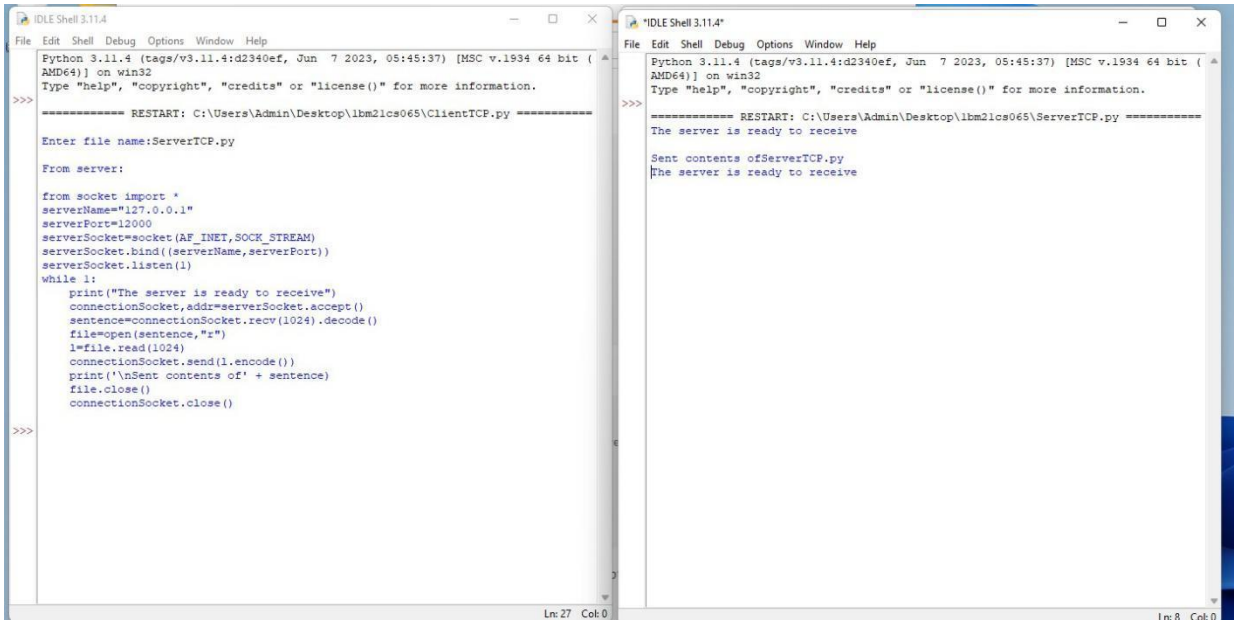
```
from socket import *
serverName = "127.0.0.1"
serverPort = 12000
clientSocket = socket(AF_INET, SOCK_STREAM)
clientSocket.connect((serverName,serverPort))
sentence = input("\nEnter file name: ")
clientSocket.send(sentence.encode())
filecontents = clientSocket.recv(1024).decode()
print ("\nFrom Server:\n")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

ServerTCP.py

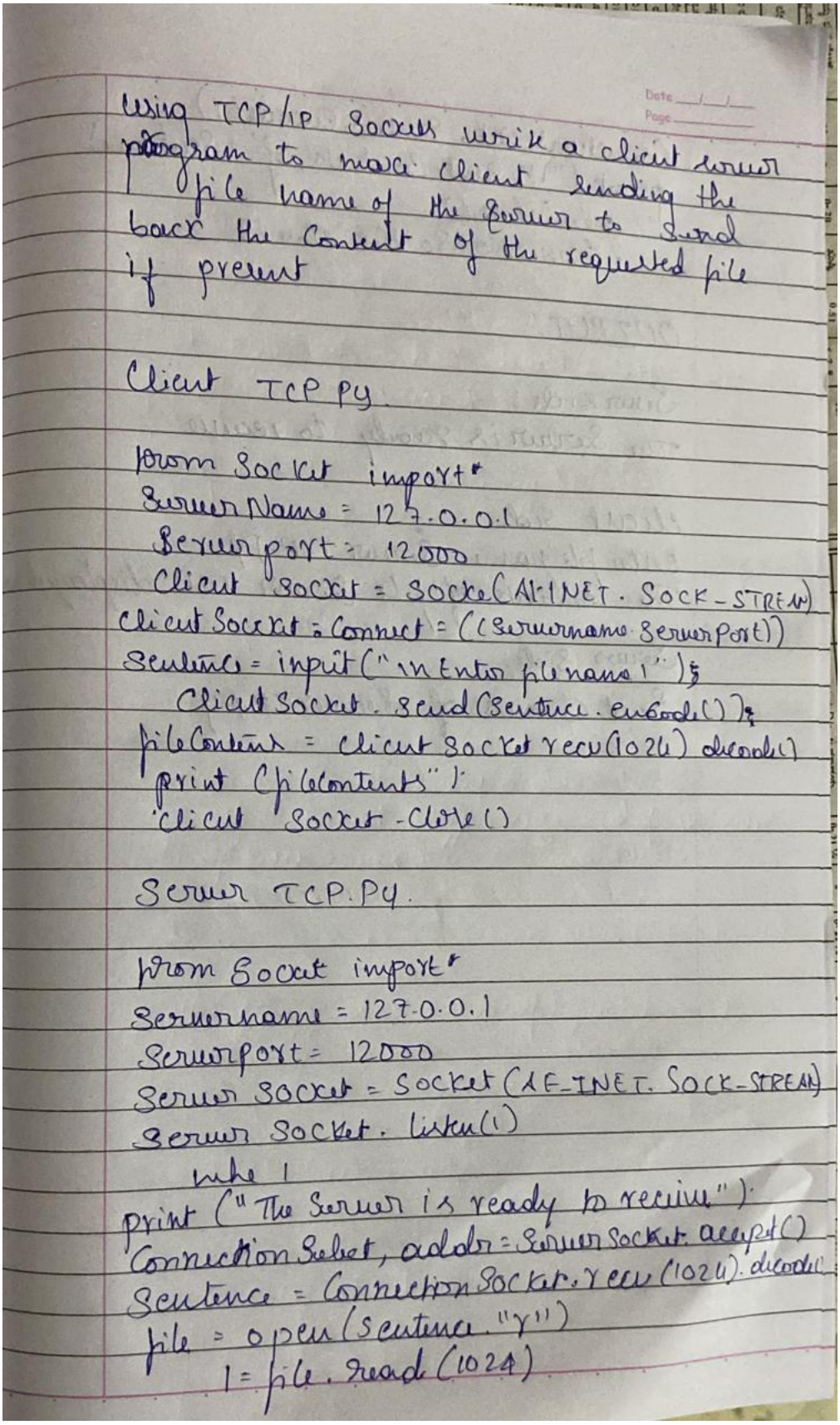
```
from socket import *
serverName="127.0.0.1"
serverPort = 12000
serverSocket =
socket(AF_INET,SOCK_STREAM)
serverSocket.bind((serverName,serverPort))
serverSocket.listen(1)
while 1:
print ("The server is ready to receive")
connectionSocket, addr = serverSocket.accept()
sentence = connectionSocket.recv(1024).decode()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

print (“\nSent contents of “ + sentence)
file.close()
connectionSocket.close()

OUTPUT:



OBSERVATION:



Using TCP/IP Sockets write a client server program to make client sending the file name of the server to send back the content of the requested file if present

Client TCP.py

```
from Socket import*
ServerName = 127.0.0.1
Serverport = 12000
Client Socket = Socket(AF_INET. SOCK_STREAM)
Client Socket.Connect((Servername.ServerPort))
Sentence = input("\n Enter filename:")
   Client Socket. send(Sentence. encode())
fileContents = client Socket recv(1024) decode()
print (fileContents")
client Socket.Close()
```

Server TCP.Py

```
from Socket import*
Servername = 127.0.0.1
Serverport = 12000
Server Socket = Socket(AF_INET. SOCK_STREAM)
Server Socket. listen(1)
   while 1
print ("The Server is ready to receive").
Connection Socket, addr = Server Socket. accept()
Sentence = Connection Socket. recv (1024). decode
file = open(Sentence."r")
   l = file. read (1024)
```

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

OUTPUT:-

Server side:
The server is ready to receive

Client side:
Enter file name: ServerTCP.py
The contents of file ServerTCP displayed

Server side:
Sent contents of ServerTCP.py

WEEK 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

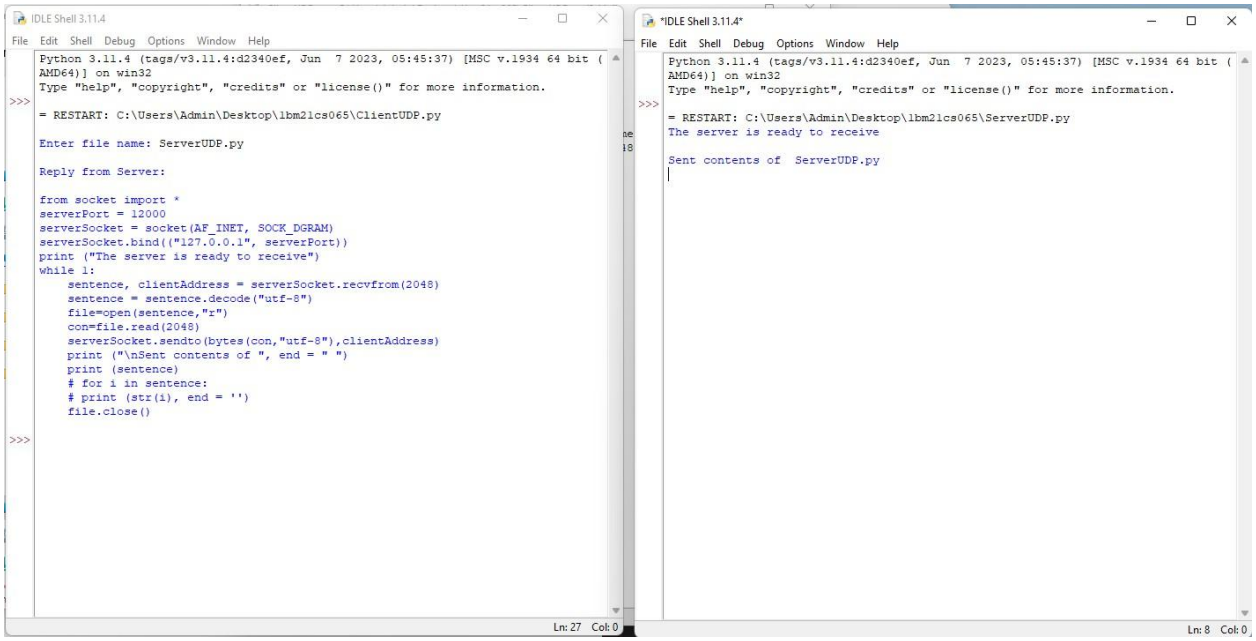
```
from socket import *
serverName = "127.0.0.1"
serverPort = 12000
clientSocket = socket(AF_INET, SOCK_DGRAM)
sentence = input("\nEnter file name: ")
clientSocket.sendto(bytes(sentence,"utf-8"),(serverName, serverPort))
filecontents,serverAddress = clientSocket.recvfrom(2048)
print ("\nReply from Server:\n")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

ServerUDP.py

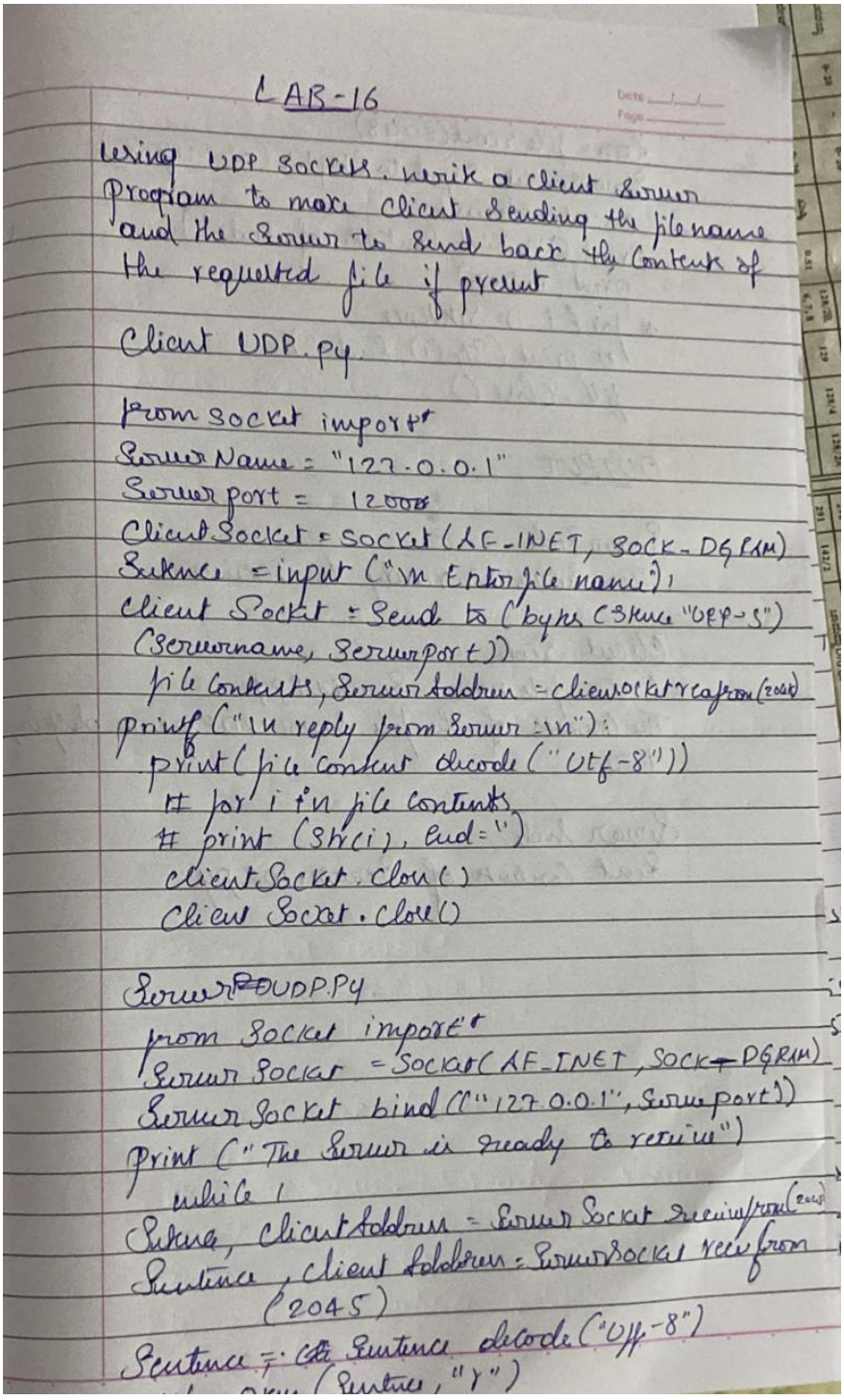
```
from socket import *
serverPort = 12000
serverSocket = socket(AF_INET, SOCK_DGRAM)
serverSocket.bind(("127.0.0.1", serverPort))
print ("The server is ready to receive")
while 1:
sentence, clientAddress = serverSocket.recvfrom(2048)
sentence = sentence.decode("utf-8")
file=open(sentence,"r")
```

```
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:



OBSERVATION:



LAB-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

Client UDP.py

```
from socket import*
ServerName = "127.0.0.1"
Serverport = 12000
ClientSocket = socket(AF_INET, SOCK_DGRAM)
Sentence = input("\n Enter file name:")
ClientSocket = Send to (bytes(Sentence,"UTF-8")
(Servername, Serverport))
file contents, ServerAddress = clientSocket.recvfrom(2048)
print("\n reply from Server :\n");
print(file content.decode("utf-8"))
# for i in file contents
# print (str(i), end="")
clientSocket.close()
Client Socket.close()
```

Server UDP.py

```
from socket import*
Server Socket = Socket(AF_INET, SOCK_DGRAM)
Server Socket bind(("127.0.0.1", Serverport))
print("The Server is ready to receive")
while 1
Sentence, clientAddress = ServerSocket.recvfrom(2048)
Sentence, client Address = ServerSocket.recvfrom
(2045)
Sentence = Sentence.decode("Utf-8")
(Sentence, "Y")
```

connectionSocket.send(1, encode())
print ("\n Sent Contents of" + sentence)
file.close()
connectionSocket.close()

OUTPUT:-

Server side:
The server is ready to receive

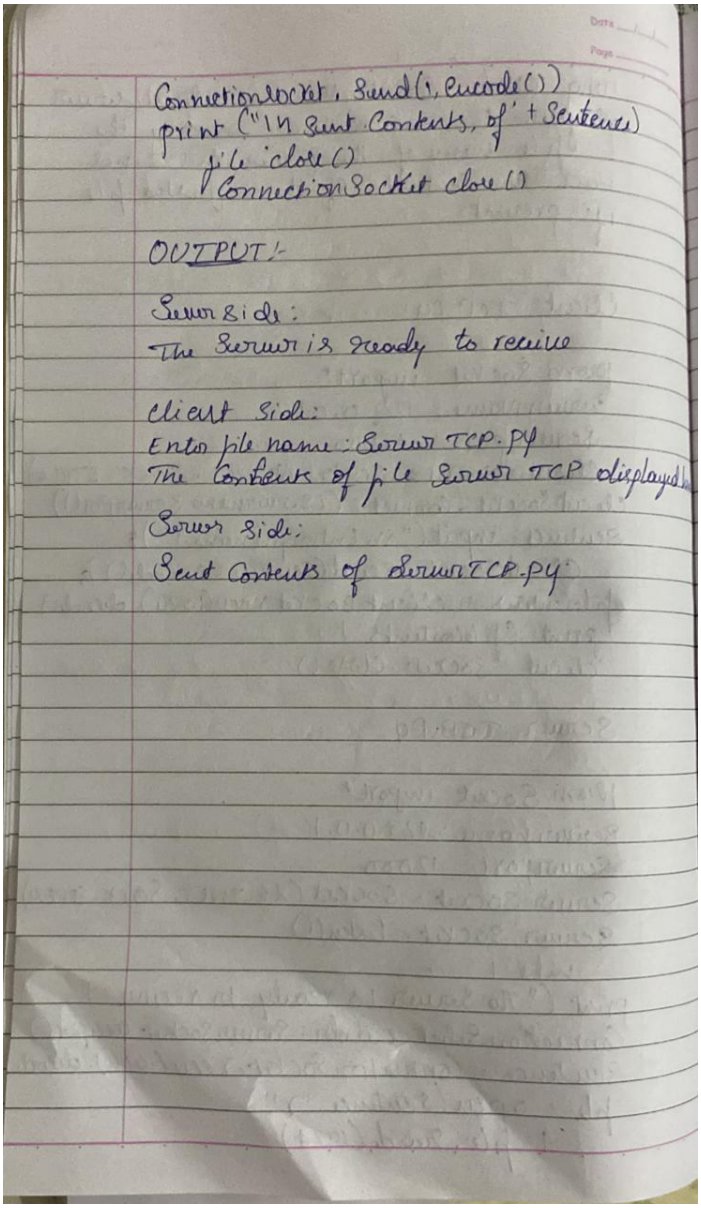
client side:
Enter file name: ServerTCP.py
The contents of file ServerTCP displayed

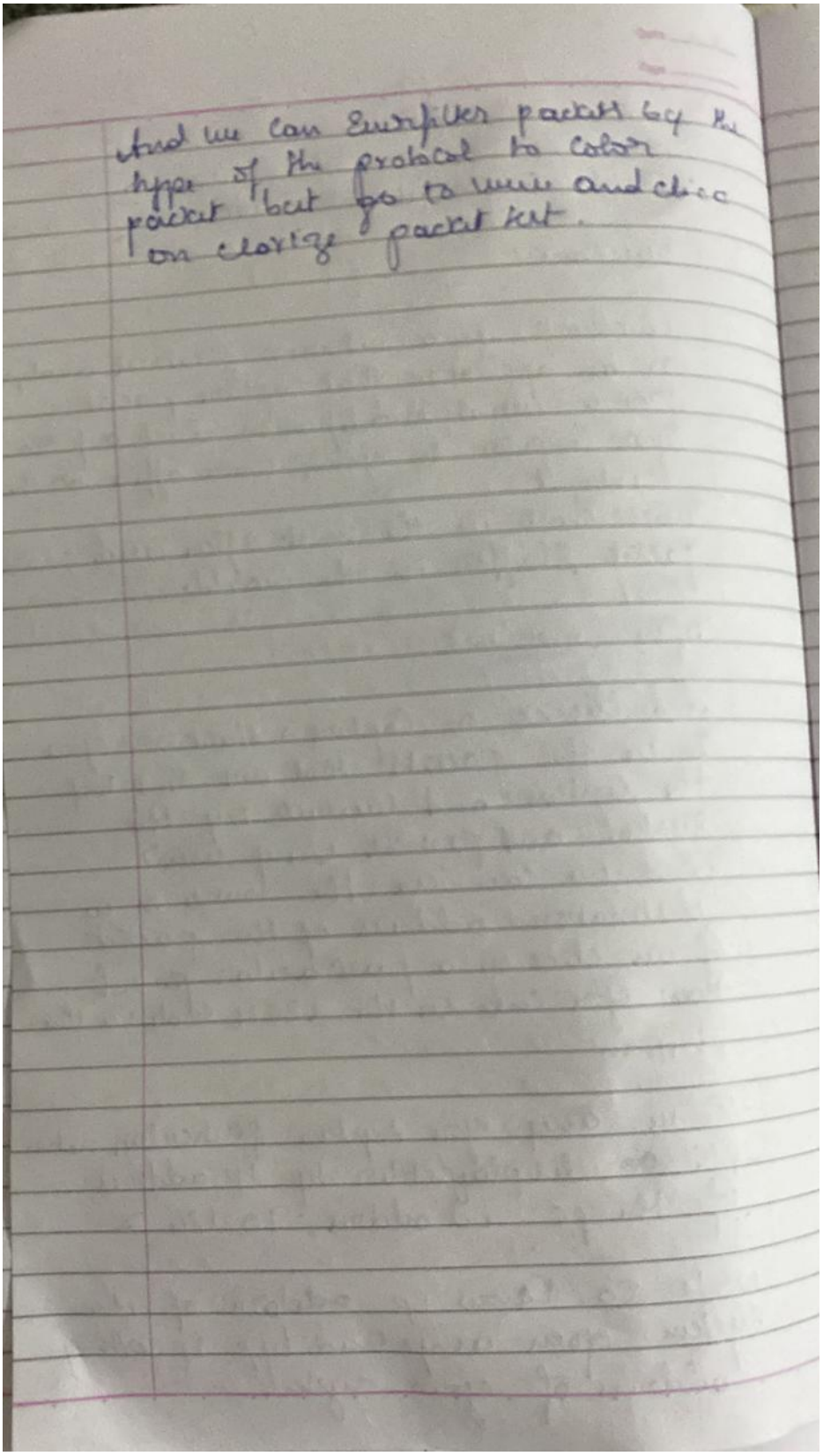
Server side:
Sent contents of ServerTCP.py

WEEK 17

Tool Exploration –Wireshark

OBSERVATION:




And we can Eurnfilter packets by the
type of the protocol to color
packet but go to www and click
on clorize packet list.