VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"JnanaSangama", Belgaum -590014, Karnataka.

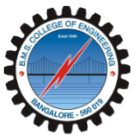
LAB REPORT
on

COMPUTER NETWORKS

Submitted by

ANITHA K J (1BM22CS401)

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 **ANITHA K J (1BM22CS401),** 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.

Dr. Latha N R
Assistant Professor
Department of CSE
BMSCE, Bengaluru

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

`

Index

WEEK 1

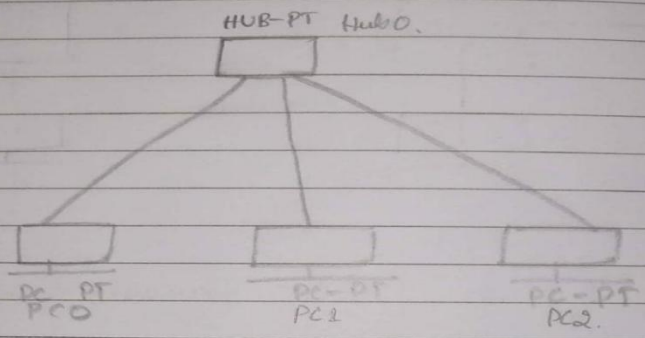
Create a topology and simulate sending a simple PDU from source to destination using hub and switch as connecting devices and demonstrate ping message.Create a topology and simulate sending a simple PDU from source to destination using hub and switch as connecting devices and demonstrate ping message.

15/06/2023

Q1 Create a topology and simulate sending a simple PDU from source to destination using a simple hub and switch as connecting domains.

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

Hub :-



Step 1 :- Select end devices and choose generic and choose PC0, PC1, PC2 (PC-PT)

Step 2 :- Go to hubs and select generic hub

Step 3 :- Go-to connection and select coffee straight through wire, then connect all PC's to the hub. (Select port number and PC)

Step 4 :- Click on PC, go to config and select fast ethernet then set IP address for the PC. Do the same for all the PC's
10.0.0.1 10.0.0.2 10.0.0.3.

Step 5 :- Add simple PDU

Step 6:- click source and destination symbol systems

Step 7:- Then go to simulation mode, Auto capture/play, Then the packets will start to transfer.

Step 8:- Click on PC, go to 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.

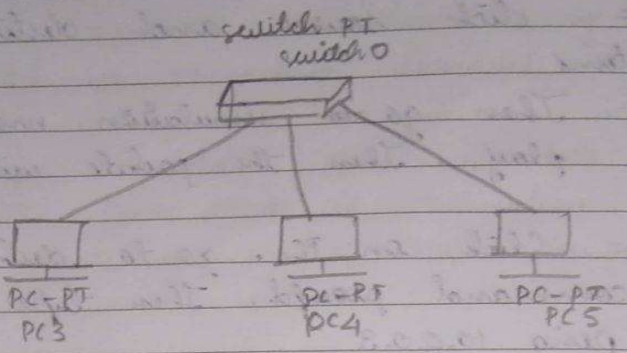
Ping statistics for 10.0.0.3:
packets: Sent=4, Received=4, Lost=0. (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 4ms, Average = 2ms

Procedure Observation:-

When the source device send a packet to the hub it will broadcast or send the packet to all the devices which are connected to the hub. And the destination device will receive the packet and others will reject the packet.

And destination device will send the acknowledgement and that will be distributed among all devices and the source will accept and others will discard.

Switch



Procedure :-

Step 1 :- Select switch and 3 PC's

step 2 :- Set IP address for all the PC's 10.0.0.4 10.0.0.5 10.0.0.6
PC → config → fast ethernet → IP address

step 3 :- connect PC's to the switch by selecting copper straight through

step 4 :- Add simple PDU
select source and destination

step 5 :- Go to simulation mode and click on Auto capture/play

step 6 :- 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=4ms TTL=128
Reply from 10.0.0.6 : bytes=32 time=4ms TTL=128

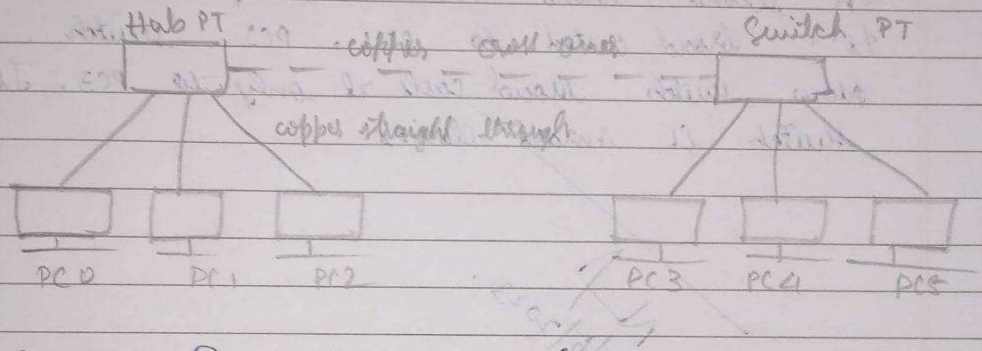
ping statistics for 10.0.0.6:
packets: sent=4, Received=4 lost=0 (0% loss),
Approximate round trip time in milli-seconds:
Minimum=4ms, Maximum=4ms, Average=4ms

(Procedure:) Observation:-

When the first time the packet is sent the switch will distribute the packet with all the devices.

Once it learns about the IP address it will only send packet to the destination and send acknowledgement to the source.

Switch - Hub Connection.



Step 1 :- Previously drawn hub topology and switch topology are connected through copper cross over.
In hub port 3 is used in switch fast ethernet 3/1 is used

Step 2 :- Add simple PDC from PC0 to PC3

ping 10.0.0.4.

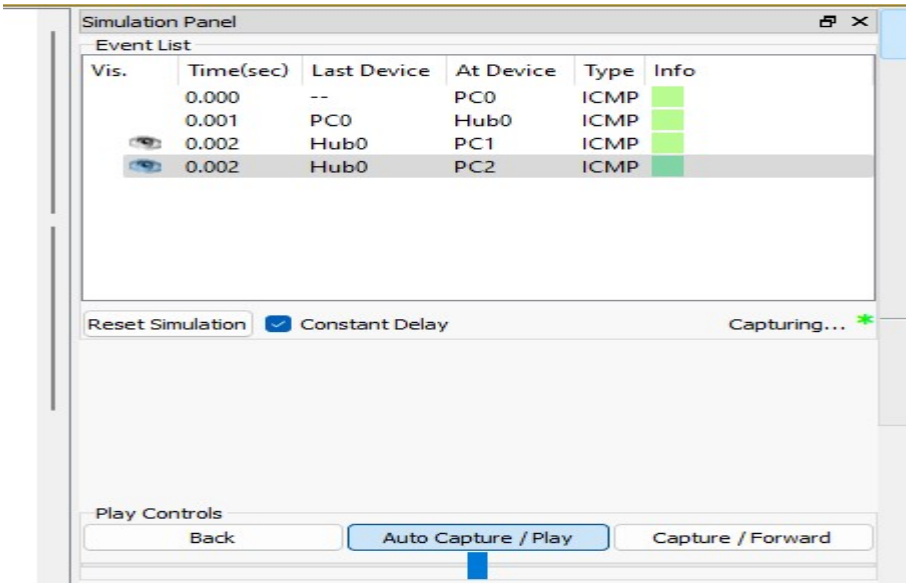
pinging 10.0.0.4 with 32 bytes of data :
Reply from 10.0.0.4 : bytes = 32 time = 1ms TTL = 128
Reply from 10.0.0.4 : bytes = 32 time = 1ms TTL = 128
Reply from 10.0.0.4 : bytes = 32 time = 1ms TTL = 128
Reply from 10.0.0.4 : bytes = 32 time = 1ms TTL = 128
ping statistics for 10.0.0.4.
packets : sent = 4 Received = 4 lost = 0 (0% loss)
Approximate round trip times in milli-seconds
Minimum = 4ms Maximum = 4 ms Average = 4ms

Observation :-

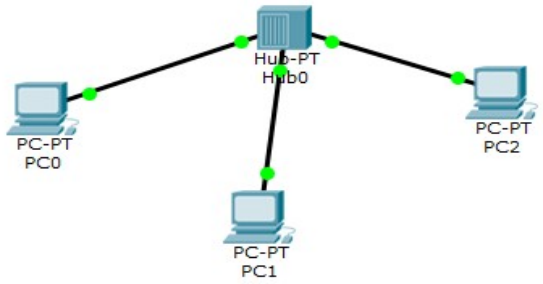
In simulation mode PC0 sends packet to hub hub sends it to PC1, PC2 and switch broad casts it to PC3, PC4 and PC5.
PC1 PC2 PC4 and PC5 discards them.
PC3 accepts and sends acknowledgement to hub through switch.
Hub is broad casts it to all 3 PCs
only PC0 accepts it and others discard.
In second round PC0 sends packet to hub it broad casted to PC1, PC2 switch.
Now switch broad casts it only to PC3. Thus switch is smart device.

NL
15/6/2023.

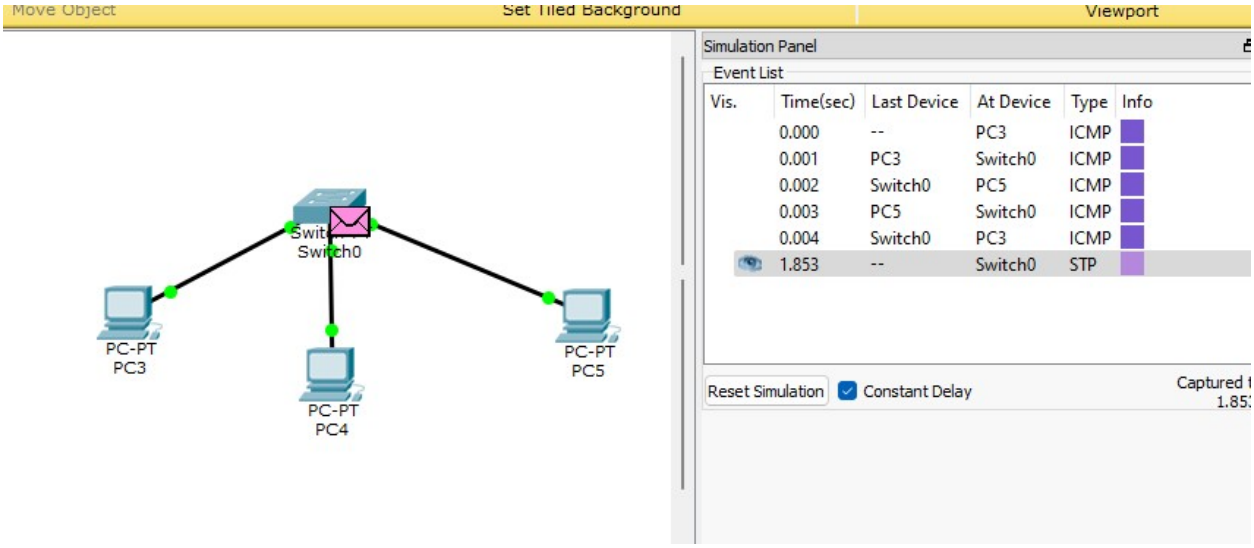
OUTPUT SCREENS

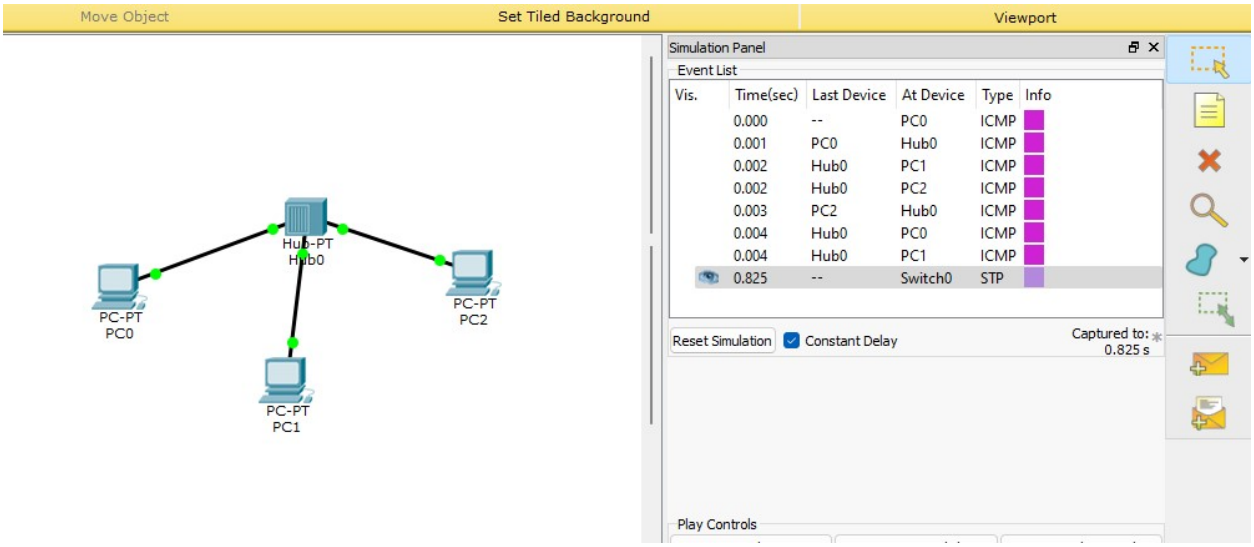


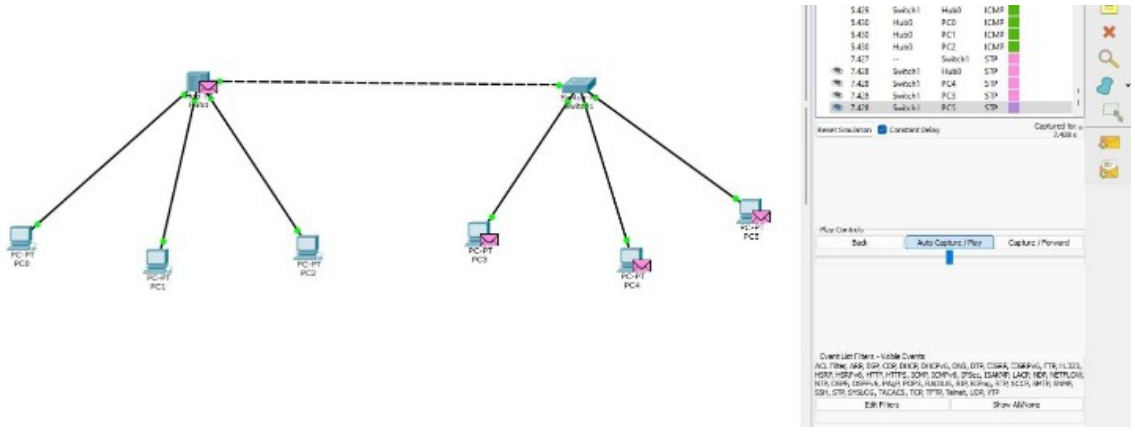
HUB



SWITCH


Move Object
Set Tiled Background
Viewport
Switch0
PC-PT
PC3
PC-PT
PC4
PC-PT
PC5
Simulation Panel
Event List
Vis. Time(sec) Last Device At Device Type Info
0.000 -- PC3 ICMP
0.001 PC3 Switch0 ICMP
0.002 Switch0 PC5 ICMP
0.003 PC5 Switch0 ICMP
0.004 Switch0 PC3 ICMP
1.853 -- Switch0 STP
Reset Simulation
Constant Delay


Move Object
Set Tiled Background
Viewport
Hub-PT
Hub0
PC-PT
PC0
PC-PT
PC2
PC-PT
PC1
Simulation Panel
Event List
Vis. Time(sec) Last Device At Device Type Info
0.000 -- PC0 ICMP
0.001 PC0 Hub0 ICMP
0.002 Hub0 PC1 ICMP
0.002 Hub0 PC2 ICMP
0.003 PC2 Hub0 ICMP
0.004 Hub0 PC0 ICMP
0.004 Hub0 PC1 ICMP
0.825 -- Switch0 STP
Reset Simulation
Constant Delay
Captured to:
0.825 s
Play Controls



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=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>
```

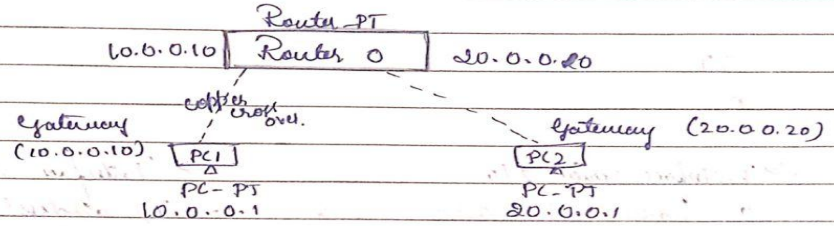
WEEK 2

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

OBSERVATION:

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

2a) Aim :- Configure IP address to routers and exploring ping messages



Procedure:-

Step 1 : Select Routers - PT and place it in workspace

step 2 : Take 2 end devices as PC-PT and drop them in workplace.

step 3 : connect fast ethernet 0/0 of PC1 to fast ethernet 0/0 of router and fast ethernet 0/0 of PC2 to fast ethernet 0/0 of routers using copper cross-over

step 4 : set IP address of of PC1 as 10.0.0.1 and PC2 as 20.0.0.1

step 5 : In setting set gateway of PC1 as 10.0.0.10 and PC2 as 20.0.0.10

step 6 : setup the interface of routers using the following steps :-

To configure routers commandline interface (CLI) is used.....

Router 0 CLI

(Press no)

Router > enable

Router # config t. or config terminal

STEPS

{ interface serial 2/0
IP address
No shut
Exit
Exit
Show IP route

CLI

n.

→ Interface serial 2/0
IP address 20.0.0.20 →
No shut.
Exit

→ Interface serial 3/0
→ IP address 30.0.0.10
→ No shut
→ exit.

→ Router enable
→ Interface serial 2/0
→ Set IP address

Router #: show IP route

→ click on PC.
→ Ping 10.0.0.1

Static Routing :-

cmd:- subnet mask

IP route 30.0.0.0 255.0.0.0 20.0.0.20
IP route 40.0.0.0 255.0.0.0 20.0.0.20

Router 2: IP route 10.0.0.0 255.0.0.0 20.0.0.10
IP route 40.0.0.0 255.0.0.0 30.0.0.20
exit
Show IP route

Router5 :- IP route 10.0.0.0 255.0.0.0 30.0.0.10
IP route 20.0.0.0 255.0.0.0 30.0.0.10
exit
show IP route.

ping 40.0.0.1 &.

Router (config) # interface fastethernet 0/0
Router (config-if) # ip address 10.0.0.10 255.0.0.0
Router (config-if) # no shut.
exit
Router (config) # interface fastethernet 1/0.
Router (config-if) # ip address 20.0.0.10 255.0.0.0
Router (config-if) # no shut.
exit.
Router (config) # exit
Router #
show ip route.

C 10.0.0.0/8 is directly connected, Fast Ethernet 0/0
C 20.0.0.0/8 is directly connected, Fast Ethernet 1/0

Step 7 :- Observation
Green lights appear on wire when no shut commands are written which indicate that they are ready for data transmission.

Ping output in PC0 :-
PC> ping 20.0.0.1
pinging 20.0.0.1 with 32 bytes of data.
Request timed out.
Reply from 20.0.0.1 : bytes = 32 time = 0ms TTL = 127.

Reply from 20.0.0.1 bytes=32 time=0ms TTL=127
Reply from 20.0.0.1 bytes=32 time=0ms TTL=127
ping statistics for 20.0.0.1
packets: Sent = 4 Received = 3, lost = 1 (25% loss),
Approximate round trip times in milli-seconds
minimum = 0ms maximum = 1ms Average.

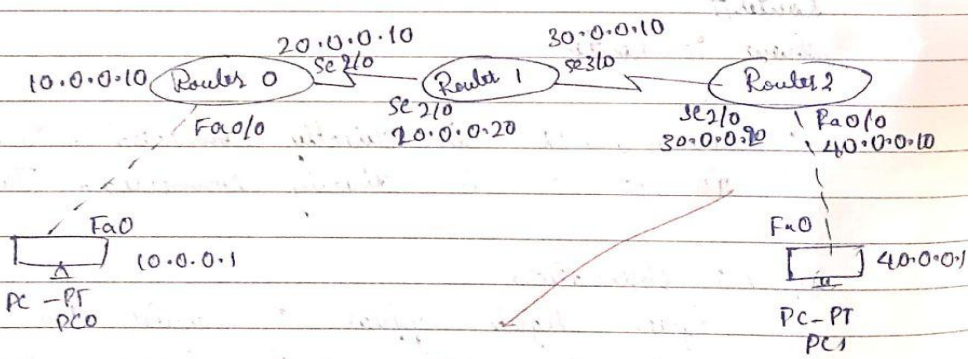
Observation

On pinging in PC0 for the first time there is a 25% loss

From next ping there are no losses

2b) Aim :- Configure using 3 routers and 2 PCs

Topology :-



Procedure

Step 1 : The network is started by selecting end devices PC0 & PC1 i.e., generic PCs and placing them in workspace.

Step 2 : select 3 router-PT and place them as router 0, router 1 and router 2 in workspace.

step 3: PC0 & PC1 are connected to router 0 and router 2 respectively using copper crossover

step 4: Connect router 0 to router 1, router 1 to router 2 using

step 5: set up IP address of PC0 to 10.0.0.1 PC1 to 40.0.0.1. Set up gateway of PC0 as 10.0.0.10 and PC1 as 40.0.0.10

Configure the routers by opening CLI:

In router 0

Router > enable
Router # config t.
Router (config) # interface fastethernet 0/0.
Router (config-if) # ip address 10.0.0.10 255.0.0.0
Router (config-if) # no shut.
exit.
Router (config) # interface serial 2/0.
Router (config-if) # ip address 20.0.0.10 255.0.0.0
Router (config-if) # no shut.
exit. exit

In Router 1

Router > enable
Router # config t.
Router (config) # interface serial 2/0
Router (config-if) # ip address 20.0.0.20 255.0.0.0
Router (config-if) # no shut
exit
Router (config) # interface serial 3/0
Router (config-if) # ip address 30.0.0.10 255.0.0.0
Router (config-if) # no shut.
exit.
Router (config) # exit

In Router 2

Router > enable.
Router # config t.
Router (config) # interface serial 2/0
Router (config-if) # ip address 30.0.0.90 255.0.0.0
Router (config-if) # no shut.
exit
Router (config) # interface fast ethernet 0/0
Router (config-if) # ip address 40.0.0.0 255.0.0.0
Router (config-if) # no shut.
exit.
Router (config) exit

IP Routes table:-

Router 0:

Router # show ip route
C 10.0.0.0/8 is directly connected, Fast ethernet 0/0
C 20.0.0.0/8 is directly connected, serial 2/0

Router 1:
Router # show ip route
C 20.0.0.0/8 is directly connected, serial 2/0
C 30.0.0.0/8 is directly connected, serial 3/0

Router 2:
Router # show ip route.
C 30.0.0.0/8 is directly connected, serial 2/0
C 40.0.0.0/8 is directly connected, Fast ethernet 0/0

ping output in PC0.

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

ping statistics for 40.0.0.1:
packets : sd:4 received = 0 lost = 4 (100% loss)

Observation :

Green lights appear on the wires when no thing is written.

Now configure the routes which does not have data of other network. Add the network in CLI In all 3 Routers CLI write config t then set route.

Router 0 :

ip route 30.0.0.0 255.0.0.0 20.0.0.20
ip route 40.0.0.0 255.0.0.0 20.0.0.20

Router 1 :

ip route 10.0.0.0 255.0.0.0 20.0.0.10
ip route 40.0.0.0 255.0.0.0 30.0.0.20

Router 2 :

ip route 20.0.0.0 255.0.0.0 30.0.0.10
ip route 10.0.0.0 255.0.0.0 30.0.0.10

new IP route table
exit

Router 0.

C 10.0.0.0/8 is directly connected, Fast ethernet 0/0
C 20.0.0.0/8 is directly connected, serial 2/0
S 30.0.0.0/8 [1/0] via 20.0.0.20
S 40.0.0.0/8 [1/0] via 20.0.0.20.

Router 1.

S 10.0.0.0/8 [1/0] via 20.0.0.10.
C 20.0.0.0/8 is directly connected serial 2/0
C 30.0.0.0/8 is directly connected serial 3/0
S 40.0.0.0/8 [1/0] via 30.0.0.20

Router 2.

S 10.0.0.0/8 [1/0] via 30.0.0.10
S 20.0.0.0/8 [1/0] via 30.0.0.10
C 30.0.0.0/8 is directly connected, serial 2/0
C 40.0.0.0/8 is directly connected, Fast ethernet 0/0.

Ping messages

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=2ms 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 = 2ms, Average = 2ms

Observation:

In first ping destination host was unreachable as Router 0 has no knowledge about the network 30.0.0.0 and 40.0.0.0 and the packets got stuck or lost.

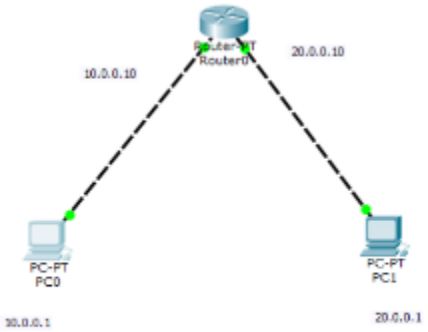
After this ip route is explicitly

Now on pinging there is 25% loss in first time, the following one's has no loss

OUTPUT SCREENS:

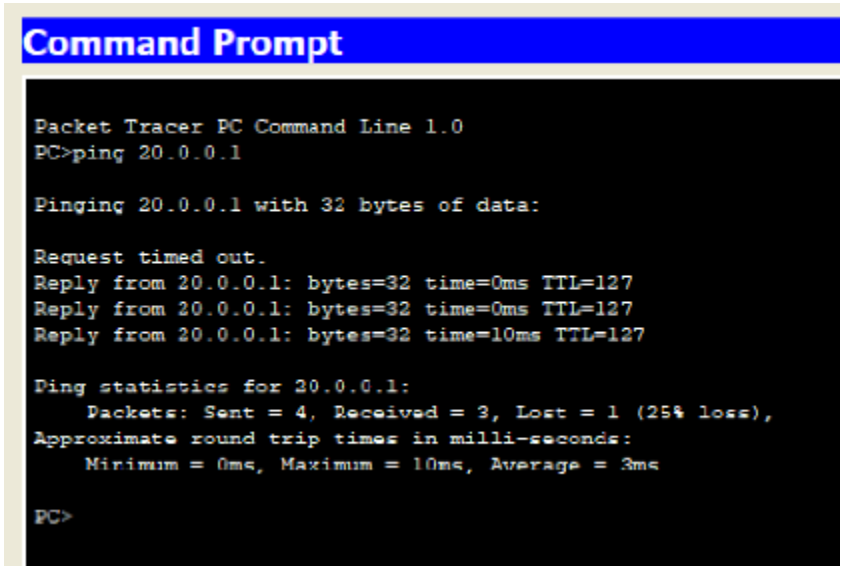
TOPOLOGY

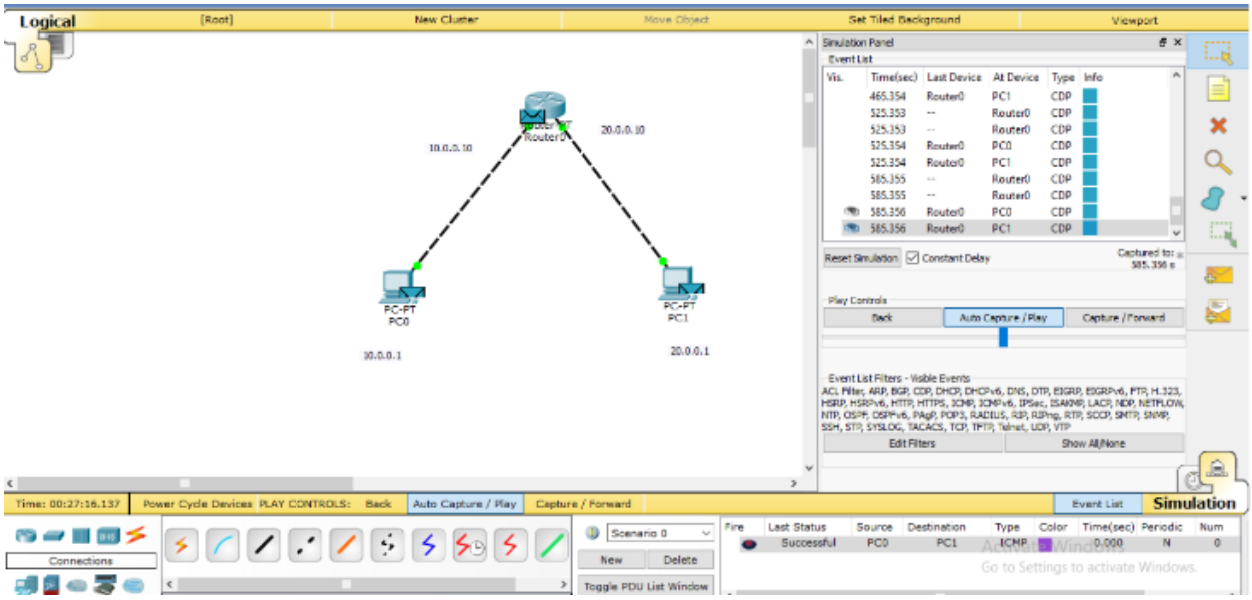
PROGRAM 2.1



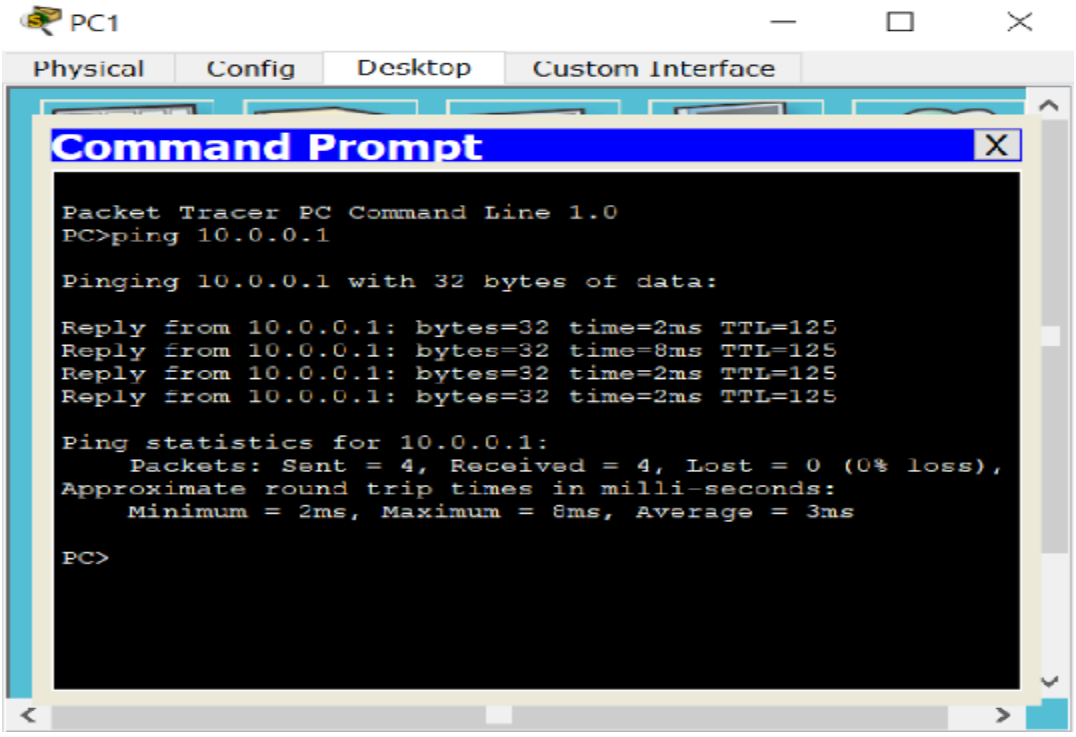
PROGRAM 2.2

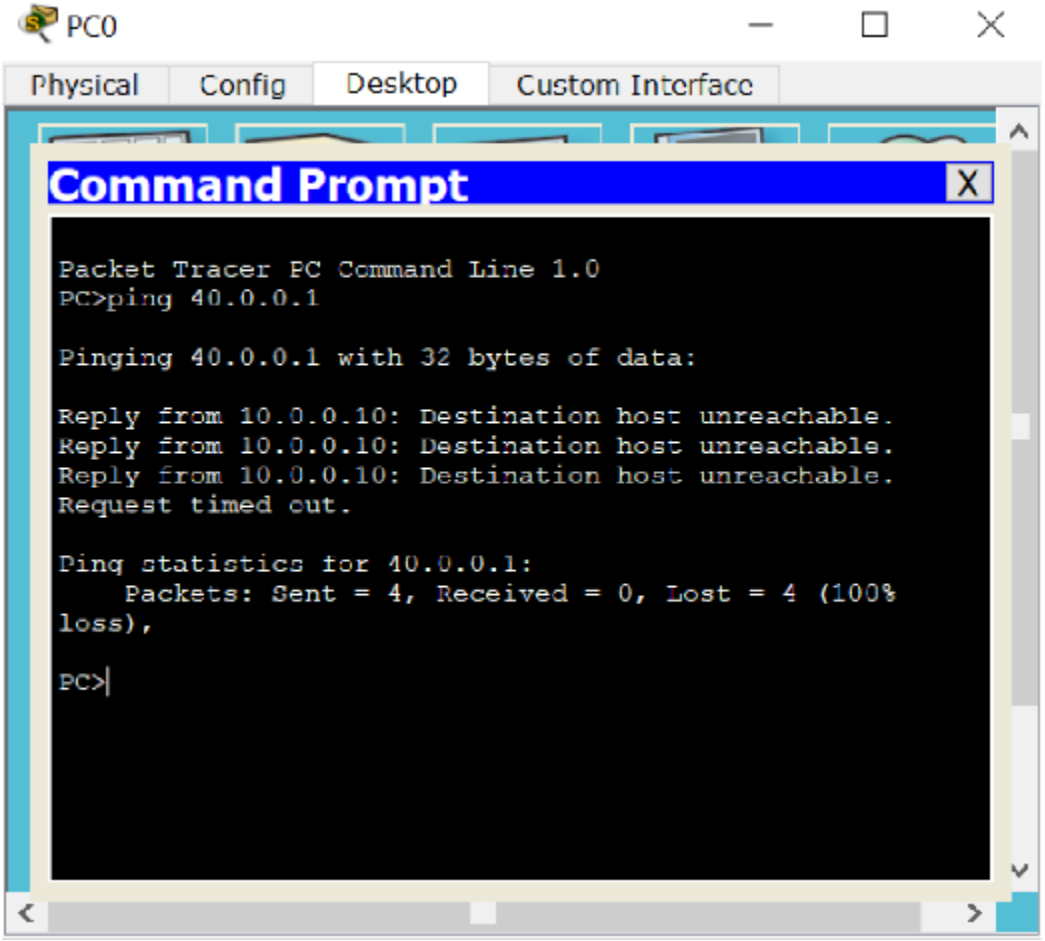
OUTPUT:
PROGRAM 2.1

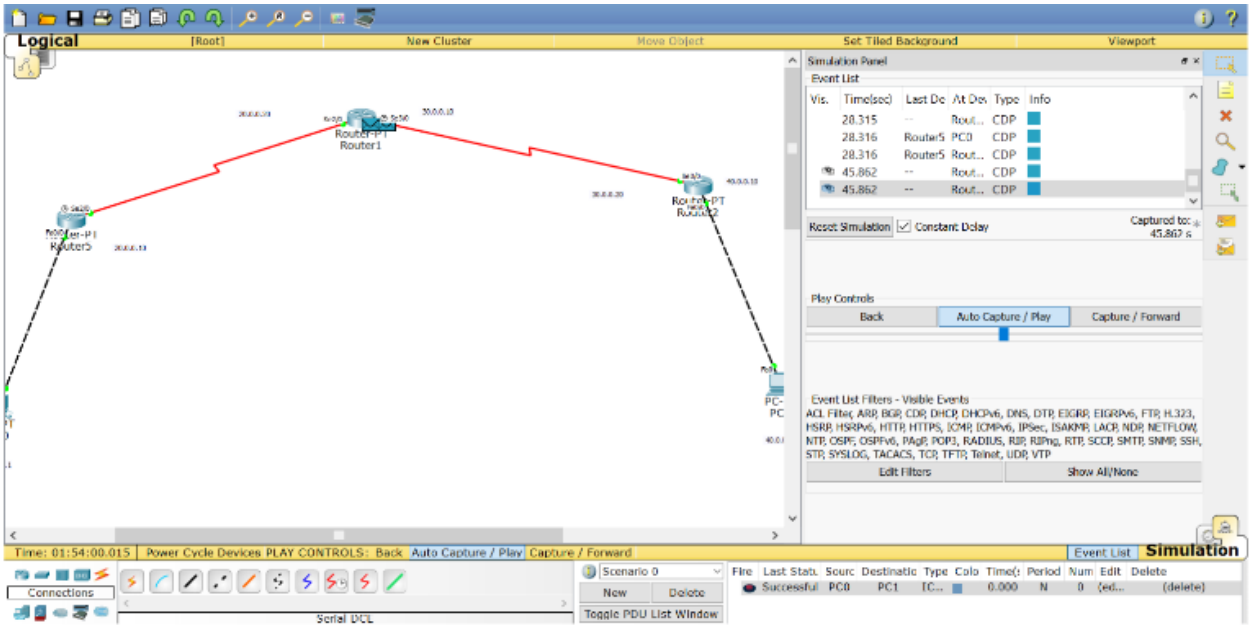




PROGRAM 2.2


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>

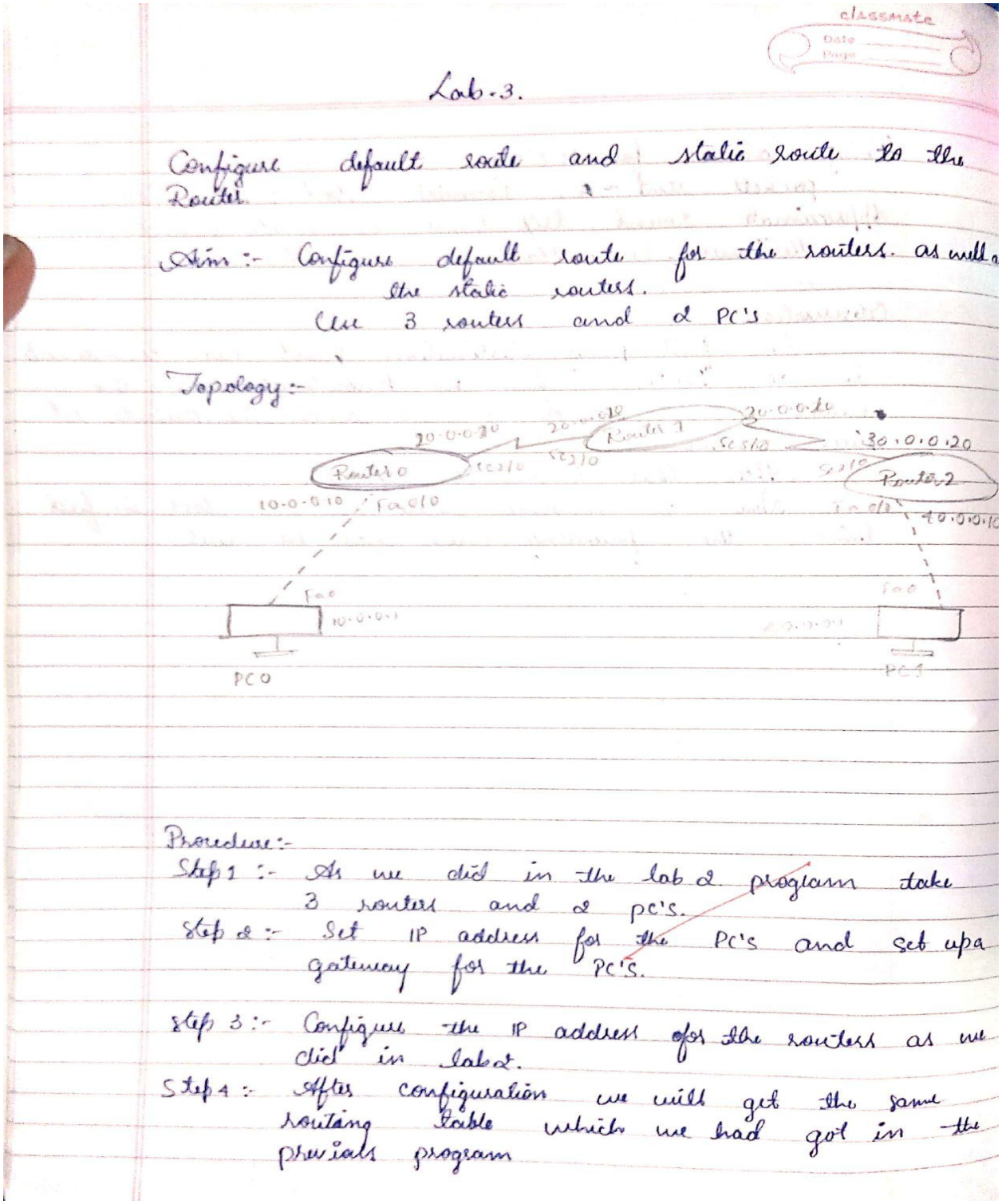

PC0
Physical
Config
Desktop
Custom Interface
Command Prompt
Packet Tracer PC Command Line 1.0
PC>ping 40.0.0.1
Pinging 40.0.0.1 with 32 bytes of data:
Reply from 10.0.0.10: Destination host unreachable.
Reply from 10.0.0.10: Destination host unreachable.
Reply from 10.0.0.10: Destination host unreachable.
Request timed out.
Ping statistics for 40.0.0.1:
Packets: Sent = 4, Received = 0, Lost = 4 (100%
loss),
PC>


Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
Router1
Router5
Simulation Panel
Event List
Vis.
Time(sec)
Last De
At Dev
Type
Info
28.315
28.316
28.316
45.862
45.862
CDP
Reset Simulation
Constant Delay
Play Controls
Back
Auto Capture / Play
Capture / Forward
Event List Filters - Visible Events
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
Scenario 0
New
Delete
Toggle PDU List Window
Fire
Last Statu
Sourc
Destinatio
Type
Colo
Time(s
Period
Num
Edit
Delete
Successful
PC0
PC1
0.000
(edit)
(delete)

WEEK 3

Configure default route, static route to the Router

OBSERVATION:



Lab-3.

Configure default route and static route to the Router.

Aim :- Configure default route for the routers. as well as the static routers.
Use 3 routers and 2 PC's

Topology :-

Procedure :-

Step 1 :- As we did in the lab 2 program take 3 routers and 2 pc's.

Step 2 :- Set IP address for the PC's and set upa gateway for the PC's.

Step 3 :- Configure the IP address for the routers as we did in lab 2.

Step 4 :- After configuration we will get the same routing table which we had got in the previals program

Ex:- Configure Router 0 IP address

Go to CLI.

Router > enable

Router # config t.

Router (config t) # interface fastethernet 0/0

Router (config t-if) # ip address 10.0.0.10 255.0.0.0

Router (config -if) # no shut

exit

Router (config) # interface serial 2/0

Router (config -if) # ip address 20.00.10 255.0.0.0

Router (config -if) # no shut

exit.

Ripeate this for other 2 routers as well.

IP Routing table:-

Router 0:

Router # show ip route

C 10.0.0.0/8 is directly connected, Fast Ethernet 0/0

C 20.0.0.0/8 is directly connected, serial 2/0.

Router 1: Same output as we got in Lab2

Router 2: —x—

Now ping output in PC0.

PC > ping 40.0.0.1.

Reply from 10.0.0.10 : destination host unreachable

ping statistics for 40.0.0.1 :

packets : sent 4 received = 0 lost = 4
(100% loss)

Observation :-

PC1 is not reachable since we haven't set up the ip route yet.

Now set up default route in router 0 and router 2. But for router 1 configure it has we did in lab 2 (By giving the ip address which it does not have knowledge about other networks).

In all 3 Routers CLI write config t then set route

Router 0 :

ip route 0.0.0.0 0.0.0.0 20.0.0.20

Router 1 :-

ip route 40.0.0.0 255.0.0.0 30.0.0.20

ip route 10.0.0.0 255.0.0.0 20.0.0.10

Router 2:

IP route 0.0.0.0 0.0.0.0 30.0.0.10

new IP route table.

Route 0.

C 10.0.0.0/8 is directly connected, FastEthernet 0/0

C 20.0.0.0/8 is directly connected, serial 2/0

S* 0.0.0.0/0 [1/0] via 20.0.0.20

Router 1.

S - 10.0.0.0/8 [1/0] via 20.0.0.10

C 20.0.0.0/8 is directly connected, serial 2/0

C 30.0.0.0/8 is directly connected, serial 3/0

S 40.0.0.0/8 [1/0] via 30.0.0.20

Router 2.

C 30.0.0.0/8 is directly connected, Serial2/0
C 40.0.0.0/8 is directly connected, FastEthernet0/0.
S* 0.0.0.0/0 [1/0] via 30.0.0.10

Ping messages:

ping in PC0.

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=2ms TTL=125.
Reply from 40.0.0.1 bytes=32 time=2ms TTL=125
Reply from 40.0.0.1 bytes=32 time=2ms TTL=125
Reply from 40.0.0.1 bytes=32 time=2ms TTL=125

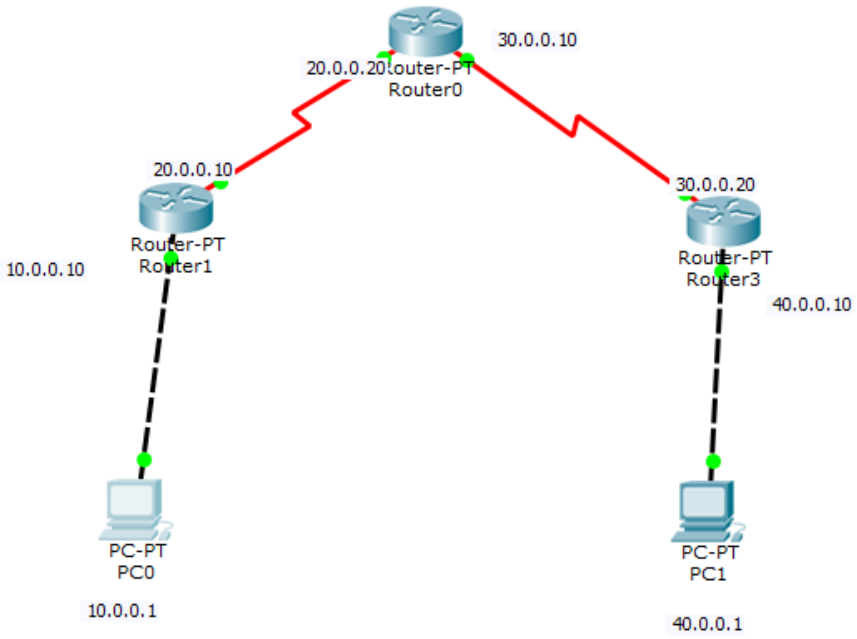
ping statistics for 40.0.0.1
packets sent=4, Received=4, lost=0 (0% loss).
Approximate round trip times in milli-seconds
Minimum=2ms, Maximum=2ms. Average=2ms.

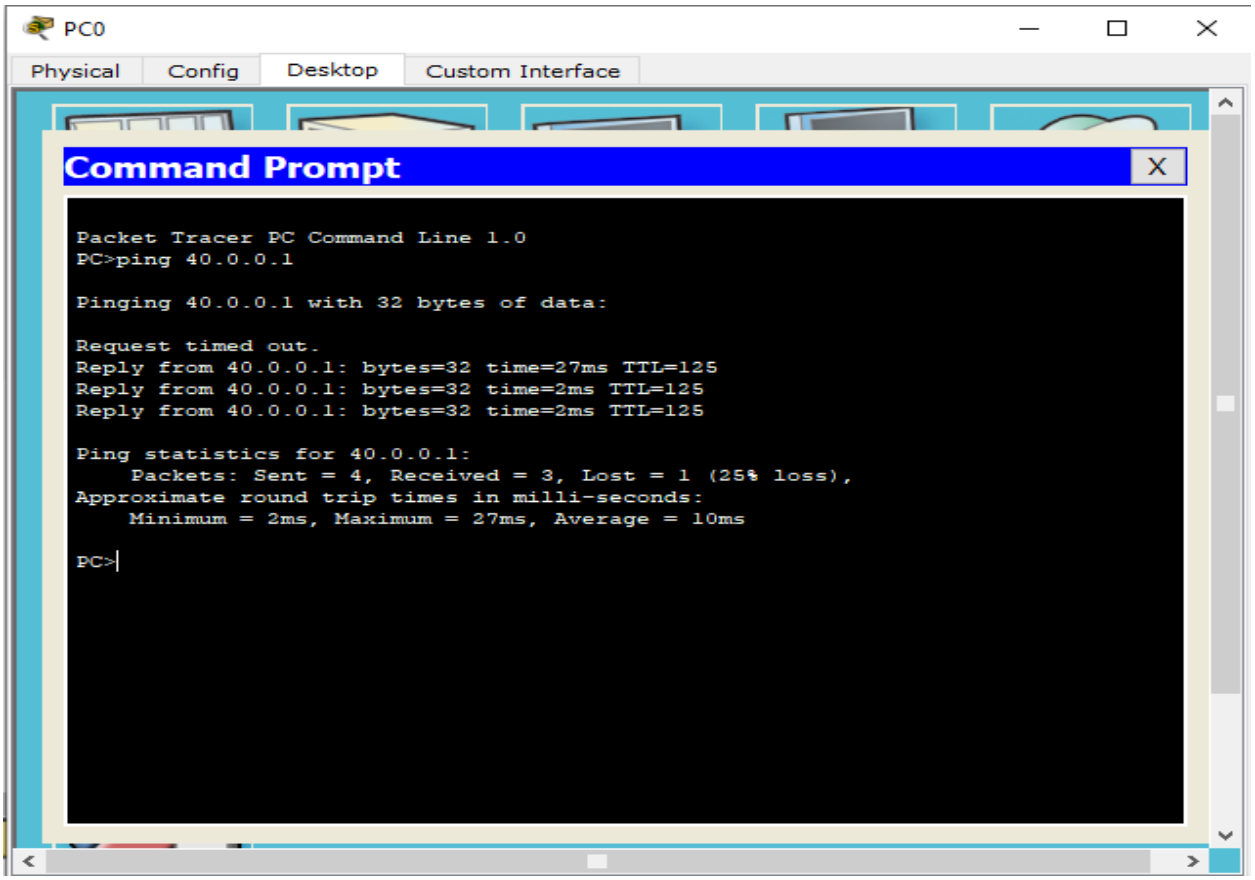
Observation:-

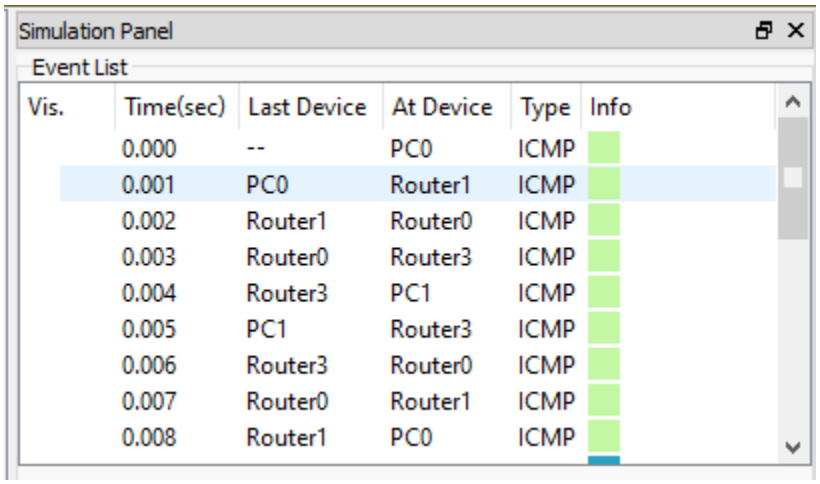
After setting a default route in routes 0 and router 2 whatever packet we send it will be redirected to the router 1. Then router 1 has the complete knowledge about all other networks.

Then Router 1 will send the packet to the respective network (Router). Then the receiver or destination device will receive the data (packet), it will send the acknowledgement after receiving the packet.

OUTPUT SCREENS:



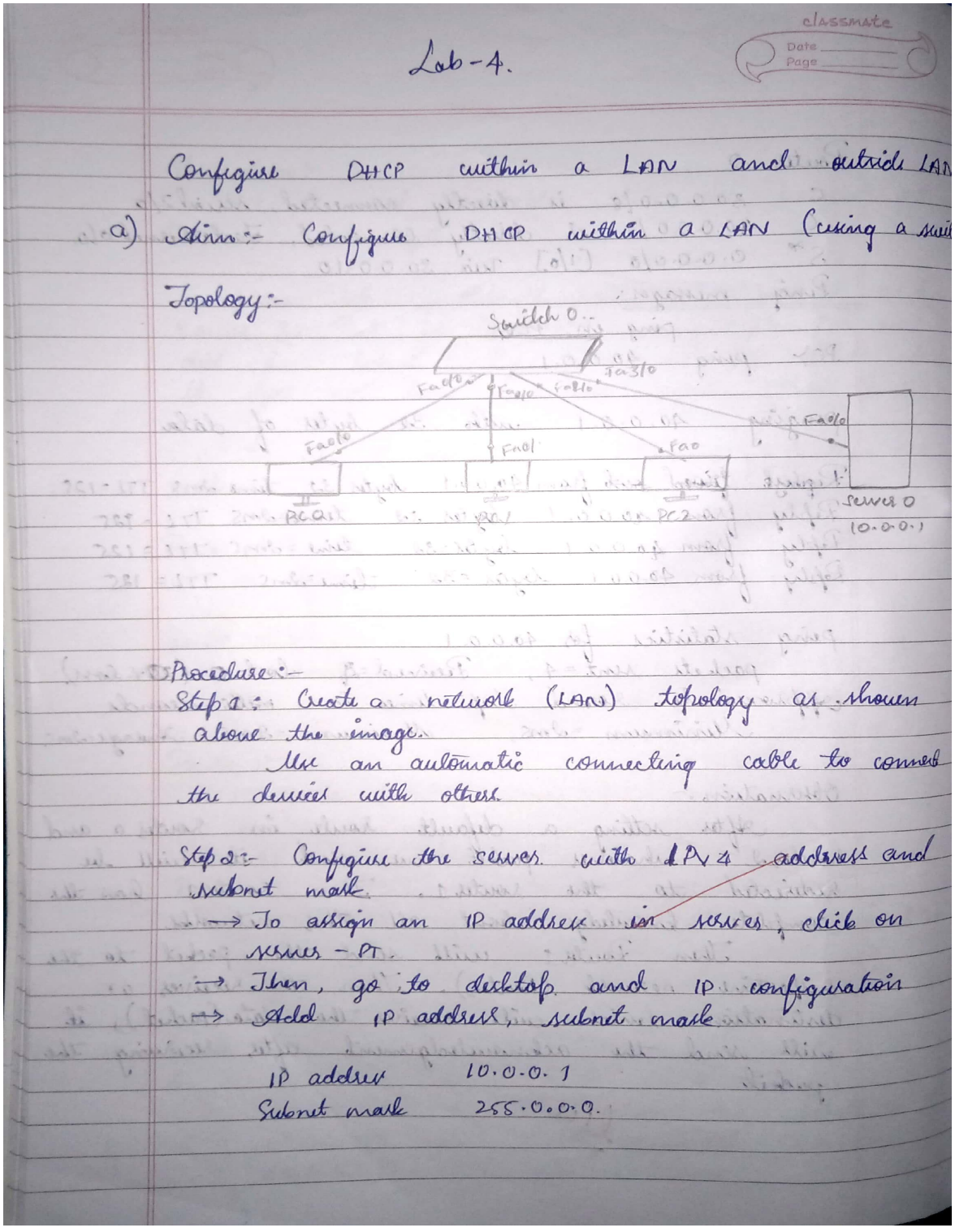



Simulation Panel
Event List
Vis.
Time(sec)
Last Device
At Device
Type
Info
0.000 -- PC0 ICMP
0.001 PC0 Router1 ICMP
0.002 Router1 Router0 ICMP
0.003 Router0 Router3 ICMP
0.004 Router3 PC1 ICMP
0.005 PC1 Router3 ICMP
0.006 Router3 Router0 ICMP
0.007 Router0 Router1 ICMP
0.008 Router1 PC0 ICMP

WEEK 4

Configure DHCP within a LAN and outside LAN.

OBSERVATION:



Lab-4.

Configure DHCP within a LAN and outside LAN

a) Aim:- Configure DHCP within a LAN (using a switch

Topology:-

Procedure:-

Step 1:- Create a network (LAN) topology as shown above the image.

Use an automatic connecting cable to connect the devices with others.

Step 2:- Configure the servers with IPv4 address and subnet mask.

→ To assign an IP address in server, click on server - PT.

→ Then, go to desktop and IP configuration

→ Add IP address, subnet mask

IP address 10.0.0.1

Subnet mask 255.0.0.0

Step3 :- Configuring DHCP server

To configure the DHCP server first.

→ click on server PT then, go to services

→ select DHCP and set the service on

→ Then set start IP address then save

Step4 :- Configure the PC's and changing the IP configuration.

→ To assign an IP address in PC0, click on PC0

→ Then, go to desktop and IP configuration and there in IP configuration (Not IPV6). change its state from static to DHCP.

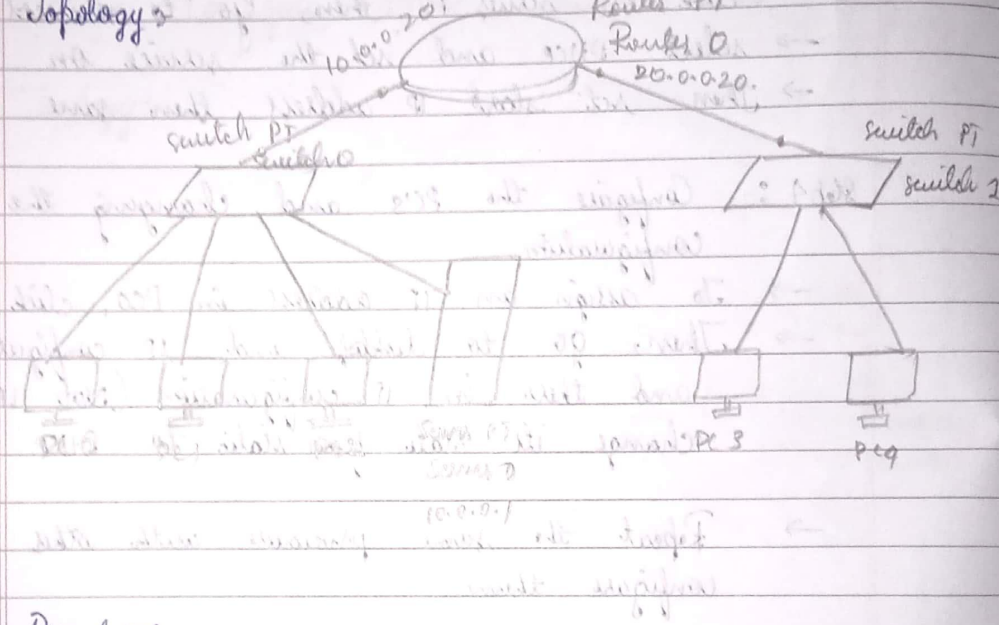
→ Repeat the same procedure with other PCs to configure them.

Observation :-

When we select a PDU and ~~start PC and~~ source PC and destination PC. The packet will be sent to the destination PC. The DHCP server will allocate a IP address for all the PC's. That is why even without saving or configuring the IP address of PC's the LAN will work properly.

b) Aim :- Configure DHCP outside LAN

Topology :-



Procedure :-

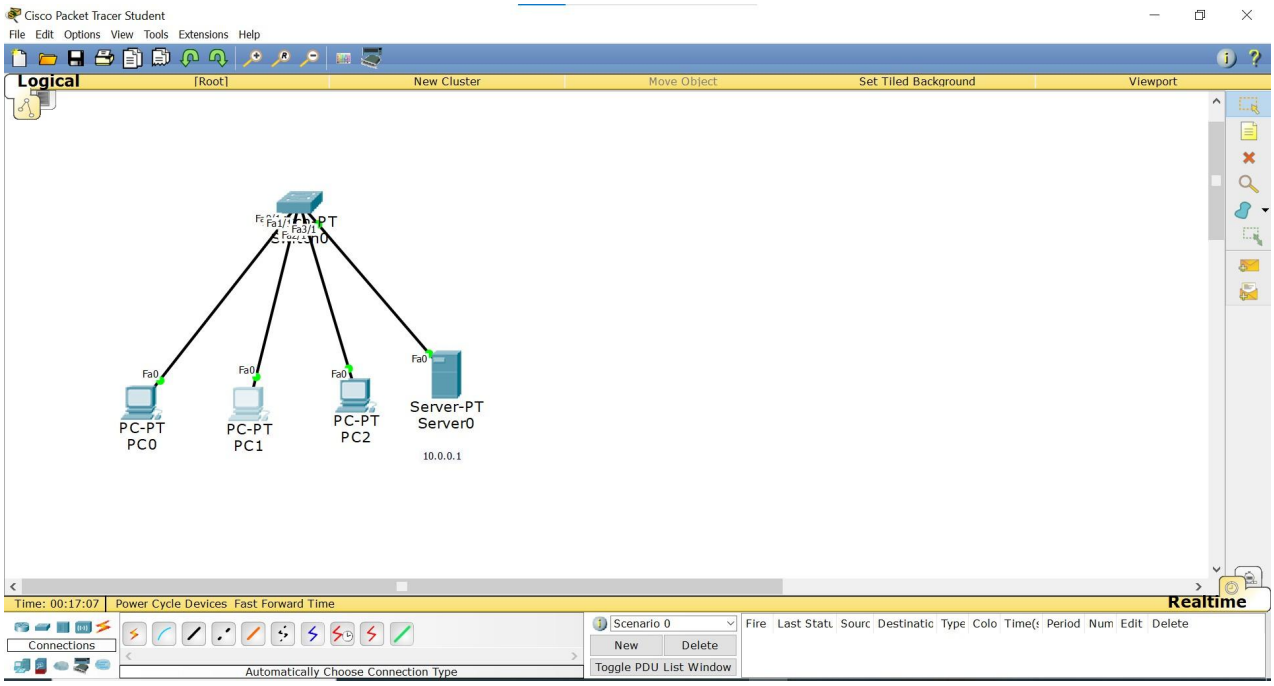
Step 1 :-
Create a above network topology.
Use an automatic cable to connect the devices with other.

Step 2 :- Configure the server with IPv4 address and subnet mask and gateway [basically repeating the steps of a)].
→ click on server
→ go to desktop → IP configuration
→ Add IP address, subnet mask and gateway

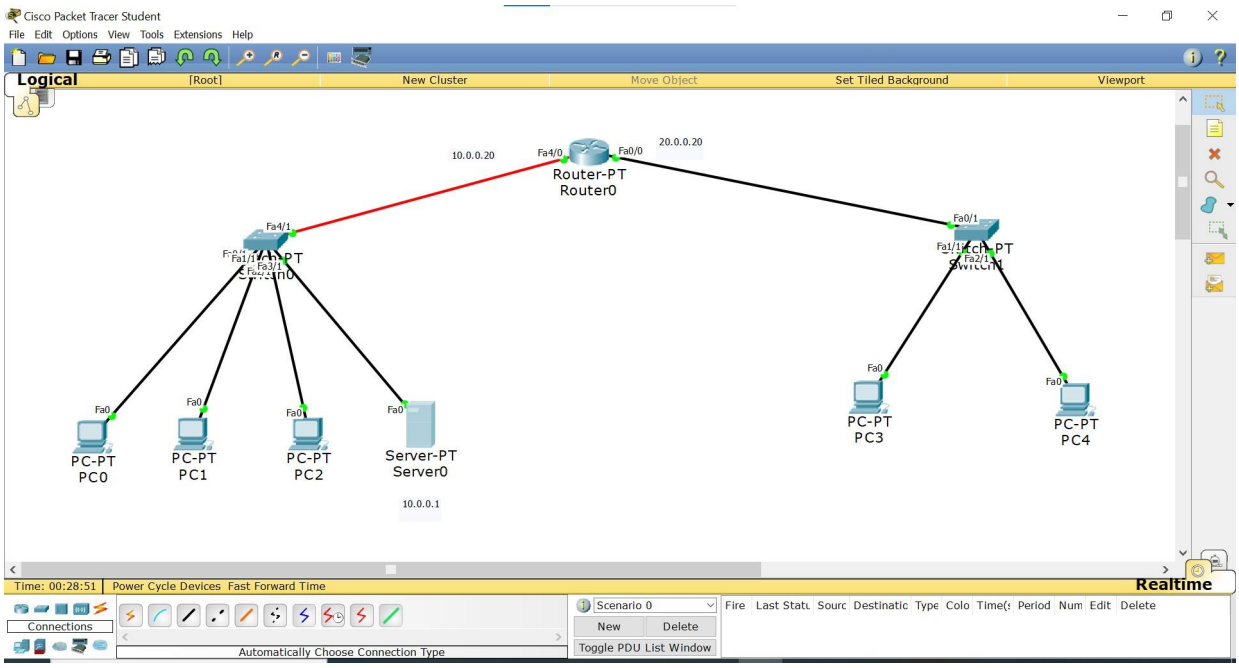
IP address	10.0.0.1
Subnet mask	255.0.0.0
Gateway	10.0.0.20

TOPOLOGY:

PROGRAM 4.1:

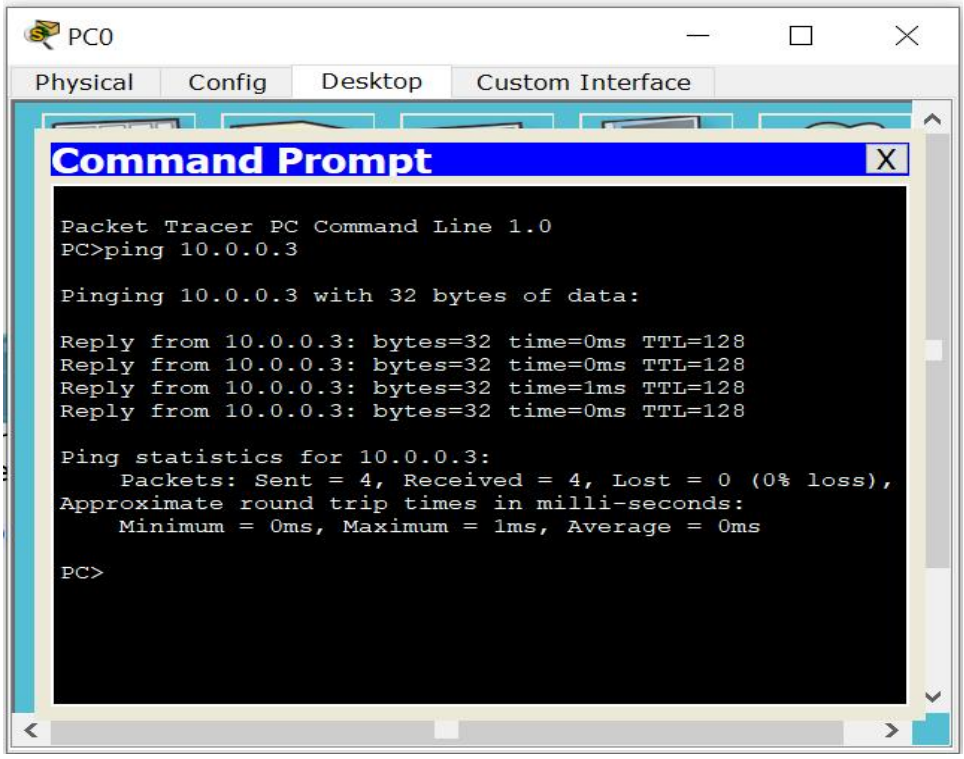


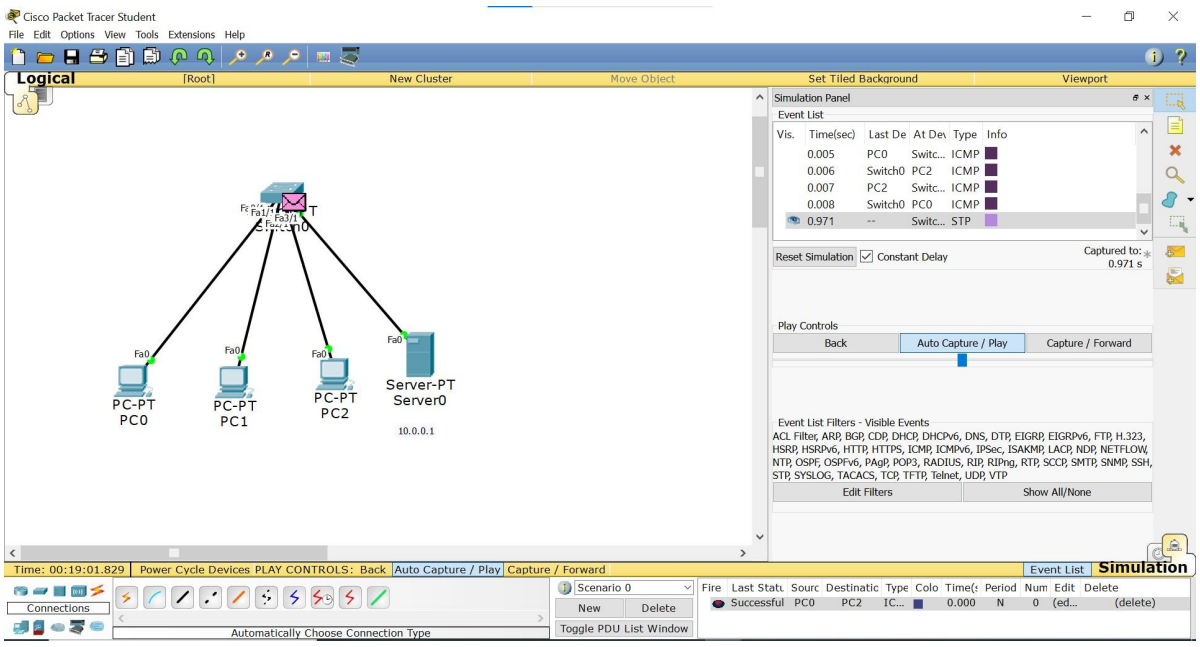
PROGRAM 4.2:



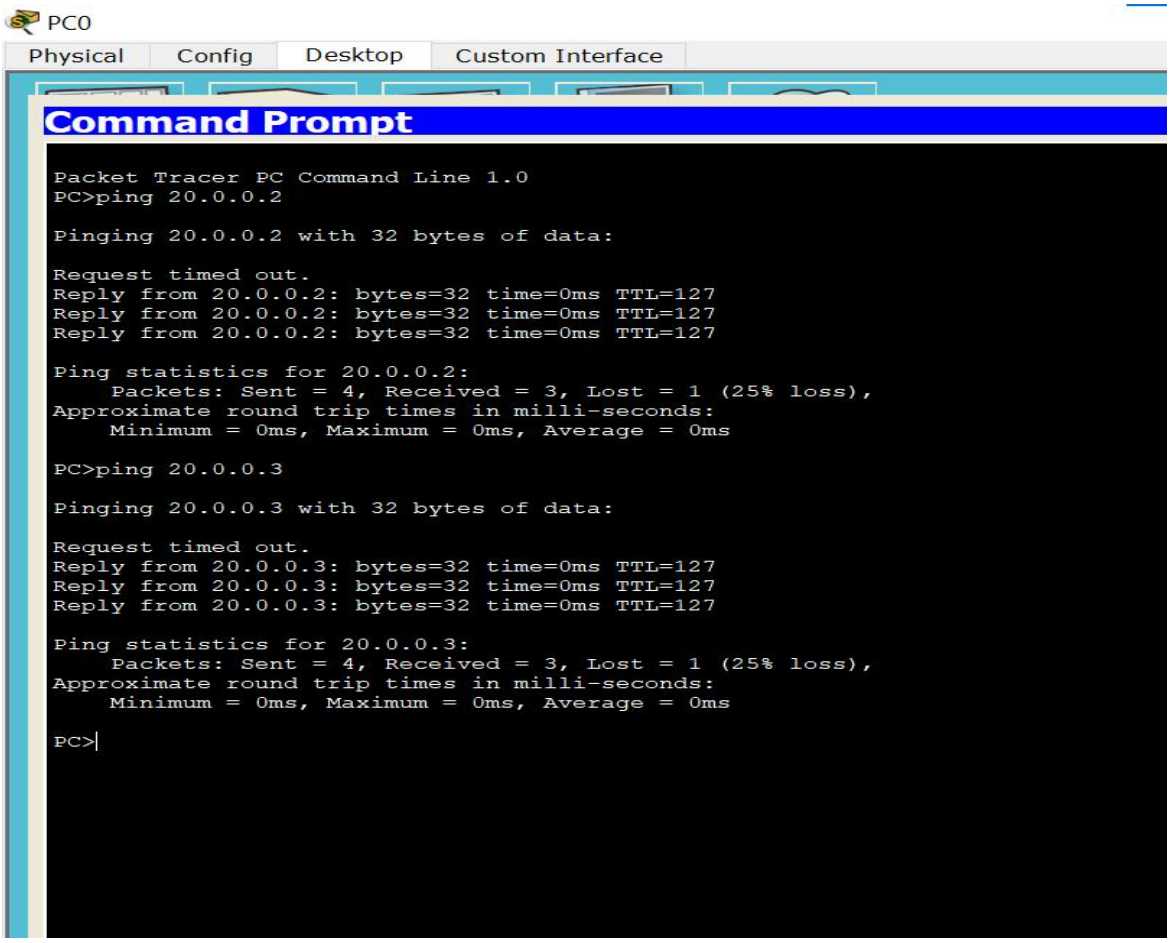
OUTPUT:

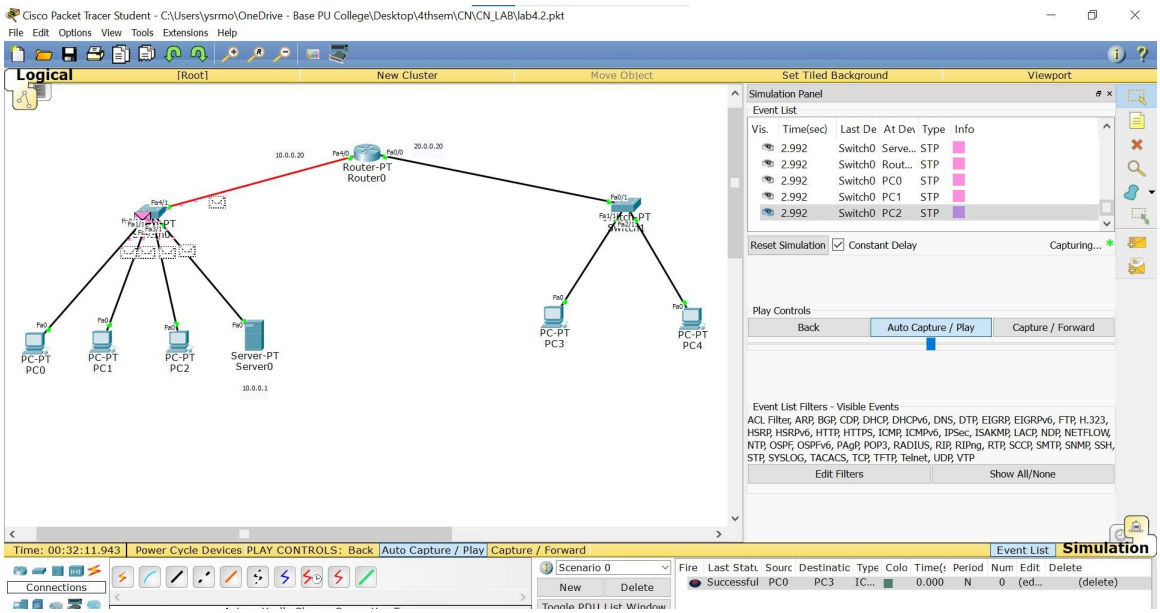
PROGRAM 4.1:





PROGRAM 4.2:

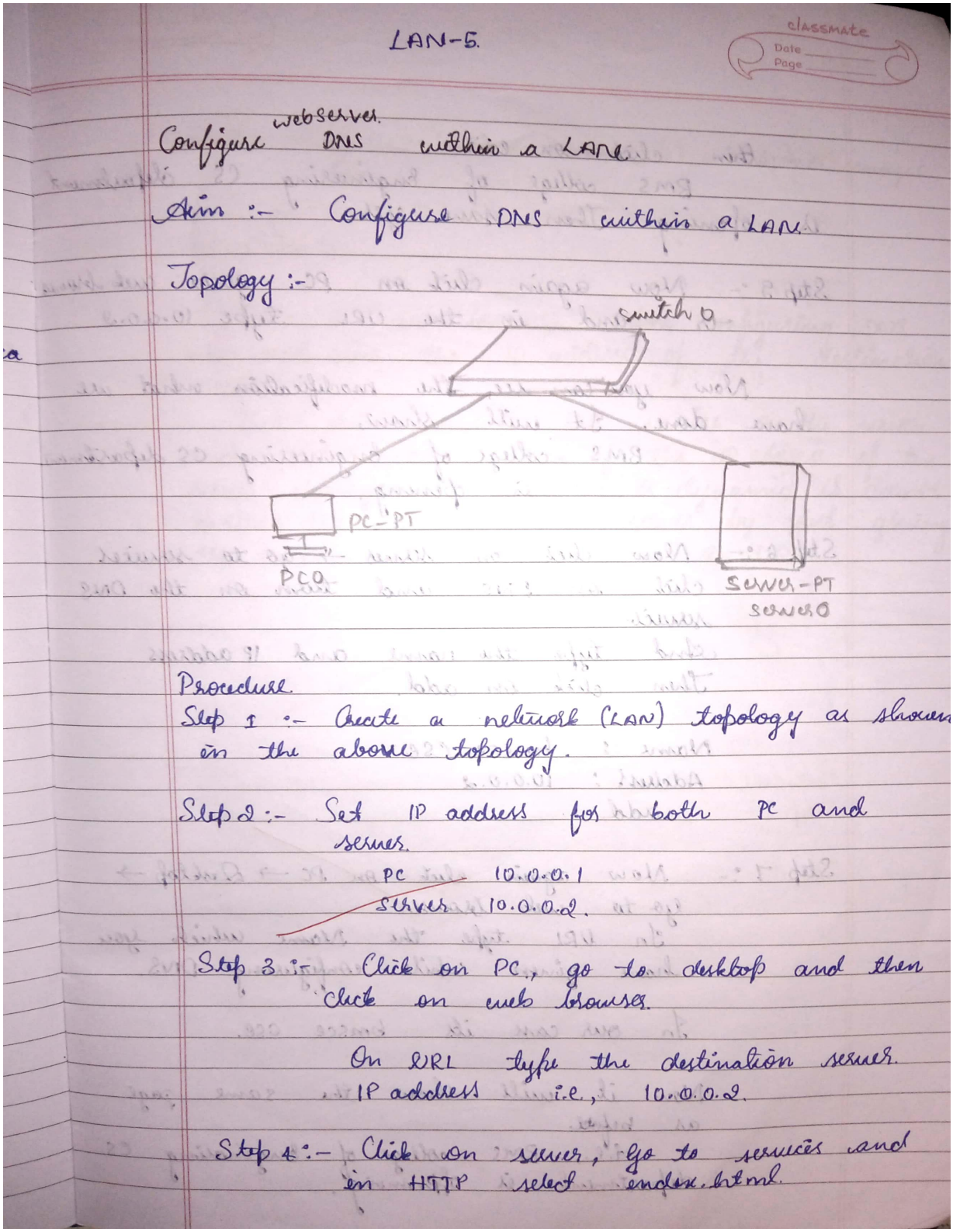

PC0
Physical
Config
Desktop
Custom Interface
Command Prompt
Packet Tracer PC Command Line 1.0
PC>ping 20.0.0.2
Pinging 20.0.0.2 with 32 bytes of data:
Request timed out.
Reply from 20.0.0.2: bytes=32 time=0ms TTL=127
Reply from 20.0.0.2: bytes=32 time=0ms TTL=127
Reply from 20.0.0.2: bytes=32 time=0ms 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
Reset Simulation
Constant Delay
Capturing...
Play Controls
Back
Auto Capture / Play
Capture / Forward
Event List Filters - Visible Events
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

WEEK 5

Configure Web Server, DNS within a LAN.

OBSERVATION:



LAN-5.

Configure webserver DNS within a LAN

Aim :- Configure DNS within a LAN.

Topology :-

Procedure.

Step 1 :- Create a network (LAN) topology as shown in the above topology.

Step 2 :- Set IP address for both PC and server.

PC 10.0.0.1

Server 10.0.0.2.

Step 3 :- Click on PC, go to desktop and then click on web browser.

On URL type the destination server IP address i.e., 10.0.0.2.

Step 4 :- Click on server, go to services and in HTTP select index.html.

Then click on edit
BMS college of Engineering CS department is opening Then save yes.

Step 5 :- Now again click on PC go to web browser and in the URL type 10.0.0.2.

Now you can see the modification what we have done. It will show,
BMS college of Engineering CS department is opening.

Step 6 :- Now click on server → go to services click on DNS and turn on the DNS service.
And type the name and IP address Then click on add.

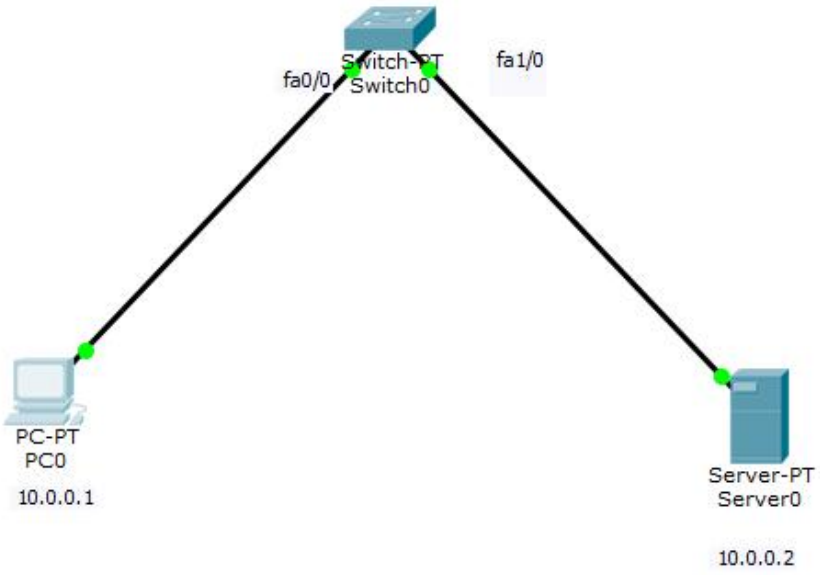
Name : bmscse.cse
Address : 10.0.0.2
add.

Step 7 :- Now again click on PC → Desktop → go to web browser
In URL type the Name which you have given while configuring DNS.

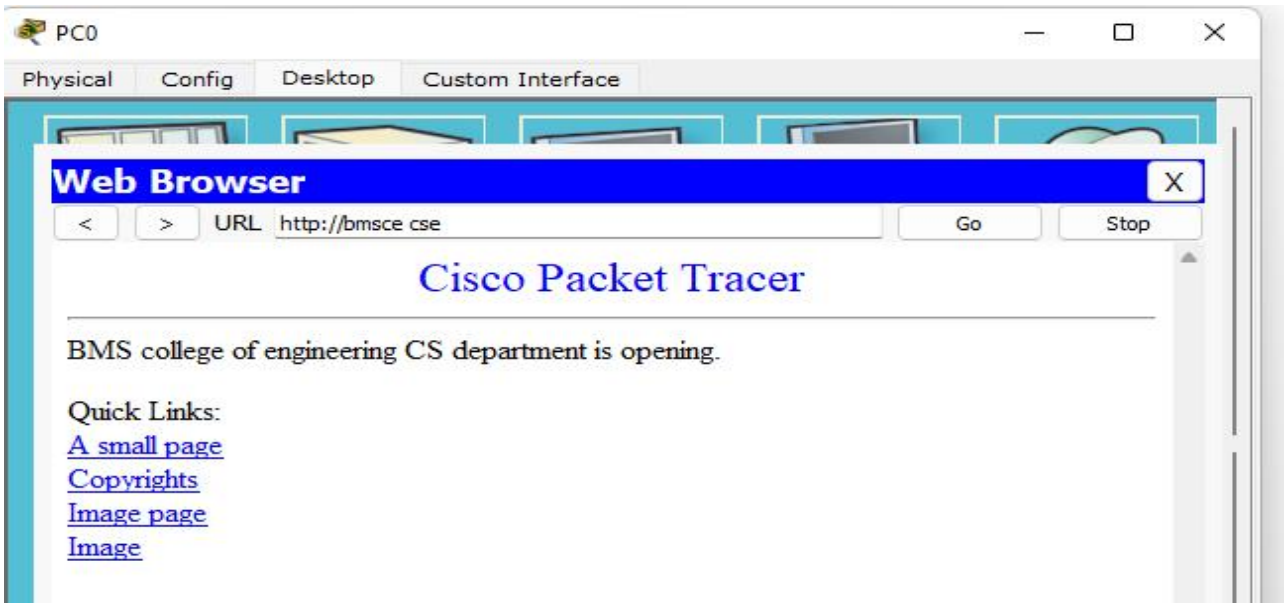
In our case its bmscse.cse.

Now it will show the same page as before.
i.e., BMS college of Engineering CS department is opening.

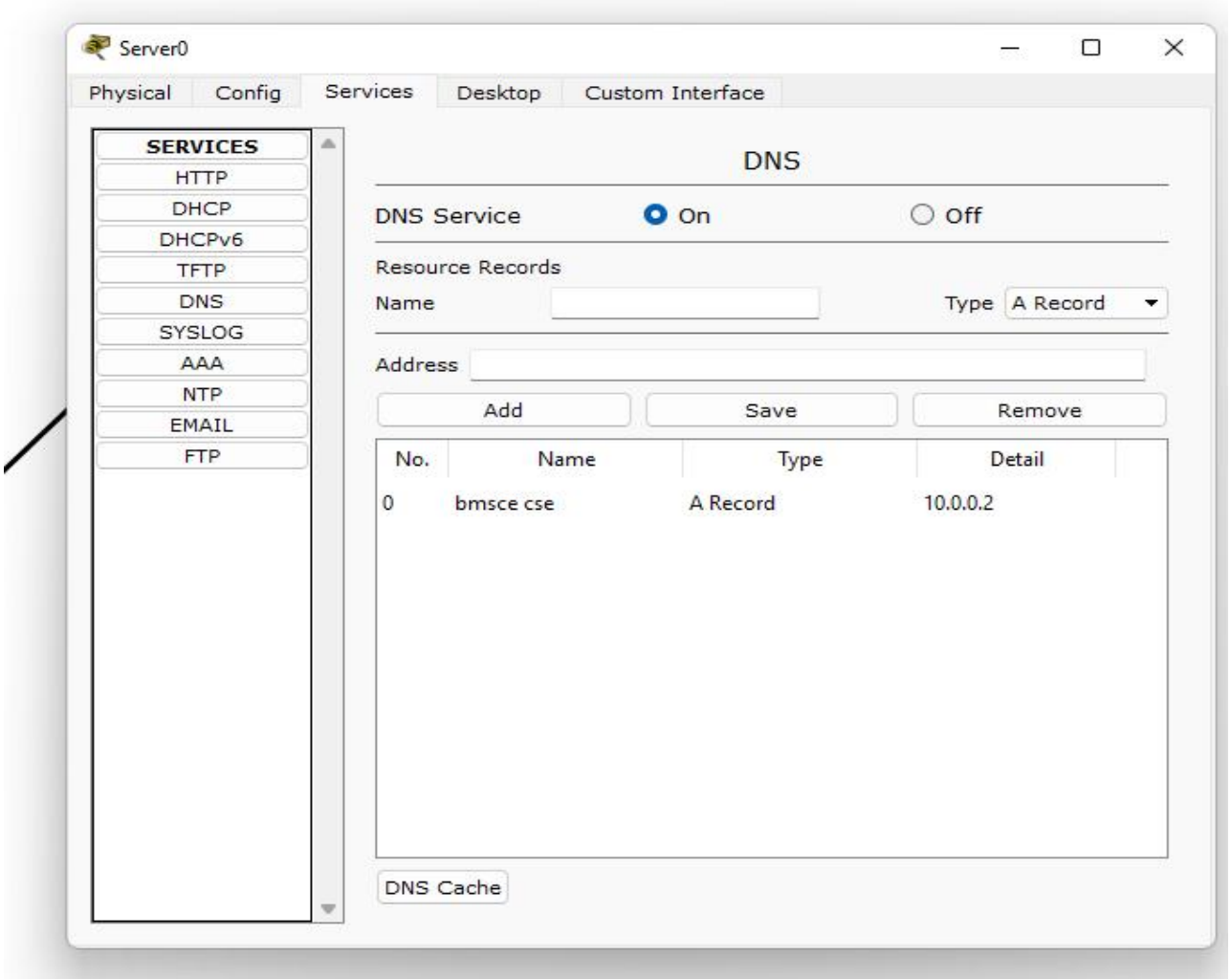
TOPOLOGY:



OUTPUT:



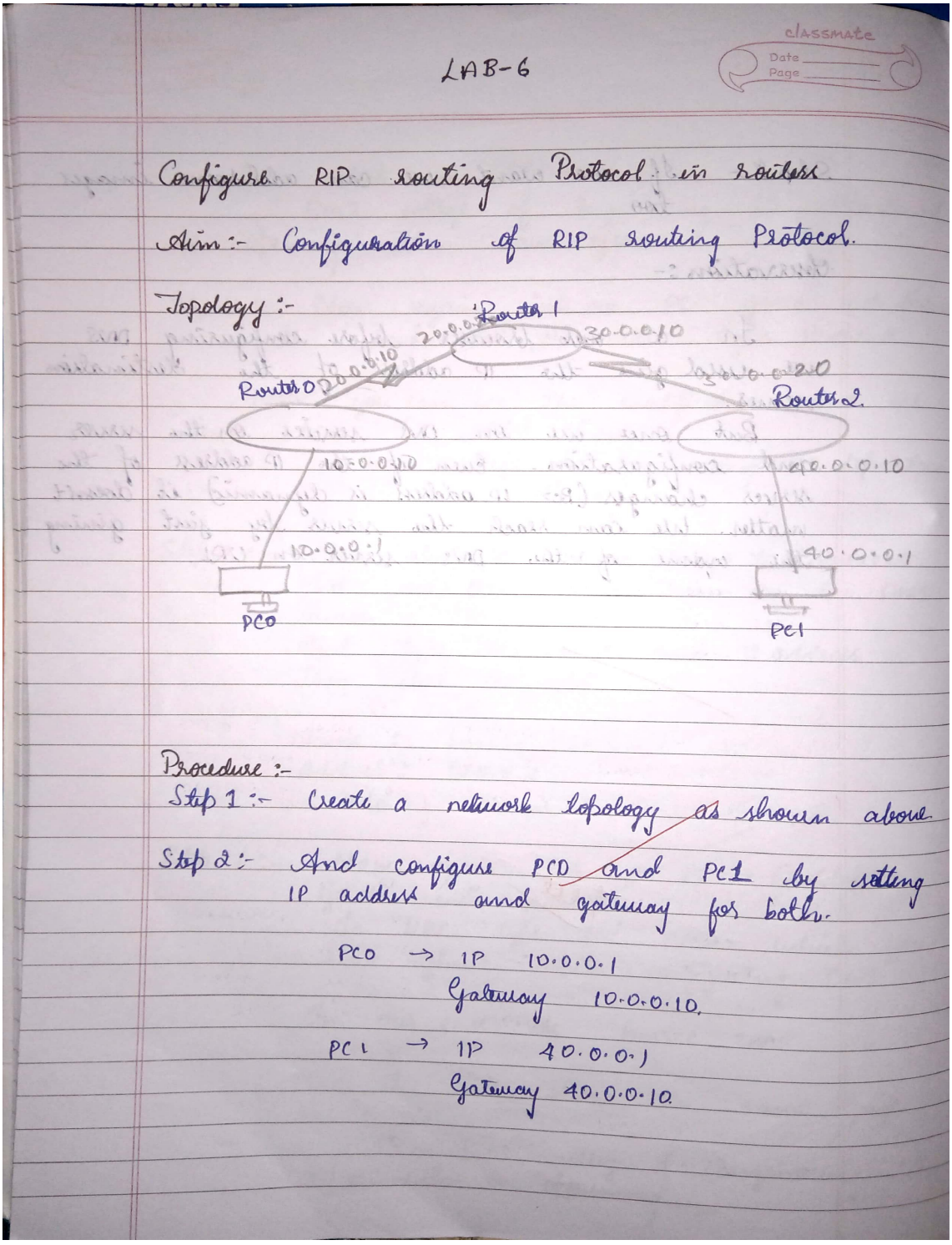

PC0
Physical
Config
Desktop
Custom Interface
Web Browser
X
<
>
URL
http://10.0.0.2
Go
Stop
Cisco Packet Tracer
BMS college of engineering CS department is opening.
Quick Links:
A small page
Copyrights
Image page
Image


Server0
Physical
Config
Services
Desktop
Custom Interface
SERVICES
HTTP
DHCP
DHCPv6
TFTP
DNS
SYSLOG
AAA
NTP
EMAIL
FTP
DNS
DNS Service
On
Off
Resource Records
Name
Type
A Record
Address
Add
Save
Remove
No.
Name
Type
Detail
0
bmsce cse
A Record
10.0.0.2
DNS Cache

WEEK 6

Configure RIP routing Protocol in Routers.

OBSERVATION:



LAB-6

Configure RIP routing Protocol in routers

Aim :- Configuration of RIP routing Protocol.

Topology :-

Procedure :-

Step 1 :- Create a network topology as shown above.

Step 2 :- And configure PC0 and PC1 by setting IP address and gateway for both.

PC0 → IP 10.0.0.1
Gateway 10.0.0.10

PC1 → IP 40.0.0.1
Gateway 40.0.0.10

Step 3:- Configure the routers.

Router 0.

enable
config t.
Router (config)# int Fa0/0.
Router (config-if)# ip address 10.0.0.10 255.0.0.0.
Router (config-if)# no shut.
exit.

int se2/0.
ip address 20.0.0.10 255.0.0.0.
encapsulation PPP
clockrate 64000
no shut
exit.

NOTE :- The encapsulation ppp should be given to all the routers and 'clockrate 64000' command should be only given to the clock symboled ports of the routers (i.e. open circle).

→ For making the routers to know about the other devices, in the previous 2 experiments we used static and the others with dynamic address but here we use routing protocol algorithm that itself makes the router to know other devices.

→ router rip
→ network 20.0.0.0 } router 2
network 30.0.0.0.

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:-

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 : 0.ms TTL: 126
Reply from 40.0.0.1 bytes = 32 time : 0ms TTL: 128
Reply from 40.0.0.1 bytes = 32 time : 0ms TTL: 128
Reply from 40.0.0.1 bytes = 32 time : 0ms TTL: 128.

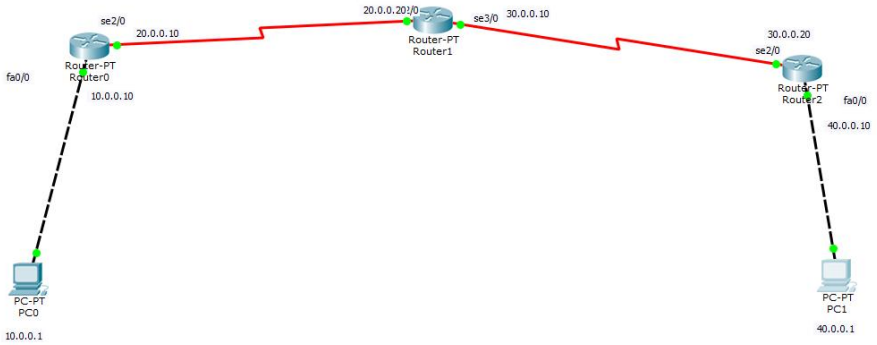
ping statistics from 40.0.0.1
packets sent = 4 Received 4 lost = 0 (0% loss)
Approximate round trip times in ms
minimum = 0ms maximum = 0ms Average = 0ms

Observation:-

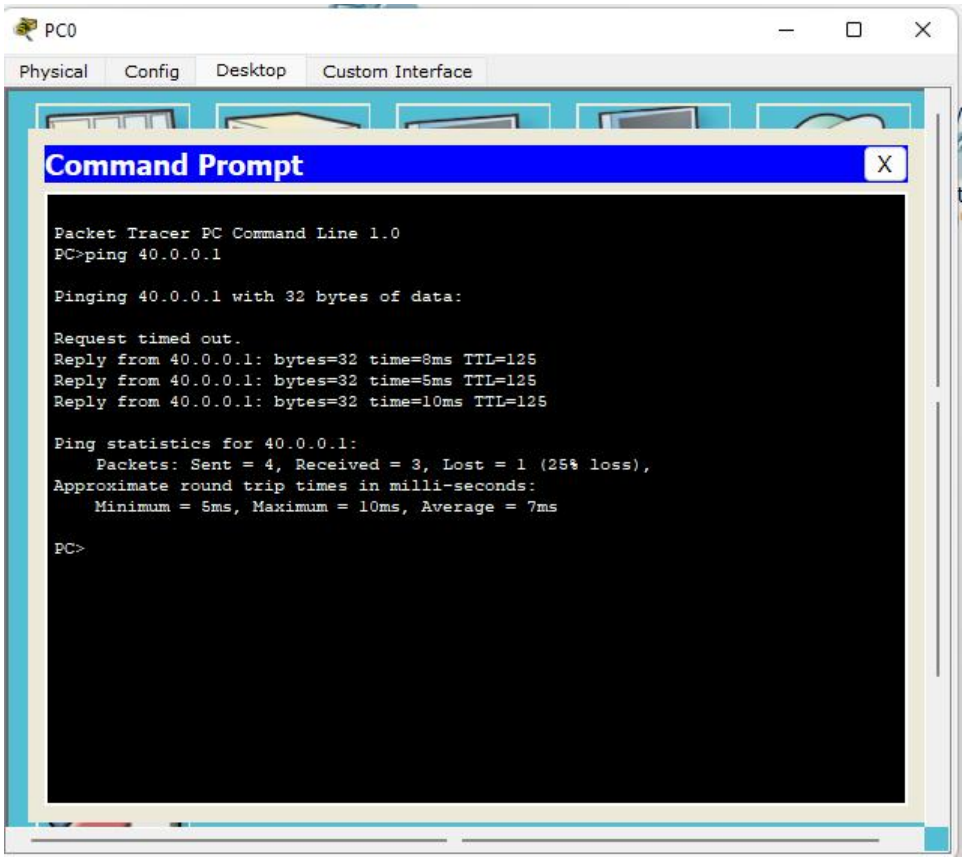
RIP is the routing information protocol it a diffusing vector protocol that uses hop count as its primary metric. RIP defines how traffic routers should share information when moving traffic among an interconnected group of local area networks.

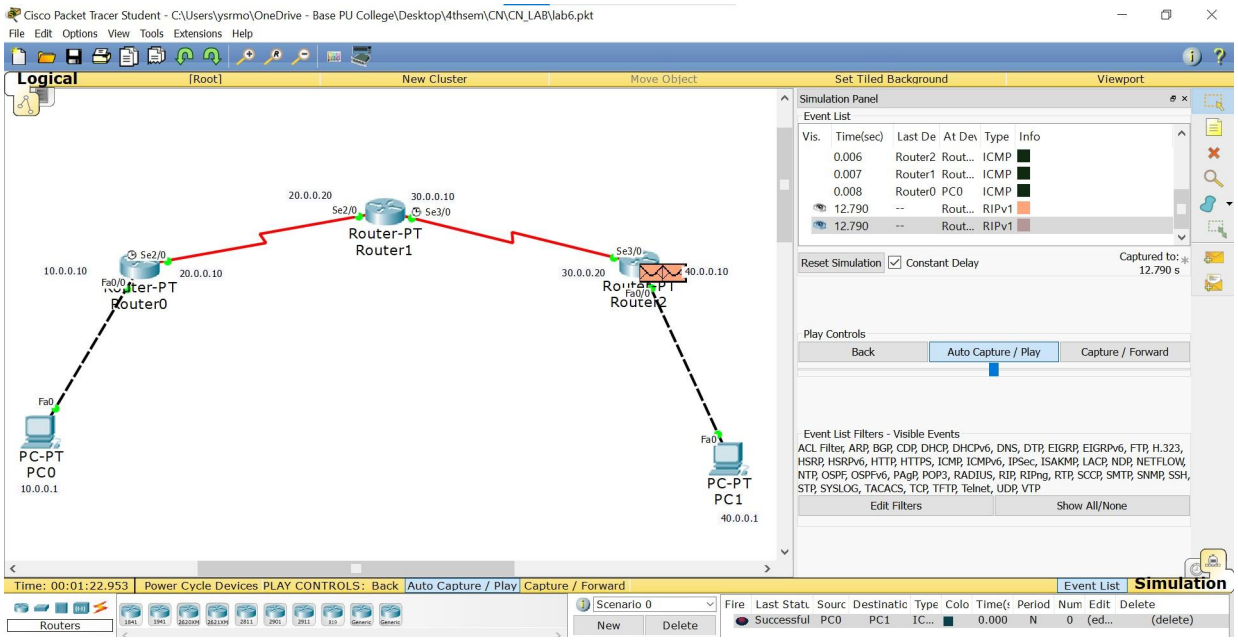
→ The RIP protocol here, used to connect the routers to one other and PC's using RIP protocol and message is pinged successfully

TOPOLOGY:



OUTPUT:

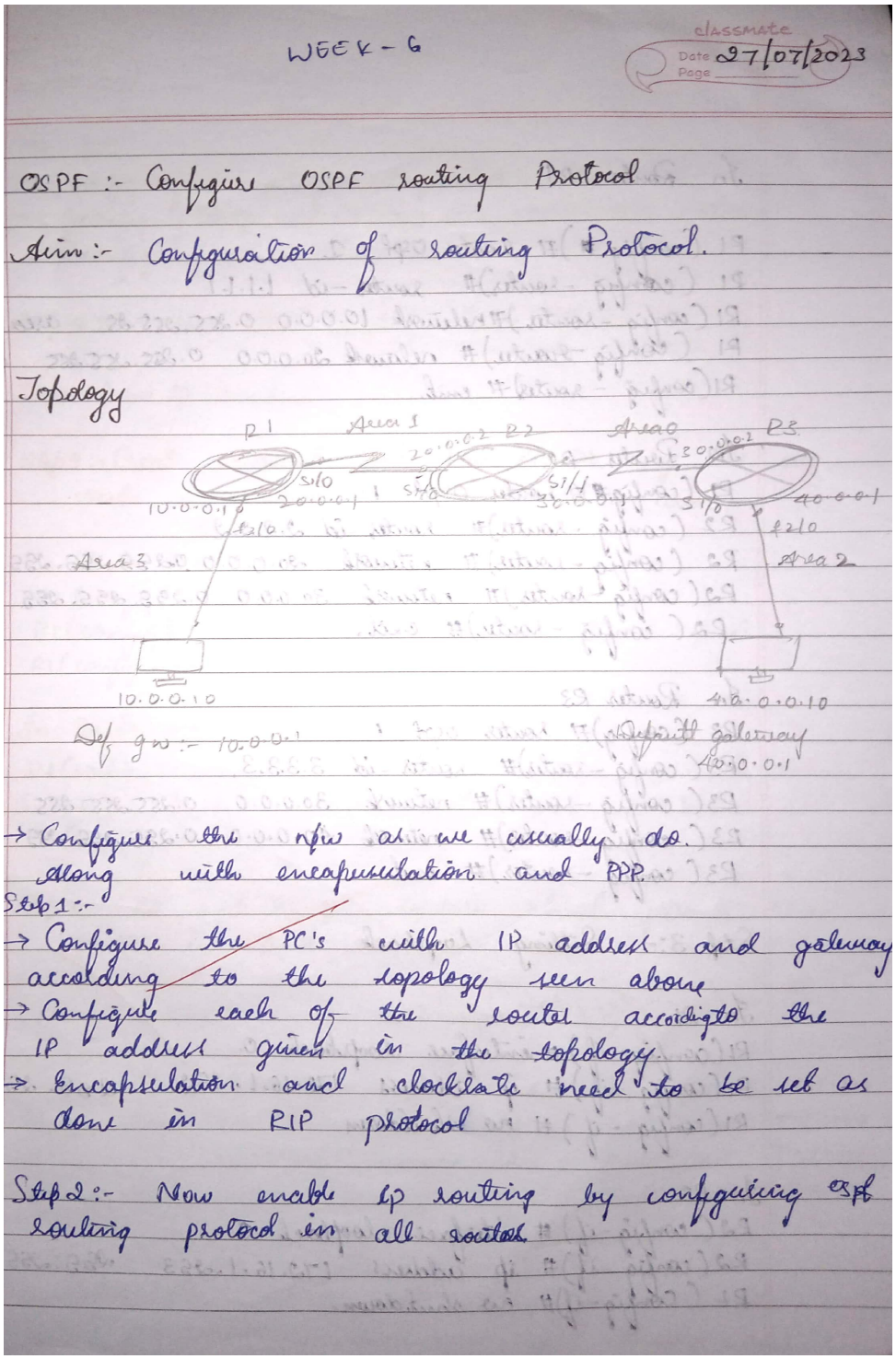



Cisco Packet Tracer Student - C:\Users\ysrmo\OneDrive - Base PU College\Desktop\4thsem\CN\CN_LAB\lab6.pkt
File Edit Options View Tools Extensions Help
Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
20.0.0.20
30.0.0.10
Se2/0
Se3/0
Router-PT
Router1
Se2/0
10.0.0.10
20.0.0.10
Fa0/0
Router0
Se3/0
30.0.0.20
40.0.0.10
Fa0/0
Router2
Fa0
PC-PT
PC0
10.0.0.1
Fa0
PC-PT
PC1
40.0.0.1
Simulation Panel
Event List
Vis. Time(sec) Last De At Dev Type Info
0.006 Router2 Rout... ICMP
0.007 Router1 Rout... ICMP
0.008 Router0 PC0 ICMP
12.790 -- Rout... RIPv1
12.790 -- Rout... RIPv1
Reset Simulation
Constant Delay
Captured to:
12.790 s
Play Controls
Back
Auto Capture / Play
Capture / Forward
Event List Filters - Visible Events
ACL Filter, ARP, BGP, CDP, DHCP, DHCPv6, DNS, DTP, EIGRP, EIGRPv6, FTP, H.323, HSRP, HSRPv6, HTTP, HTTPS, ICMP, ICMPv6, IPSec, ISAKMP, LACP, NDP, NETFLOW, NTP, OSPF, OSPFv6, PAgP, POP3, RADIUS, RIP, RIPng, RTP, SCCP, SMTP, SNMP, SSH, STP, SYSLOG, TACACS, TCP, TFTP, Telnet, UDP, VTP
Edit Filters
Show All/None
Time: 00:01:22.953
Power Cycle Devices PLAY CONTROLS: Back
Auto Capture / Play
Capture / Forward
Event List
Simulation
Routers
Scenario 0
New
Delete
Fire Last Statu Sourc Destinatic Type Colo Time(s Period Num Edit Delete
Successful PC0 PC1 IC... 0.000 N 0 (ed... (delete)

WEEK 7

Configure OSPF routing protocol.

OBSERVATION:



WEEK - 6

Date 27/07/2023

OSPF :- Configure OSPF routing Protocol

Aim :- Configuration of routing Protocol.

Topology

→ Configure the nfw as we usually do. Along with encapsulation and PPP.

Step 1 :-

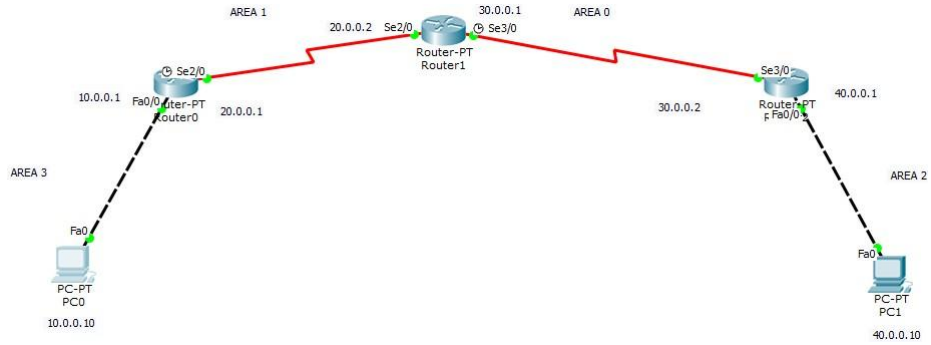
→ Configure the PC's with IP address and gateway according to the topology seen above

→ Configure each of the router according to the IP address given in the topology.

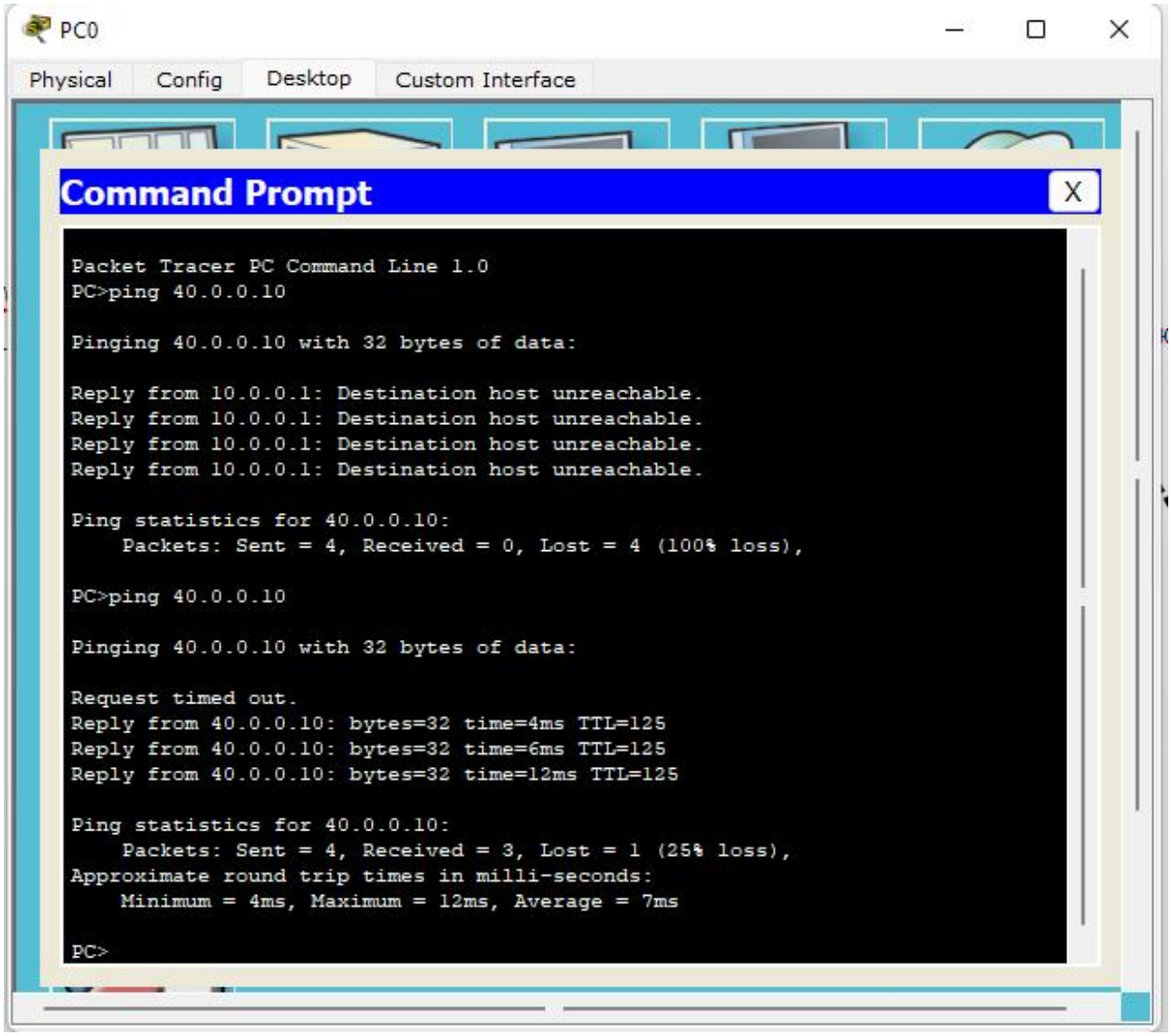
→ Encapsulation and clockrate need to be set as done in RIP protocol

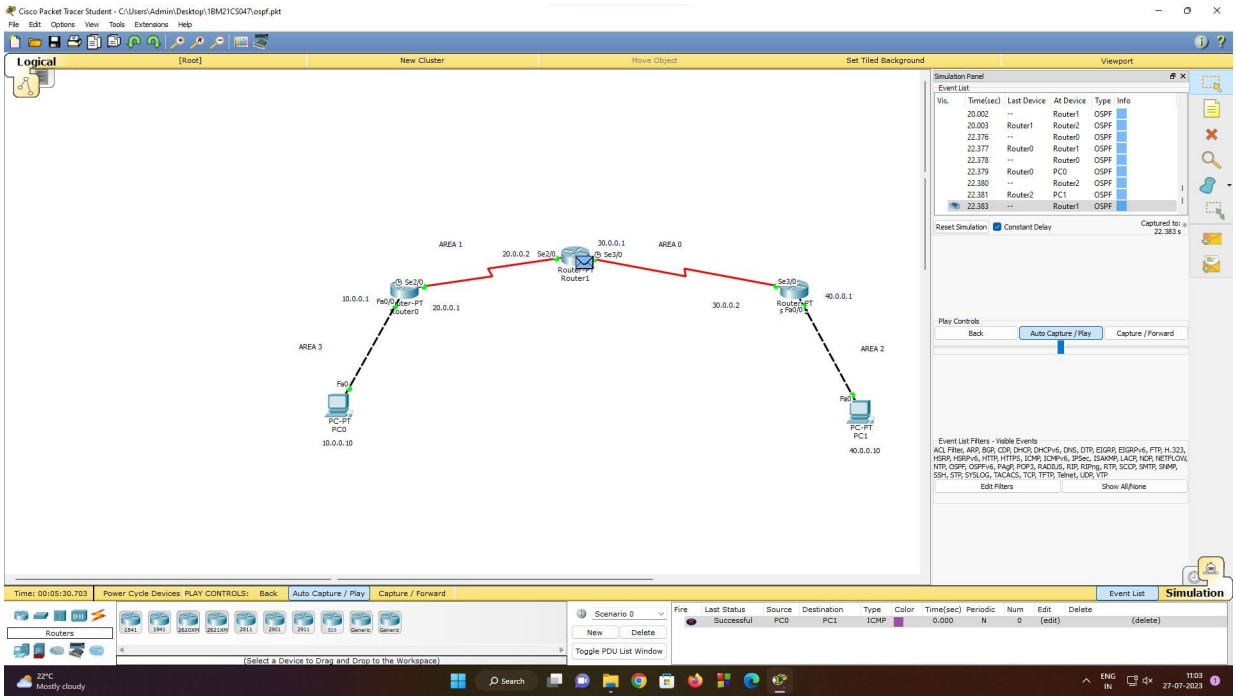
Step 2 :- Now enable ip routing by configuring ospf routing protocol in all routers

TOPOLOGY:



OUTPUT:

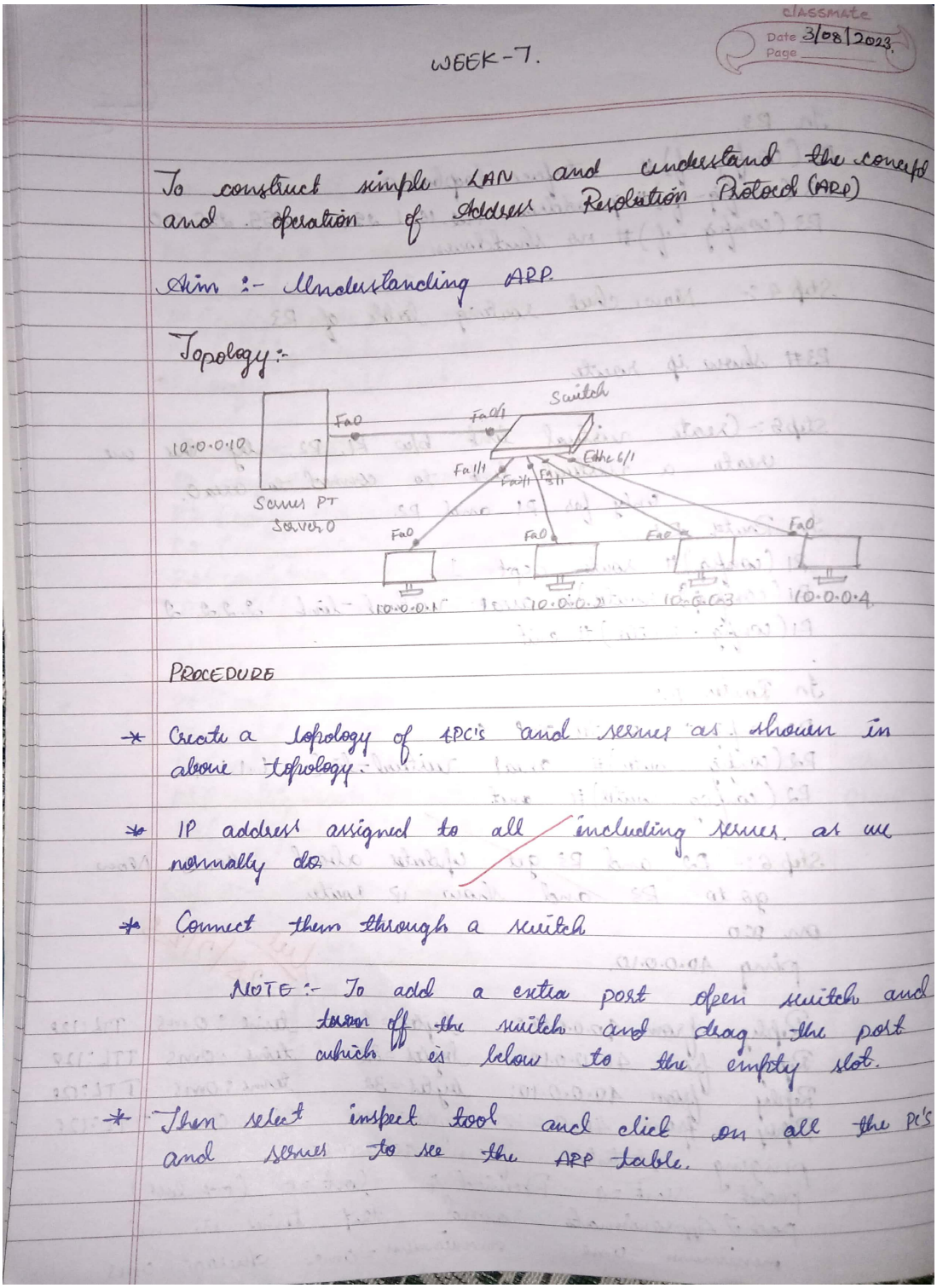



Cisco Packet Tracer Student - C:\Users\Admin\Desktop\1BM21CS047\ospf.pkt
File Edit Options View Tools Extensions Help
Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
AREA 1
AREA 0
AREA 3
AREA 2
30.0.0.1
20.0.0.2
Se2/0
Se3/0
Router-PT
Router1
Router-PT
Router0
10.0.0.1
Fa0/0
20.0.0.1
30.0.0.2
Router-PT
Fa0/0
40.0.0.1
Fa0
PC-PT
PC0
10.0.0.10
PC-PT
PC1
40.0.0.10
Simulation Panel
Event List
Vis. Time(sec) Last Device At Device Type Info
20.002 -- Router1 OSPF
20.003 Router1 Router2 OSPF
22.376 -- Router0 OSPF
22.377 Router0 Router1 OSPF
22.378 -- Router0 OSPF
22.379 Router0 PC0 OSPF
22.380 -- Router2 OSPF
22.381 Router2 PC1 OSPF
22.383 -- Router1 OSPF
Reset Simulation
Constant Delay
Captured to: 22.383 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: 00:05:30.703
Power Cycle Devices PLAY CONTROLS: Back
Auto Capture / Play
Capture / Forward
Event List
Simulation
Routers
(Select a Device to Drag and Drop to the Workspace)
Scenario 0
New
Delete
Toggle PDU List Window
Fire Last Status Source Destination Type Color Time(sec) Periodic Num Edit Delete
Successful PC0 PC1 ICMP 0.000 N 0 (edit) (delete)
22°C
Mostly cloudy
Search
ENG
IN
11:03
27-07-2023

WEEK 8

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

OBSERVATION:



WEEK-7.

Date 3/08/2023

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

Aim :- Understanding ARP

Topology :-

PROCEDURE

* Create a topology of PC's and server as shown in above topology.
* IP address assigned to all including server as we normally do.
* Connect them through a switch

NOTE :- To add a extra port open switch and turn off the switch and drag the port which is below to the empty slot.

* Then select inspect tool and click on all the PC's and server to see the ARP table.

* Then select PC go to command prompt give arp -a.
→ Initially ARP table is empty.

* Also in CLI of 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

* Observe the switch as well as the nodes update the ARP Table as and when a new communication starts.

Ping output:-

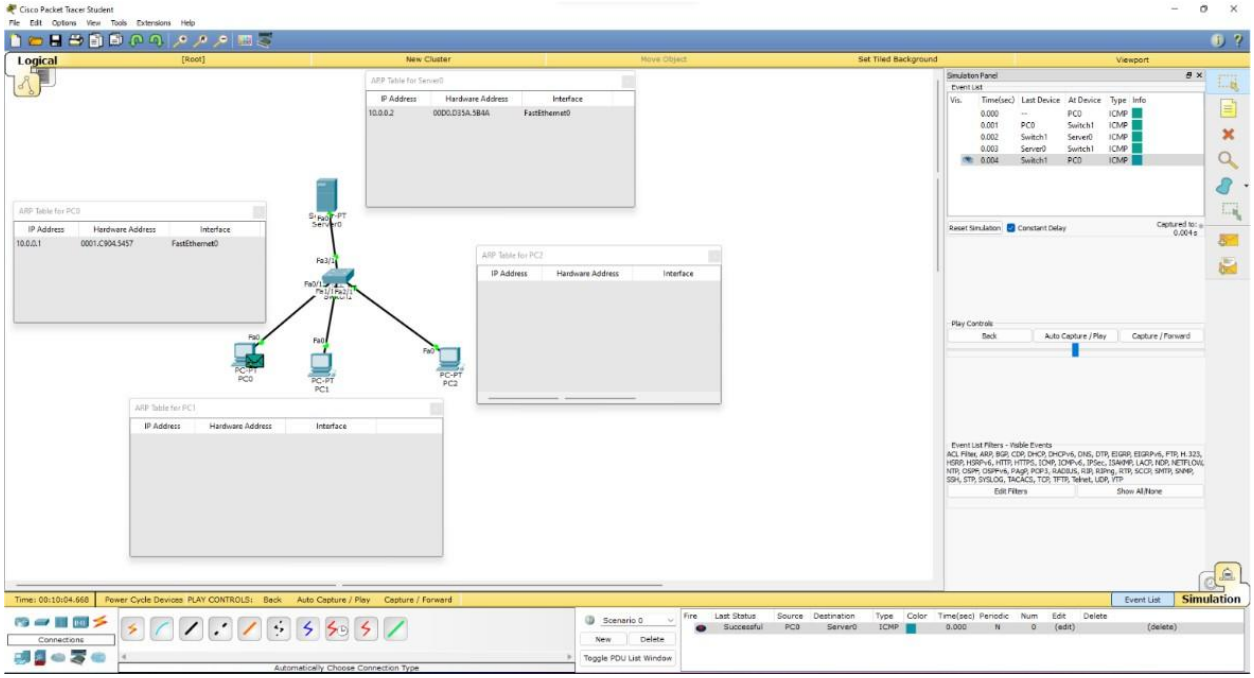
PC> ping 10.0.0.4.

Pinging 10.0.0.4 with 32 bytes of data.
Reply from 10.0.0.4 : bytes=32 time=0ms TTL=128
Reply from 10.0.0.4 : bytes=32 time=0ms TTL=128
Reply from 10.0.0.4 : bytes=32 time=0ms TTL=128
Reply from 10.0.0.4 : bytes=32 time=0ms TTL=128

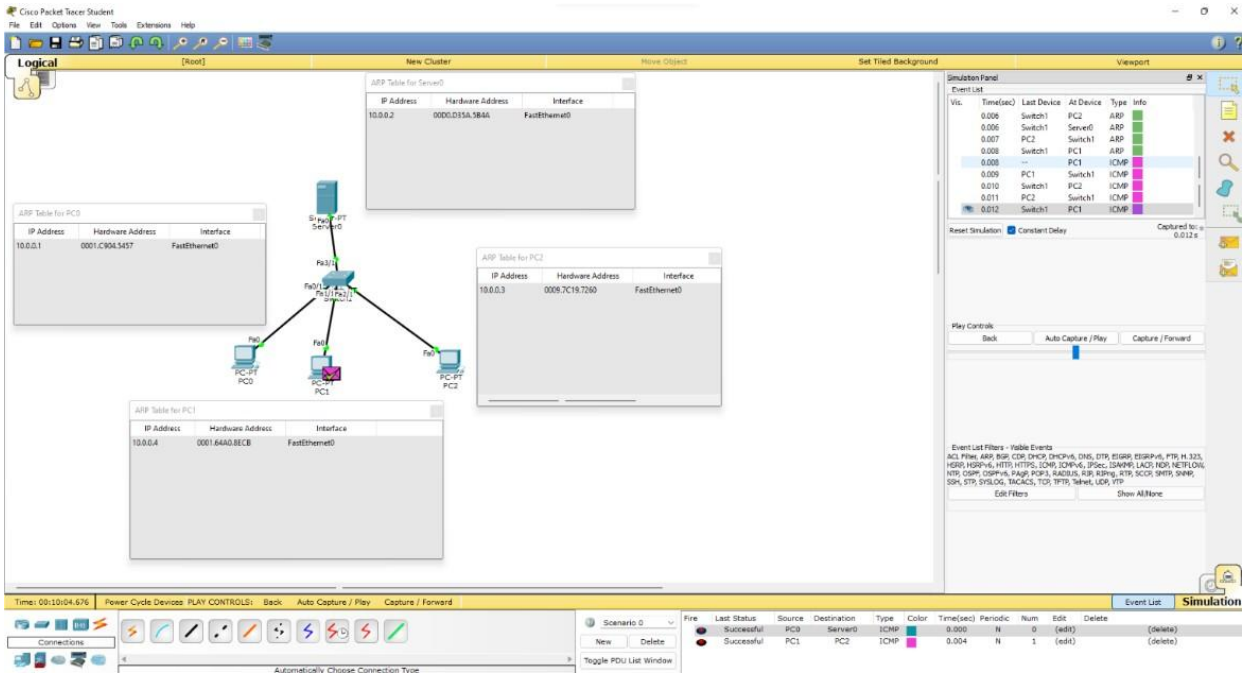
ping statistics for 10.0.0.4.
Packets: Sent=4, Recieved=4, Lost=0 (0% loss)
Approximate round trip times in milliseconds
Minimum=0ms Maximum=0ms Average=0ms

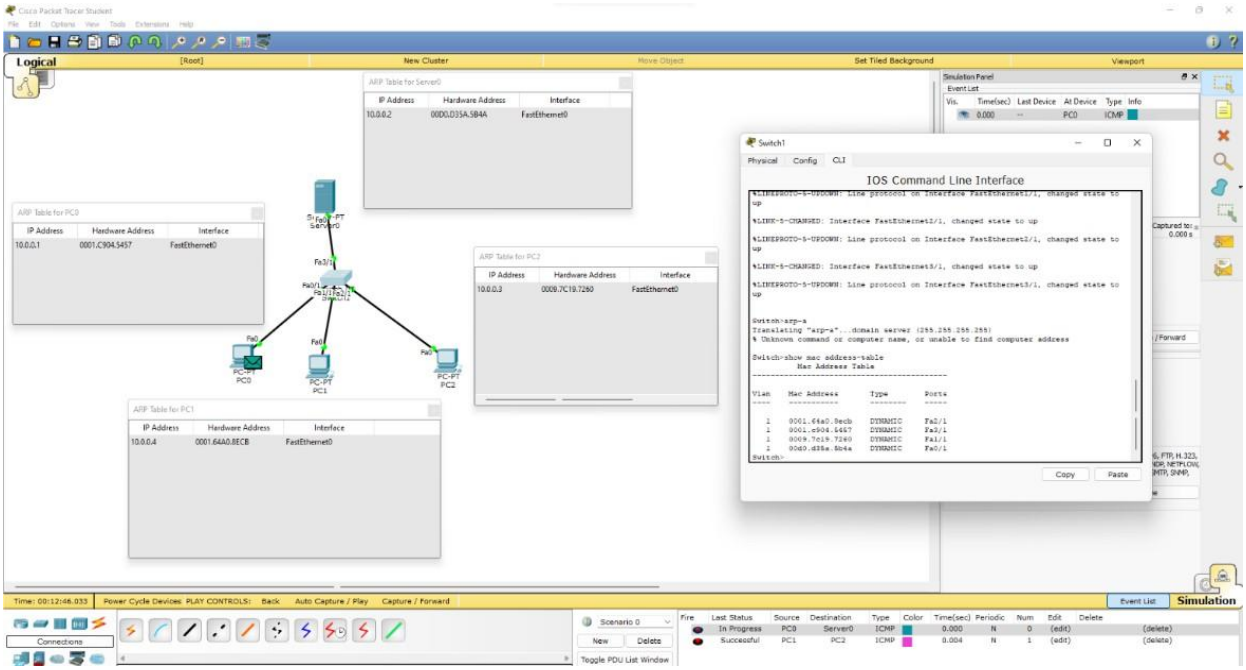
PC> arp -a.

TOPOLOGY:



OUTPUT:

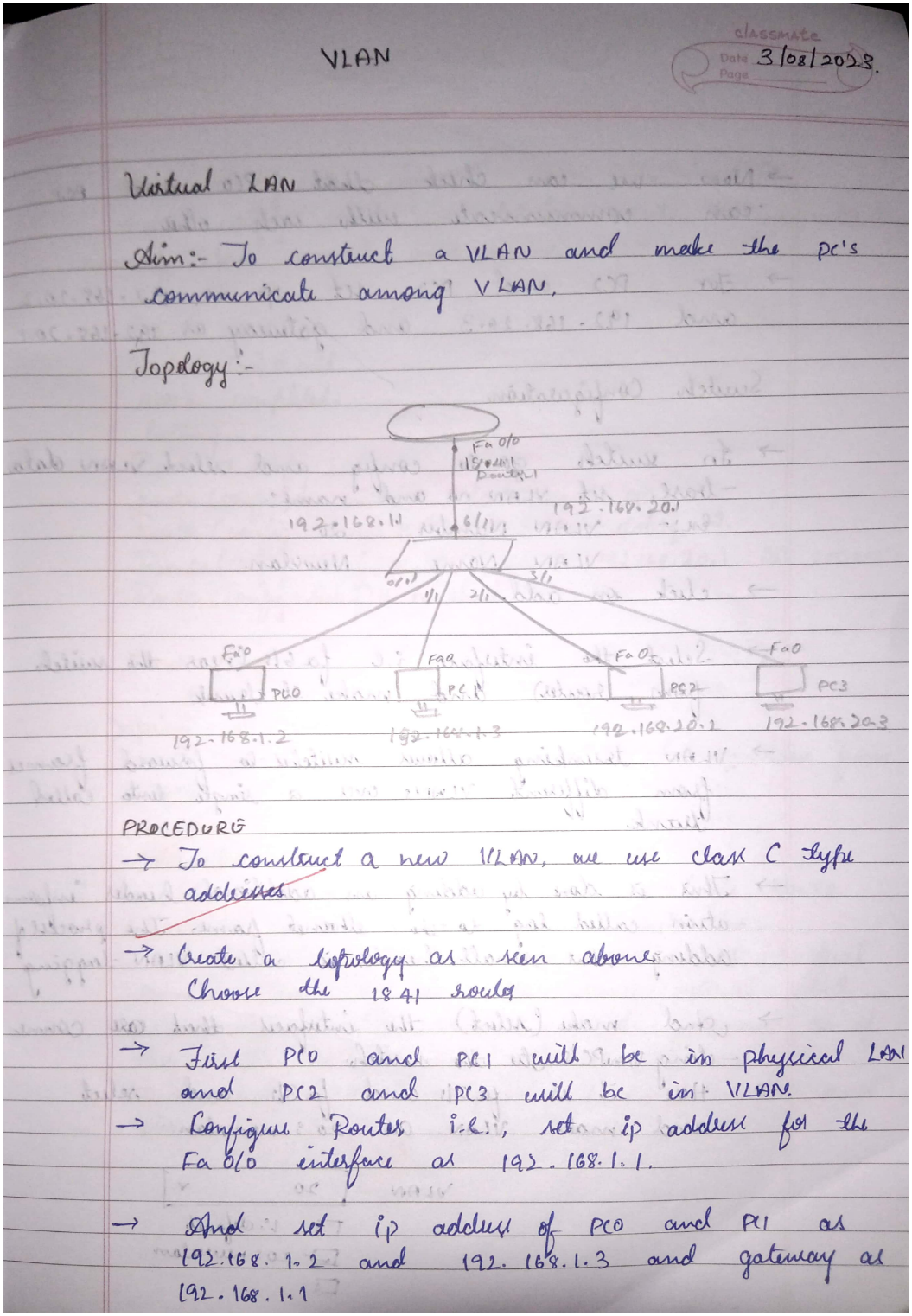



Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
ARP Table for Server0
IP Address Hardware Address Interface
10.0.0.2 00D0.D35A.5B4A FastEthernet0
ARP Table for PC0
IP Address Hardware Address Interface
10.0.0.1 0001.C904.5457 FastEthernet0
ARP Table for PC2
IP Address Hardware Address Interface
10.0.0.3 0009.7C19.7260 FastEthernet0
ARP Table for PC1
IP Address Hardware Address Interface
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 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.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: 00:12:46.033
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 we can check that PC0 and PC1 can communicate with each other

→ For PC2 and PC3 set IP as 192.168.20.2 and 192.168.20.3 and gateway as 192.168.20.

Switch Configuration

→ In switch go to config and select VLAN data -base. set VLAN no and name:-
Ex:- VLAN Number 20
VLAN Name Newvlan

→ click on add.

→ Select the interface i.e fa 6/1 (near the switch from router) and make it trunk.

→ VLAN trunking allows switches 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 frame. The process of adding this small header is called VLAN tagging

→ And make (select) the interfaces that are connecting VLAN PC's to the switch.

→ Here it is fa 2/1 and fa 3/1 and select and make VLAN as 20 : newvlan.

VLAN [20]
☐ 1: default
☑ 20: newvlan
☐

Router configuration:

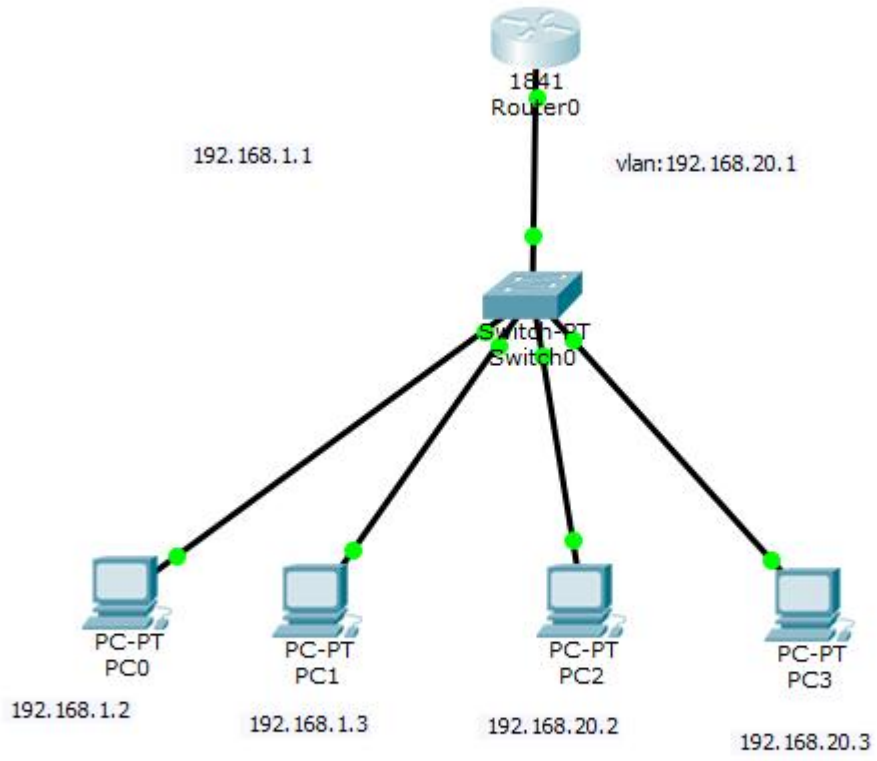
→ open config select VLAN database enter the number & name of vlan created ~~and name~~

Go to CLI
Router(vlan)# exit
APPLY completed.
Exiting....
Router# config t.
Router(config)# interface fastethernet 0/0.1
Router(config-subif)# encapsulation dot1q 2.
Router(config-subif)# ip address 192.168.20.1 255.255.255.0
Router(config-subif)# no shut

Now ping.
from pc0 to pc3.
> ping 192.168.20.3
You will get a succesful transmission from pc0 to pc3.

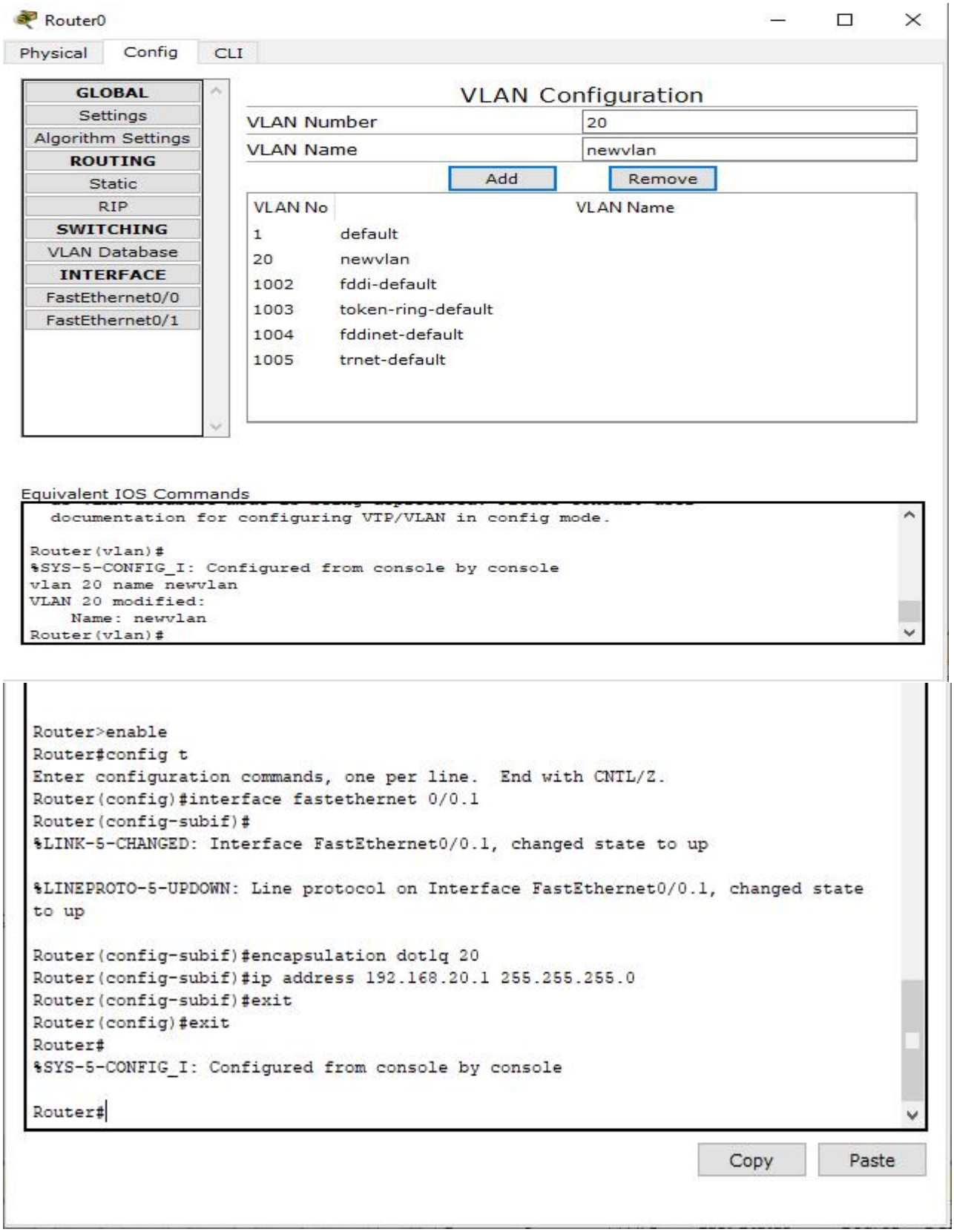
Observation :-
Even though we are using a single router. We can use multiple different networks.
And those networks will works as virtual networks. They are in the same network but has different Network IP addresses and gateway as if they are in different network. And we can communicate from physical LAN to VLAN and vice versa.

TOPOLOGY

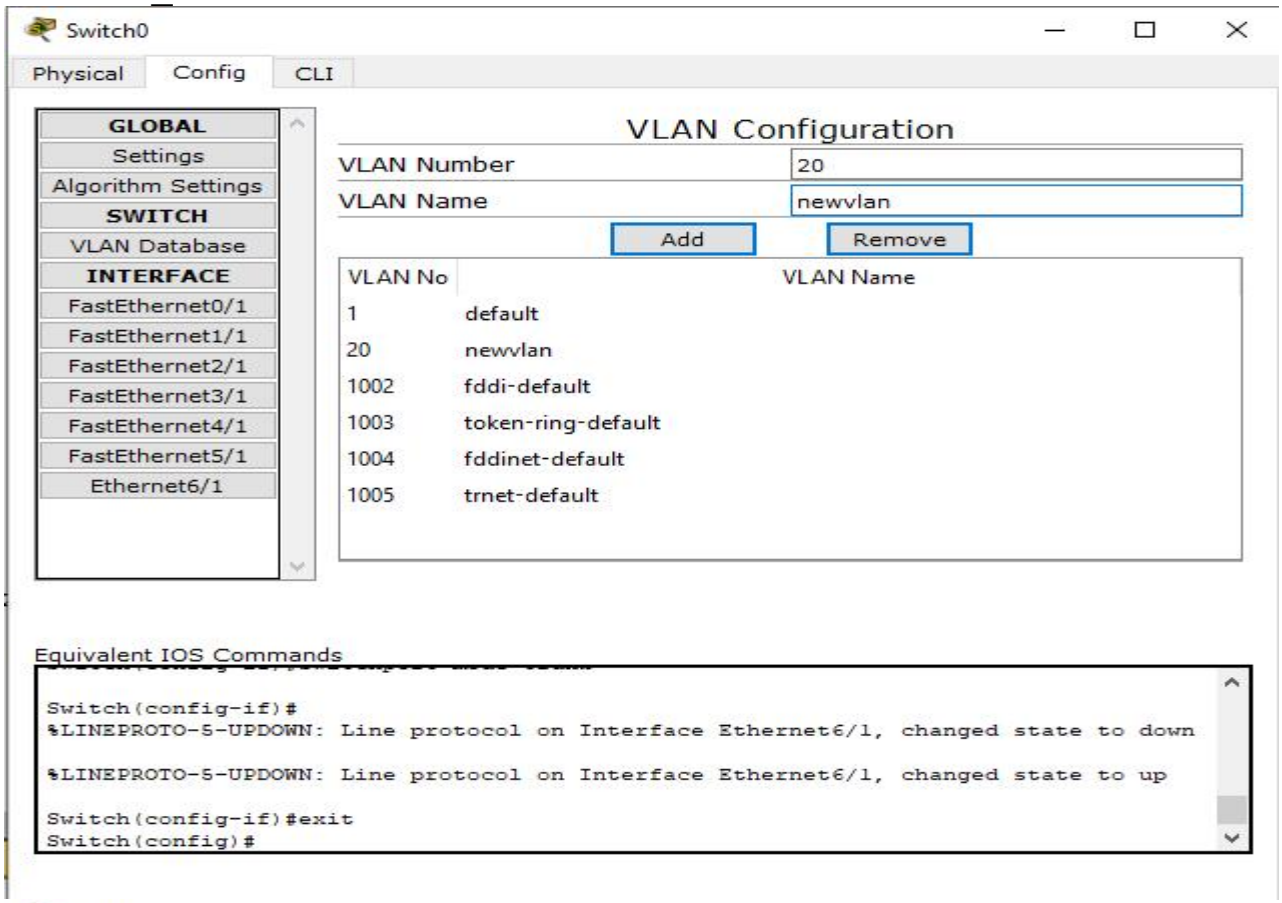


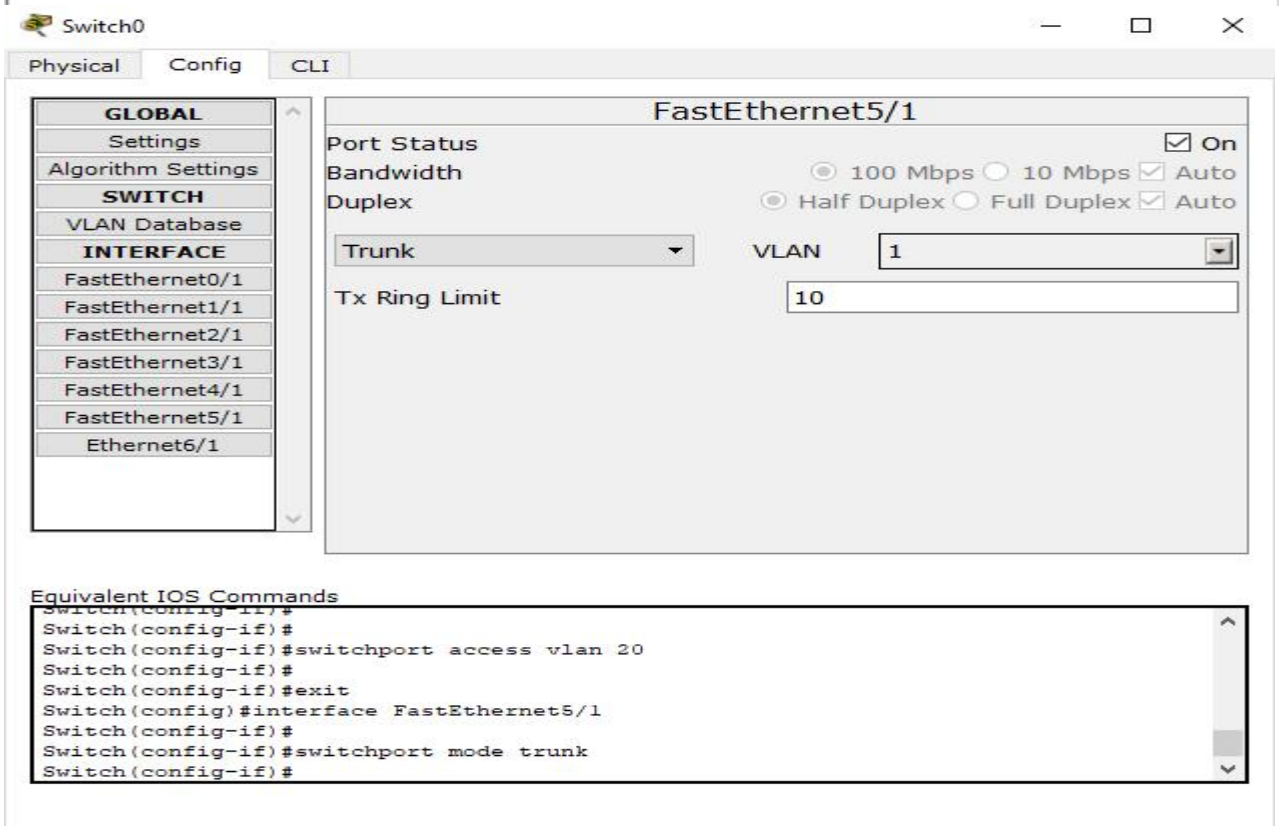
OUTPUT:

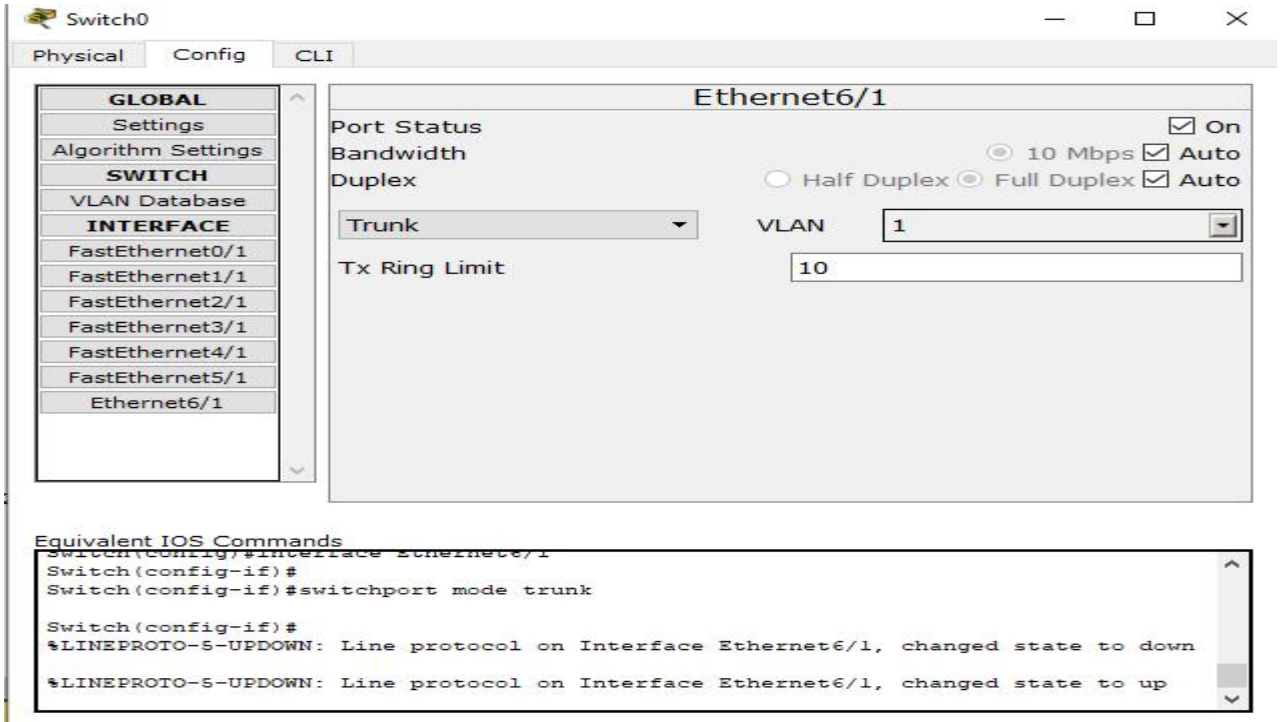
ROUTER CONFIGURATION:

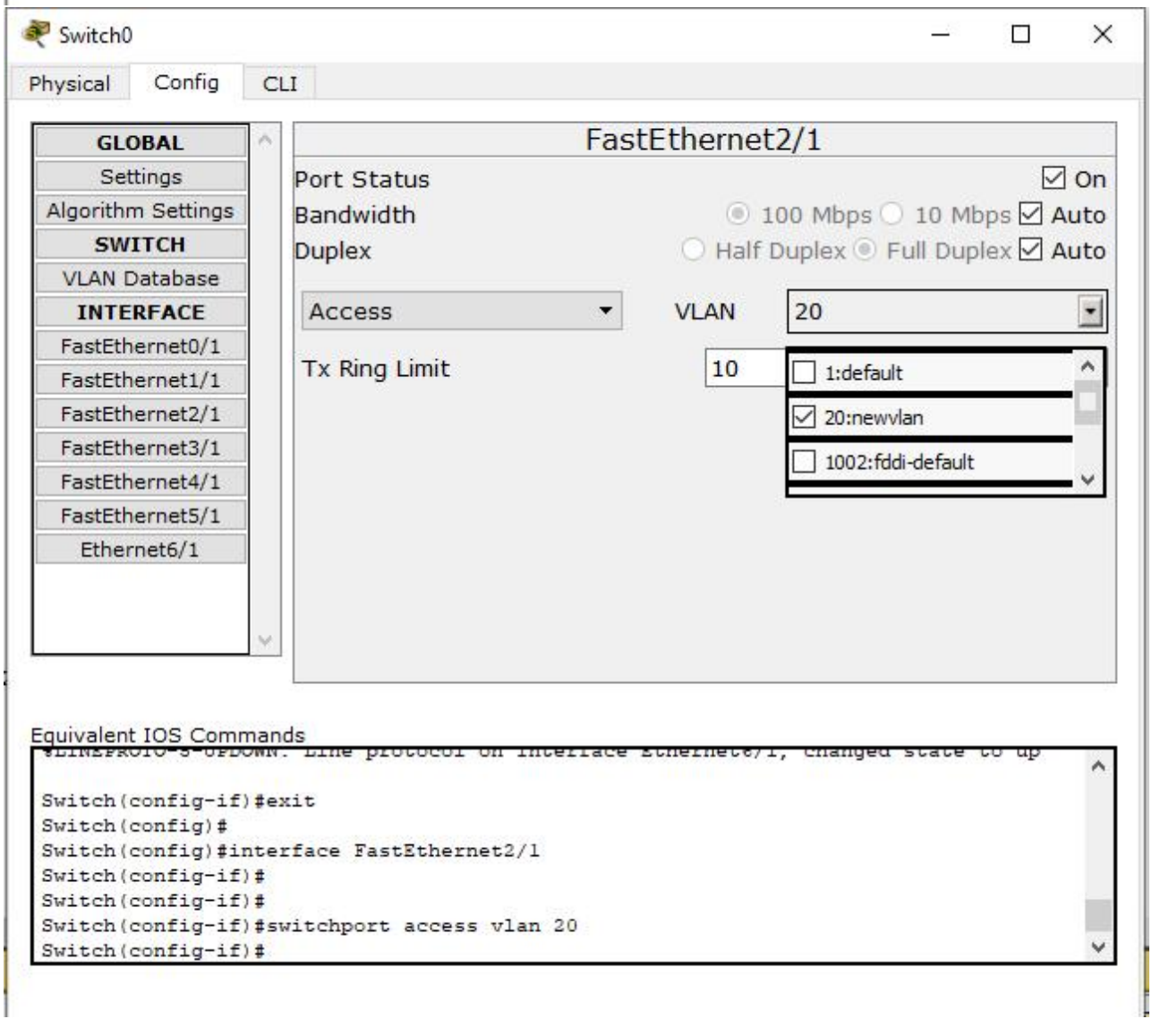

Router0
Physical
Config
CLI
GLOBAL
Settings
Algorithm Settings
ROUTING
Static
RIP
SWITCHING
VLAN Database
INTERFACE
FastEthernet0/0
FastEthernet0/1
VLAN Configuration
VLAN Number
20
VLAN Name
newvlan
Add
Remove
VLAN No
VLAN Name
1 default
20 newvlan
1002 fddi-default
1003 token-ring-default
1004 fddinet-default
1005 trnet-default
Equivalent IOS Commands
documentation for configuring VTP/VLAN in config mode.
Router(vlan)#
%SYS-5-CONFIG_I: Configured from console by console
vlan 20 name newvlan
VLAN 20 modified:
Name: newvlan
Router(vlan)#
Router>enable
Router#config t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface fastethernet 0/0.1
Router(config-subif)#
%LINK-5-CHANGED: Interface FastEthernet0/0.1, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0.1, changed state to up
Router(config-subif)#encapsulation dot1q 20
Router(config-subif)#ip address 192.168.20.1 255.255.255.0
Router(config-subif)#exit
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console
Router#
Copy
Paste

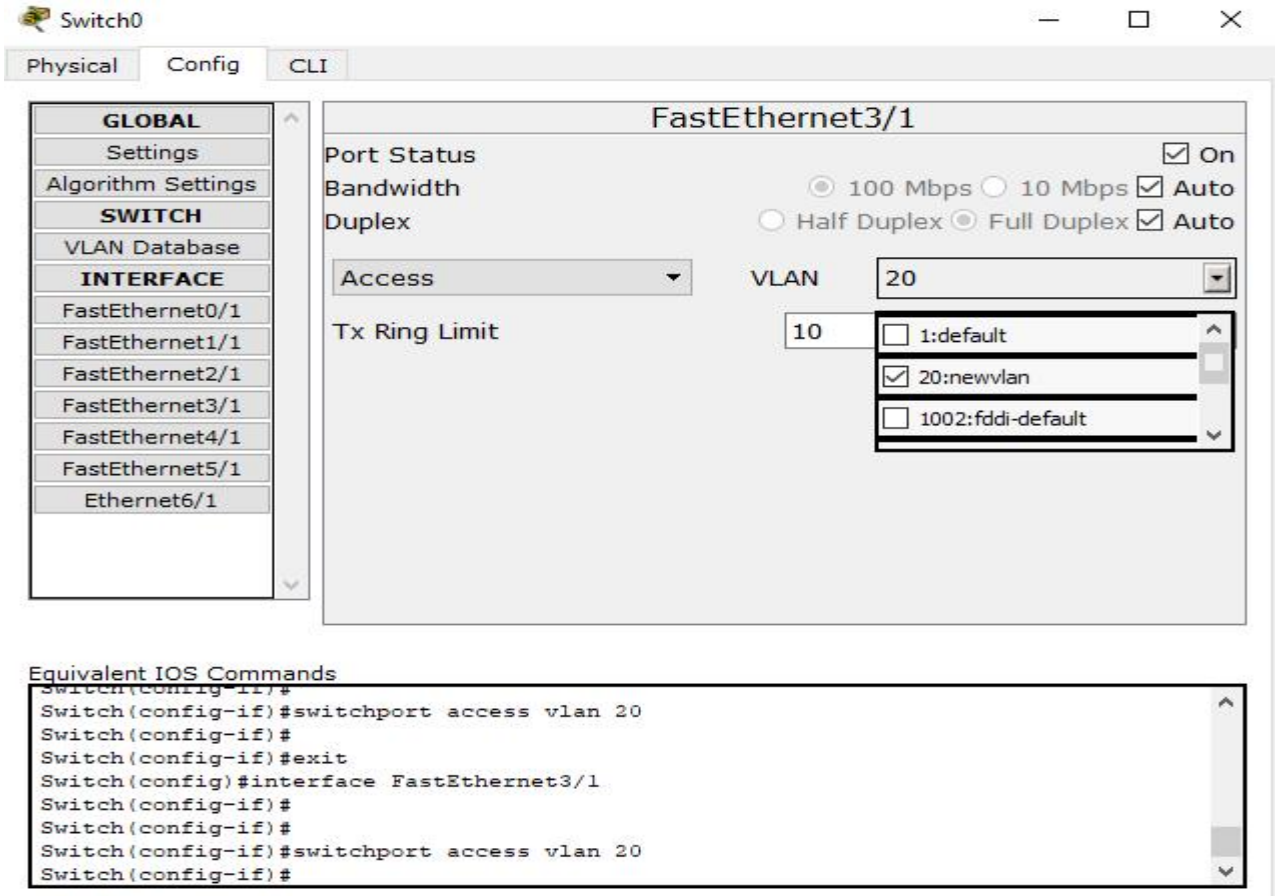
SWITCH_CONFIGURATION:

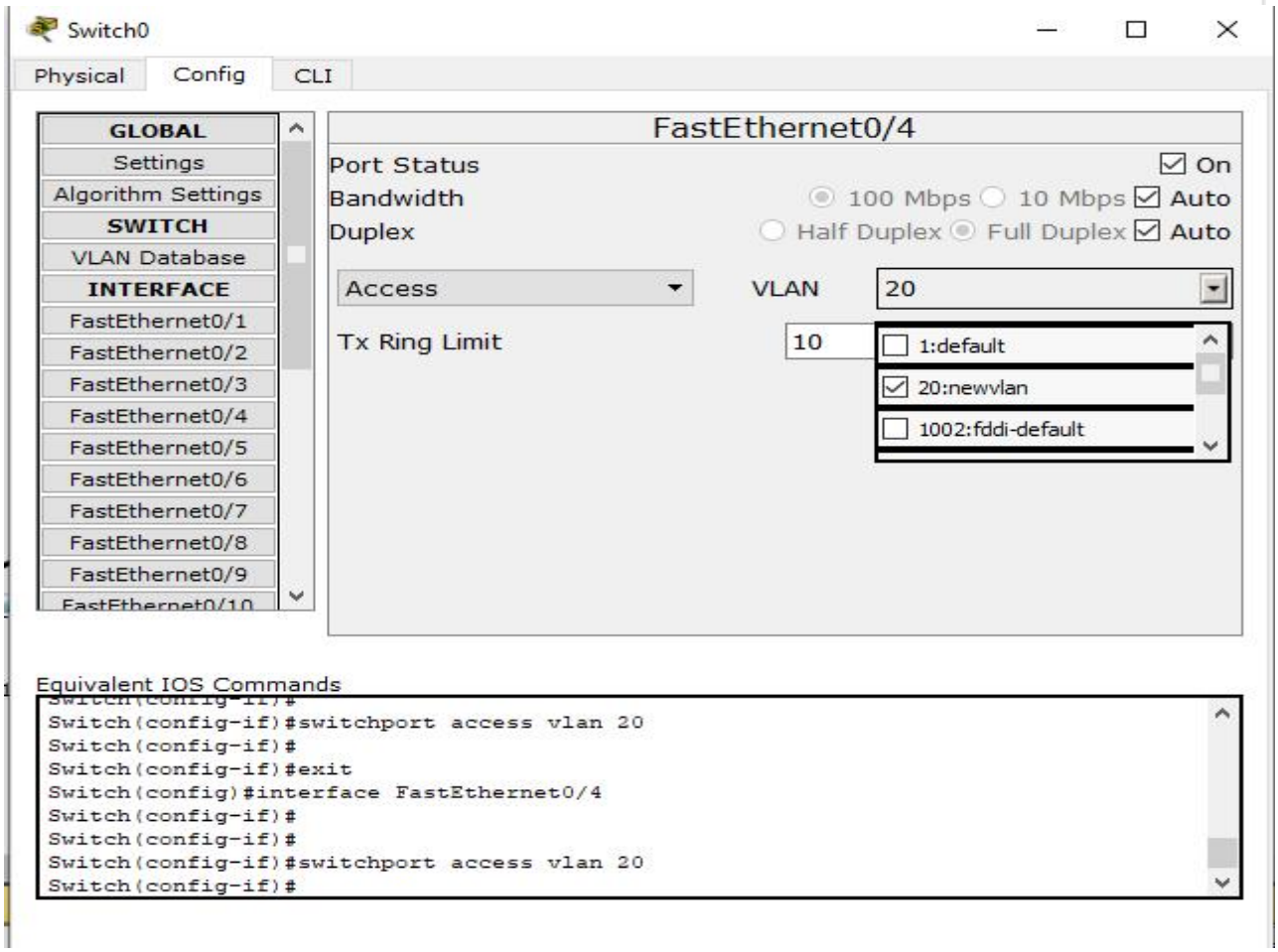


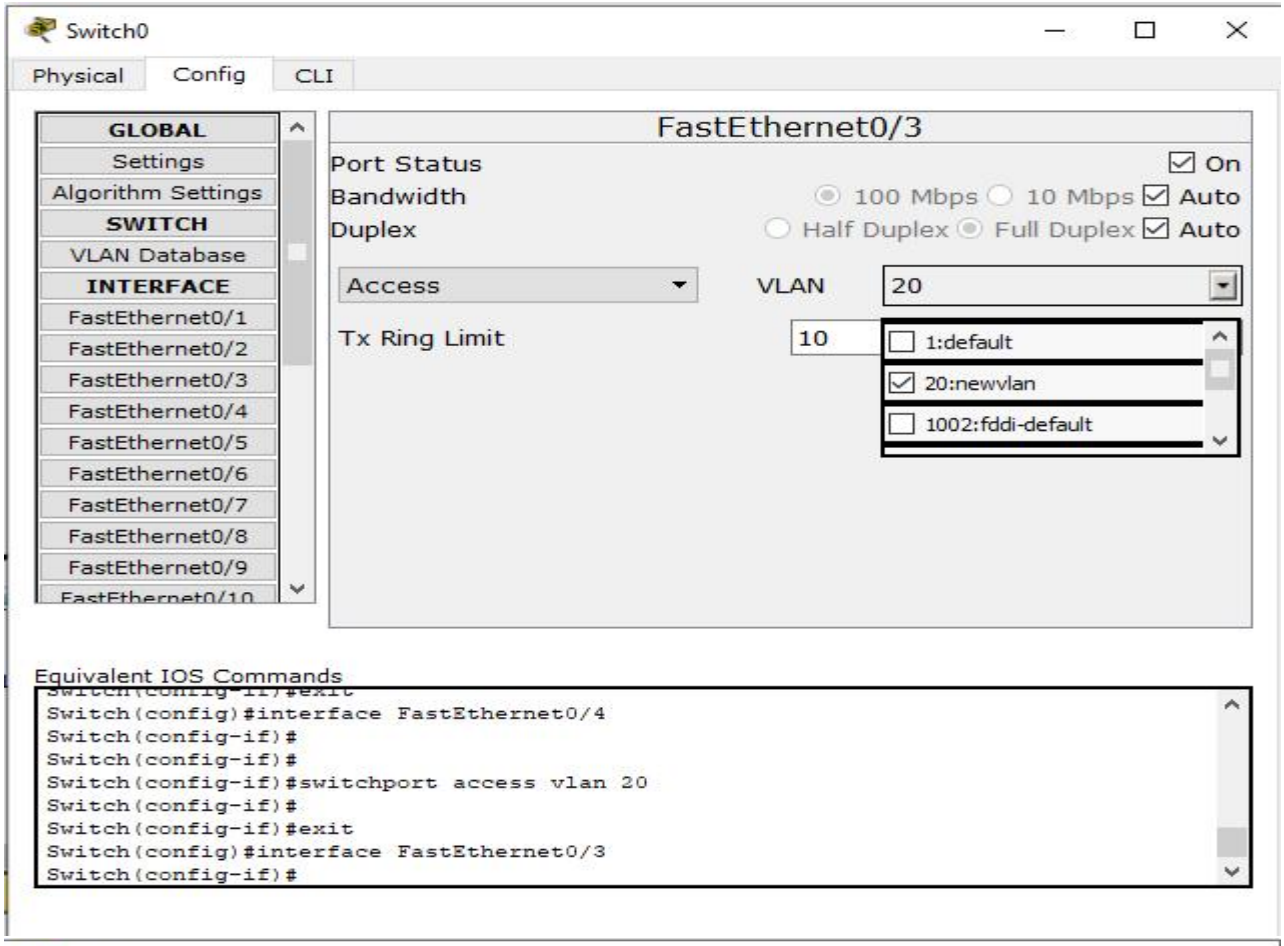


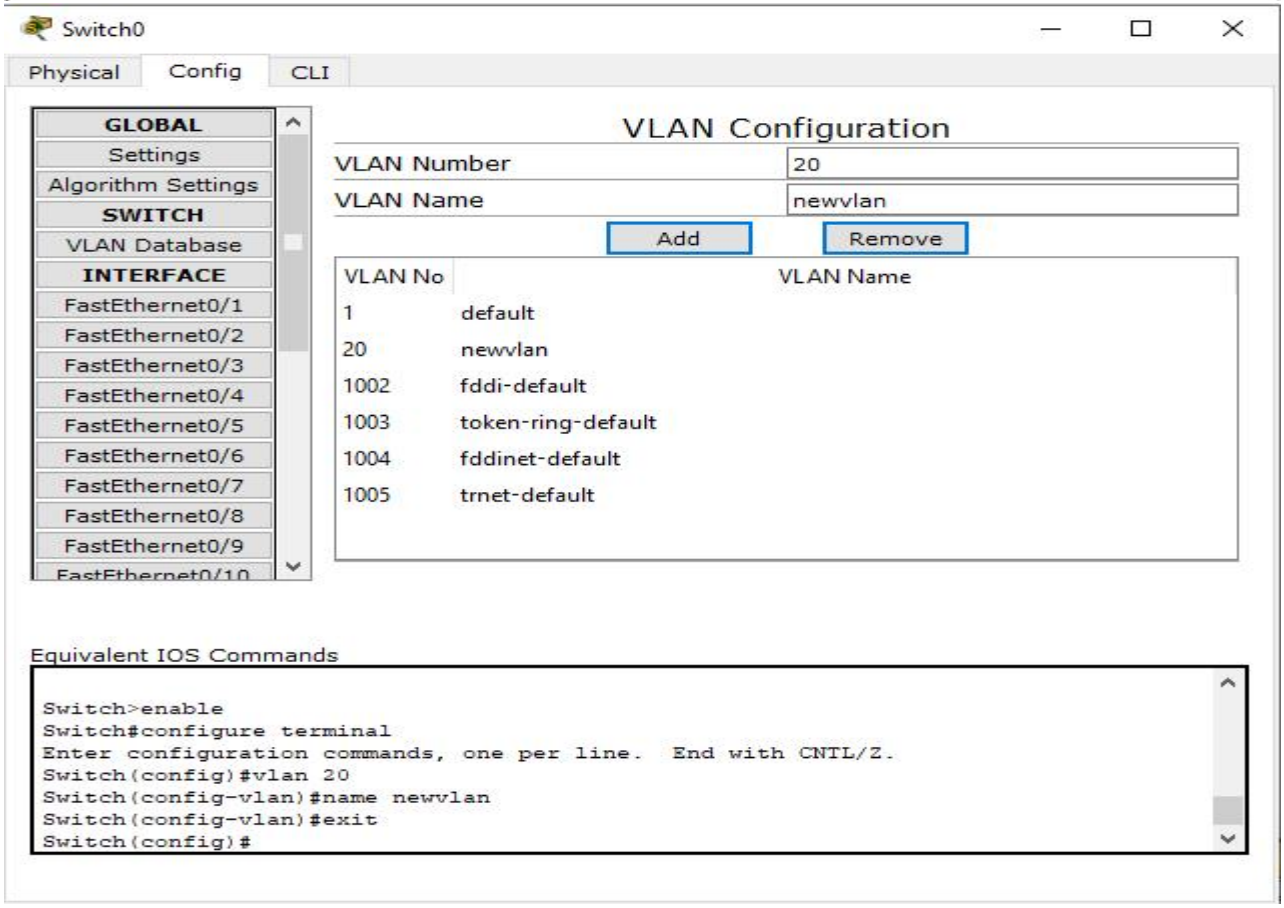

Switch0
Physical
Config
CLI
GLOBAL
Settings
Algorithm Settings
SWITCH
VLAN Database
INTERFACE
FastEthernet0/1
FastEthernet1/1
FastEthernet2/1
FastEthernet3/1
FastEthernet4/1
FastEthernet5/1
Ethernet6/1
Ethernet6/1
Port Status
On
Bandwidth
10 Mbps
Auto
Duplex
Half Duplex
Full Duplex
Auto
Trunk
VLAN
1
Tx Ring Limit
10
Equivalent IOS Commands
Switch(config-if)#
Switch(config-if)#switchport mode trunk
Switch(config-if)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface Ethernet6/1, changed state to down
%LINEPROTO-5-UPDOWN: Line protocol on Interface Ethernet6/1, changed state to up

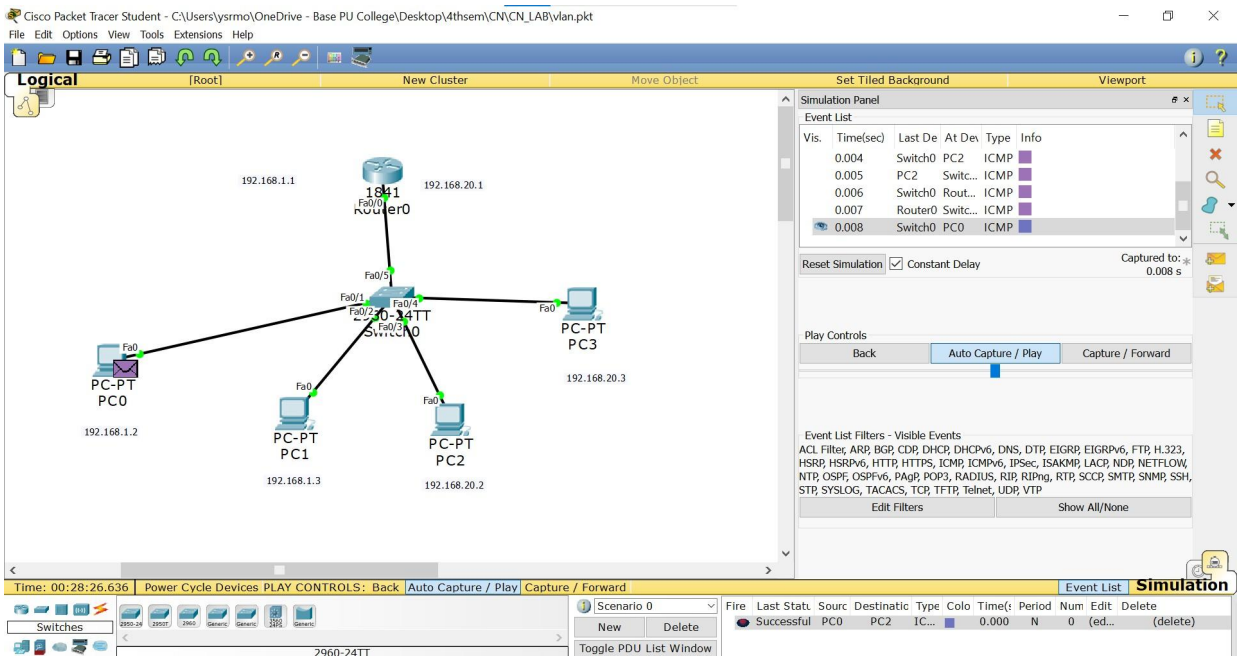

Switch0
Physical
Config
CLI
GLOBAL
Settings
Algorithm Settings
SWITCH
VLAN Database
INTERFACE
FastEthernet0/1
FastEthernet1/1
FastEthernet2/1
FastEthernet3/1
FastEthernet4/1
FastEthernet5/1
Ethernet6/1
FastEthernet2/1
Port Status
On
Bandwidth
100 Mbps
10 Mbps
Auto
Duplex
Half Duplex
Full Duplex
Auto
Access
VLAN
20
Tx Ring Limit
10
1:default
20:newvlan
1002:fddi-default
Equivalent IOS Commands
%LINEPROTO-5-UPDOWN: Line protocol on Interface Ethernet6/1, changed state to up
Switch(config-if)#exit
Switch(config)#
Switch(config)#interface FastEthernet2/1
Switch(config-if)#
Switch(config-if)#
Switch(config-if)#switchport access vlan 20
Switch(config-if)#


Switch0
Physical
Config
CLI
GLOBAL
Settings
Algorithm Settings
SWITCH
VLAN Database
INTERFACE
FastEthernet0/1
FastEthernet1/1
FastEthernet2/1
FastEthernet3/1
FastEthernet4/1
FastEthernet5/1
Ethernet6/1
FastEthernet3/1
Port Status
On
Bandwidth
100 Mbps
10 Mbps
Auto
Duplex
Half Duplex
Full Duplex
Auto
Access
VLAN
20
Tx Ring Limit
10
1:default
20:newvlan
1002:fddi-default
Equivalent IOS Commands
Switch(config-if)#switchport access vlan 20
Switch(config-if)#
Switch(config-if)#exit
Switch(config)#interface FastEthernet3/1
Switch(config-if)#
Switch(config-if)#
Switch(config-if)#switchport access vlan 20
Switch(config-if)#

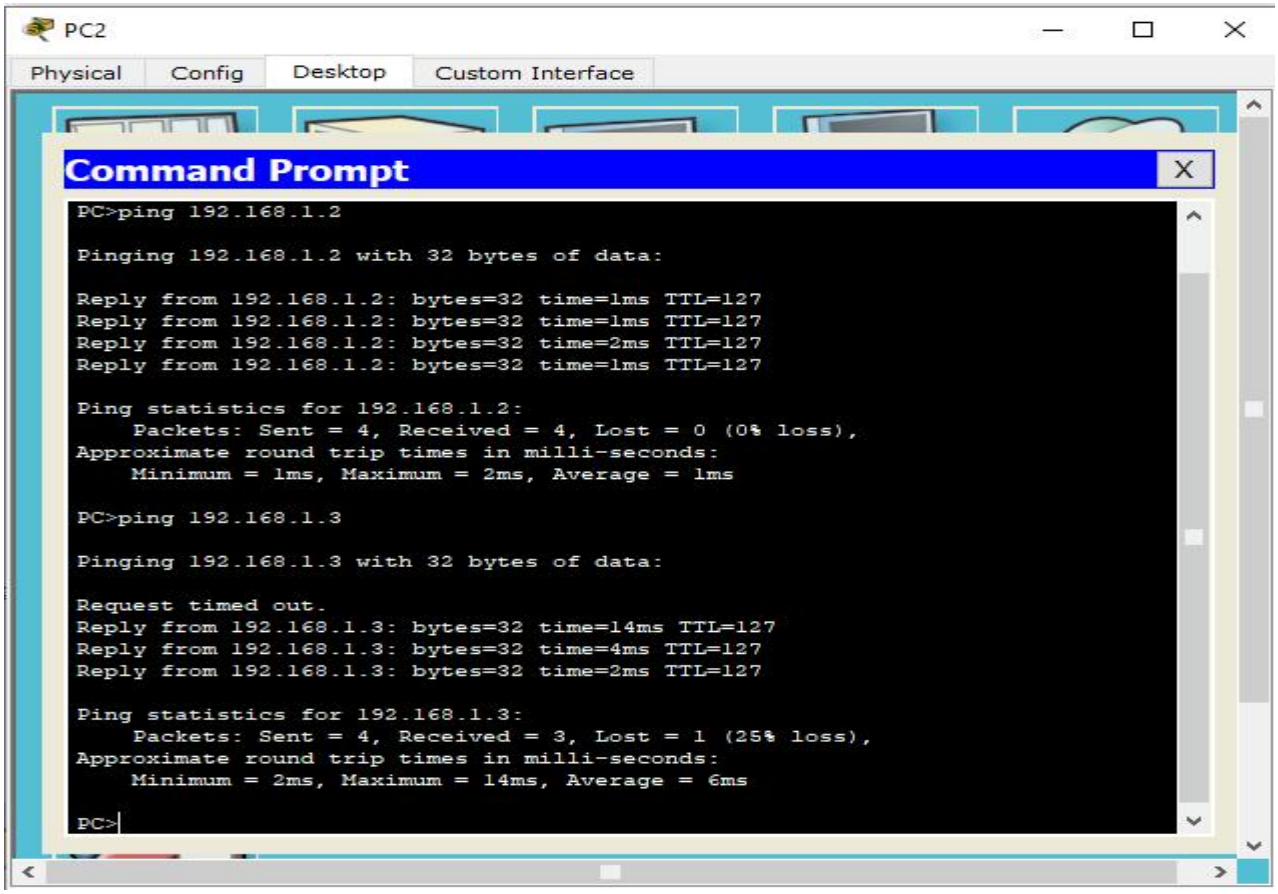

Switch0
Physical
Config
CLI
GLOBAL
Settings
Algorithm Settings
SWITCH
VLAN Database
INTERFACE
FastEthernet0/1
FastEthernet0/2
FastEthernet0/3
FastEthernet0/4
FastEthernet0/5
FastEthernet0/6
FastEthernet0/7
FastEthernet0/8
FastEthernet0/9
FastEthernet0/10
FastEthernet0/4
Port Status
On
Bandwidth
100 Mbps
10 Mbps
Auto
Duplex
Half Duplex
Full Duplex
Auto
Access
VLAN
20
Tx Ring Limit
10
1:default
20:newvlan
1002:fddi-default
Equivalent IOS Commands
Switch(config-if)#switchport access vlan 20
Switch(config-if)#
Switch(config-if)#exit
Switch(config)#interface FastEthernet0/4
Switch(config-if)#
Switch(config-if)#
Switch(config-if)#switchport access vlan 20
Switch(config-if)#

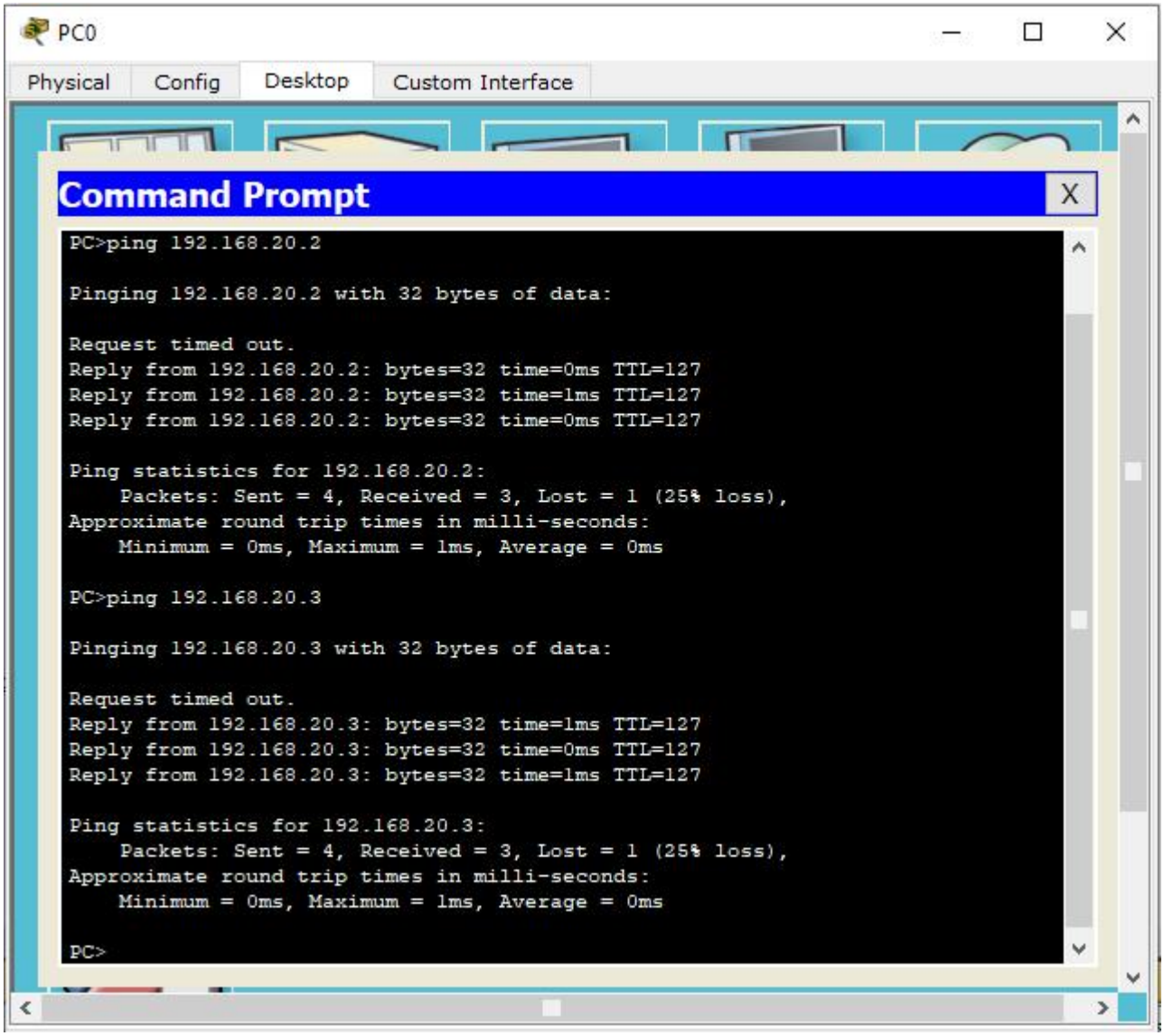

Switch0
Physical
Config
CLI
GLOBAL
Settings
Algorithm Settings
SWITCH
VLAN Database
INTERFACE
FastEthernet0/1
FastEthernet0/2
FastEthernet0/3
FastEthernet0/4
FastEthernet0/5
FastEthernet0/6
FastEthernet0/7
FastEthernet0/8
FastEthernet0/9
FastEthernet0/10
FastEthernet0/3
Port Status
On
Bandwidth
100 Mbps
10 Mbps
Auto
Duplex
Half Duplex
Full Duplex
Auto
Access
VLAN
20
Tx Ring Limit
10
1:default
20:newvlan
1002:fddi-default
Equivalent IOS Commands
Switch(config-if)#exit
Switch(config)#interface FastEthernet0/4
Switch(config-if)#
Switch(config-if)#
Switch(config-if)#switchport access vlan 20
Switch(config-if)#
Switch(config-if)#exit
Switch(config)#interface FastEthernet0/3
Switch(config-if)#


Switch0
Physical
Config
CLI
GLOBAL
Settings
Algorithm Settings
SWITCH
VLAN Database
INTERFACE
FastEthernet0/1
FastEthernet0/2
FastEthernet0/3
FastEthernet0/4
FastEthernet0/5
FastEthernet0/6
FastEthernet0/7
FastEthernet0/8
FastEthernet0/9
FastEthernet0/10
VLAN Configuration
VLAN Number
20
VLAN Name
newvlan
Add
Remove
VLAN No
VLAN Name
1
default
20
newvlan
1002
fddi-default
1003
token-ring-default
1004
fddinet-default
1005
trnet-default
Equivalent IOS Commands
Switch>enable
Switch#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)#vlan 20
Switch(config-vlan)#name newvlan
Switch(config-vlan)#exit
Switch(config)#


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
Fa0/4
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
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 Statu Sourc Destinatic Type Colo Time(Period Num Edit Delete
Successful PC0 PC2 IC... 0.000 N 0 (ed... (delete)

PING OUTPUT:

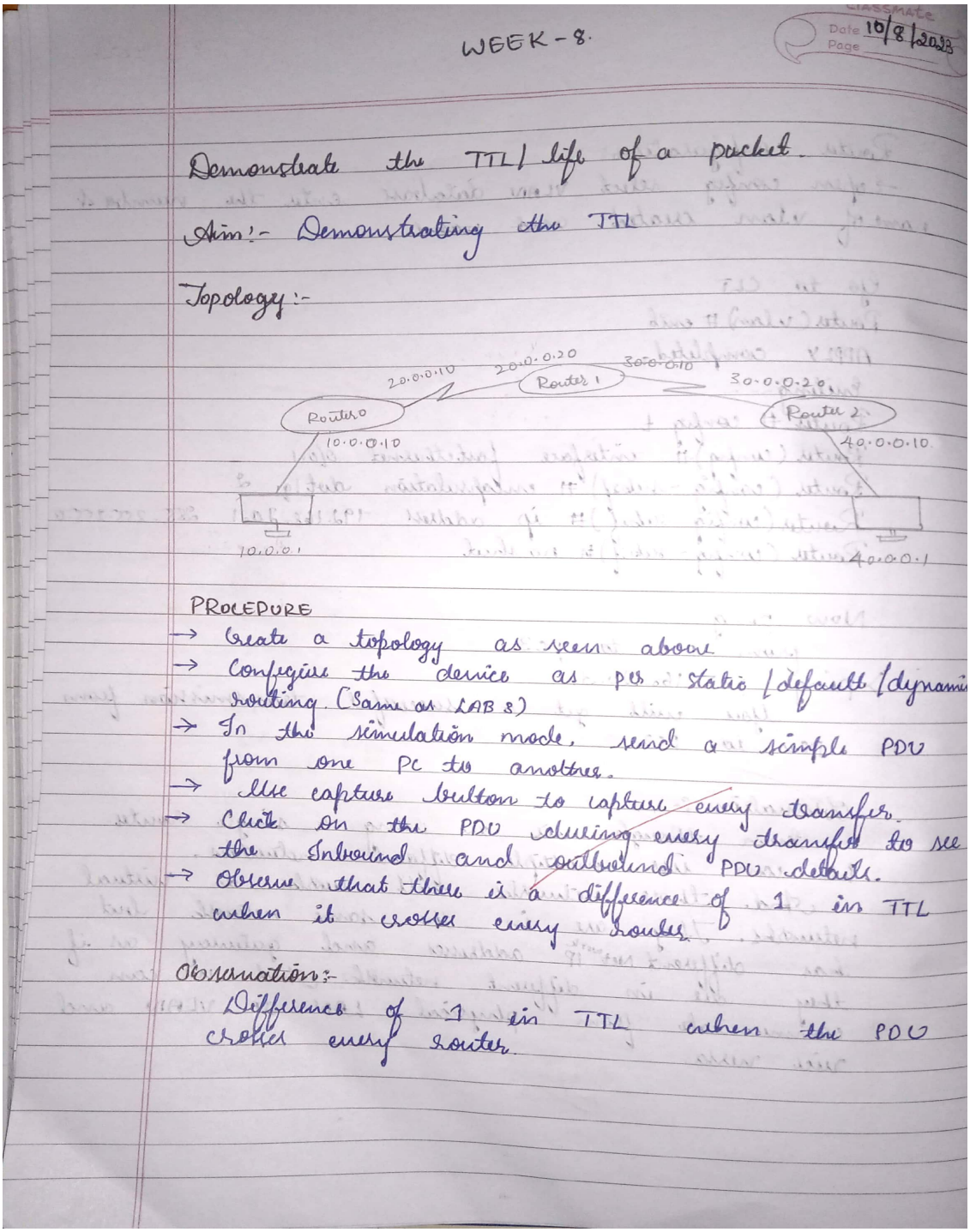

PC2
Physical
Config
Desktop
Custom Interface
Command Prompt
PC>ping 192.168.1.2
Pinging 192.168.1.2 with 32 bytes of data:
Reply from 192.168.1.2: bytes=32 time=1ms TTL=127
Reply from 192.168.1.2: bytes=32 time=1ms TTL=127
Reply from 192.168.1.2: bytes=32 time=2ms TTL=127
Reply from 192.168.1.2: bytes=32 time=1ms TTL=127
Ping statistics for 192.168.1.2:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 1ms, Maximum = 2ms, Average = 1ms
PC>ping 192.168.1.3
Pinging 192.168.1.3 with 32 bytes of data:
Request timed out.
Reply from 192.168.1.3: bytes=32 time=14ms TTL=127
Reply from 192.168.1.3: bytes=32 time=4ms TTL=127
Reply from 192.168.1.3: bytes=32 time=2ms TTL=127
Ping statistics for 192.168.1.3:
Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
Minimum = 2ms, Maximum = 14ms, Average = 6ms
PC>


PC0
Physical
Config
Desktop
Custom Interface
Command Prompt
X
PC>ping 192.168.20.2
Pinging 192.168.20.2 with 32 bytes of data:
Request timed out.
Reply from 192.168.20.2: bytes=32 time=0ms TTL=127
Reply from 192.168.20.2: bytes=32 time=1ms TTL=127
Reply from 192.168.20.2: bytes=32 time=0ms TTL=127
Ping statistics for 192.168.20.2:
Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 1ms, Average = 0ms
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=1ms TTL=127
Reply from 192.168.20.3: bytes=32 time=0ms TTL=127
Reply from 192.168.20.3: bytes=32 time=1ms 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 = 1ms, Average = 0ms
PC>

WEEK 10

Demonstrate the TTL/ Life of a Packet.

OBSERVATION:



WEEK-8.

Date 10/8/2023

Demonstrate the TTL/ life of a packet.

Aim:- Demonstrating the TTL

Topology:-

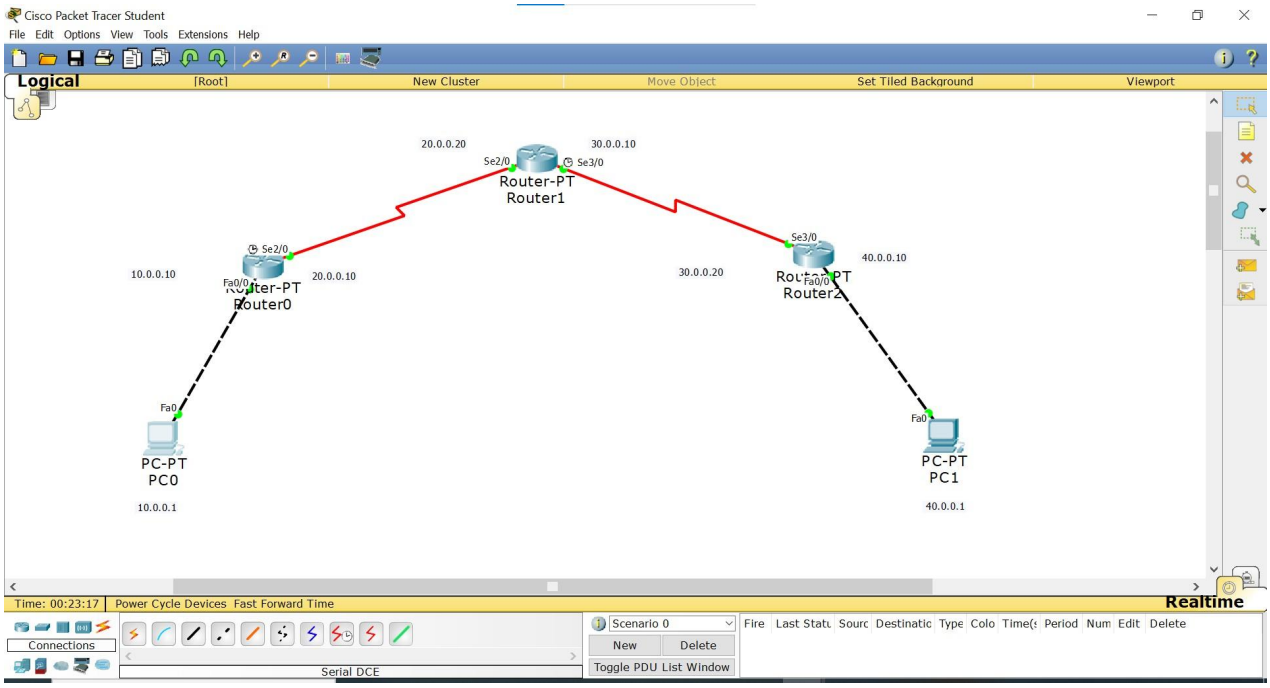
PROCEDURE

→ Create a topology as seen above
→ Configure the device as per static / default / dynamic routing. (Same as LAB 8)
→ In the simulation mode, send a simple PDU from one PC to another.
→ Use capture button to capture every transfer.
→ Click on the PDU during every transfer to see the Inbound and outbound PDU details.
→ Observe that there is a difference of 1 in TTL when it crosses every router.

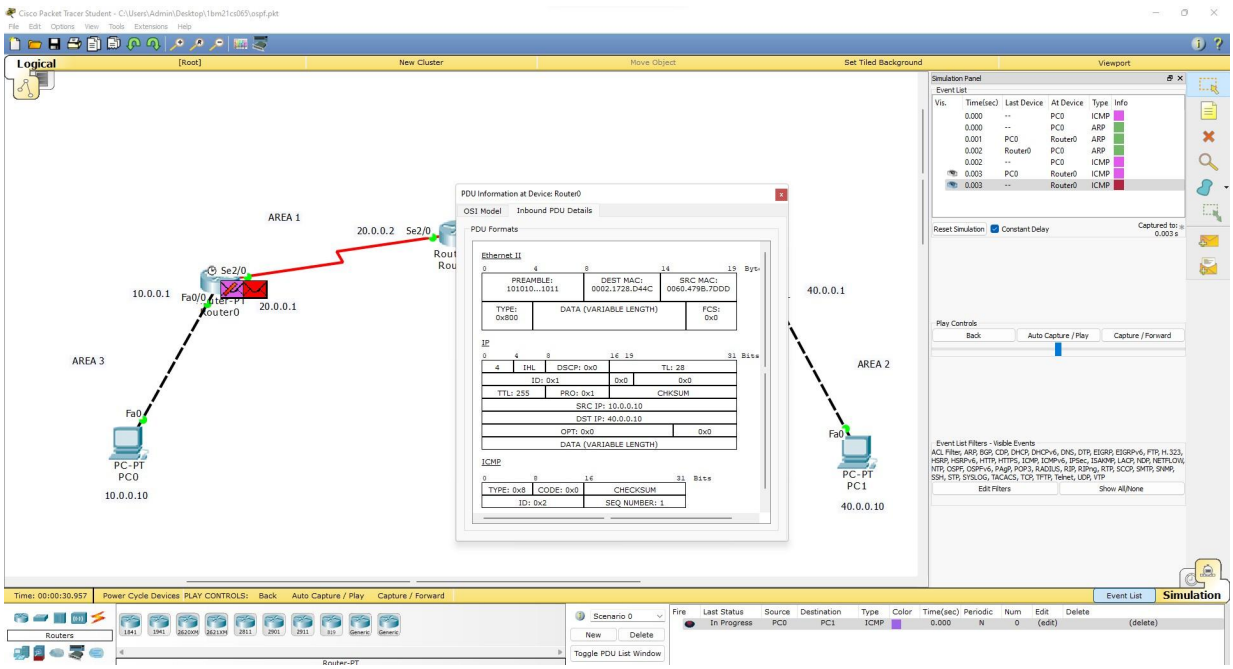
Observation:-

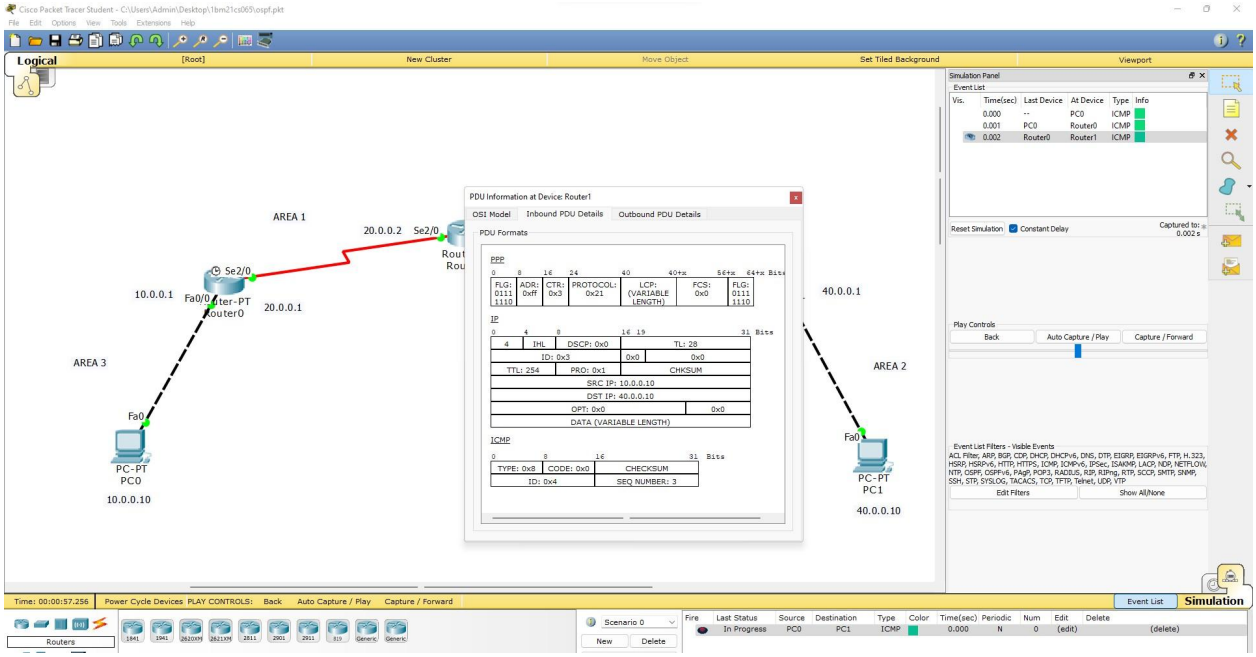
Difference of 1 in TTL when the PDU crosses every router.

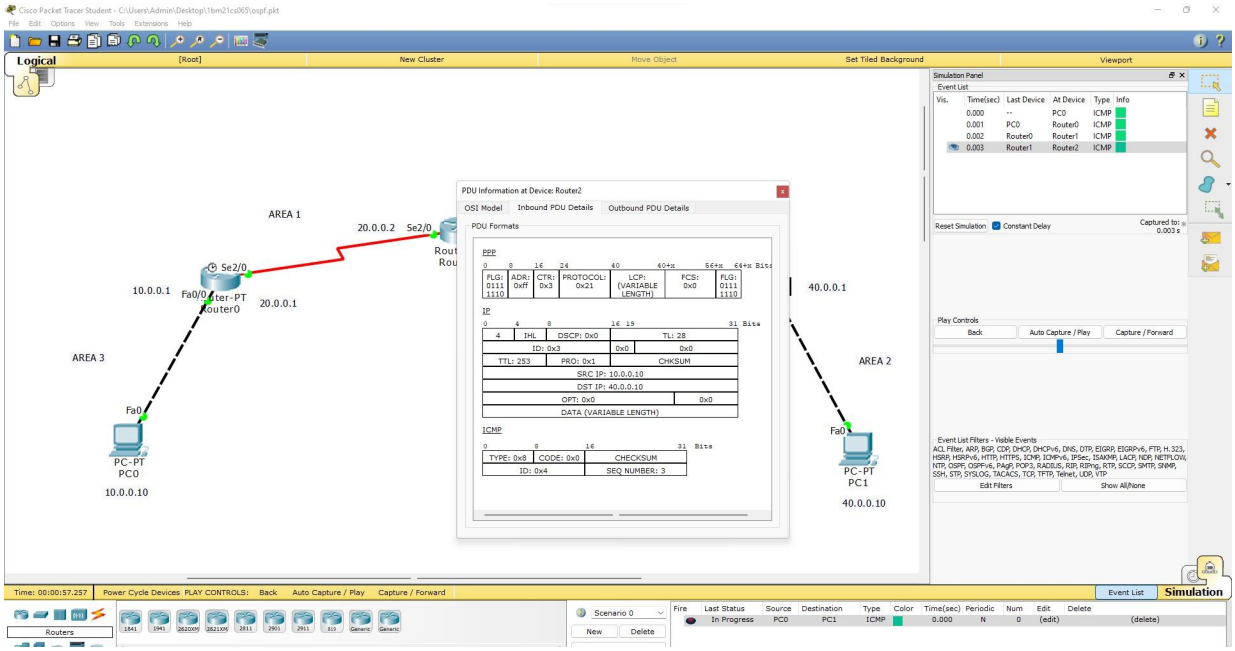
TOPOLOGY:

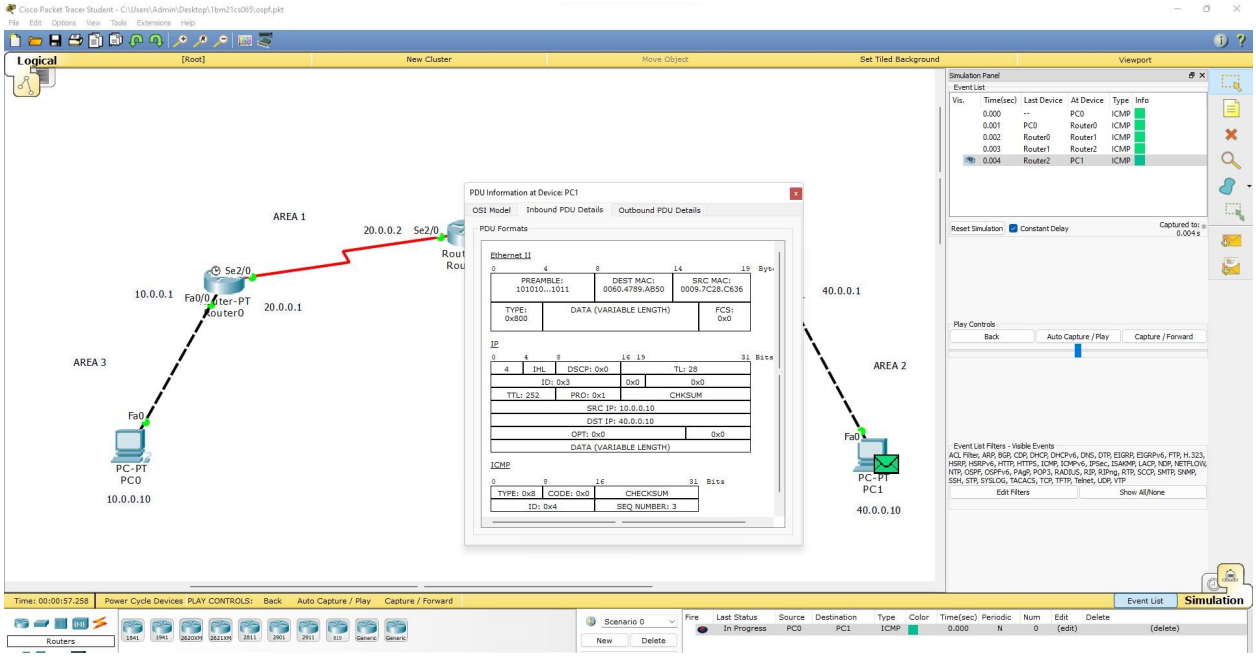


OUTPUT:




Logical
AREA 1
AREA 2
AREA 3
20.0.0.2
Se2/0
10.0.0.1
Fa0/0
Router-PT
Router0
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: Router1
OSI Model
Inbound PDU Details
Outbound PDU Details
PDU Formats
PPP
IP
ICMP
Simulation Panel
Event List
Play Controls
Back
Auto Capture / Play
Capture / Forward
Simulation

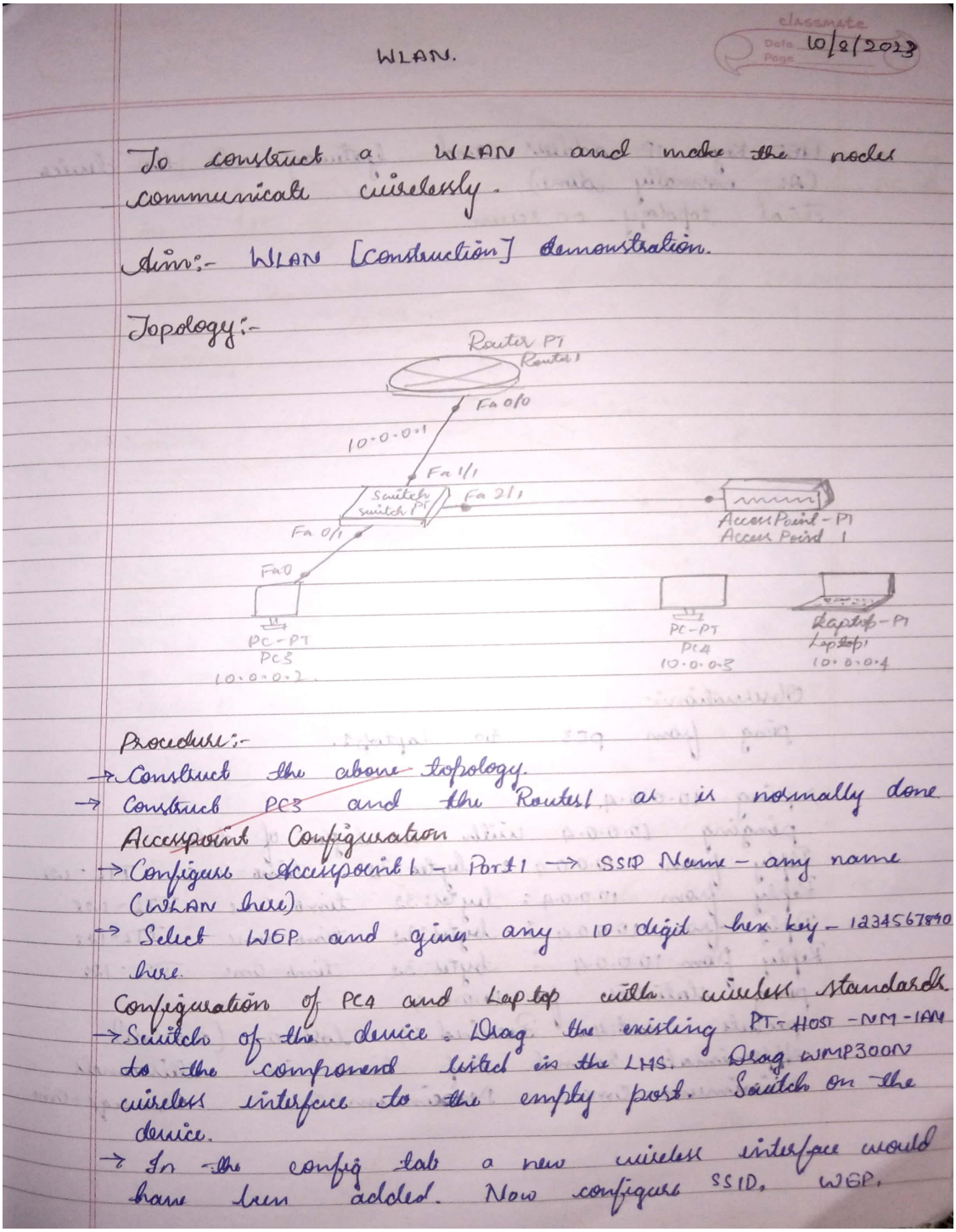

Logical
AREA 1
AREA 2
AREA 3
20.0.0.2
Se2/0
10.0.0.1
Fa0/0
Router-PT
Router0
20.0.0.1
40.0.0.1
Fa0
PC-PT
PC0
10.0.0.10
PC-PT
PC1
40.0.0.10
PDU Information at Device: Router2
OSI Model
Inbound PDU Details
Outbound PDU Details
PDU Formats
PPP
IP
ICMP
Simulation Panel
Event List
Play Controls
Back
Auto Capture / Play
Capture / Forward
Simulation


Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
AREA 1
20.0.0.2
Se2/0
10.0.0.1
Fa0/0
Router-PT
Router0
20.0.0.1
AREA 3
Fa0
PC-PT
PC0
10.0.0.10
40.0.0.1
AREA 2
Fa0
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
IHL
DSCP: 0x0
TL: 28
ID: 0x3
TTL: 252
PRO: 0x1
CHKSUM
SRC IP: 10.0.0.10
DST IP: 40.0.0.10
OPT: 0x0
DATA (VARIABLE LENGTH)
ICMP
TYPE: 0x8
CODE: 0x0
CHECKSUM
ID: 0x4
SEQ NUMBER: 3
Simulation Panel
Event List
Vis.
Time(sec)
Last Device
At Device
Type
Info
0.000 -- PC0 ICMP
0.001 PC0 Router0 ICMP
0.002 Router0 Router1 ICMP
0.003 Router1 Router2 ICMP
0.004 Router2 PC1 ICMP
Reset Simulation
Constant Delay
Play Controls
Back
Auto Capture / Play
Capture / Forward
Event List Filters - Visible Events
Edit Filters
Show All/None
Time: 00:00:57.258
Power Cycle Devices
PLAY CONTROLS:
Back
Auto Capture / Play
Capture / Forward
Event List
Simulation
Routers
Scenario 0
New
Delete
Fire
Last Status
Source
Destination
Type
Color
Time(sec)
Periodic
Num
Edit
Delete
In Progress
PC0
PC1
ICMP
0.000
N
0
(edit)
(delete)

WEEK 11

To construct a WLAN and make the nodes communicate wirelessly

OBSERVATION:



WLAN.

Date 10/8/2023

To construct a WLAN and make the nodes communicate wirelessly.

Aim:- WLAN [construction] demonstration.

Topology:-

Procedure:-

→ Construct the above topology.

→ Construct PC3 and the Router1 as is normally done

Accesspoint Configuration

→ Configure Accesspoint1 – Port1 → SSID Name – any name (WLAN here)

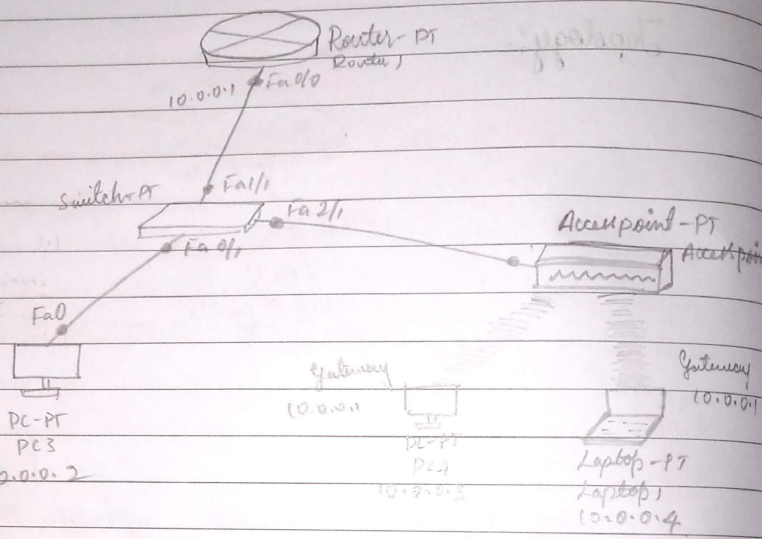
→ Select WEP and giving any 10 digit hex key – 1234567890 here.

Configuration of PC4 and Laptop with wireless standards

→ Switch off the device. Drag the existing PT-HOST-NM-1AM to the component listed in the LHS. Drag WMP300N wireless interface to the empty port. Switch on the device.

→ In the config tab a new wireless interface would have been added. Now configure SSID, WEP.

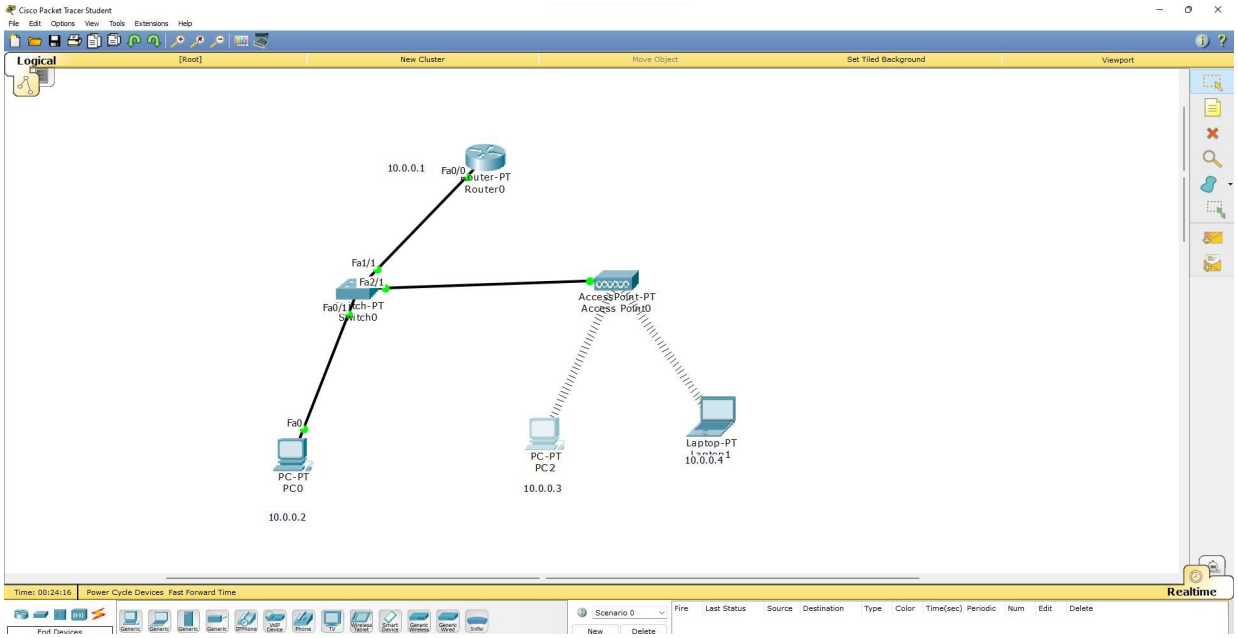
WEP Key, IP address and gateway to the devices (As normally done).
Final topology on screen.

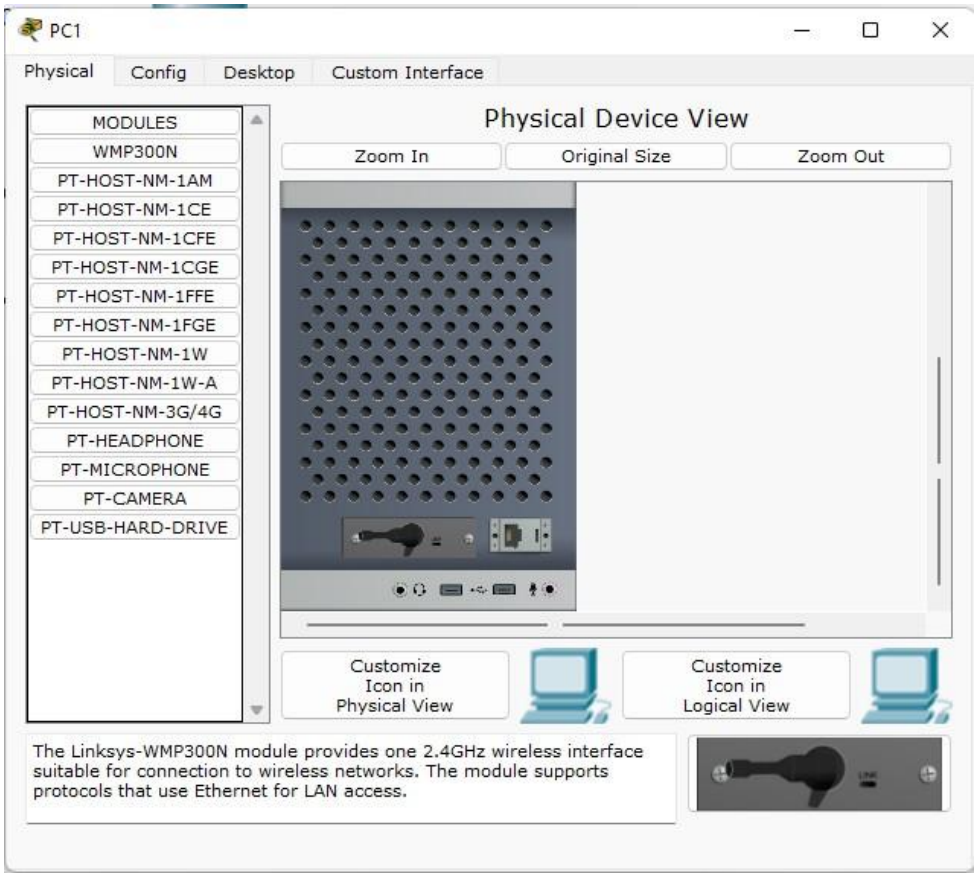


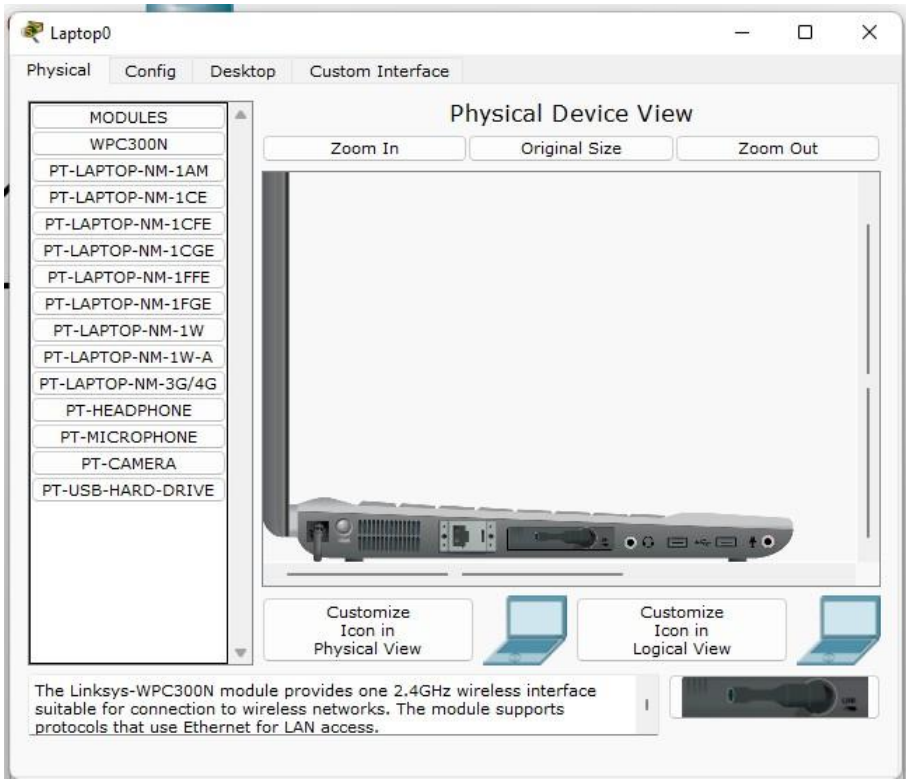
Observation:-
ping from pc3 to laptop1.

>ping 10.0.0.4.
pinging 10.0.0.4 with 32 bytes of data.
Reply from 10.0.0.4 : bytes:32 time=0ms TTL=128.
Reply from 10.0.0.4 : bytes:32 time=0ms TTL=128
Reply from 10.0.0.4 : bytes:32 time=0ms TTL=128
Reply from 10.0.0.4 : bytes:32 time=0ms TTL=128
ping statistics for 10.0.0.4
packets sent:4, Received :4 lost=0 (0% loss)
Approximate round trip times in milliseconds
Minimum=0ms Maximum=0ms Average=0ms

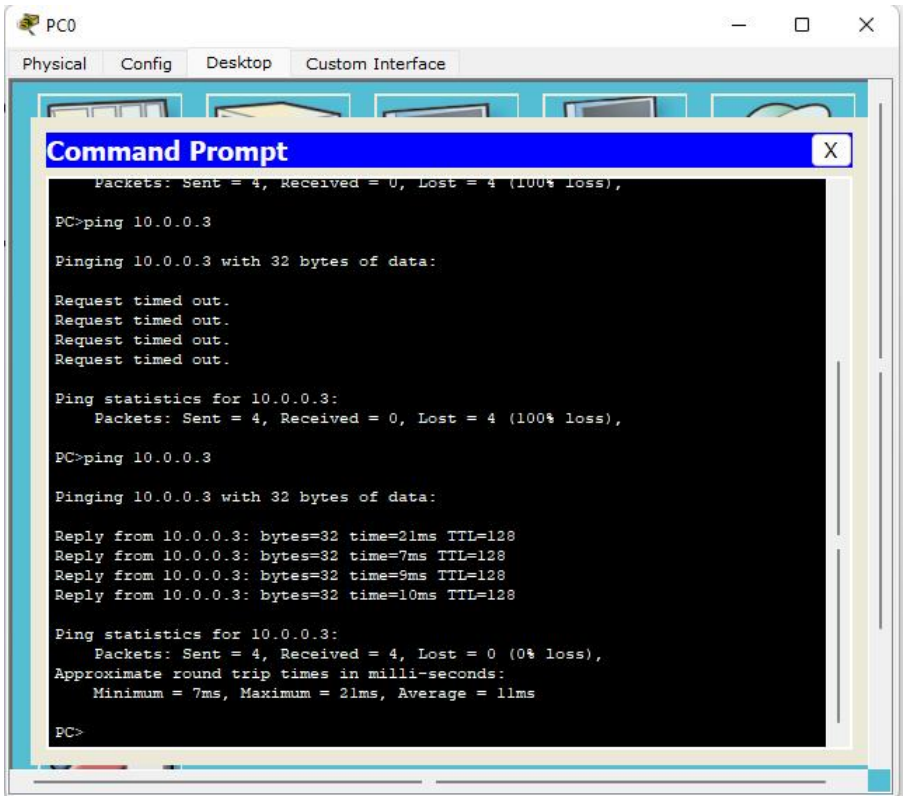
TOPOLOGY:


Cisco Packet Tracer Student
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.

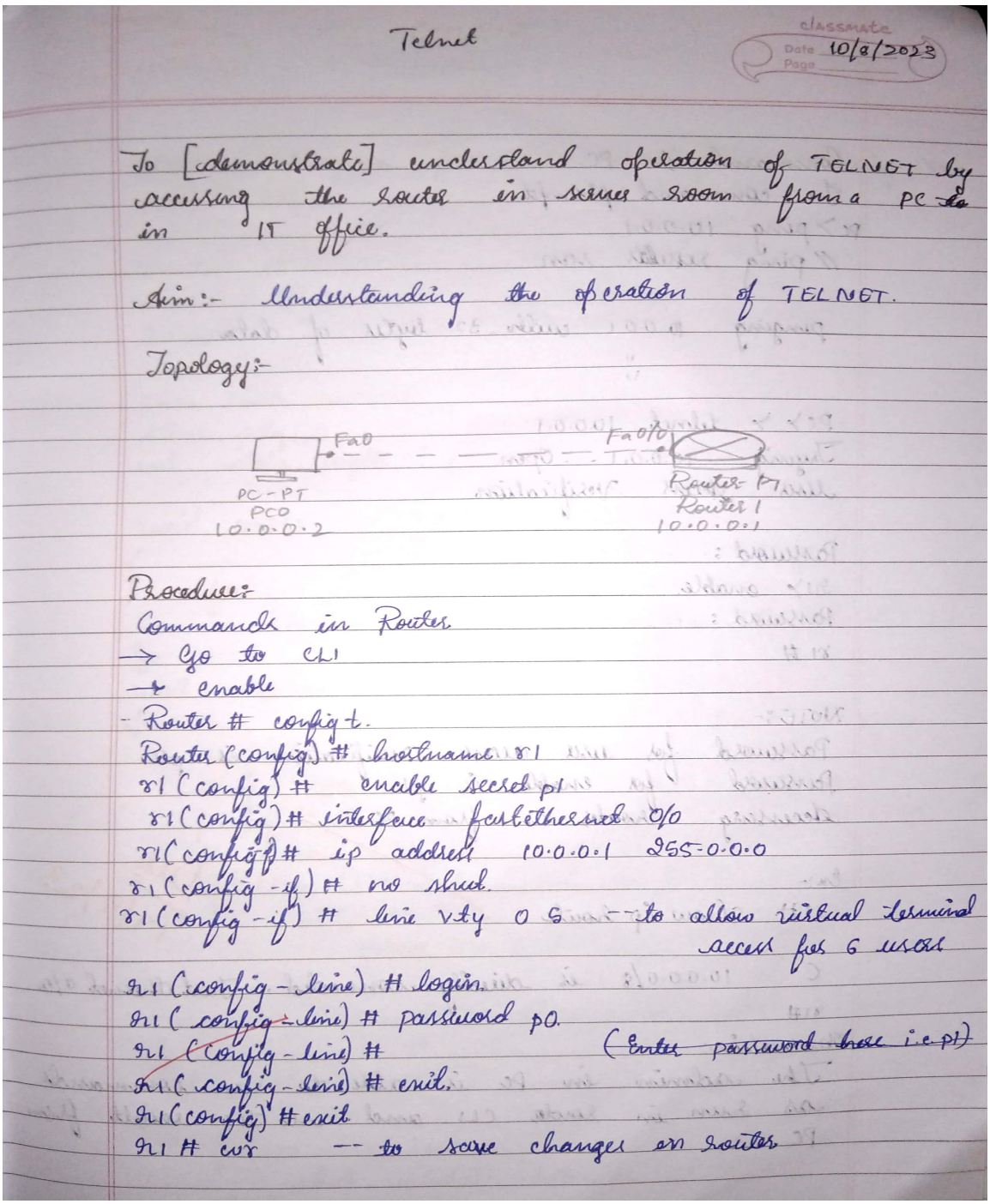
OUTPUT:


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:



Telnet

classmate Date 10/8/2023 Page

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

Aim:- Understanding the operation of TELNET.

Topology:-

Procedure:

Commands in Router.

→ Go to CLI

→ enable

- Router # config t.

Router(config) # hostname r1

r1(config) # enable secret p1

r1(config) # interface fastethernet 0/0

r1(config-if) # ip address 10.0.0.1 255.0.0.0

r1(config-if) # no shut.

r1(config-if) # line vty 0 5 -- to allow virtual terminal access for 6 users

r1(config-line) # login.

r1(config-line) # password p0.

r1(config-line) # (Enter password here i.e. p1)

r1(config-line) # exit.

r1(config) # exit

r1 # cw -- to save changes on router

Commands in PC.
In command prompt.

PC> ping 10.0.0.1
// ping results seen.

pinging 10.0.0.1 with 32 bytes of data.

PC> > telnet 10.0.0.1
Trying 10.0.0.1 ... Open
User access Verification

Password :
R1> enable
Password :
R1#

NOTE :-
Password for user access verification is p0.
Password for enable is p1
Accessing router CLI from PC

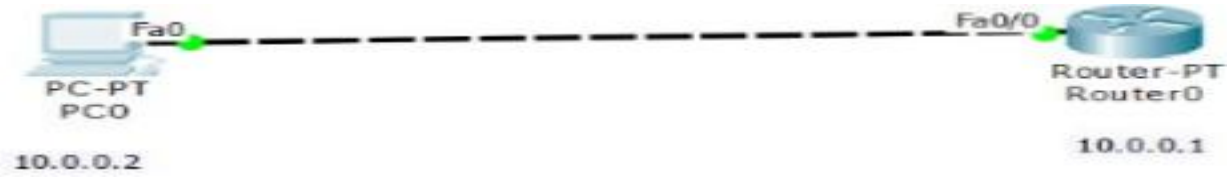
Ex:-
R1# show ip route

C 10.0.0.0/8 is directly connected, Fast Ethernet 0/0.
R1#

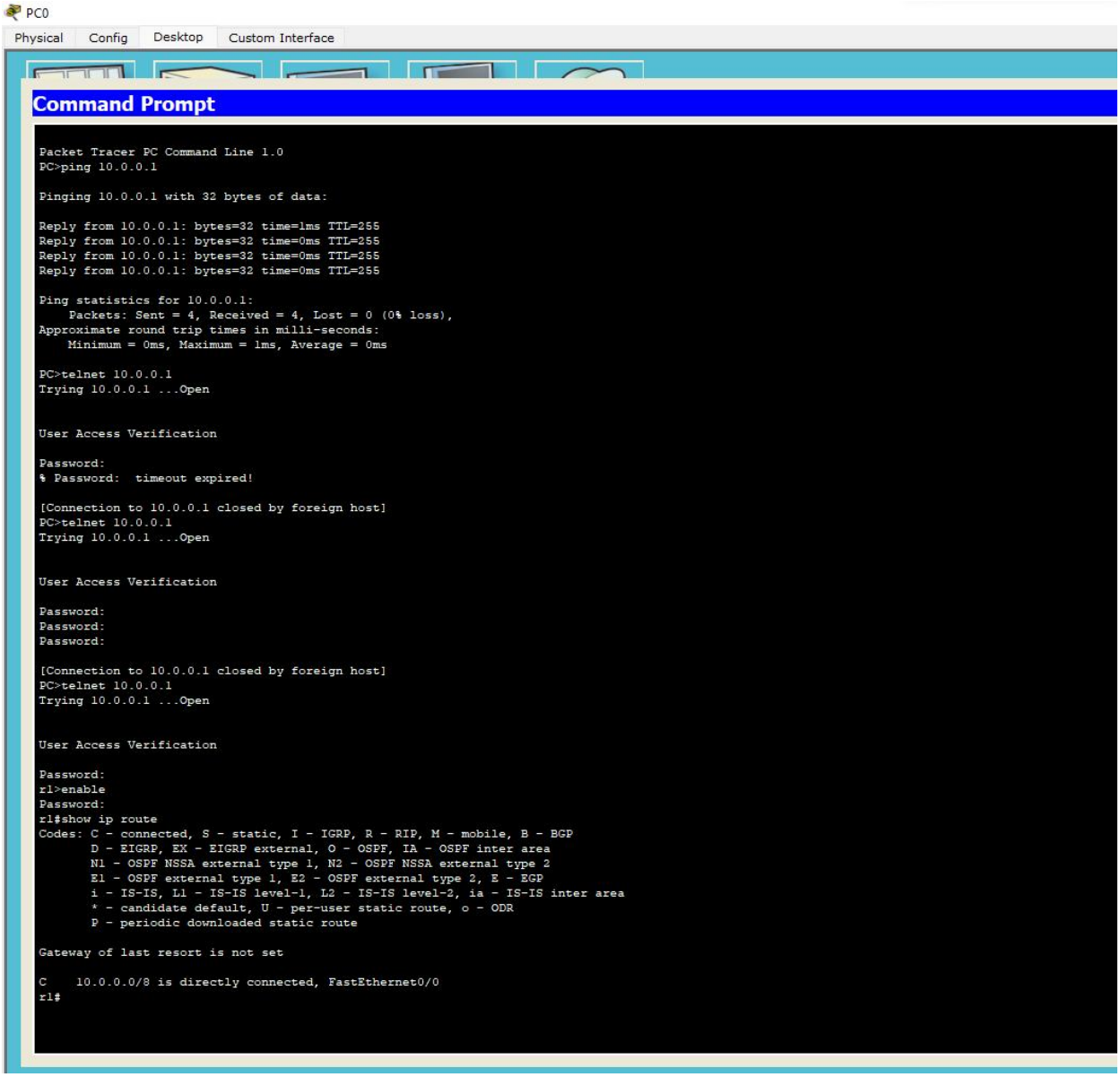
17/8/23

Observation:
The admin in PC is able to run commands as run in router CLI and see the result from PC.

TOPOLOGY:



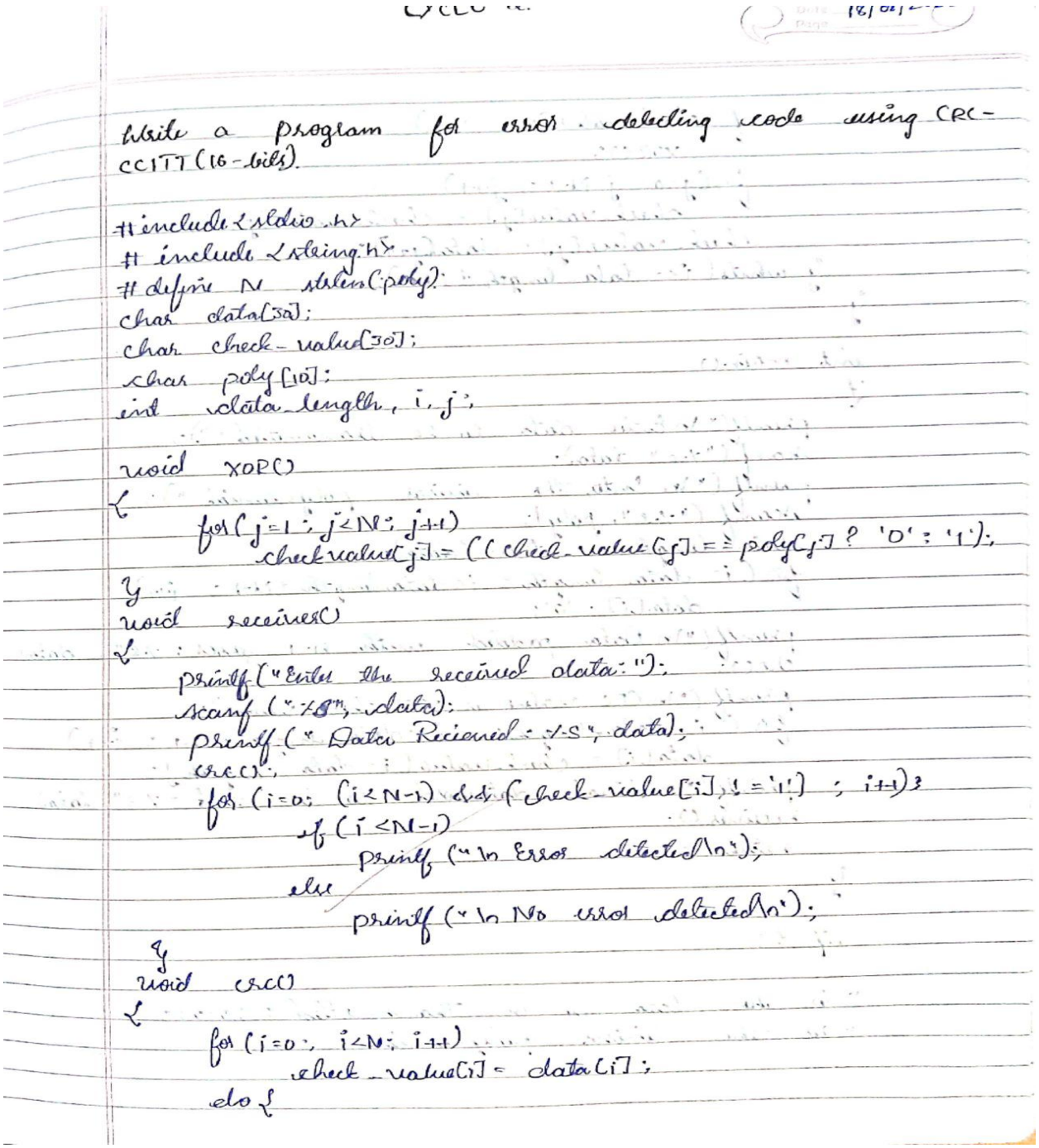
OUTPUT



WEEK 13
Program 1

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

Observation:



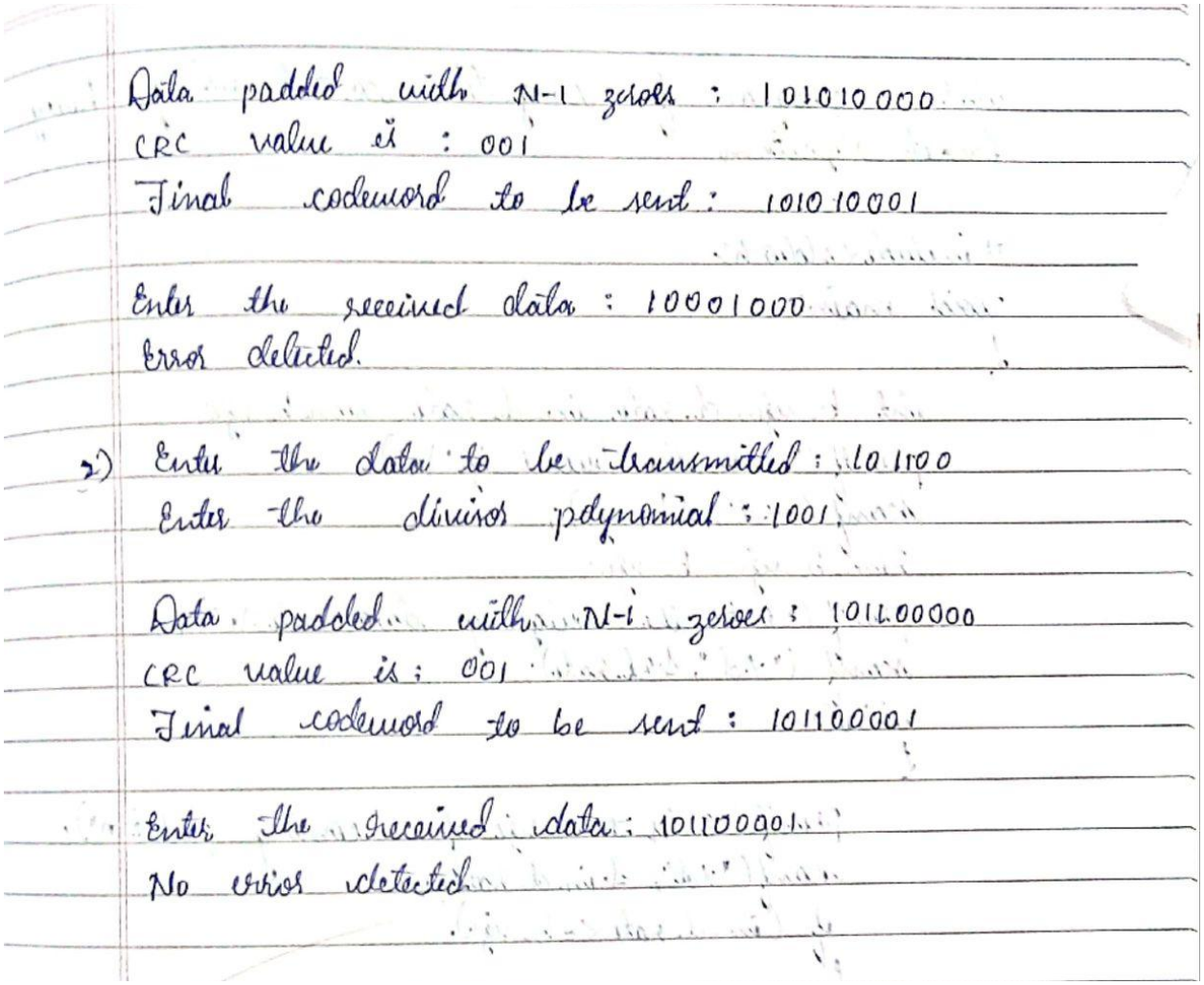
Write a program for error detecting code using CRC-CCITT (16-bits).

```
#include<stdio.h>
#include<string.h>
#define N strlen(poly)
char data[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 Recieved = %s", data);
    crc();
    for(i=0; (i<N-1)&&(check_value[i]!='1'); i++);
        if(i<N-1)
            printf("\n Error detected\n");
        else
            printf("\n No error detected\n");
}
void crc()
{
    for(i=0; i<N; i++)
        check_value[i]=data[i];
    do{
```

```
        if (check_value[0] == '1')
            XOR();
        for(j=0; j<N-1; j++)
            check_value[j] = check_value[j+1];
        check_value[j] = data[j+1];
    } while(i<= data_length + N-1);
}

int main()
{
    printf("\nEnter data to be transmitted: ");
    scanf("%s", data);
    printf("\nEnter the divisor polynomial: ");
    scanf("%s", poly);
    data_length = strlen(data);
    for(i= data_length; i<data_length+N-1; i++)
        data[i] = '0';
    printf("\nData padded with n-1 zeros : %s", data);
    crc();
    printf("\nCRC value is %s", check_value);
    for(i= data_length; i<data_length+N-1; i++)
        data[i] = check_value[i-data_length];
    printf("\nFinal datawords to be sent = %s", data);
    receiver();
    return 0;
}
```

Output :

Enter the data to be transmitted : 101010
Enter the divisor polynomial : 101



CODE:

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

}
```

```
void receiver(){
   printf("Enter the received data: ");
   scanf("%s", data);
   printf("Data received: %s", data);
   crc();
   for(i=0;(i<N-1) && (check_value[i]!='1');i++);
      if(i<N-1)
         printf("\nError detected\n\n");
      else
         printf("\nNo error detected\n\n");
}

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

int main()
{
   printf("\nEnter data to be transmitted: ");
   scanf("%s",data);
   printf("\n Enter the Generating polynomial: ");
   scanf("%s",gen_poly);
   data_length=strlen(data);
   for(i=data_length;i<data_length+N-1;i++)
      data[i]='0';
   printf("\n Data padded with n-1 zeros : %s",data);
   crc();
```

```
    printf("\nCRC or Check value is : %s",check_value);
    for(i=data_length;i<data_length+N-1;i++)
        data[i]=check_value[i-data_length];
    printf("\n Final data to be sent : %s",data);
    receiver();
        return 0;
}
```

OUTPUT:

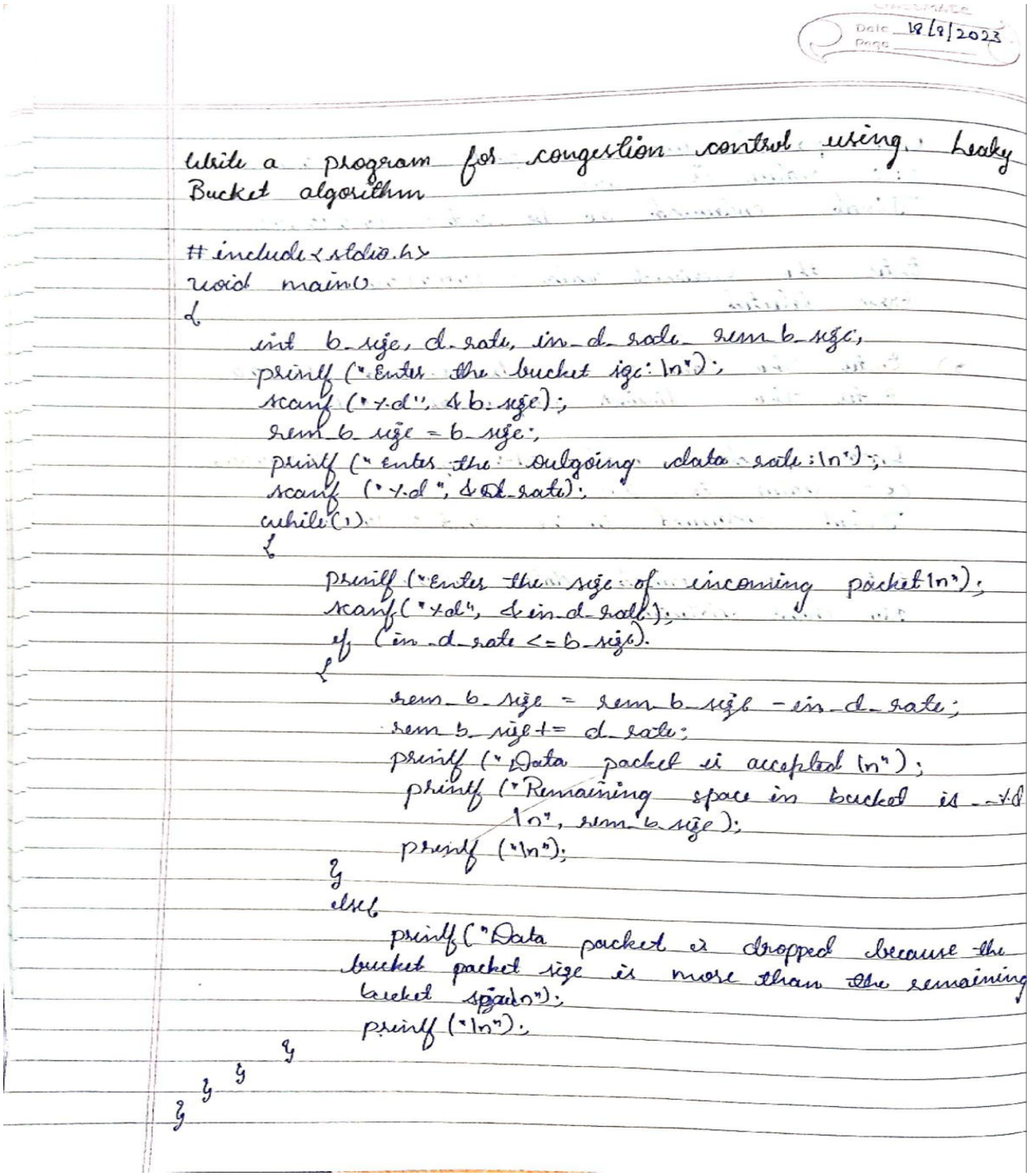
```
Enter data to be transmitted: 10001000000100001

 Enter the Generating polynomial: 1011

 Data padded with n-1 zeros : 10001000000100001000
CRC or Check value is : 100
 Final data to be sent : 10001000000100001100
Enter the received data: 10001000000100001100
Data received: 10001000000100001100
No error detected
```

```
  Enter data to be transmitted: 10001000000100001

   Enter the Generating polynomial: 1011

   Data padded with n-1 zeros : 10001000000100001000
  CRC or Check value is : 100
   Final data to be sent : 10001000000100001100
  Enter the received data: 10010000000100001100
  Data received: 10010000000100001100
  Error detected
```

Program 2

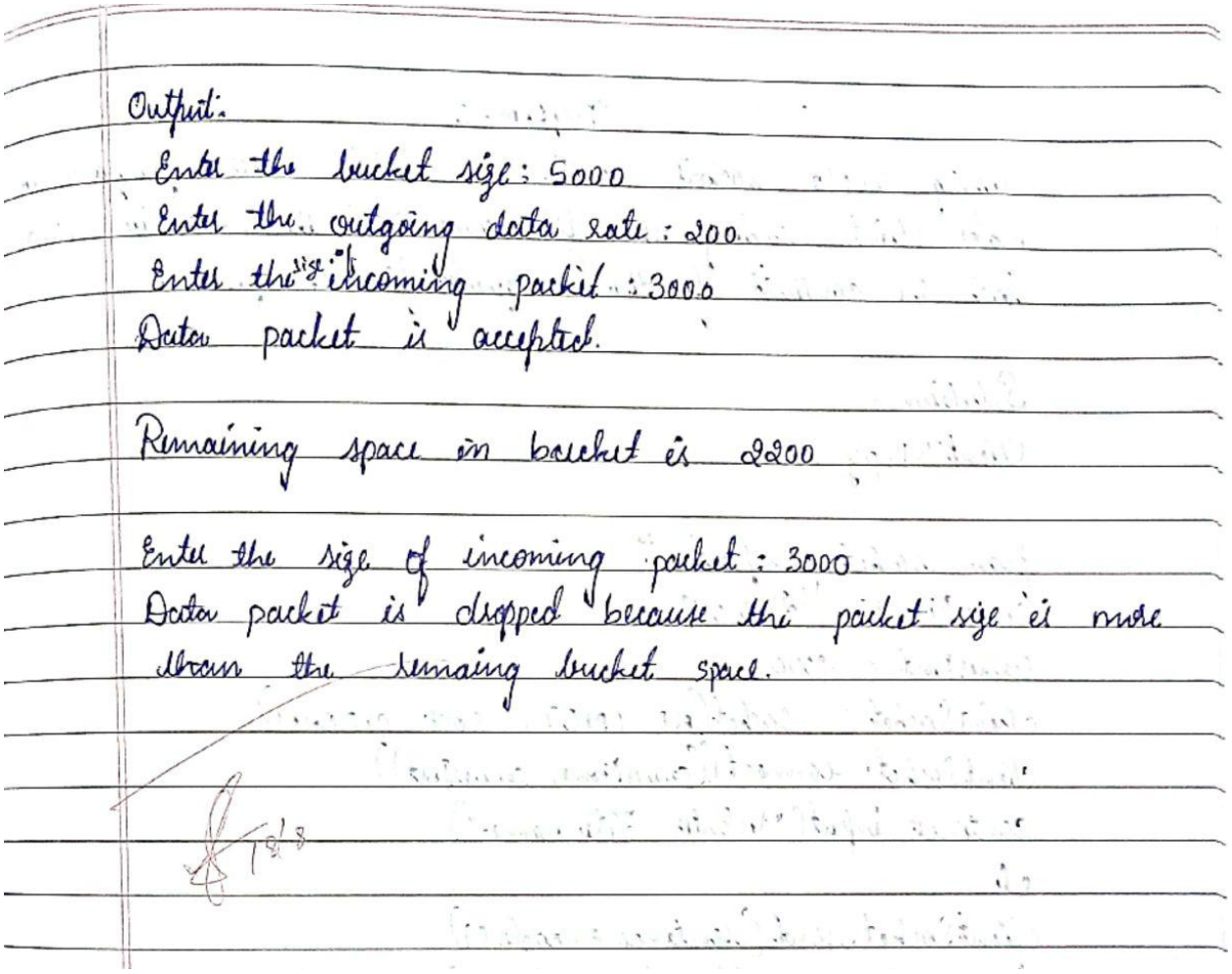
Write a program for congestion control using Leaky bucket algorithm.

Observation:



Write a program for congestion control using Leaky Bucket algorithm

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

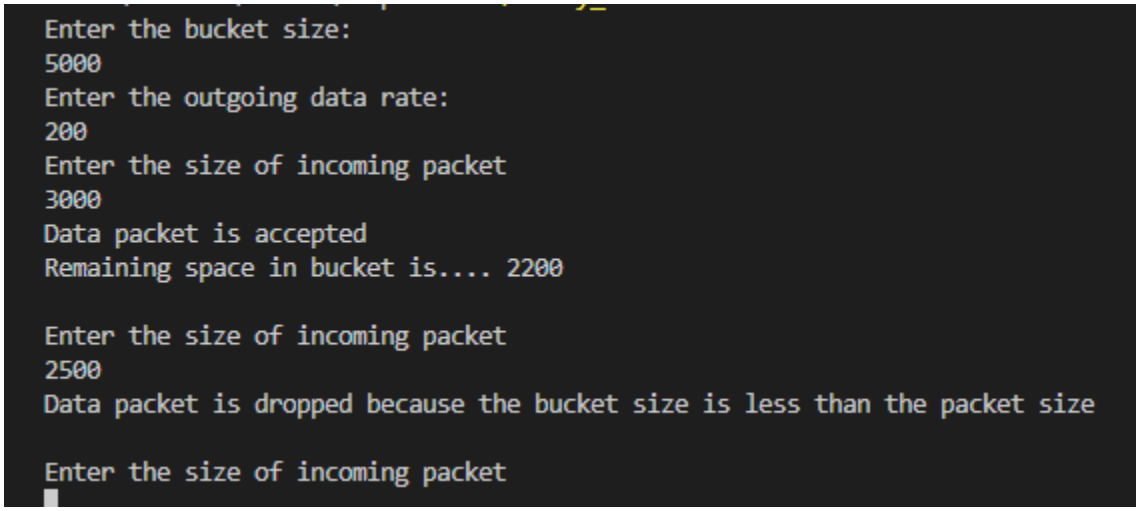


CODE:

```
#include<stdio.h>
void main()
{
   int b_size,d_rate,in_d_rate,rem_b_size;
   printf("Enter the bucket size:\n");
   scanf("%d",&b_size);
   rem_b_size=b_size;
   printf("Enter the outgoing data rate:\n");
   scanf("%d",&d_rate);
   while(1)
   {
```

```
    printf("Enter the size of incoming packet\n");
    scanf("%d",&in_d_rate);
    if(in_d_rate<=b_size)
    {
       if(in_d_rate<=rem_b_size)
       {
       rem_b_size=rem_b_size-in_d_rate;
       rem_b_size=rem_b_size+d_rate;
       printf("Data packet is accepted\n");
       printf("Remaining space in bucket is.... %d\n",rem_b_size);
       printf("\n");
       }
       else{
       printf("Data packet is dropped because the bucket size is less than the packet
size\n");
       printf("\n");
    }
    }
    }
}
```

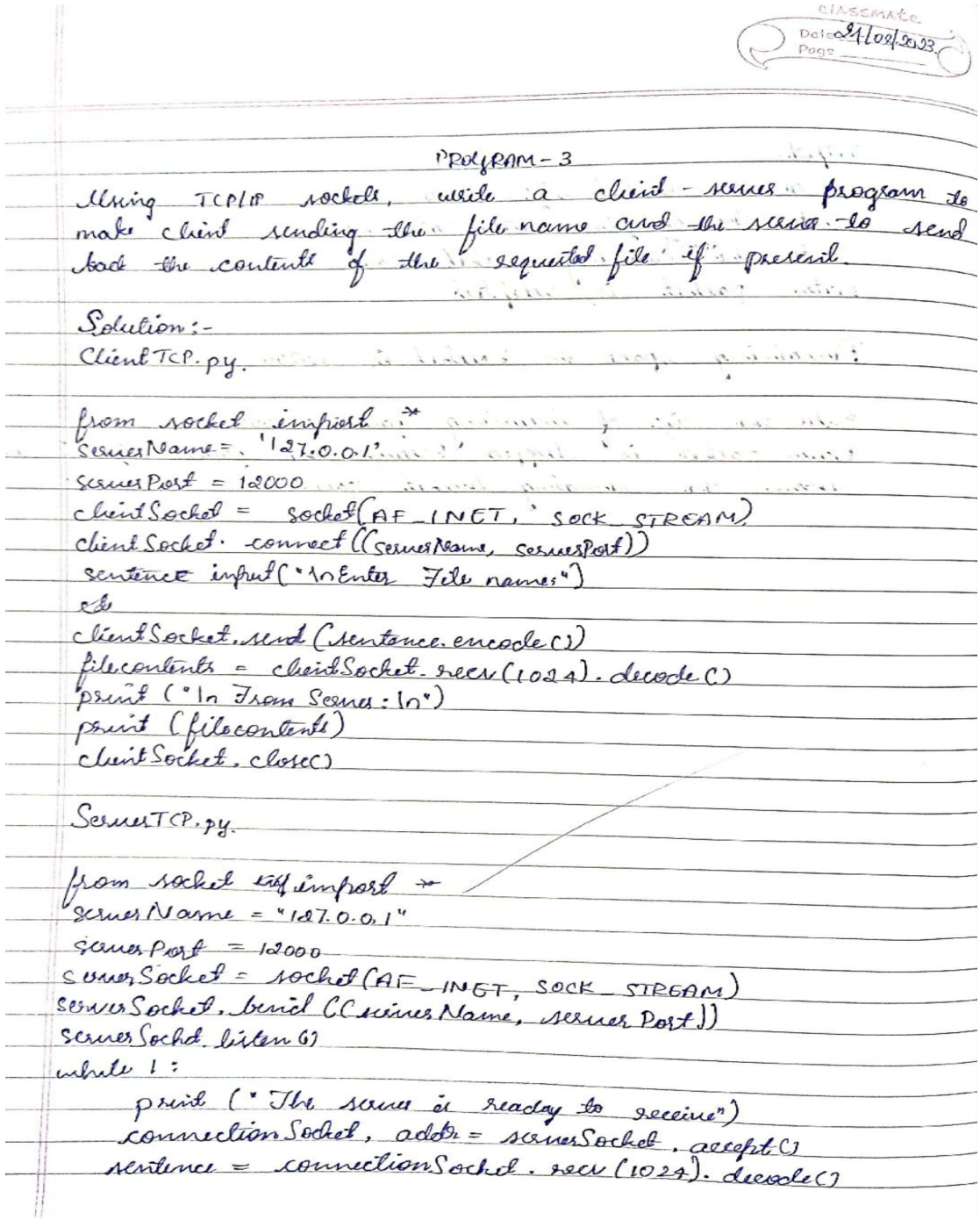
OUTPUT:



WEEK 14
Program 3

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

Observation:



PROGRAM - 3

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

Solution:-

ClientTCP.py.

```
from socket import *
serverName = '127.0.0.1'
serverPort = 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("\n From 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:

First Run serverTCP.py then run clientTCP.py.

output on serverTCP.

The server is ready to receive.

sent contents of serverTCP.py
The server is ready to receive.
:

Output on clientTCP terminal.

Enter file name: serverTCP.py.

From Server:
Contents in serverTCP.py is displayed here.

SOLUTION:

ClientTCP.py

```
from socket import *
serverName = '127.0.0.1'
serverPort = 12000
clientSocket = socket(AF_INET, SOCK_STREAM)
clientSocket.connect((serverName,serverPort))
sentence = input("\nEnter file name: ")

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

ServerTCP.py

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

    file=open(sentence,"r")
    l=file.read(1024)

    connectionSocket.send(l.encode())
    print ('\nSent contents of ' + sentence)
```

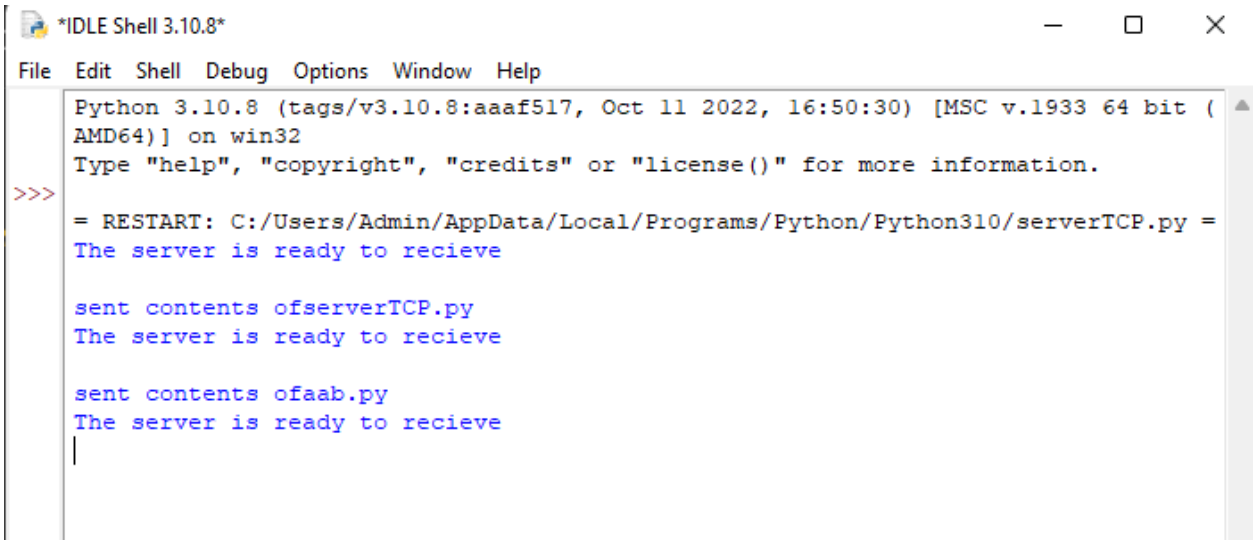
file.close()
connectionSocket.close()

OUTPUT:

Client:

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

Enter file name:aab.py

From Server:

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

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

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

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

        y.right=z
        z.left=T3

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

        return y

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

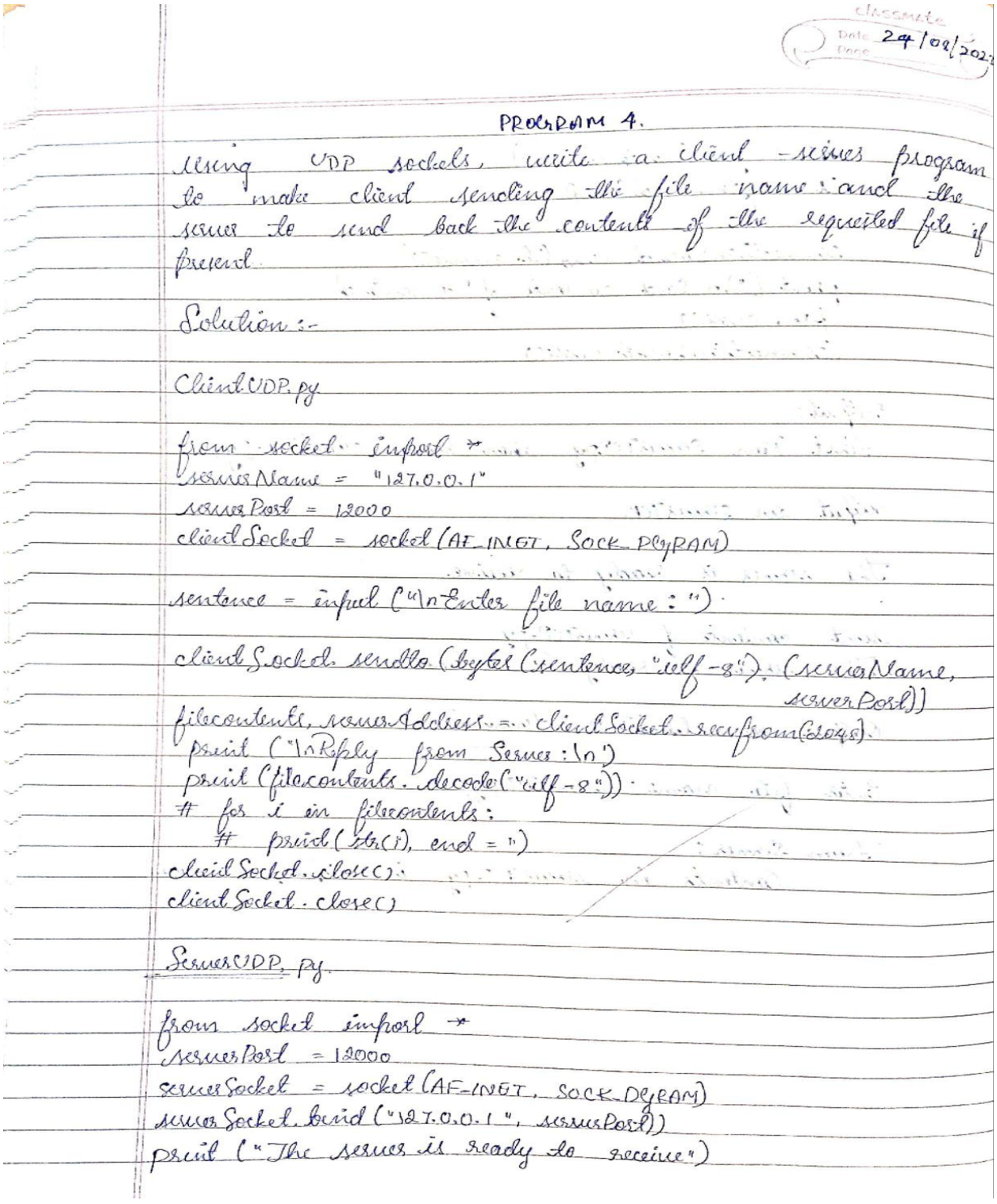
Server:



Program 4

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

Observation:



PROGRAM 4.

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

Solution :-

ClientUDP.py

```
from socket import *
serverName = "127.0.0.1"
serverPort = 12000
clientSocket = socket(AF_INET, SOCK_DGRAM)

sentence = input("\nEnter file name : ")

clientSocket.sendto(bytes(sentence, "utf-8"), (serverName,
                                               serverPort))

fileContents, serverAddress = clientSocket.recvfrom(2048)
print("\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:-

First run serverUDP.py then run clientUDP.py

Output on serverUDP.py

The server is ready to receive.

Sent contents of serverUDP.py

Output on clientUDP.py

Enter file name : serverUDP.py

Reply from server:

contents of serverUDP.py is read. Be displayed here

Same as serverUDP.py file

SOLUTION:

ClientUDP.py

```
from socket import *
serverName = "127.0.0.1"
serverPort = 12000
clientSocket = socket(AF_INET, SOCK_DGRAM)

sentence = input("\nEnter file name:  ")

clientSocket.sendto(bytes(sentence,"utf-8"),(serverName, serverPort))

filecontents,serverAddress = clientSocket.recvfrom(2048)
print ('\nReply from Server:\n')
print (filecontents.decode("utf-8"))
# for i in filecontents:
   # print(str(i), end = '')
clientSocket.close()
clientSocket.close()
```

ServerUDP.py

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

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

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

OUTPUT:

Client:

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

Enter file name:  serverUDP.py

Reply from Server:

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

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

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

>>>
```

Server:

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

Sent contents of  serverUDP.py
```

WEEK 15

Tool Exploration -Wireshark

Observation:

Date 31/08/2023
Page

PROGRAMS

Tool Exploration - Wireshark

Wireshark :-

Wireshark is a network protocol analyzer, or an application that captures packets from a network connection, such as from your computer to your home office or the internet.

Wireshark is the most often-used packet sniffer in the world.

→ Open Browser :-

And type wireshark download.

and select the appropriate software for your system and download it.

And install it.

Uses of the wireshark :-

1. It is used by network security, engineers, to examine security problems
2. It allows users to watch all the traffic being passed over the network
3. It is used by network engineers to troubleshoot network issues.
4. It also helps to troubleshoot latency and malicious activities on your network
5. It can also analyze dropped packets
6. It helps us to know how all the devices, like laptop, mobile phones, desktop, switch, routers etc. communicate in a local network or the rest of the world.

→ Open wireshark
And click on capture → start
Now we can see the packets that are sent by the system and received by the system and the protocol being used.

→ And we can view the source and destination address of the packet.
→ If we click on a particular packet. now you can see the ASCII code in the bottom

→ And wireshark uses different colours for the different protocols to represent

→ To see only your system participation in network in display filter type ip.addr == (ip address of the PC

ex:- ipaddr = 10.124.7.1

NOTE : To know the ip address of the system open cmd and type ipconfig.
You can now see the ip address of your system

31/8/2023

→ And we can even filter packets by the type of the protocol

Go to Analyze and display filters to see the filters
To stop capturing go to capture and click on stop

To colorize packet list go to view and click on colorize packet list