

VISVESVARAYA TECHNOLOGICAL UNIVERSITY

“JnanaSangama”, Belgaum -590014, Karnataka.



LAB REPORT
On

Computer Networks

Submitted by

SKANDA M SHASTRY (1BM21CS212)

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

May-2023 to July-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 **SKANDA M SHASTRY (1BM21CS212)**, 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 academic semester May- 2023 to July-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.

Prameetha Pai
Assistant Professor
Department of CSE
BMSCE, Bengaluru

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

Index Sheet

Lab Program No.	CYCLE 1	Page No.
	Program Details	
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.	5
2	Configure IP address to routers in packet tracer. Explore the following messages: ping responses, destination unreachable, request timed out and reply.	6
3	Configure default route, static route to the Router.	10
4	Configure DHCP within a LAN and outside LAN.	21
5	Configure RIP routing Protocol in Routers.	25
6	Configure OSPF routing protocol.	33
7	Demonstrate the TTL/ Life of a Packet.	46
8	Configure Web Server, DNS within a LAN.	52
9	To construct simple LAN and understand the concept and operation of Address Resolution Protocol (ARP).	54
10	To understand the operation of TELNET by accessing the router in server room from a PC in IT office.	61
11	To construct a WLAN and make the nodes communicate wirelessly.	65
12	To construct a VLAN and make the PC's communicate among a VLAN.	69

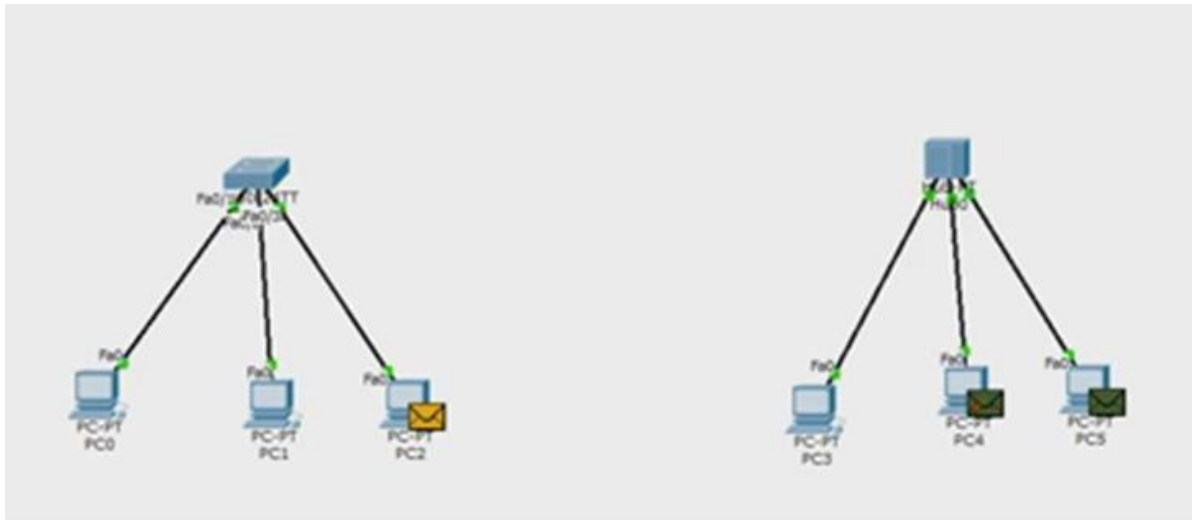
Lab Program No.	CYCLE 2	Page No.
	Program Details	
1	Write a program for congestion control using Leaky bucket algorithm.	73
2	Using TCP/IP sockets, write a client-server program to make client sending the file name and the server to send back the contents of the requested file if present.	75
3	Using 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.	77
4	Write a program for error detecting code using CRCCCITT (16-bits).	79
5	WIRESHARK TOOL EXPLORATION	84

Course Outcome

CO1	Apply the fundamental concepts of communication in networking.
CO2	Analyze the various protocols, techniques in TCP/IP network architecture.
CO3	Develop programs that demonstrate the functionalities of physical, Data Link, Network, Transport or Application layer.

1) Create a topology and simulate sending a simple PDU from source to destination using hub and switch as connecting devices.

Topology:

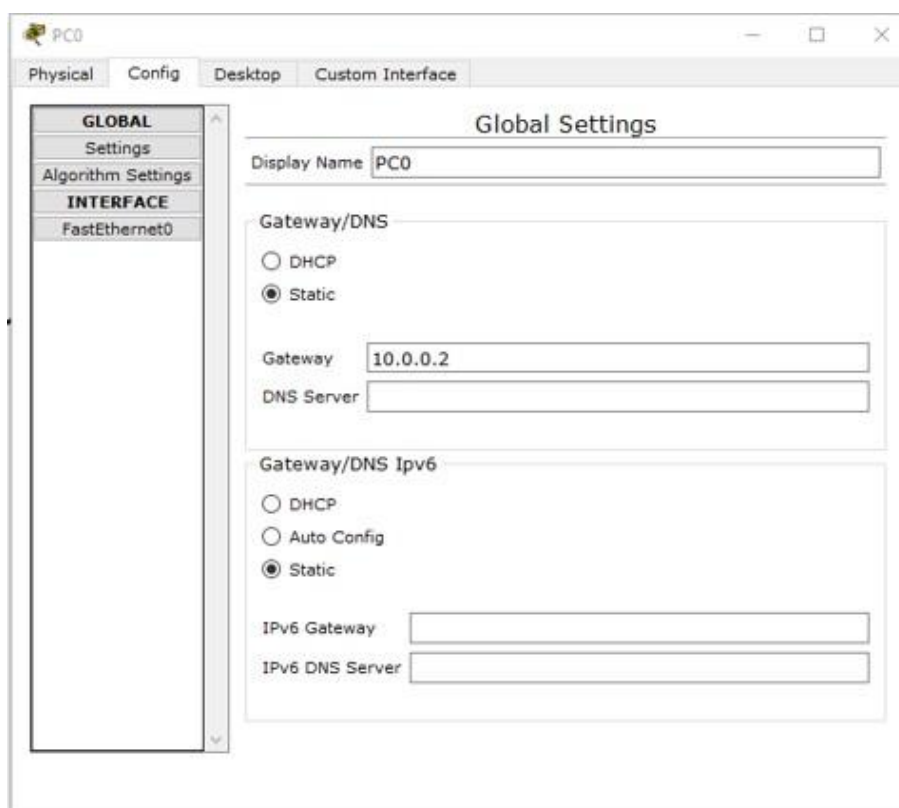
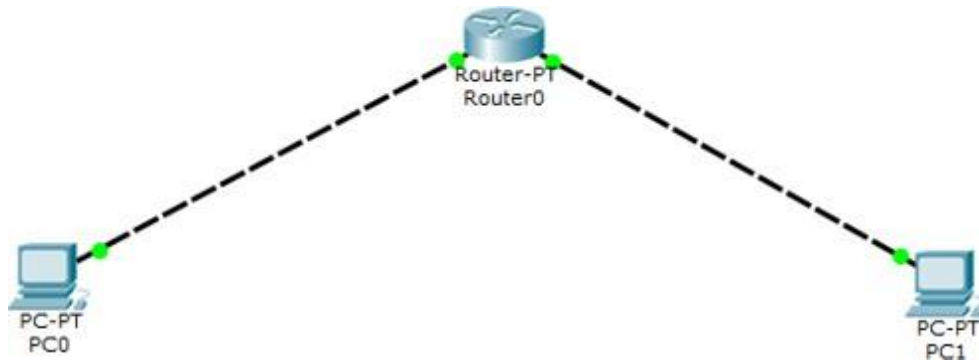


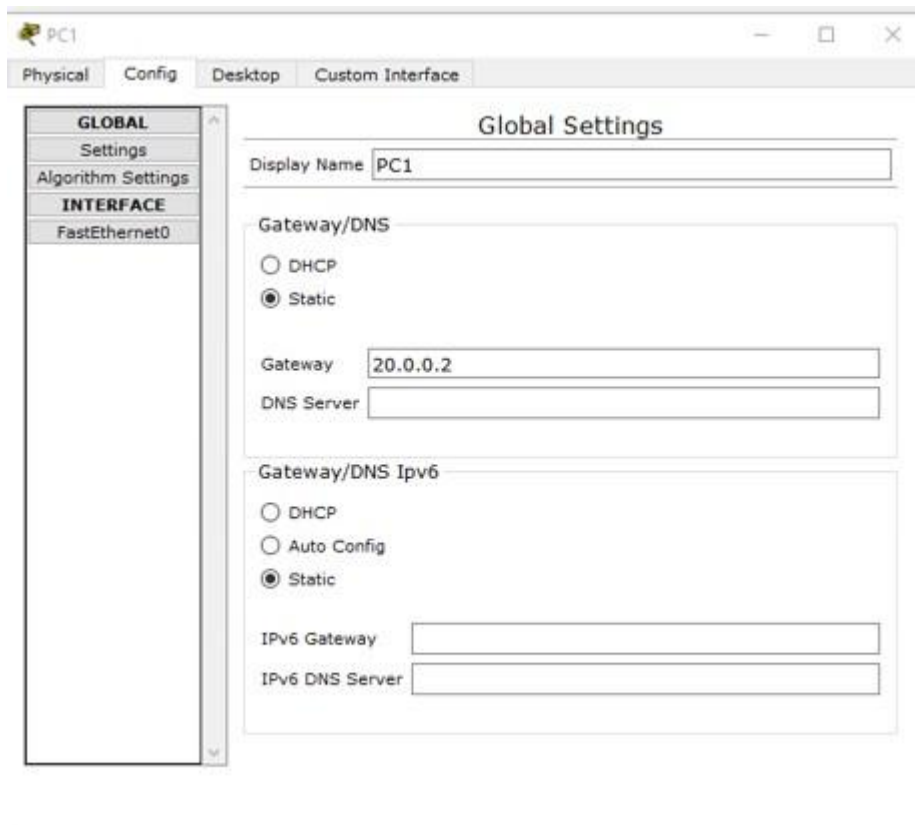
OUTCOME:

- We were able to figure out how the Switches and hub transfer the data from one end device to another.

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

Topology:





Now configure router interface with ip address and subnet mask then give no shutdown to make this interface and line protocol up(i.e. Carefully configure ip address with proper interfaces in this case f0/0 and f1/0,f is short form of fastethernet.

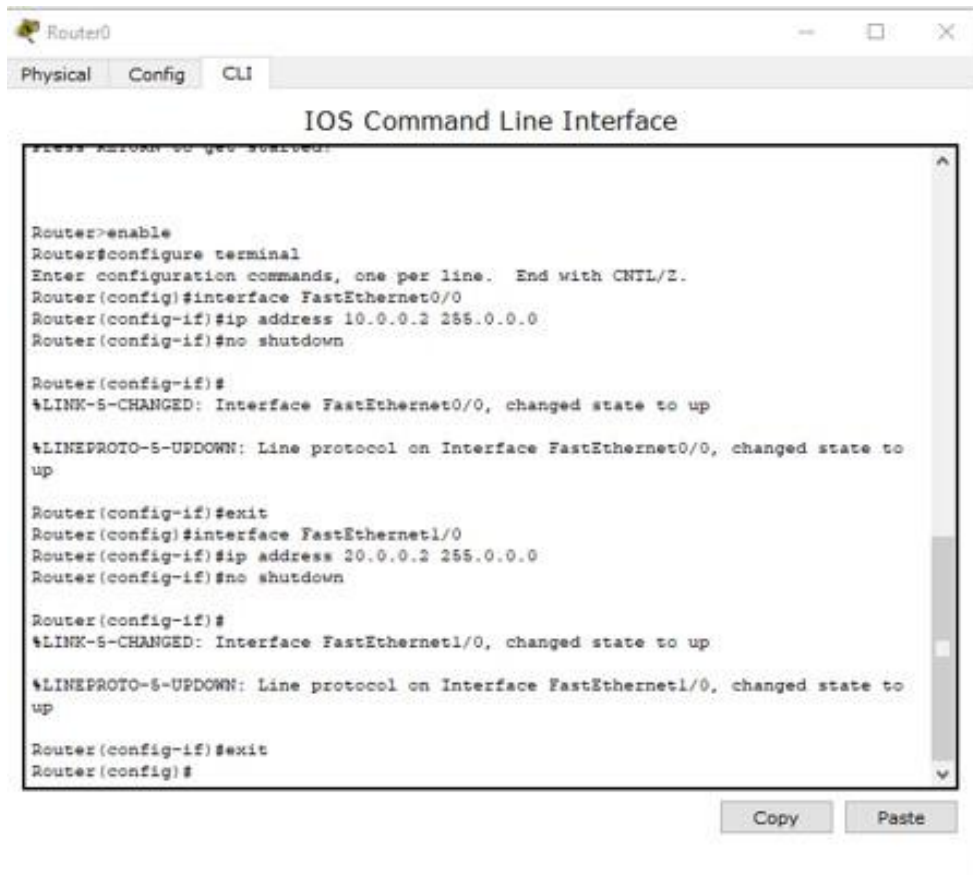
```
Router(config)#interface fastEthernet 0/0
Router(config-if)#ip address 10.0.0.2 255.0.0.0
Router(config-if)#no shutdown
Router(config-if)#exit
```

Interface Line protocol on FastEthernet0/0, changed state to up

```
Router(config)#interface fastethernet 1/0
Router(config-if)#ip address 20.0.0.2 255.0.0.0
Router(config-if)#no shutdown
Router(config-if)#exit
```

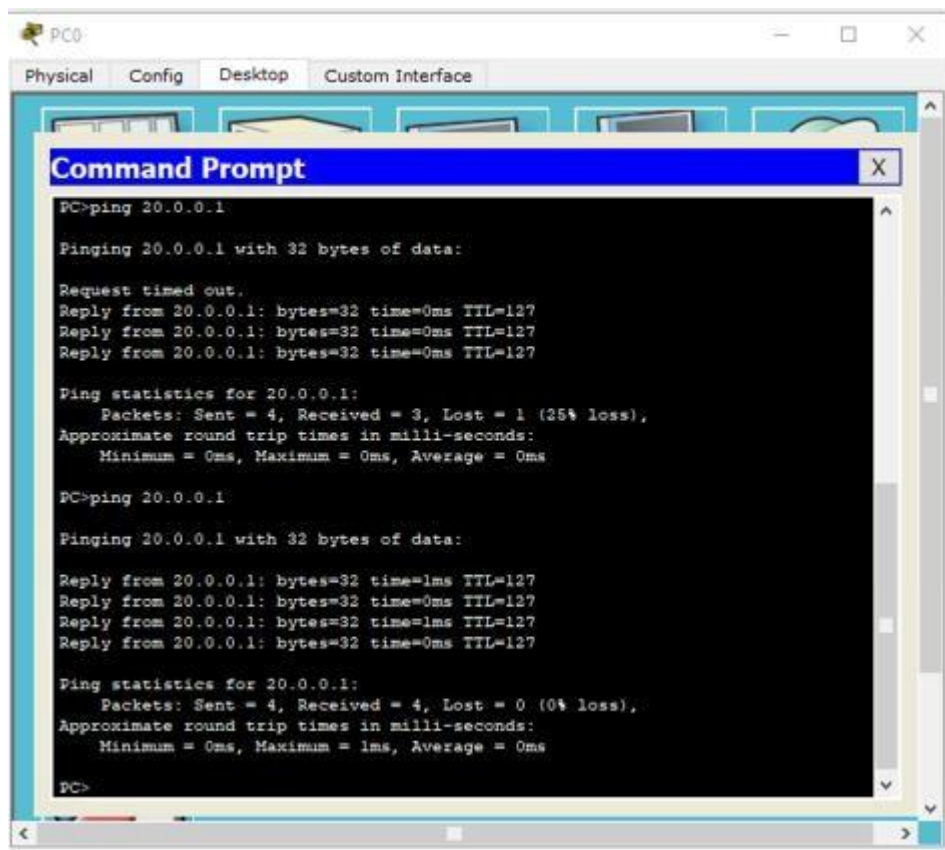
Interface Line protocol on FastEthernet1/0, changed state to up

Now lights on all ports become green from red.Now click on PC1->Desktop->Command Prompt.



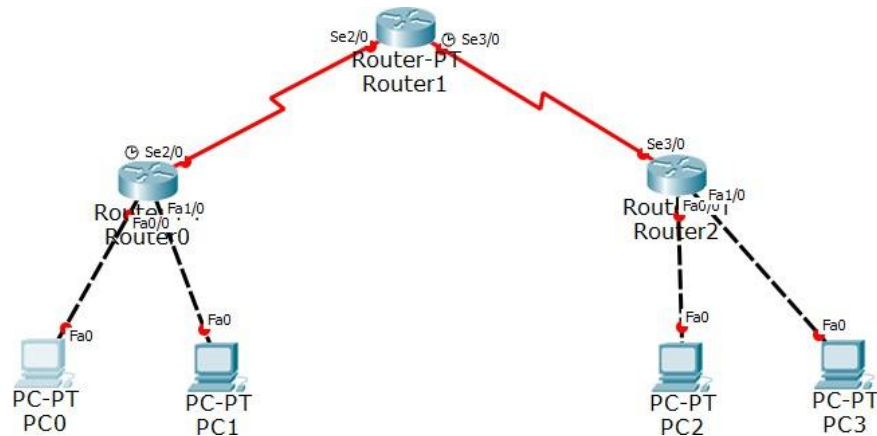
Now give this command "ping 20.0.0.1" and press enter. you will get, connectivity between 10.0.0.1 and 20.0.0.1 is ok. Now PC1 communicates with PC2

Another way of checking connectivity is, select "simple PDU packet" from right side of packet tracer and select source PC and Destination PC. You will get response at right bottom of the packet tracer window.



3) Configure default, static route to the router.

Static routing:



Topology

Configure IP address and default gateway of PC'S

Configure the routers as shown below

R0

```
Router0
Physical Config CLI
IOS Command Line Interface
Router(config)#interface fastEthernet0/0
Router(config-if)#ip address 10.0.0.3 255.0.0.0
Router(config-if)#no shutdown

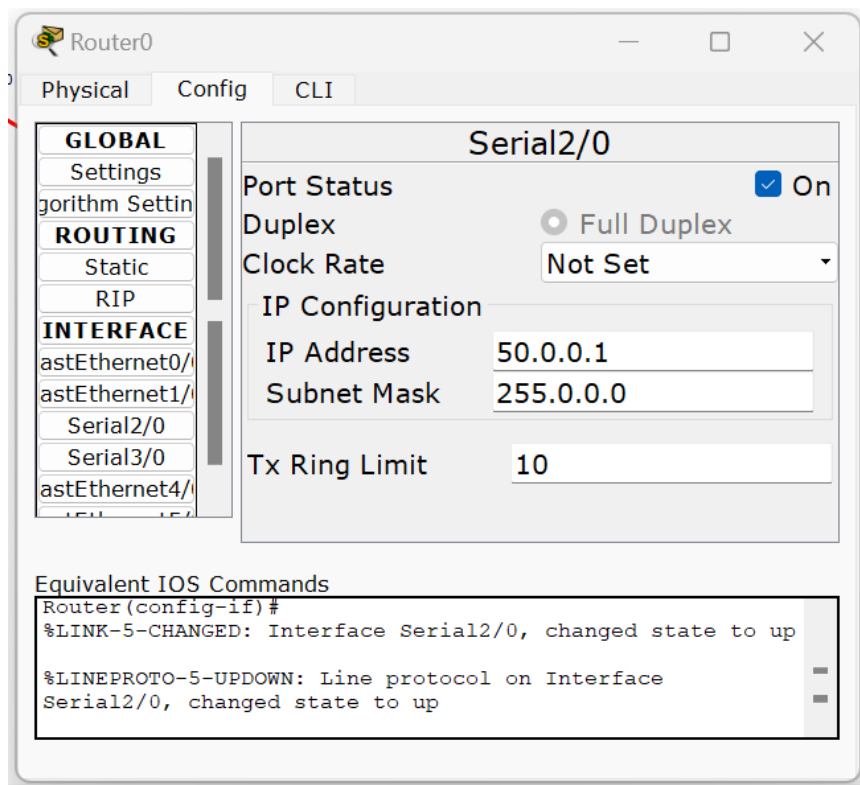
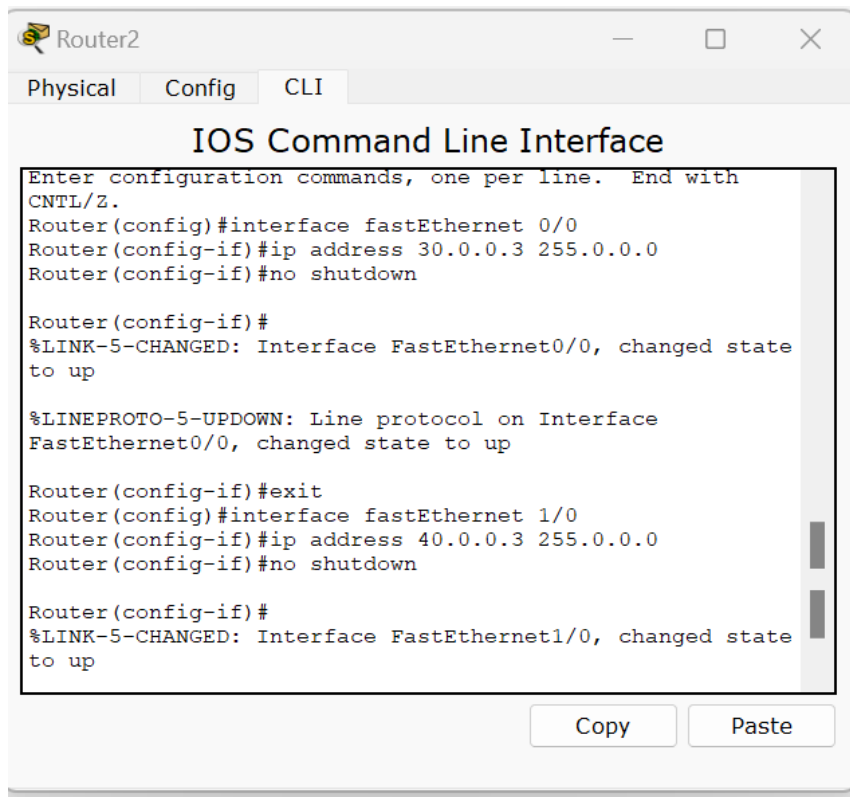
Router(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/0, changed state
to up

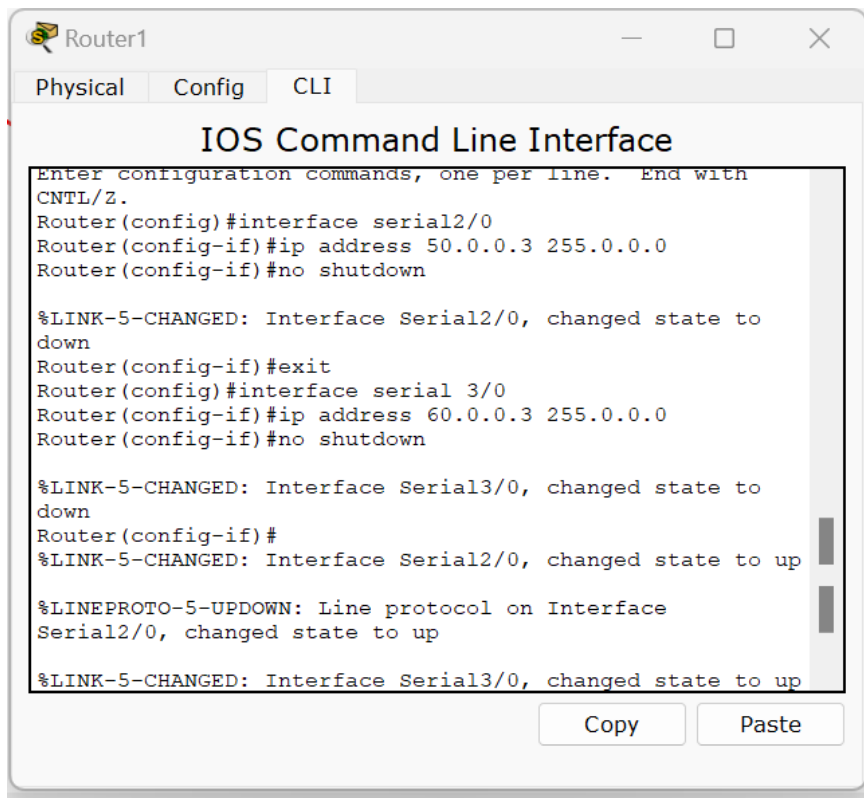
%LINEPROTO-5-UPDOWN: Line protocol on Interface
FastEthernet0/0, changed state to up

Router(config-if)#exit
Router(config)#interface fastEthernet1/0
Router(config-if)#ip address 20.0.0.3 255.0.0.0
Router(config-if)#no shutdown

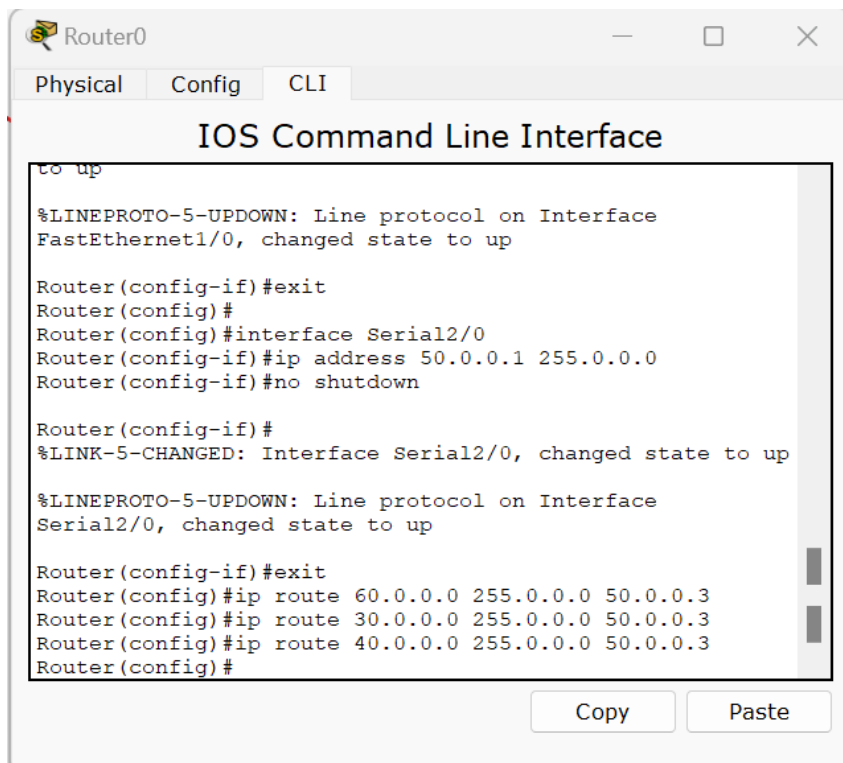
Router(config-if)#
%LINK-5-CHANGED: Interface FastEthernet1/0, changed state
to up
```

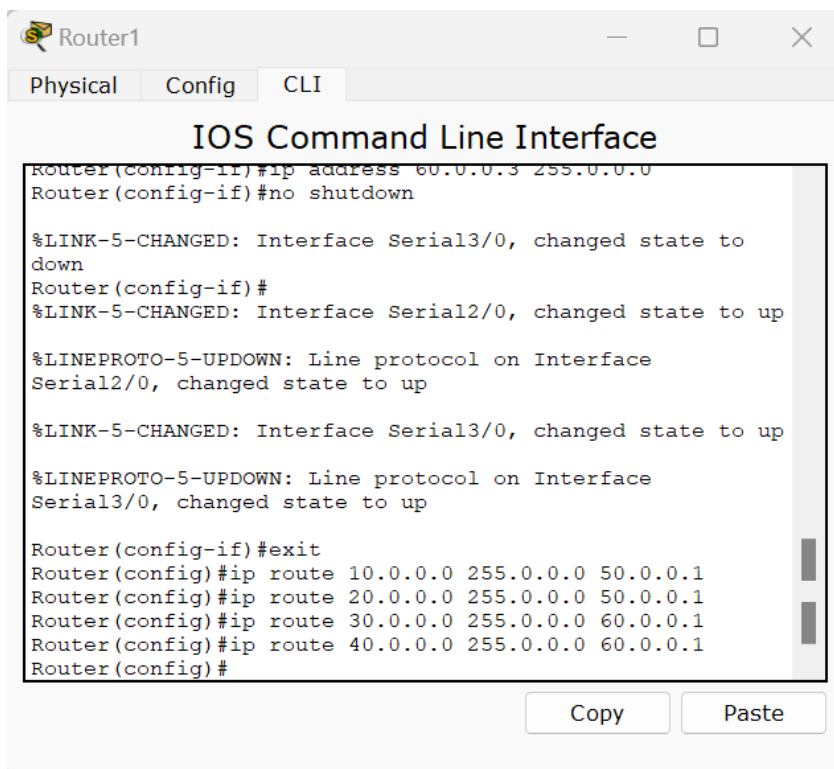
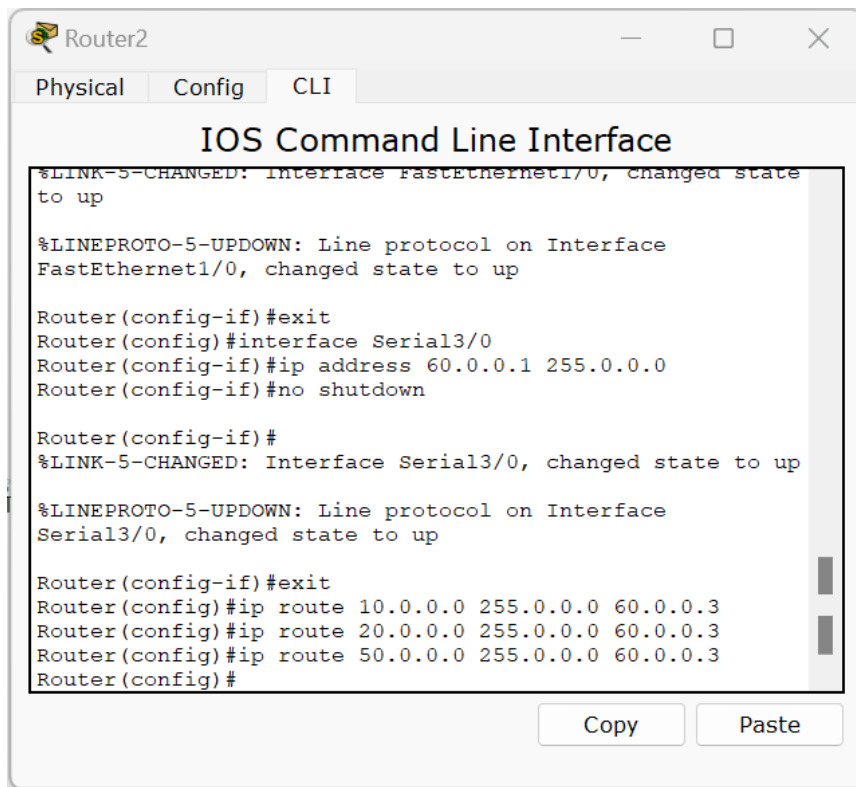
R2





IP ROUTE COMMANDS





```
Command Prompt
Ping statistics for 30.0.0.1:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 2ms, Maximum = 17ms, Average = 11ms

PC>ping 30.0.0.1

Pinging 30.0.0.1 with 32 bytes of data:

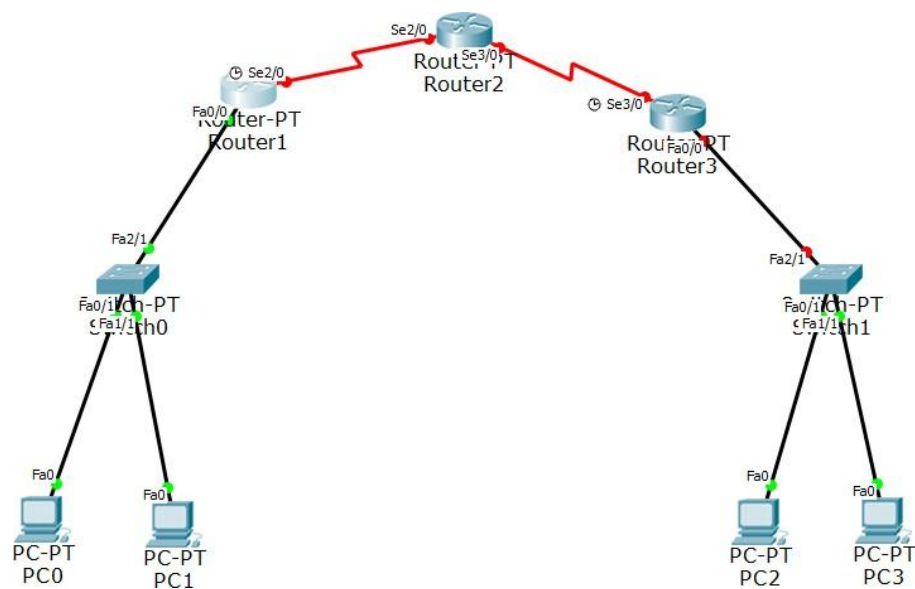
Reply from 30.0.0.1: bytes=32 time=12ms TTL=125
Reply from 30.0.0.1: bytes=32 time=11ms TTL=125
Reply from 30.0.0.1: bytes=32 time=2ms TTL=125
Reply from 30.0.0.1: bytes=32 time=2ms TTL=125

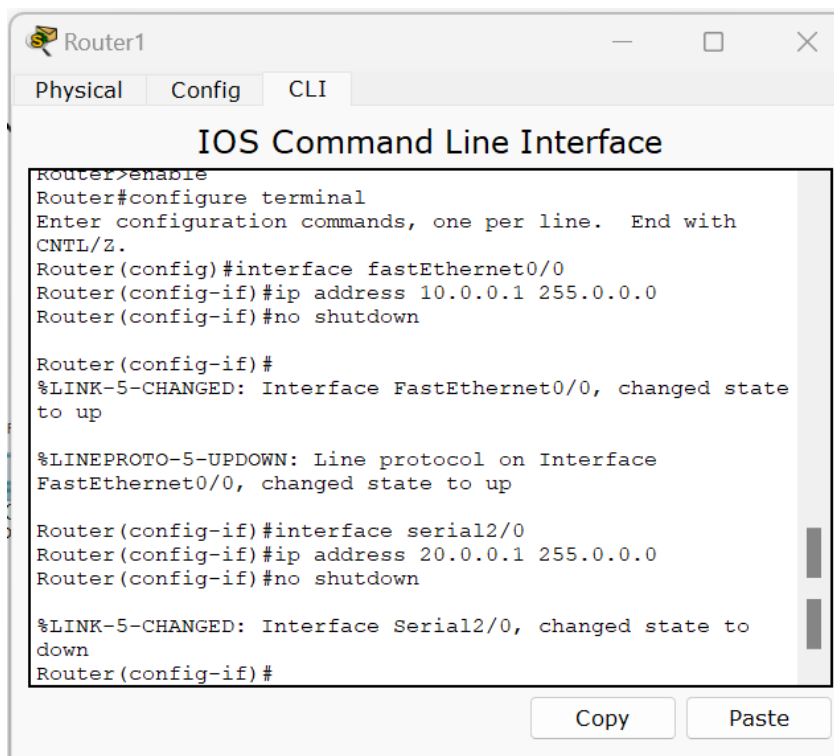
Ping statistics for 30.0.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 2ms, Maximum = 12ms, Average = 6ms

PC>
```

Default routing

Set up topology as shown





Router 2

```
System Configuration Dialog

Continue with configuration dialog? [yes/no]: no

Press RETURN to get started!

Router>enable
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#configure terminal
      ^
% Invalid input detected at '^' marker.

Router(config)#interface serial2/0
Router(config-if)#ip address 20.0.0.2 255.0.0.0
Router(config-if)#no shutdown

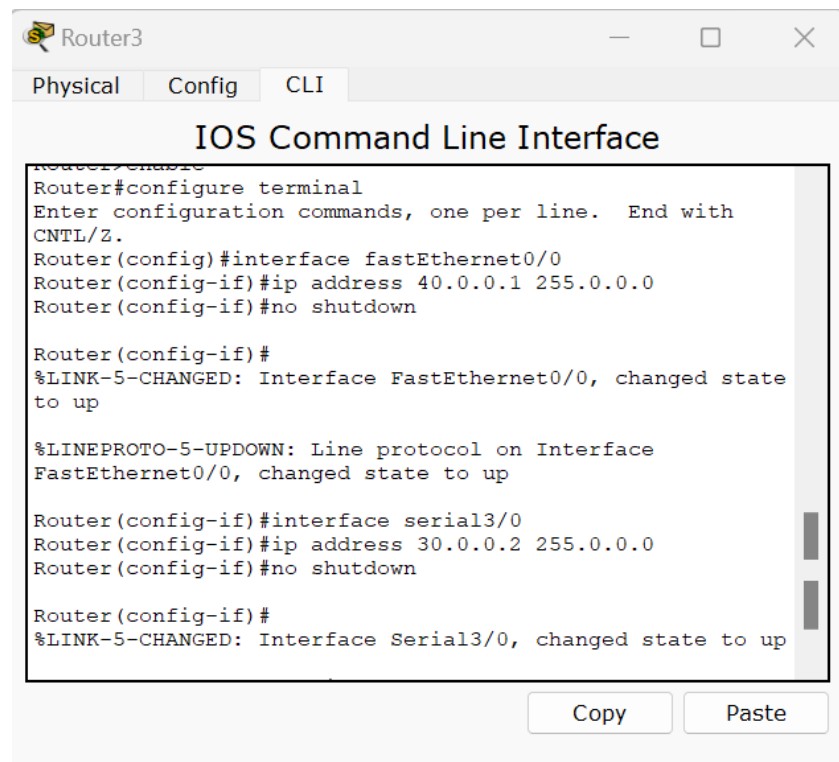
Router(config-if)#
%LINK-5-CHANGED: Interface Serial2/0, changed state to up

Router(config-if)#exit
Router(config)#
Router(config)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to up

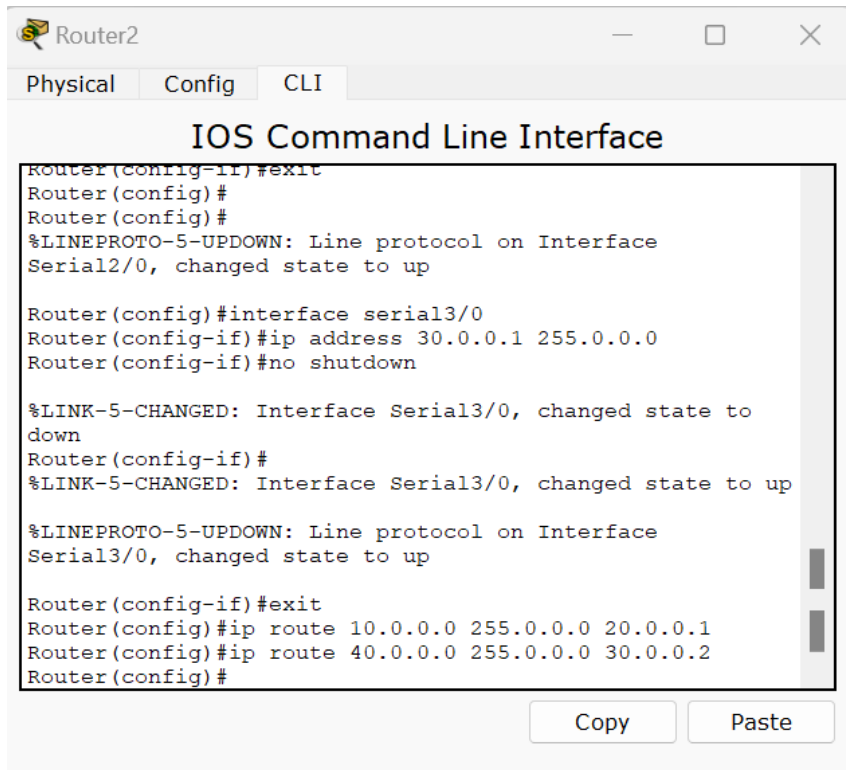
Router(config)#interface serial3/0
Router(config-if)#ip address 30.0.0.1 255.0.0.0
Router(config-if)#no shutdown

%LINK-5-CHANGED: Interface Serial3/0, changed state to down
Router(config-if)#
```

Router3



Configure static route for middle router R2 we have to do it for 40 & 10 network.



The screenshot shows the CLI window for Router2. The 'Config' tab is selected. The command history shows the following sequence of commands and their outputs:

```
Router(config-if)#exit
Router(config)#
Router(config)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface
Serial2/0, changed state to up

Router(config)#interface serial3/0
Router(config-if)#ip address 30.0.0.1 255.0.0.0
Router(config-if)#no shutdown

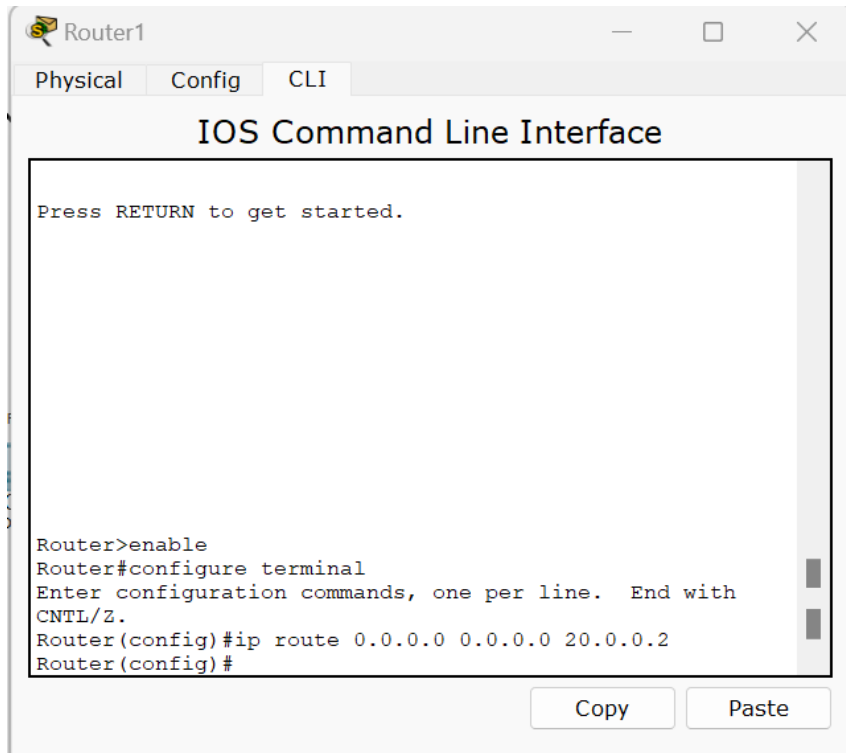
%LINK-5-CHANGED: Interface Serial3/0, changed state to
down
Router(config-if)#
%LINK-5-CHANGED: Interface Serial3/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface
Serial3/0, changed state to up

Router(config-if)#exit
Router(config)#ip route 10.0.0.0 255.0.0.0 20.0.0.1
Router(config)#ip route 40.0.0.0 255.0.0.0 30.0.0.2
Router(config)#
```

At the bottom of the window, there are 'Copy' and 'Paste' buttons.

Default routing for router 1 and 3

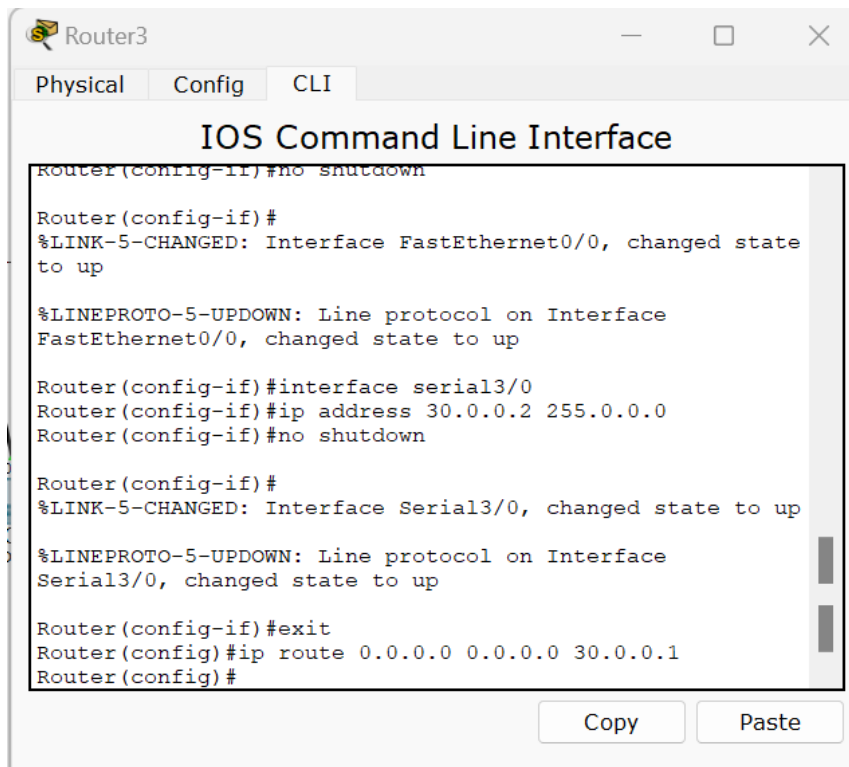


The screenshot shows the CLI window for Router1. The 'Config' tab is selected. The command history shows the following sequence of commands and their outputs:

```
Press RETURN to get started.

Router>enable
Router#configure terminal
Enter configuration commands, one per line. End with
CNTL/Z.
Router(config)#ip route 0.0.0.0 0.0.0.0 20.0.0.2
Router(config)#
```

At the bottom of the window, there are 'Copy' and 'Paste' buttons.



Router3

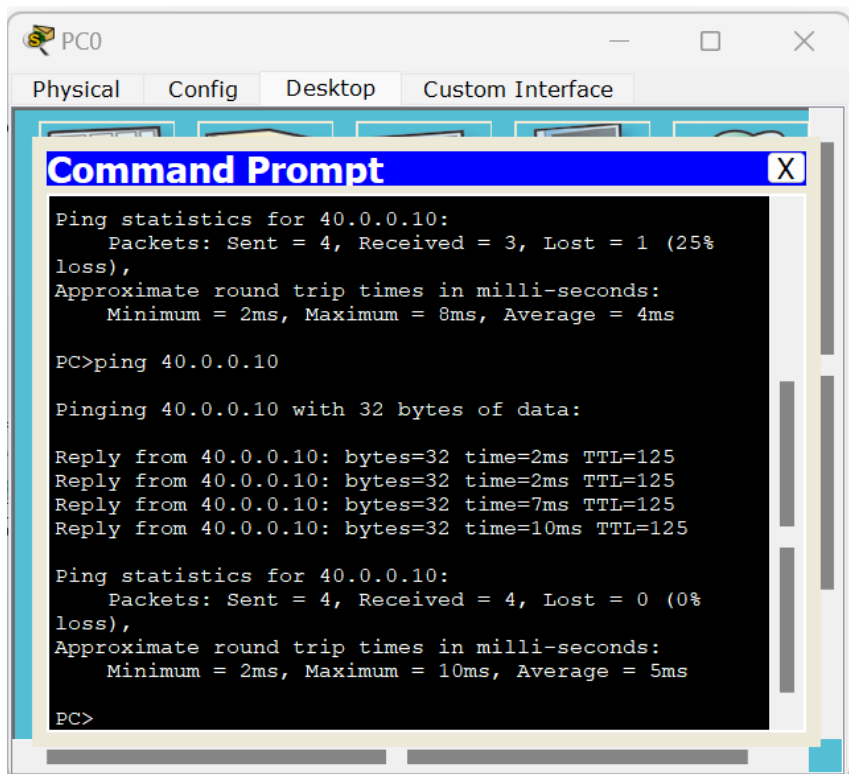
Physical Config CLI

IOS Command Line Interface

```
Router(config-if)#no shutdown
Router(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/0, changed state
to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface
FastEthernet0/0, changed state to up
Router(config-if)#interface serial3/0
Router(config-if)#ip address 30.0.0.2 255.0.0.0
Router(config-if)#no shutdown
Router(config-if)#
%LINK-5-CHANGED: Interface Serial3/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface
Serial3/0, changed state to up
Router(config-if)#exit
Router(config)#ip route 0.0.0.0 0.0.0.0 30.0.0.1
Router(config)#
```

Copy Paste

Ping from pc0 to pc2



PC0

Physical Config Desktop Custom Interface

Command Prompt

```
Ping statistics for 40.0.0.10:
  Packets: Sent = 4, Received = 3, Lost = 1 (25%
loss),
Approximate round trip times in milli-seconds:
  Minimum = 2ms, Maximum = 8ms, Average = 4ms

PC>ping 40.0.0.10

Pinging 40.0.0.10 with 32 bytes of data:

Reply from 40.0.0.10: bytes=32 time=2ms TTL=125
Reply from 40.0.0.10: bytes=32 time=2ms TTL=125
Reply from 40.0.0.10: bytes=32 time=7ms TTL=125
Reply from 40.0.0.10: bytes=32 time=10ms TTL=125

Ping statistics for 40.0.0.10:
  Packets: Sent = 4, Received = 4, Lost = 0 (0%
loss),
Approximate round trip times in milli-seconds:
  Minimum = 2ms, Maximum = 10ms, Average = 5ms

PC>
```

Router1

Physical Config CLI

IOS Command Line Interface

```
Router#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M -
mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA -
OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA
external type 2
       E1 - OSPF external type 1, E2 - OSPF external type
2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2,
ia - IS-IS inter area
       * - candidate default, U - per-user static route,
o - ODR
       P - periodic downloaded static route

Gateway of last resort is 20.0.0.2 to network 0.0.0.0

C    10.0.0.0/8 is directly connected, FastEthernet0/0
C    20.0.0.0/8 is directly connected, Serial2/0
S*   0.0.0.0/0 [1/0] via 20.0.0.2
Router#
```

Copy Paste

Router2

Physical Config CLI

IOS Command Line Interface

```
Router#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M -
mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA -
OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA
external type 2
       E1 - OSPF external type 1, E2 - OSPF external type
2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2,
ia - IS-IS inter area
       * - candidate default, U - per-user static route,
o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

S    10.0.0.0/8 [1/0] via 20.0.0.1
C    20.0.0.0/8 is directly connected, Serial2/0
C    30.0.0.0/8 is directly connected, Serial3/0
S    40.0.0.0/8 [1/0] via 30.0.0.2
Router#
```

Copy Paste

Router3

PhysicalConfigCLI

IOS Command Line Interface

```
Router#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is 30.0.0.1 to network 0.0.0.0

C    30.0.0.0/8 is directly connected, Serial3/0
C    40.0.0.0/8 is directly connected, FastEthernet0/0
S*   0.0.0.0/0 [1/0] via 30.0.0.1
Router#
```

CopyPaste

4) Configure DHCP within a LAN and outside LAN.

Step 1: Create a LAN like this,

Step 2: Router>enable

Router #config

Router(config)

#interface fastethernet0/0 Router(config-if)

#ip address 10.0.0.1 255.0.0.0 Router(config-if)

#no shutdown Router(config-if)

#exit Router(config)

Step 3: click on server-> config, then assign gateway in our example 10.0.0.1

Step 4: Then Click on Fastethernet and assign ip address and subnet mask. I am going to use 10.0.0.2 and subnet mask 255.0.0.0 for our server.

Step 5: Click on DHCP, there you can see default pool,

Step 6: Just give default gate way, here we are using 10.0.0.1.

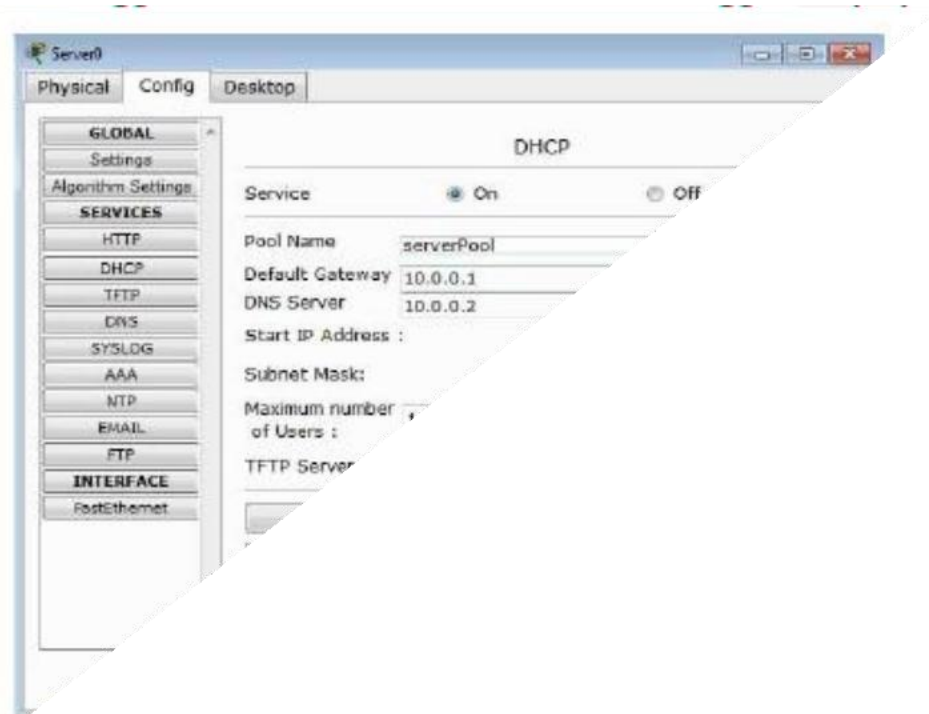
Step 7: DNS server, Just give our server ip address, 10.0.0.2.

Step 8: Then just edit start ip address. I am going to give 10.0.0.10 and subnet mask

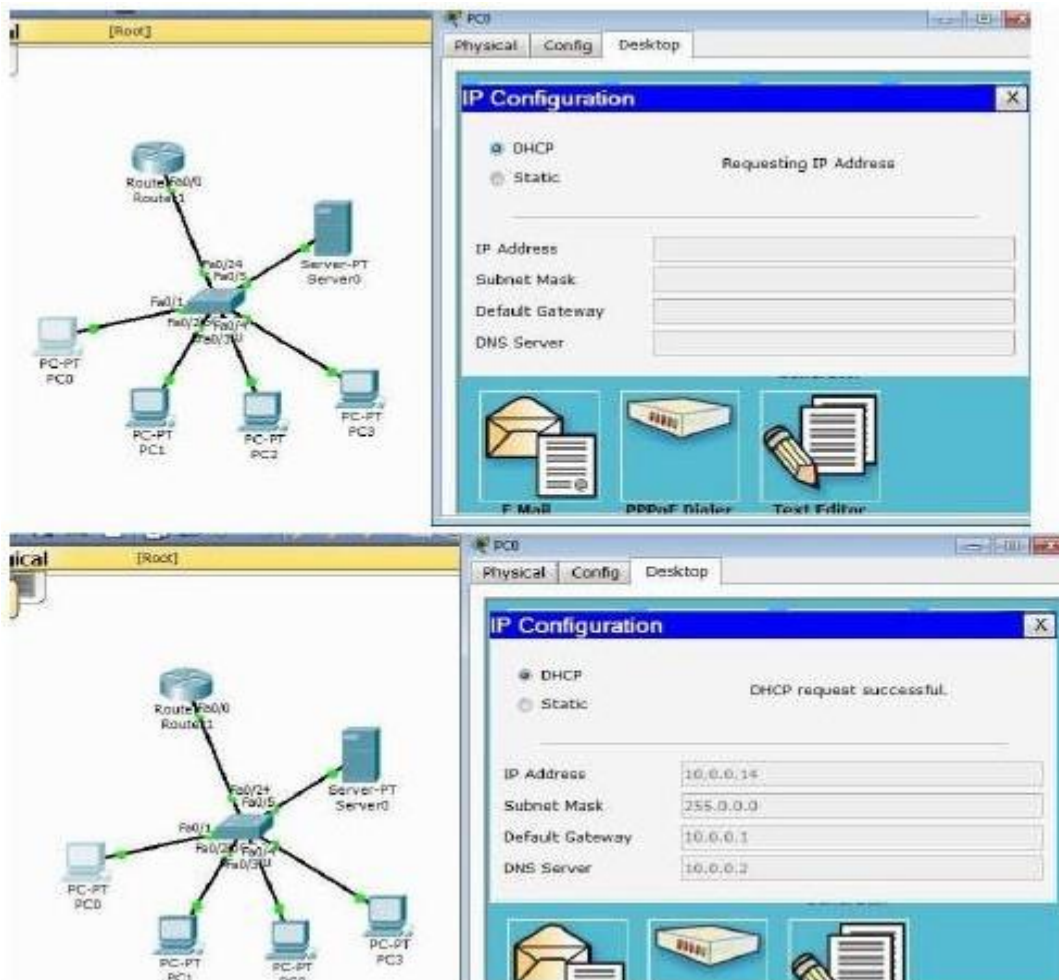
255.0.0.0 **Step 9:** In Maximum Number of Users, Here we are using Class A Network so we can use 1,67,77,216 ip address. just give how many ip address you want in this pool. I am going to give 500

Step 10: Assign TFTP server ip address, just give our server ip address, 10.0.0.2.

Step 11: And click on save. That's it...



Step 12: Now, Click on any of the PC-> then click on Desktop->Ip configuration, and Choose 'DHCP' wait for some time, if your dhcp request failed then try few more times. This is how you should get.



DHCP outside LAN

Here we are going to see, how to configure DHCP for multiple networks. Can we get IP address from DHCP that is present in other network? yes we can. Let's see how to do with help of 'ip helper-address'.

Step 1: Create a topology like this,

Step 2: Configure the router

interface fastethernet0/0 and fastethernet 0/1 with IP address .

Router>enable Router

#config terminal Router(config)

#interface fastethernet0/0 Router(config-if)

#ip address 10.0.0.1 255.0.0.0 Router(config-if)

#no shutdown Router(config-if)

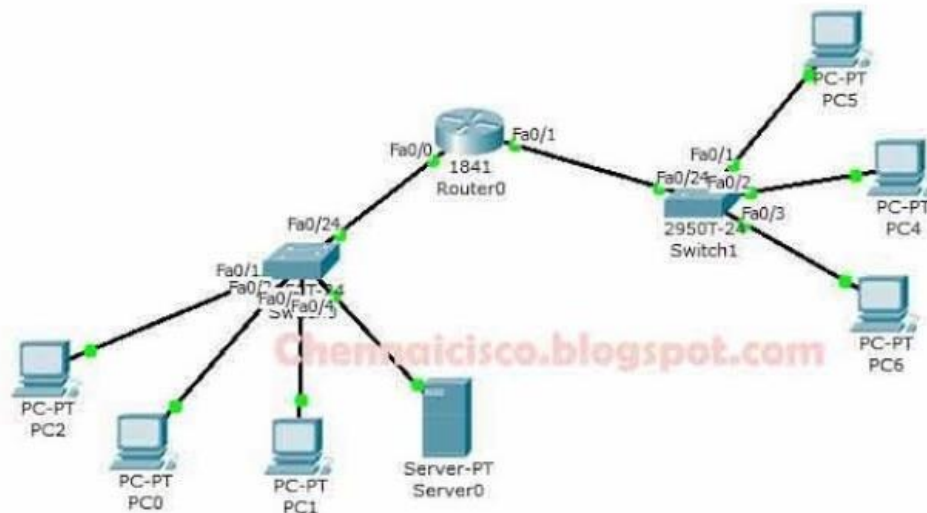
#exit Router(config)

#interface fastethernet0/1 Router(config-if)

#ip address 20.0.0.1 255.0.0.0 Router(config-if)

#no shutdown Router(config-if)

#exit



Step 3: Click on server->config->then just give the gateway ip address .Gateway for this network is 10.0.0.1

Step 4: Then click on fastethernet assign ip address.I am going to give 10.0.0.2 and subnetmask 255.0.0.0.Once we have configured the ip address for the server,DHCP server automatically assign 10 network for default pool.We don't have to create pool for 10 Network again.Just we need to give ip for DNS,Gateway and TFTP then we may configure starting ip address or leave it and Save. **Step 5:** Now,Click on Pc in a LAN with Server and Check whether DHCP working fine in this network.Click on any PC->Desktop->Ip configuration->Choose DHCP, then you will get ip from dhcp server for this PC.

Step 5: Now,we see how to get ip address for PC that is in a network without Server.For that,first we have to add network pool in a dhcp server. So,Click on Server->Config->DHCP.

Step 6: Just edit Pool Name with any other name.I am going to give 20Network.
DefaultGateway->20.0.0.1, DNS Server->10.0.0.2
Start Ip Address->20.0.0.10 Subnet Mask->255.0.0.0 Maximun Number Of Users->100
TFTP Server10.0.0.2 Then,Click on Add and Save.

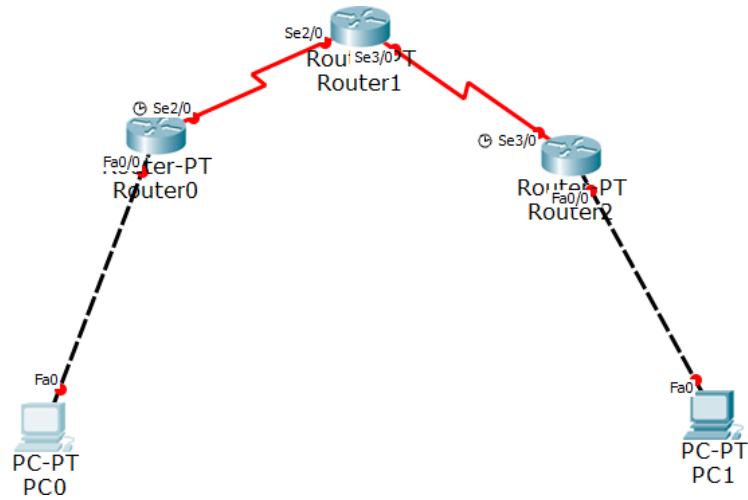
Step 7: Now go to router and give ip helper address under fastethernet0/1,that is server address here our server address is 10.0.0.2.Now we can get ip for this network also In Router,(Global configuration mode)

```
Router(config)#interface fastethernet0/1
Router(config-if)#ip helper-address 10.0.0.2
Router(config-if)#exit
```

Step 8: Now,check whether PC from network without server getting ip from the DHCP server in another Network.Click on any PC->Desktop->Ip configuration->Choose DHCP. Now we have got ip address from dhcp server.

6) Configure RIP routing Protocol in Routers

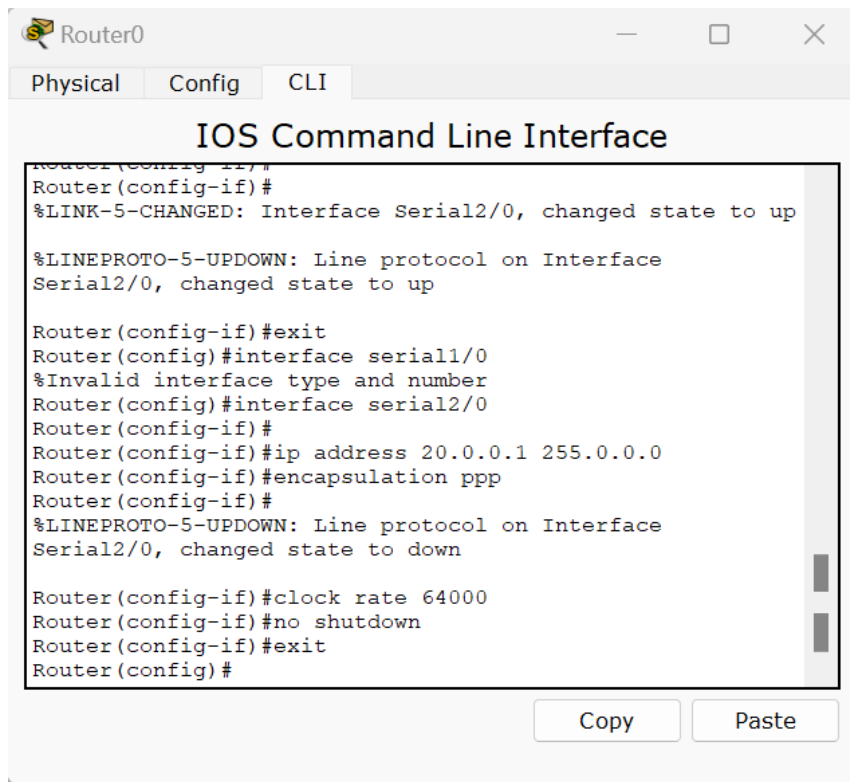
TOPOLOGY



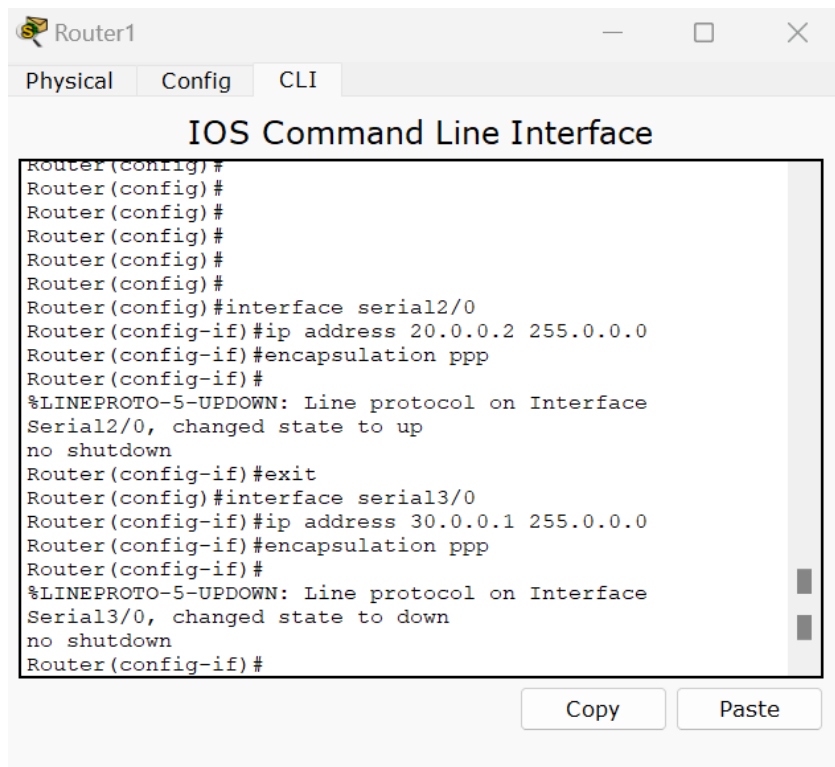
Configure ip address and gateway of PC's

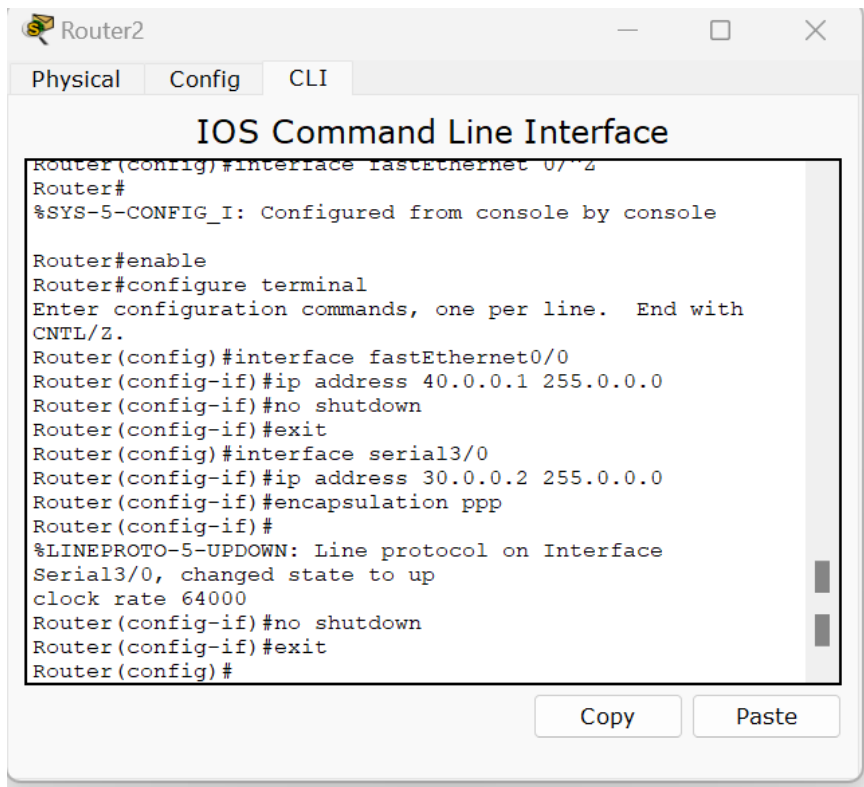
Configure routers as shown in diagram.

Now configure ppp or point to point protocol for all routers.

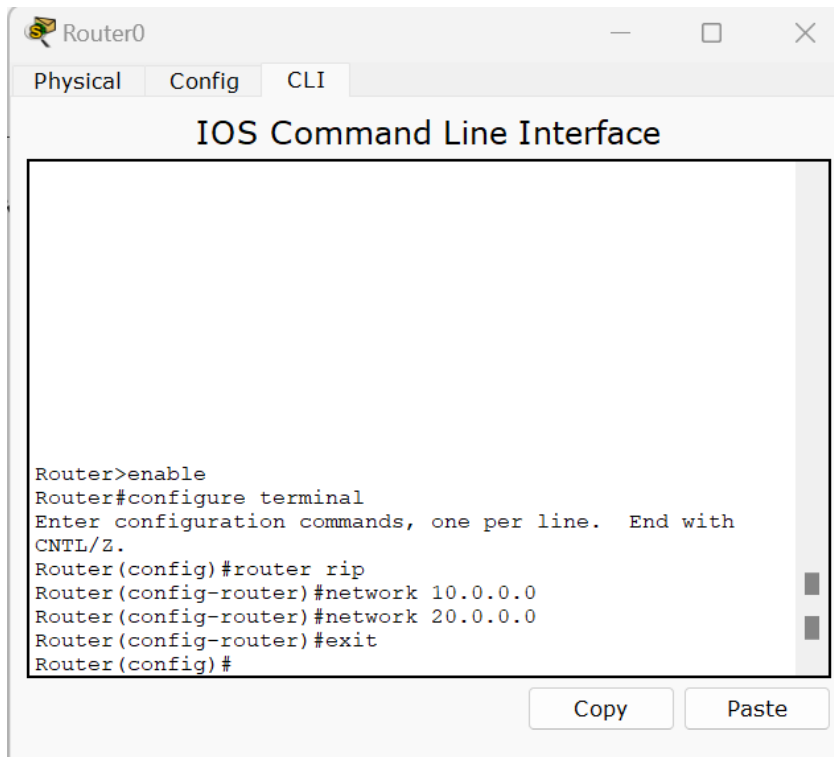


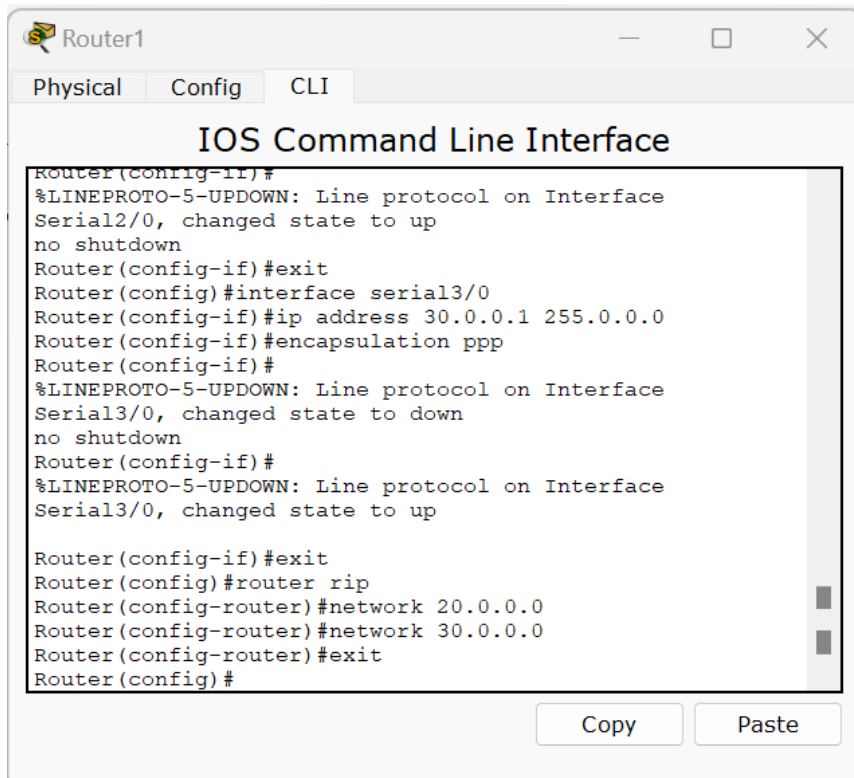
No need to give clk rate in second router





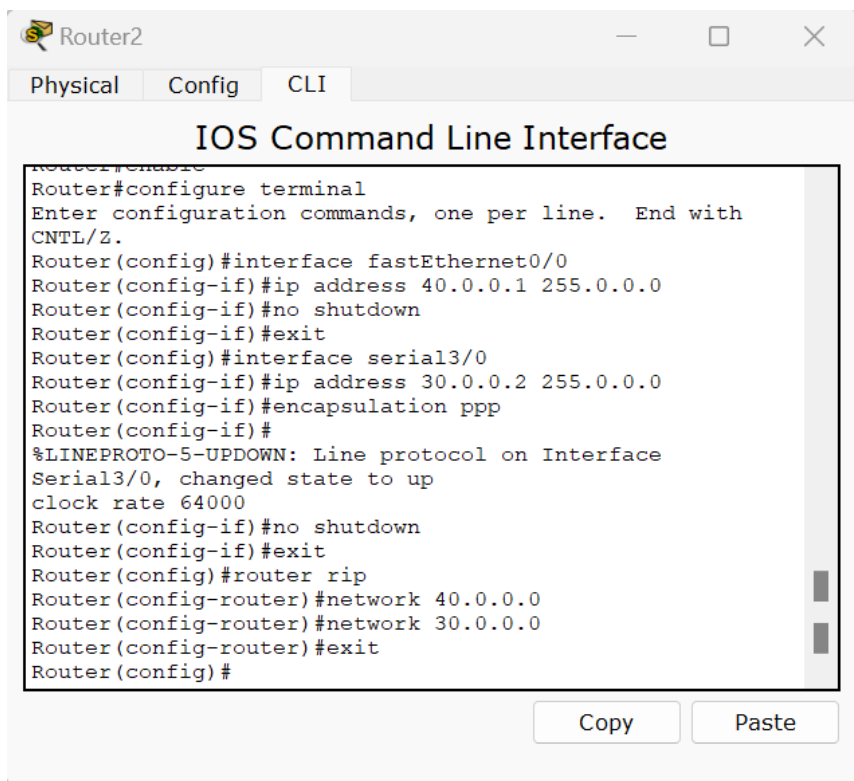
CONFIGURE RIP



A screenshot of a Cisco IOS CLI window titled "Router1". The window has three tabs: "Physical", "Config", and "CLI", with "CLI" being the active tab. The title bar includes a Cisco logo, the text "Router1", and standard window controls (minimize, maximize, close). The main content area is titled "IOS Command Line Interface" and contains a series of configuration commands and their outputs. The commands include enabling Serial2/0, configuring Serial3/0 with IP 30.0.0.1/255.0.0.0 and PPP encapsulation, and enabling RIP. The outputs show the line protocol state changing to up and down. At the bottom right of the text area are "Copy" and "Paste" buttons.

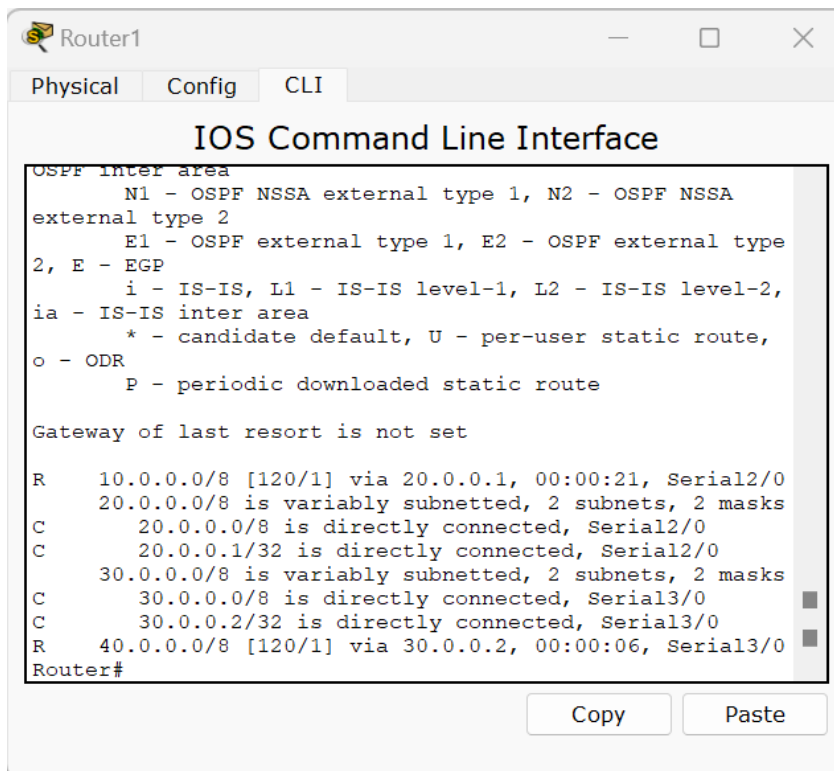
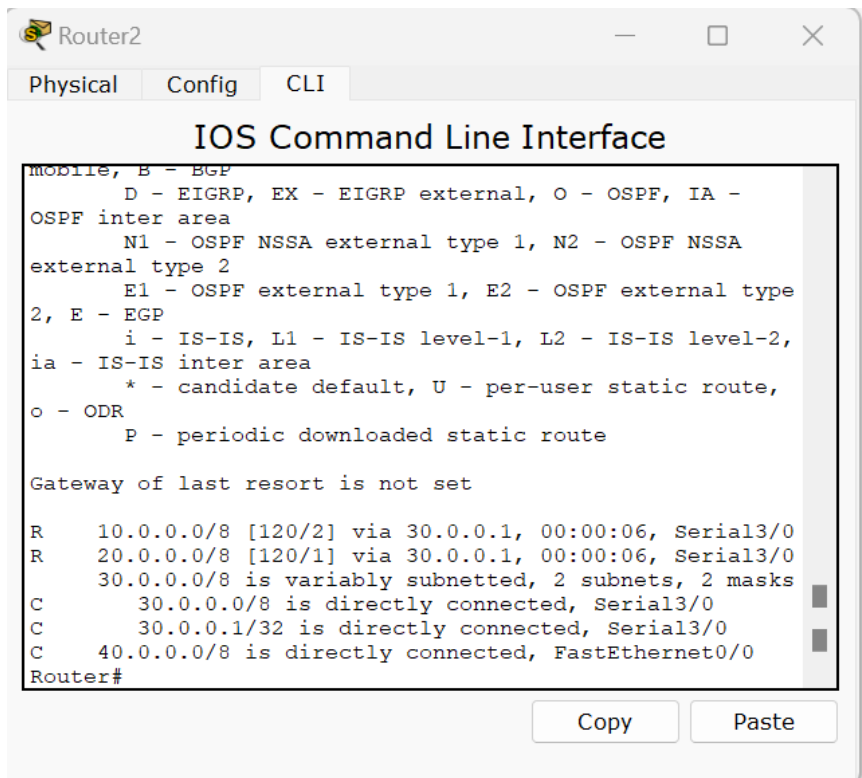
```
Router(config-if)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface
Serial2/0, changed state to up
no shutdown
Router(config-if)#exit
Router(config)#interface serial3/0
Router(config-if)#ip address 30.0.0.1 255.0.0.0
Router(config-if)#encapsulation ppp
Router(config-if)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface
Serial3/0, changed state to down
no shutdown
Router(config-if)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface
Serial3/0, changed state to up

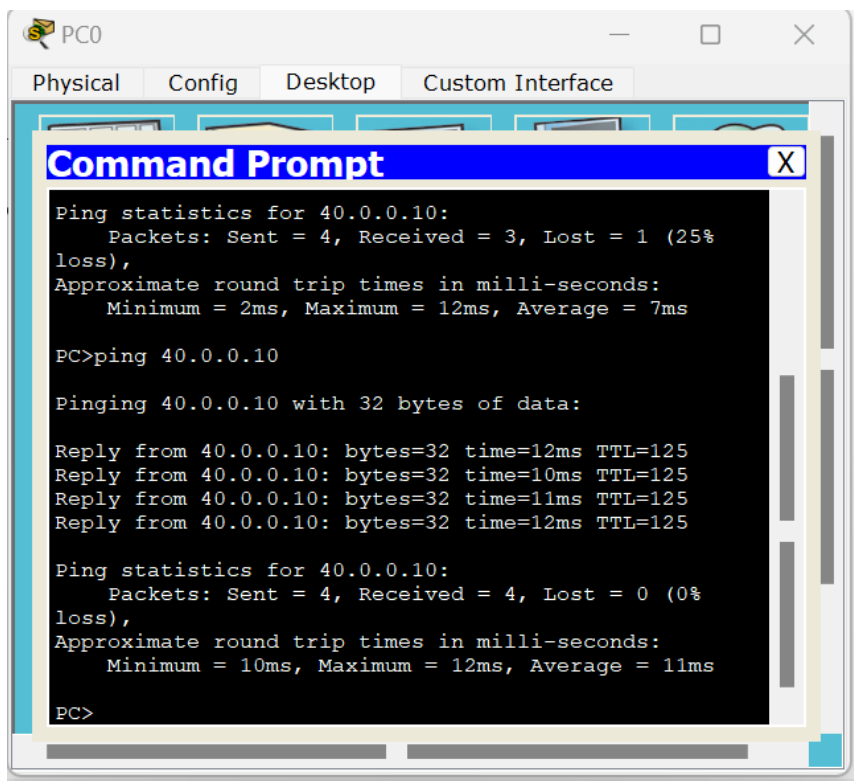
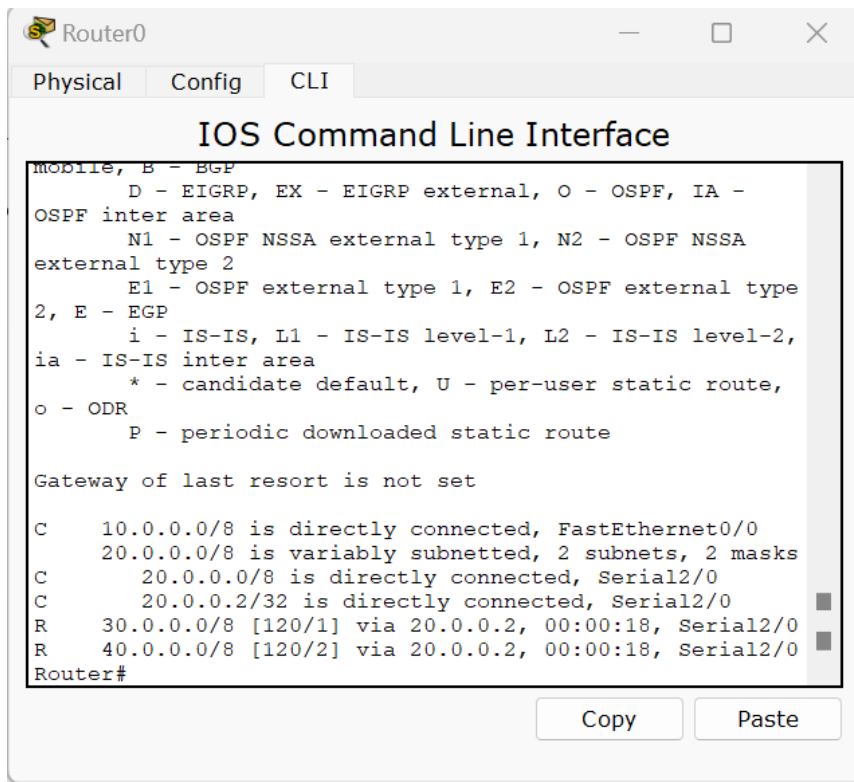
Router(config-if)#exit
Router(config)#router rip
Router(config-router)#network 20.0.0.0
Router(config-router)#network 30.0.0.0
Router(config-router)#exit
Router(config)#
```

A screenshot of a Cisco IOS CLI window titled "Router2". The window has three tabs: "Physical", "Config", and "CLI", with "CLI" being the active tab. The title bar includes a Cisco logo, the text "Router2", and standard window controls (minimize, maximize, close). The main content area is titled "IOS Command Line Interface" and contains a series of configuration commands and their outputs. The commands include enabling Router2, configuring fastEthernet0/0 with IP 40.0.0.1/255.0.0.0, configuring Serial3/0 with IP 30.0.0.2/255.0.0.0 and PPP encapsulation, and enabling RIP. The outputs show the line protocol state changing to up. At the bottom right of the text area are "Copy" and "Paste" buttons.

```
Router#enable
Router#configure terminal
Enter configuration commands, one per line. End with
CNTL/Z.
Router(config)#interface fastEthernet0/0
Router(config-if)#ip address 40.0.0.1 255.0.0.0
Router(config-if)#no shutdown
Router(config-if)#exit
Router(config)#interface serial3/0
Router(config-if)#ip address 30.0.0.2 255.0.0.0
Router(config-if)#encapsulation ppp
Router(config-if)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface
Serial3/0, changed state to up
clock rate 64000
Router(config-if)#no shutdown
Router(config-if)#exit
Router(config)#router rip
Router(config-router)#network 40.0.0.0
Router(config-router)#network 30.0.0.0
Router(config-router)#exit
Router(config)#
```

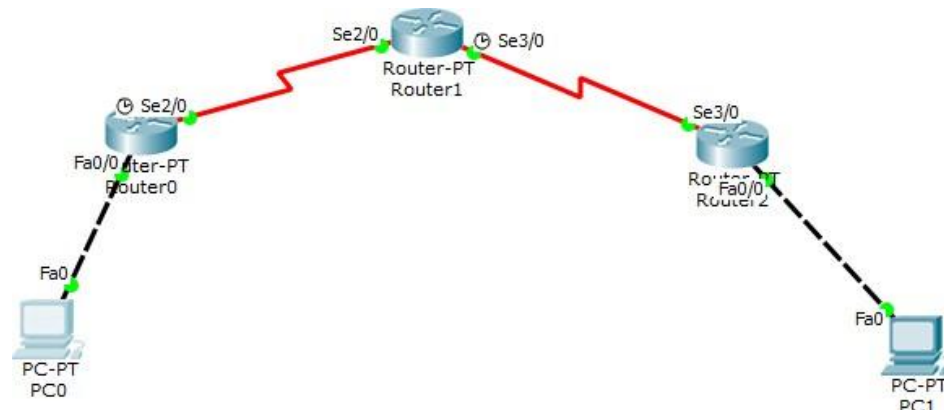
Execute show ip route





6) Configure OSPF routing protocol

Topology.



- 1.config rip.
2. config ospf

```
Router0
Physical Config CLI
IOS Command Line Interface
Router(config-if)#ip address 20.0.0.1 255.0.0.0
Router(config-if)#encapsulation ppp
Router(config-if)#clock 64000
^
% Invalid input detected at '^' marker.
Router(config-if)#clock rate 64000
Router(config-if)#no shutdown
%LINK-5-CHANGED: Interface Serial2/0, changed state to down
Router(config-if)#exit
Router(config)#
%LINK-5-CHANGED: Interface Serial2/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to up
Router(config)#router ospf1
^
% Invalid input detected at '^' marker.
Router(config)#router ospf 1
Router(config-router)#router id 1.1.1.1
^
% Invalid input detected at '^' marker.
Router(config-router)#router-id 1.1.1.1
Router(config-router)#network 10.0.0.0 0.255.255.255 area 3
Router(config-router)#network 20.0.0.0 0.255.255.255 area 1
Router(config-router)#exit
Router(config)#
```

IOS Command Line Interface

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

Router(config-if)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to up

Router(config-if)#interface serial 3/0
Router(config-if)#ip address 30.0.0.1 255.0.0.0
Router(config-if)#encapsulation ppp
Router(config-if)#clock rate 64000
Router(config-if)#no shutdown

%LINK-5-CHANGED: Interface Serial3/0, changed state to down
Router(config-if)#exit
Router(config)#
%LINK-5-CHANGED: Interface Serial3/0, changed state to up

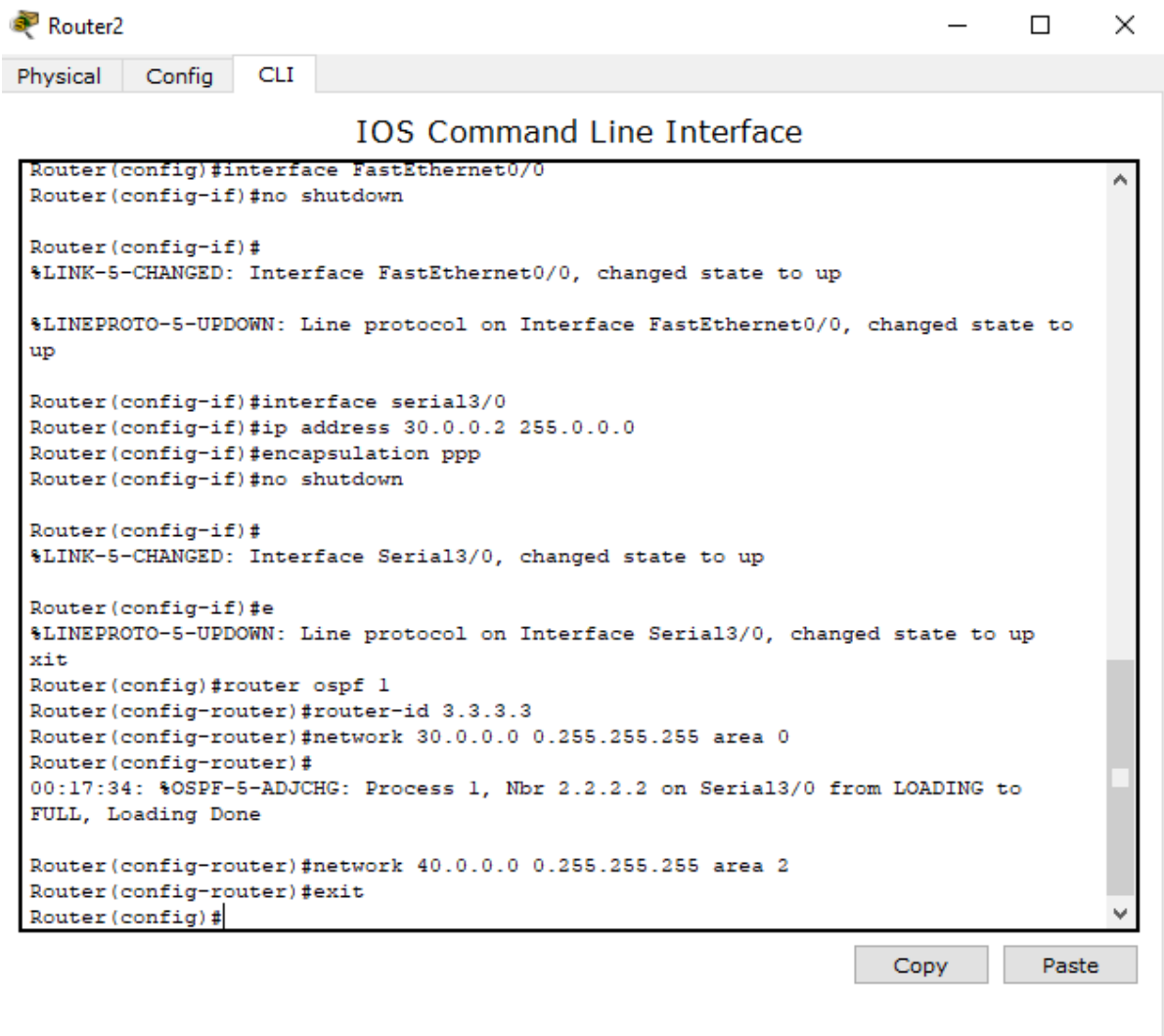
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial3/0, changed state to up

Router(config)#router ospf 1
Router(config-router)#router-id 2.2.2.2
Router(config-router)#network 20.0.0.0 0.255.255.255 area 1
Router(config-router)#
00:15:10: %OSPF-5-ADJCHG: Process 1, Nbr 1.1.1.1 on Serial2/0 from LOADING to
FULL, Loading Done

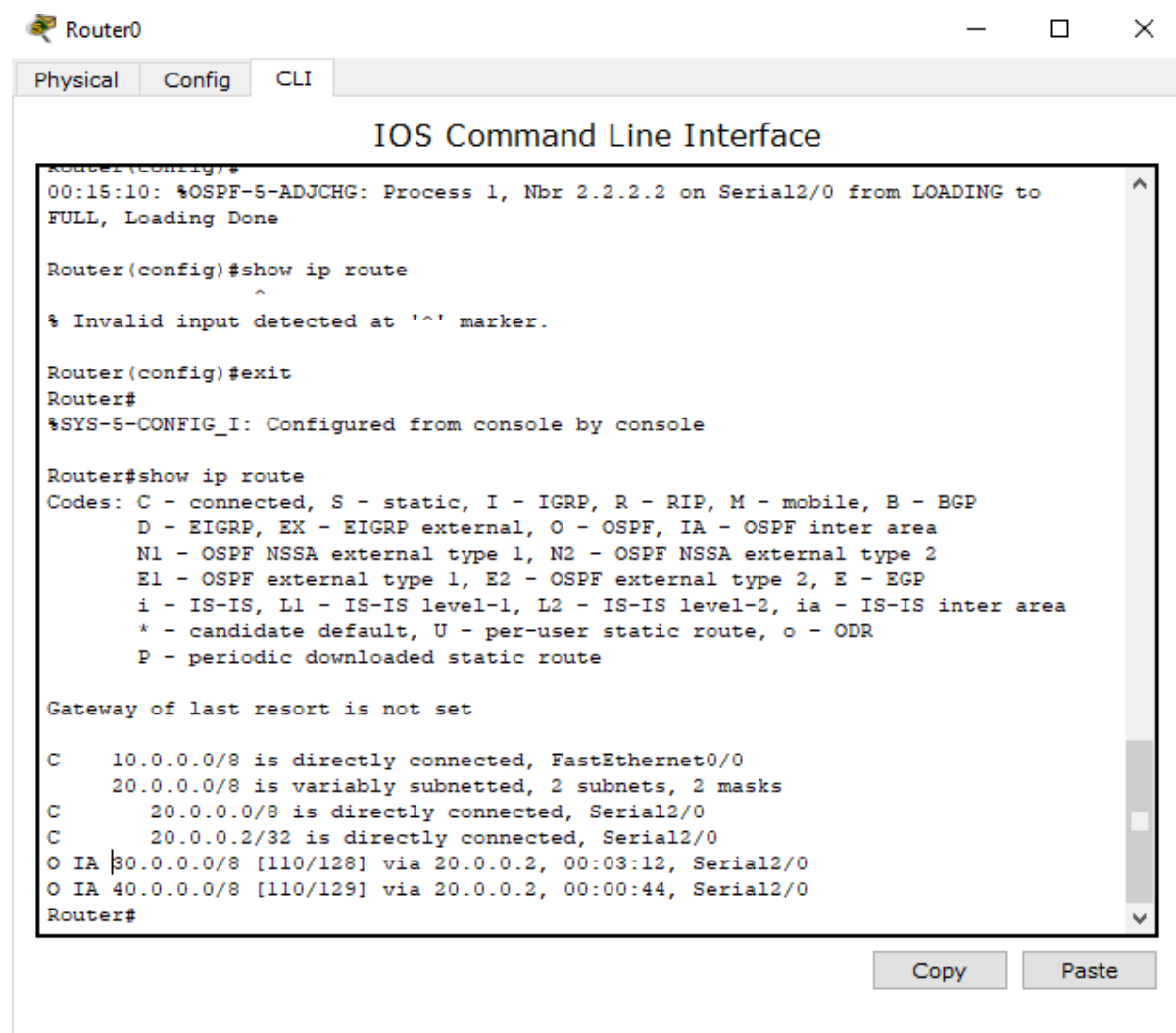
Router(config-router)#network 30.0.0.0 0.255.255.255 area 0
Router(config-router)#exit
Router(config)#
```

Copy

Paste



Show ip route



The screenshot shows a window titled "Router0" with three tabs: "Physical", "Config", and "CLI". The "CLI" tab is active, displaying the "IOS Command Line Interface". The terminal output shows the following sequence of commands and responses:

```
Router(config)#
00:15:10: %OSPF-5-ADJCHG: Process 1, Nbr 2.2.2.2 on Serial2/0 from LOADING to FULL, Loading Done

Router(config)#show ip route
^
% Invalid input detected at '^' marker.

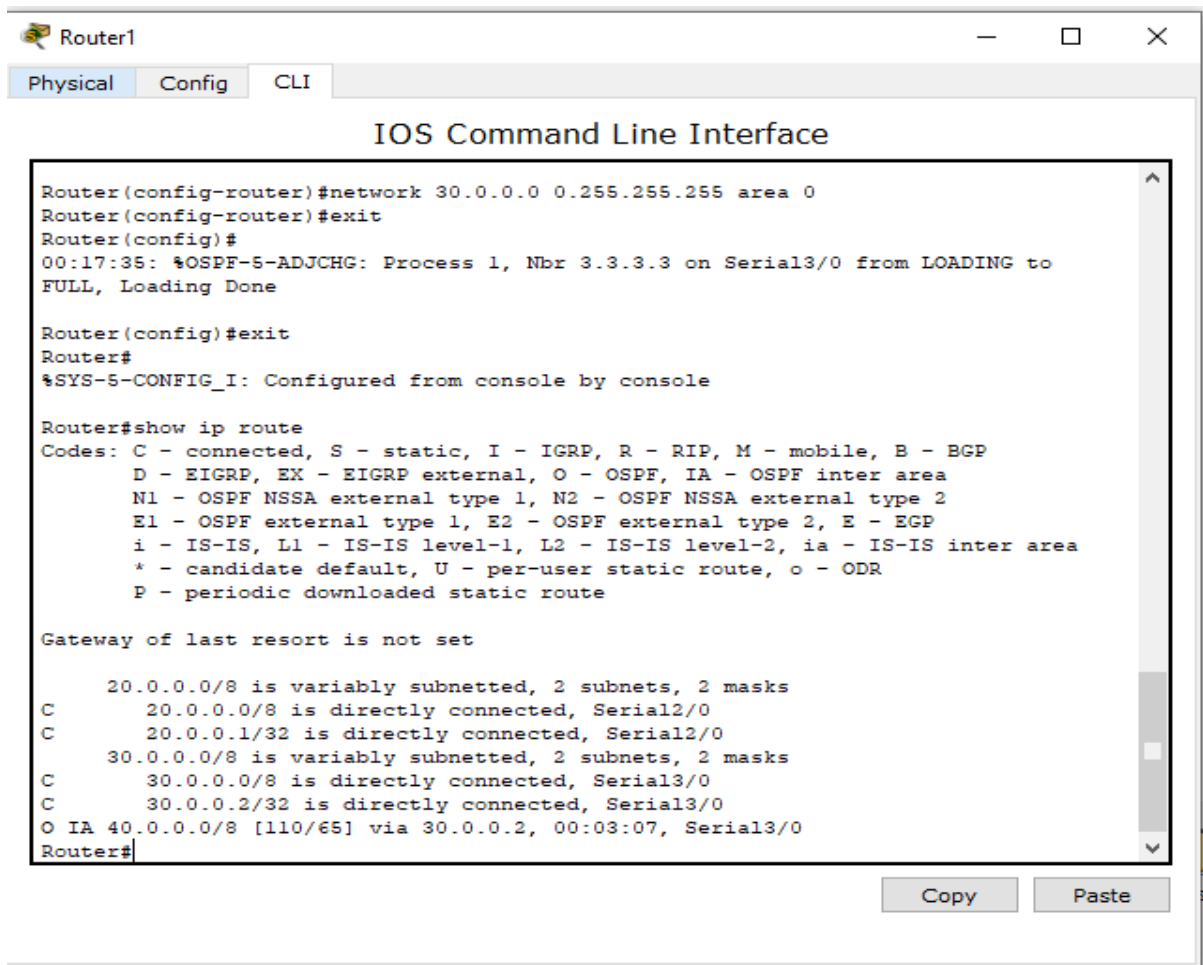
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console

Router#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

C      10.0.0.0/8 is directly connected, FastEthernet0/0
      20.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C      20.0.0.0/8 is directly connected, Serial2/0
C      20.0.0.2/32 is directly connected, Serial2/0
O IA 30.0.0.0/8 [110/128] via 20.0.0.2, 00:03:12, Serial2/0
O IA 40.0.0.0/8 [110/129] via 20.0.0.2, 00:00:44, Serial2/0
Router#
```

At the bottom right of the CLI window, there are two buttons: "Copy" and "Paste".



Router2

Physical Config CLI

IOS Command Line Interface

```
Router(config-router)#router-id 3.3.3.3
Router(config-router)#network 30.0.0.0 0.255.255.255 area 0
Router(config-router)#
00:17:34: %OSPF-5-ADJCHG: Process 1, Nbr 2.2.2.2 on Serial3/0 from LOADING to FULL, Loading Done

Router(config-router)#network 40.0.0.0 0.255.255.255 area 2
Router(config-router)#exit
Router(config)#
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console

Router#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

O IA 20.0.0.0/8 [110/128] via 30.0.0.1, 00:04:33, Serial3/0
    30.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C     30.0.0.0/8 is directly connected, Serial3/0
C     30.0.0.1/32 is directly connected, Serial3/0
C     40.0.0.0/8 is directly connected, FastEthernet0/0
Router#
```

Copy Paste

Loopback

Router0

Physical Config CLI

IOS Command Line Interface

```
Router#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

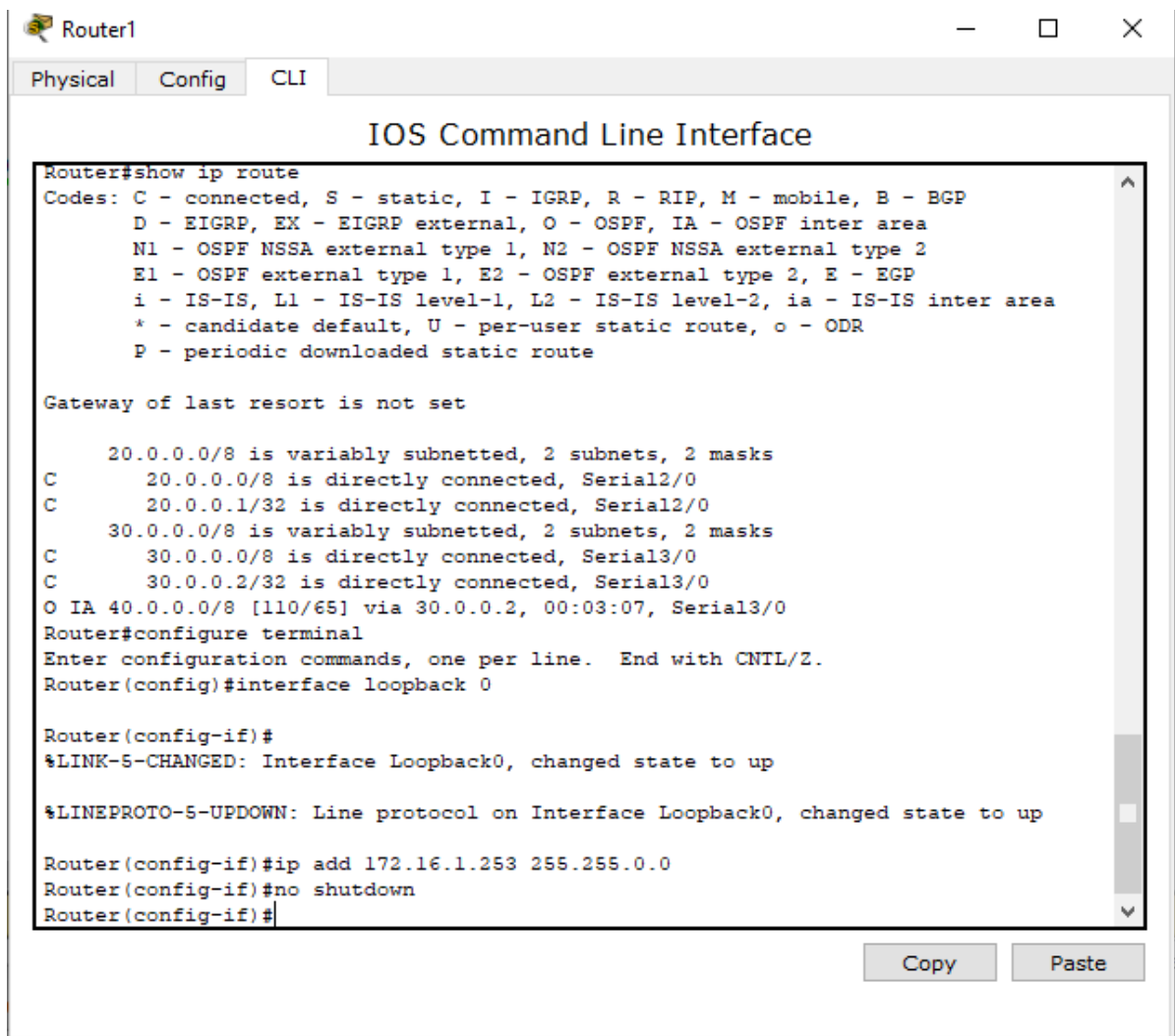
C     10.0.0.0/8 is directly connected, FastEthernet0/0
    20.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C     20.0.0.0/8 is directly connected, Serial2/0
C     20.0.0.2/32 is directly connected, Serial2/0
O IA 30.0.0.0/8 [110/128] via 20.0.0.2, 00:03:12, Serial2/0
O IA 40.0.0.0/8 [110/129] via 20.0.0.2, 00:00:44, Serial2/0
Router#
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface loopback 0

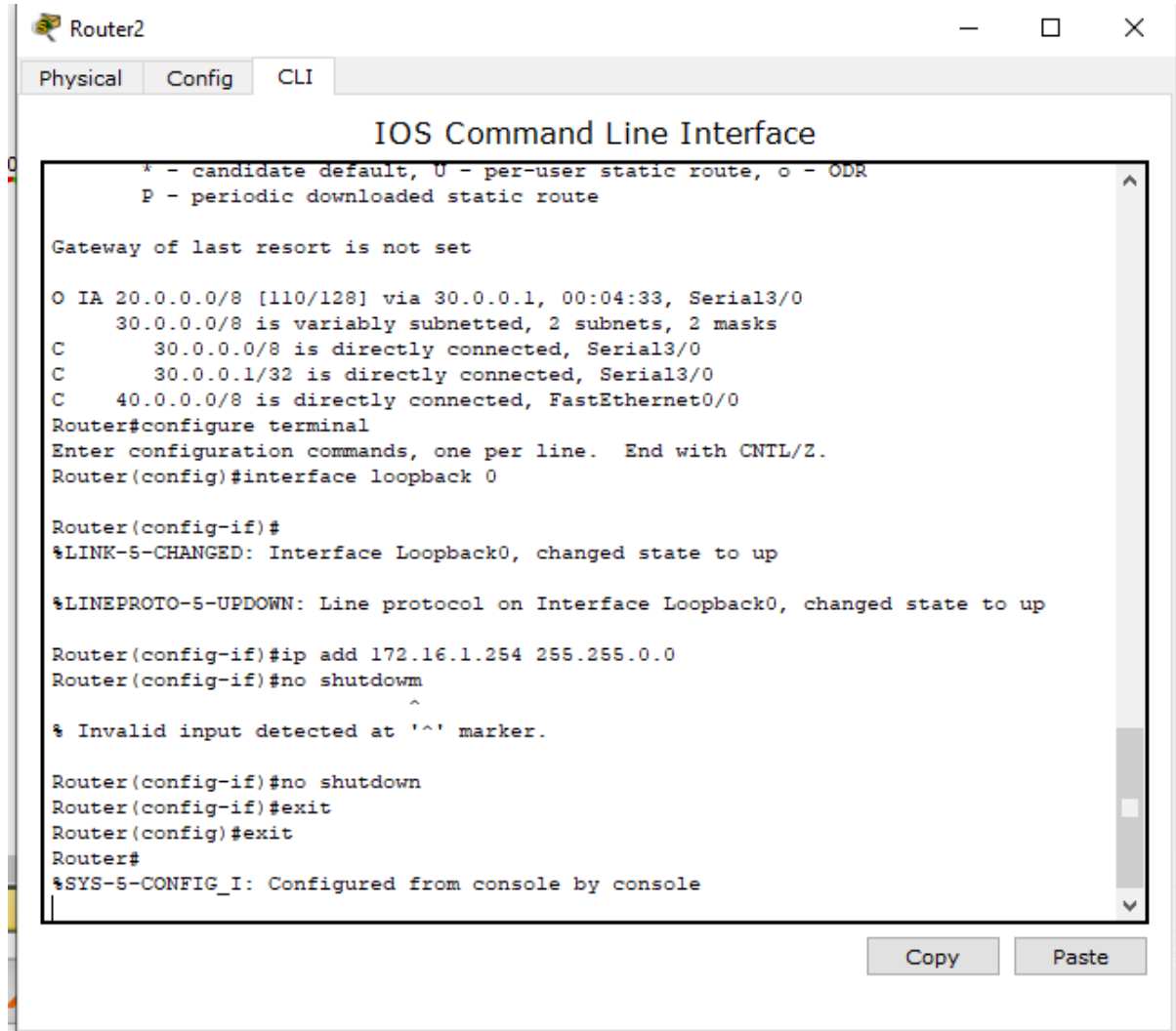
Router(config-if)#
%LINK-5-CHANGED: Interface Loopback0, changed state to up

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

Router(config-if)#ip add 172.16.1.252 255.255.0.0
Router(config-if)#no shutdown
Router(config-if)#
```

Copy Paste





Router2

Physical Config CLI

IOS Command Line Interface

```
* - candidate default, U - per-user static route, o - ODR
P - periodic downloaded static route

Gateway of last resort is not set

O IA 20.0.0.0/8 [110/128] via 30.0.0.1, 00:04:33, Serial3/0
  30.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C    30.0.0.0/8 is directly connected, Serial3/0
C    30.0.0.1/32 is directly connected, Serial3/0
C    40.0.0.0/8 is directly connected, FastEthernet0/0
Router#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#interface loopback 0

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

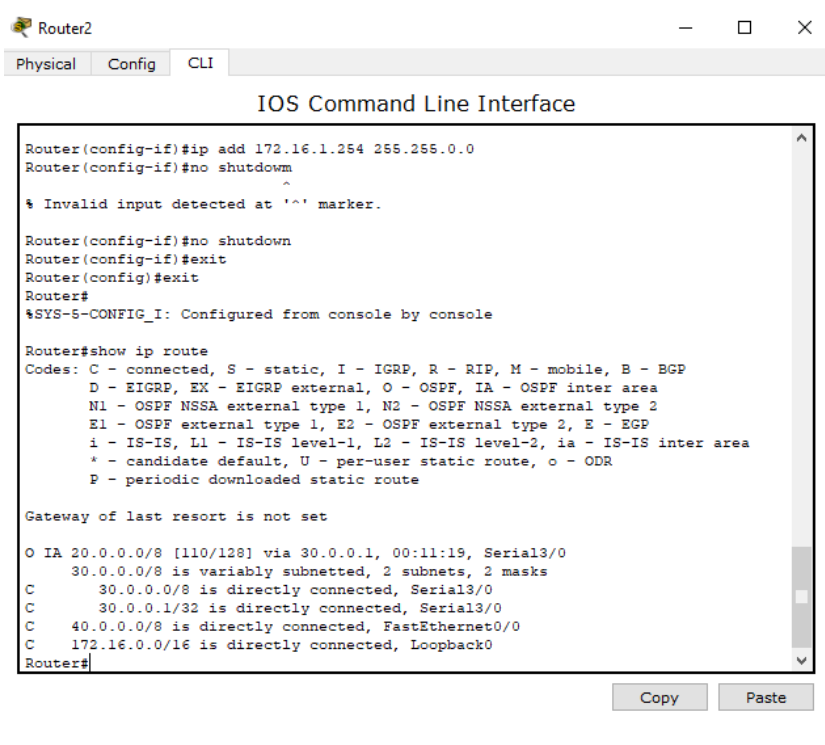
%LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, changed state to up

Router(config-if)#ip add 172.16.1.254 255.255.0.0
Router(config-if)#no shutdown
      ^
% Invalid input detected at '^' marker.

Router(config-if)#no shutdown
Router(config-if)#exit
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console
```

Copy Paste

show ip route for R2



Router2

Physical Config CLI

IOS Command Line Interface

```
Router(config-if)#ip add 172.16.1.254 255.255.0.0
Router(config-if)#no shutdown
      ^
% Invalid input detected at '^' marker.

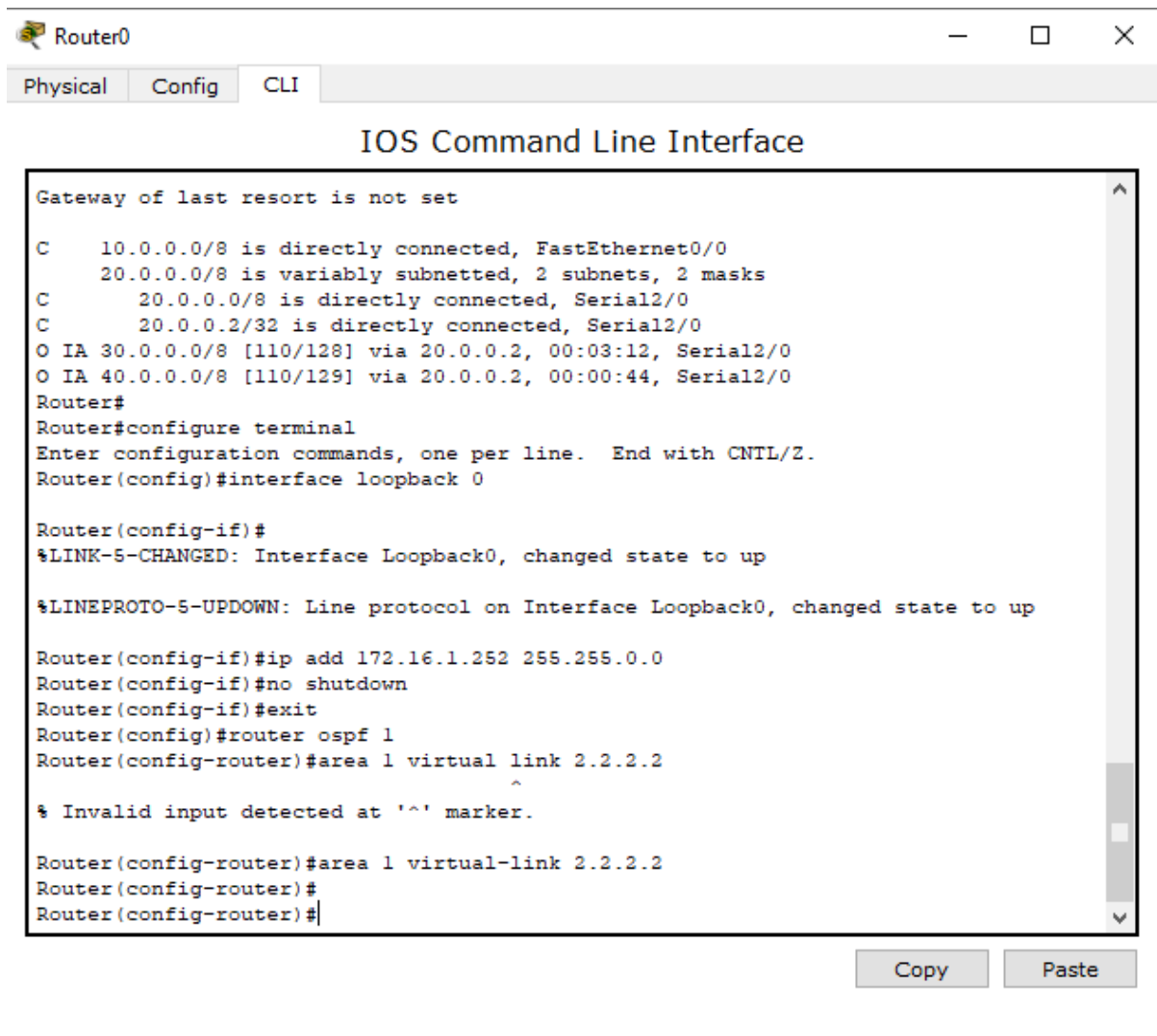
Router(config-if)#no shutdown
Router(config-if)#exit
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console

Router#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
        D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
        N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
        E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
        i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
        * - candidate default, U - per-user static route, o - ODR
        P - periodic downloaded static route

Gateway of last resort is not set

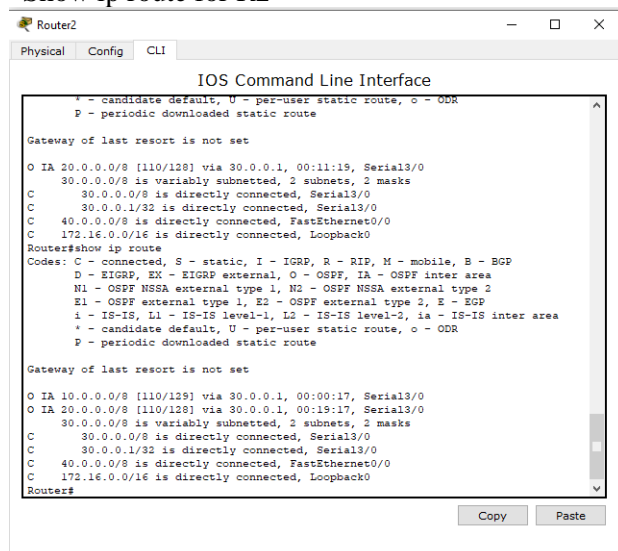
O IA 20.0.0.0/8 [110/128] via 30.0.0.1, 00:11:19, Serial3/0
  30.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C    30.0.0.0/8 is directly connected, Serial3/0
C    30.0.0.1/32 is directly connected, Serial3/0
C    40.0.0.0/8 is directly connected, FastEthernet0/0
C    172.16.0.0/16 is directly connected, Loopback0
Router#
```

Copy Paste

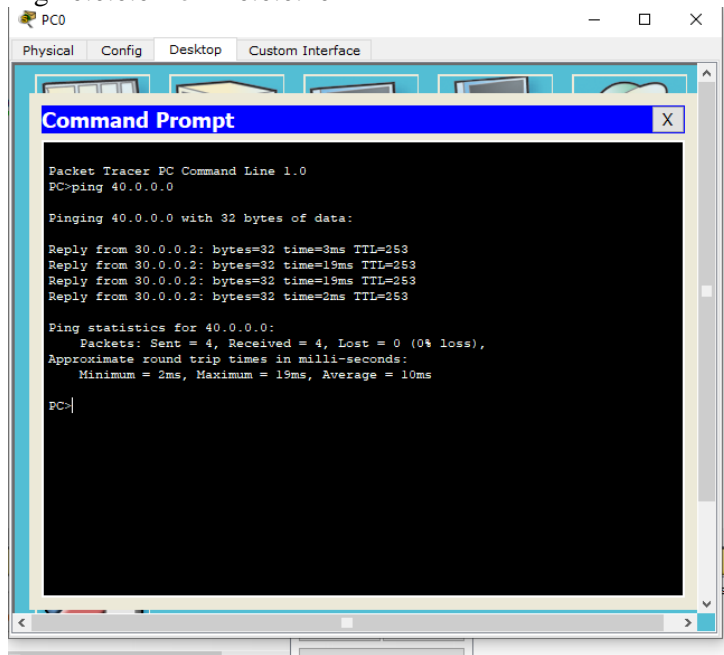


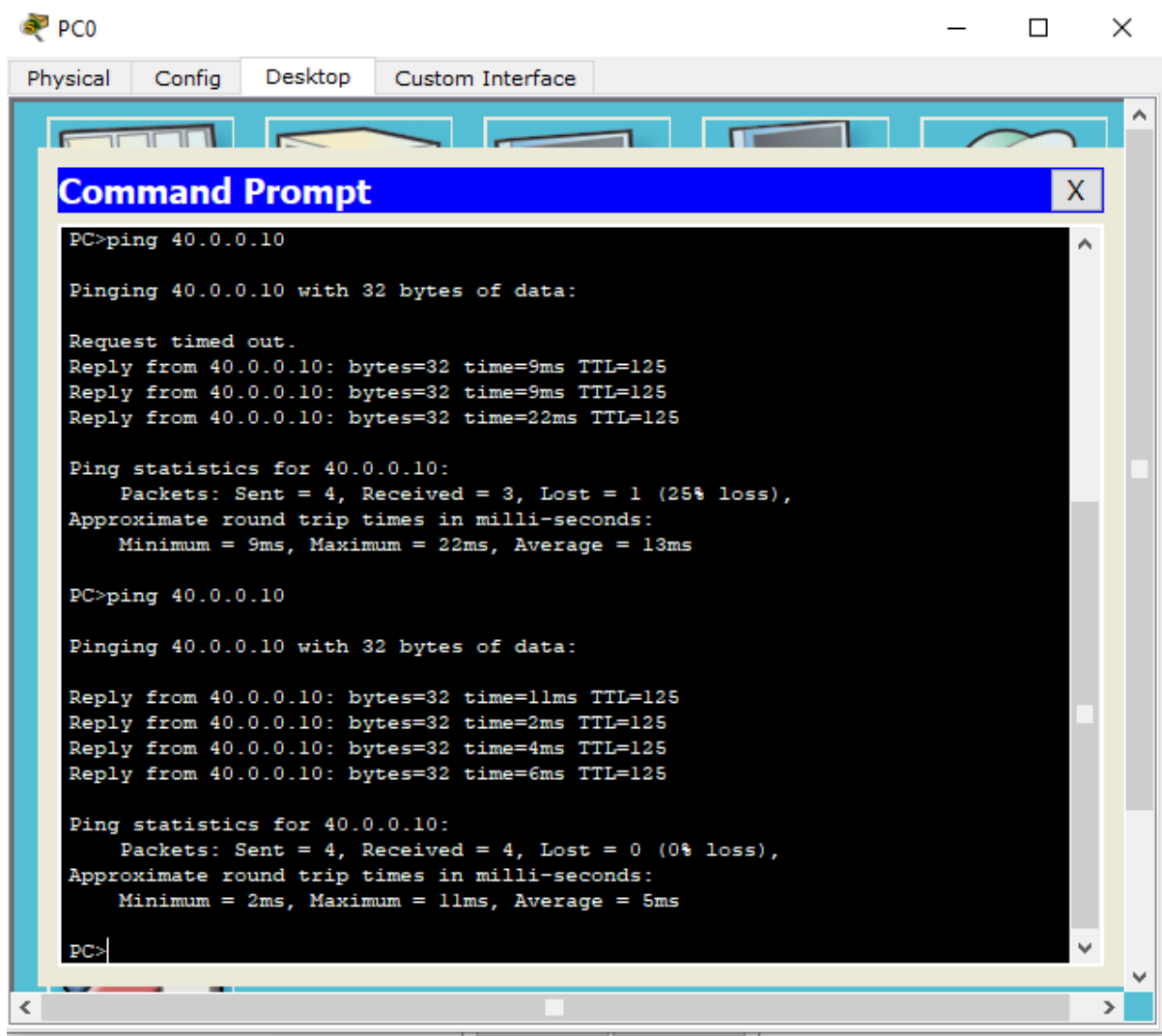
Similarly do for R1 you may face some disturbance while typing

Show ip route for R2



Ping 40.0.0.0 from 10.0.0.10





7) Demonstrate the TTL/ Life of a Packet

Create a topology as shown below with two PCs and three routers.

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

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.

The screenshot shows the Packet Tracer interface with a topology of two PCs and three routers. A PDU capture window is open for PC0, displaying the following details:

PDU Information at Device: PC0

OSI Model | **Outbound PDU Details**

PDU Formats

Ethernet II

0	4	8	14	15
PREAMBLE:	DEST	SRC		
101010...	MAC:	MAC:		
TYP	DATA (VARIABLE LENGTH)	FCS		

IP

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

ICMP

0	8	16	31
TYPE:	CODE:	CHECKSUM	
ID: 0x8	SEQ NUMBER: 7		

The Event List on the right shows a single event at 0.000s: PC0 to PC0, ICMP, with a TTL of 255.

The screenshot shows the Packet Tracer interface with the same topology. A PDU capture window is open for Router1, displaying the following details:

PDU Information at Device: Router1

OSI Model | **Inbound PDU Details** | **Outbound PDU Details**

PDU Formats

Ethernet II

0	4	8	14	15
PREAMBLE:	DEST	SRC		
101010...	MAC:	MAC:		
TYP	DATA (VARIABLE LENGTH)	FCS		

IP

0	4	8	16	19	31
IHL	DSCP:	TL:	28		
ID: 0x6	0x1	0x0			
TTL: 255	PRO: 0x1	CHKSUM			
SRC IP: 10.0.0.1					
DST IP: 40.0.0.1					
OPT: 0x0	0x0				
DATA (VARIABLE LENGTH)					

ICMP

0	8	16	31
TYPE:	CODE:	CHECKSUM	
ID: 0x7	SEQ NUMBER: 6		

The Event List on the right shows multiple events, including one at 0.001s: PC0 to Router1, ICMP, with a TTL of 254.

Cisco Packet Tracer Student - C:\Users\nanvi\Cisco Packet Tracer 6.2sv\saves\default1.pkt

File Edit Options View Tools Extensions Help

Logical [Root] New Cluster Move Object Set Tiled Background Viewport

Simulation Panel

Event List

Time(sec)	Last De	At Dev	Type	Info
0.000	--	PC0	ICMP	
0.000	--	PC0	ICMP	
0.001	PC0	Rout...	ICMP	
0.001	--	PC0	ICMP	

Set Simulation ☒ Constant Delay Captured to: 0.001 s

Back Auto Capture / Play Capture / Forward

Event List Filters - Visible Events

Filter: ARP, BGP, CDP, DHCP, DHCPv6, DNS, DTP, EIGRP, EIGRPv6, FTP, H.323, HSRPv6, HTTP, HTTPS, ICMP, ICMPv6, IPsec, ISAKMP, LACP, NBP, NETFLOW, OSPF, OSPFv6, PAgP, POP3, RADIUS, RIP, RIPng, RTP, SCCP, SMTP, SNMP, SSH, SYSLOG, TACACS, TCB, TFTP, Telnet, UDP, VTP

Edit Filters Show All/None

Event List Simulation

Sourc	Destinac	Type	Colo	Time(Period	Num	Edit	Delete
PC0	PC1	IC...		0.000	N	0	(ed...	(delete)
PC0	PC1	IC...		0.000	N	1	(ed...	(delete)

Time: 00:01:25.695 Power Cycle Devices PLAY CONTROLS: Back Auto

Connections Automatically Choose Connection Type

Toggle PDU List Window

PDU Information at Device: Router1

OSI Model Inbound PDU Details Outbound PDU Details

PDU Formats

HDLC

FL	AD	CONTR	DATA	FCS	FL
G:	R:	OL:	(VARIABLE)	0x0	G:

IP

4	IHL	DSCP	16	TL	28
ID:	0x6	0x	0x0		
TTL:	254	PRO:	0x1	CHKSUM	
SRC IP:	10.0.0.1				
DST IP:	40.0.0.1				
OPT:	0x0		0x0		
DATA (VARIABLE LENGTH)					

ICMP

0	8	16	31	bits
TYPE:	CODE:	CHECKSUM		
ID:	0x7	SEQ NUMBER:	6	

File Edit Options View Tools Extensions Help

Logical [Root] New Cluster Move Object Set Tiled Background Viewport

Simulation Panel

Event List

Time(sec)	Last De	At Dev	Type	Info
0.000	--	PC0	ICMP	
0.001	PC0	Rout...	ICMP	
0.001	--	PC0	ICMP	
0.002	PC0	Rout...	ICMP	
0.002	Router1	Rout...	ICMP	

Set Simulation ☒ Constant Delay Captured to: 0.002 s

Back Auto Capture / Play Capture / Forward

Event List Filters - Visible Events

Filter: ARP, BGP, CDP, DHCP, DHCPv6, DNS, DTP, EIGRP, EIGRPv6, FTP, H.323, HSRPv6, HTTP, HTTPS, ICMP, ICMPv6, IPsec, ISAKMP, LACP, NBP, NETFLOW, OSPF, OSPFv6, PAgP, POP3, RADIUS, RIP, RIPng, RTP, SCCP, SMTP, SNMP, SSH, SYSLOG, TACACS, TCB, TFTP, Telnet, UDP, VTP

Edit Filters Show All/None

Event List Simulation

Fire	Last Statu	Sourc	Destinac	Type	Colo	Time(Period	Num	Edit	Delete
In Progr...	PC0	PC1	IC...			0.000	N	0	(ed...	(delete)
In Progr...	PC0	PC1	IC...			0.000	N	1	(ed...	(delete)

Time: 00:01:25.696 Power Cycle Devices PLAY CONTROLS: Back Auto

Connections Automatically Choose Connection Type

Toggle PDU List Window

PDU Information at Device: Router2

OSI Model Inbound PDU Details Outbound PDU Details

PDU Formats

HDLC

FL	AD	CONTR	DATA	FCS	FL
G:	R:	OL:	(VARIABLE)	0x0	G:

IP

4	IHL	DSCP	16	TL	28
ID:	0x6	0x	0x0		
TTL:	254	PRO:	0x1	CHKSUM	
SRC IP:	10.0.0.1				
DST IP:	40.0.0.1				
OPT:	0x0		0x0		
DATA (VARIABLE LENGTH)					

ICMP

0	8	16	31	bits
TYPE:	CODE:	CHECKSUM		
ID:	0x7	SEQ NUMBER:	6	

File Edit Options View Tools Extensions Help

Logical [Root] New Cluster Move Object Set Tiled Background Viewport

Time: 00:01:25.696 Power Cycle Devices PLAY CONTROLS: Back Auto

Scenario 0 New Delete Toggle PDU List Window

PDU Information at Device: Router2

OSI Model Inbound PDU Details Outbound PDU Details

PDU Formats

HDLCL

FL	AD	CONTR	DATA:	FCS:	FL
G:	R:	OL:	(VARIABLE)	0x0	G:

IP

4	8	16	19	31	31 bits
ID: 0x6	DSCP: 0x	TL: 28			
TTL: 253	PRO: 0x1	CHKSUM			
SRC IP: 10.0.0.1					
DST IP: 40.0.0.1					
OPT: 0x0					0x0
DATA (VARIABLE LENGTH)					

ICMP

0	8	16	31	31 bits
TYPE:	CODE:	CHECKSUM		
ID: 0x7	SEQ NUMBER: 6			

Simulation Panel

Event List

Time(sec)	Last De	At Dev	Type	Info
0.000	--	PC0	ICMP	
0.001	PC0	Router1	ICMP	
0.001	--	PC0	ICMP	
0.002	PC0	Router1	ICMP	
0.002	Router1	Router1	ICMP	

Simulation

Fire Last Statu Sourc Destinatio Type Colo Time(Period Num Edit Delete

In Progr... PC0 PC1 IC... 0.000 N 0 (ed... (delete)

In Progr... PC0 PC1 IC... 0.000 N 1 (ed... (delete)

Cisco Packet Tracer Student - C:\Users\nanvi\Cisco Packet Tracer 6.2sv\saves\default1.pkt

File Edit Options View Tools Extensions Help

Logical [Root] New Cluster Move Object Set Tiled Background Viewport

Time: 00:01:25.697 Power Cycle Devices PLAY CONTROLS: Back Auto

Scenario 0 New Delete Toggle PDU List Window

PDU Information at Device: Router3

OSI Model Inbound PDU Details Outbound PDU Details

PDU Formats

HDLCL

FL	AD	CONTR	DATA:	FCS:	FL
G:	R:	OL:	(VARIABLE)	0x0	G:

IP

4	8	16	19	31	31 bits
ID: 0x6	DSCP: 0x	TL: 28			
TTL: 253	PRO: 0x1	CHKSUM			
SRC IP: 10.0.0.1					
DST IP: 40.0.0.1					
OPT: 0x0					0x0
DATA (VARIABLE LENGTH)					

ICMP

0	8	16	31	31 bits
TYPE:	CODE:	CHECKSUM		
ID: 0x7	SEQ NUMBER: 6			

Simulation Panel

Event List

Time(sec)	Last De	At Dev	Type	Info
0.001	--	PC0	ICMP	
0.002	PC0	Router1	ICMP	
0.002	Router1	Router1	ICMP	
0.003	Router1	Router1	ICMP	
0.003	Router2	Router1	ICMP	

Simulation

Fire Last Statu Sourc Destinatio Type Colo Time(Period Num Edit Delete

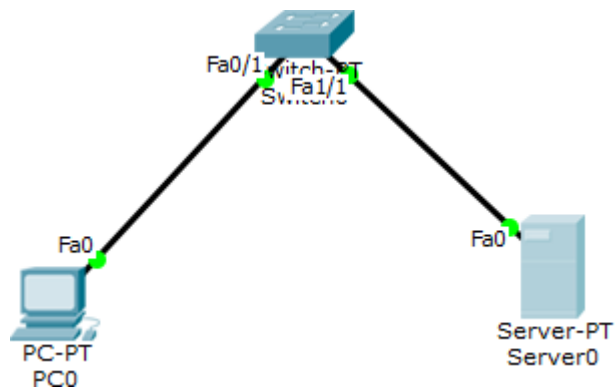
In Progr... PC0 PC1 IC... 0.000 N 0 (ed... (delete)

In Progr... PC0 PC1 IC... 0.000 N 1 (ed... (delete)

8) Configure Web Server, DNS within a LAN.

DNS

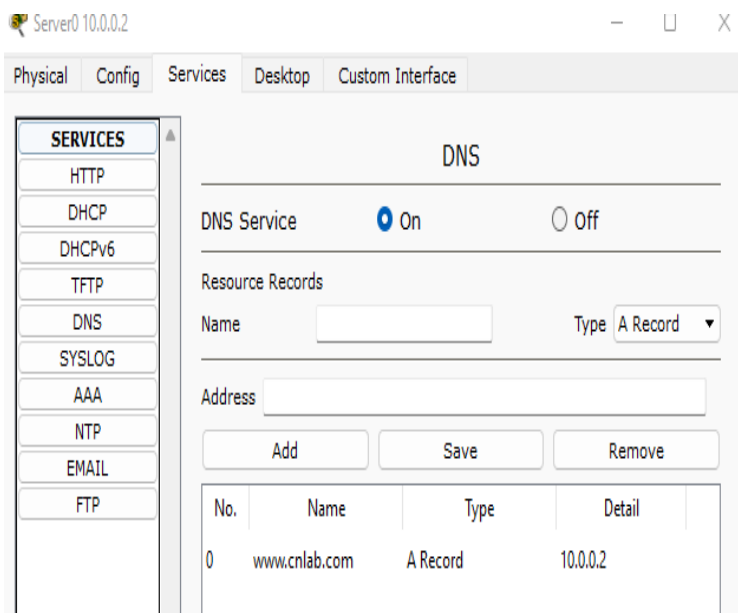
Topology:

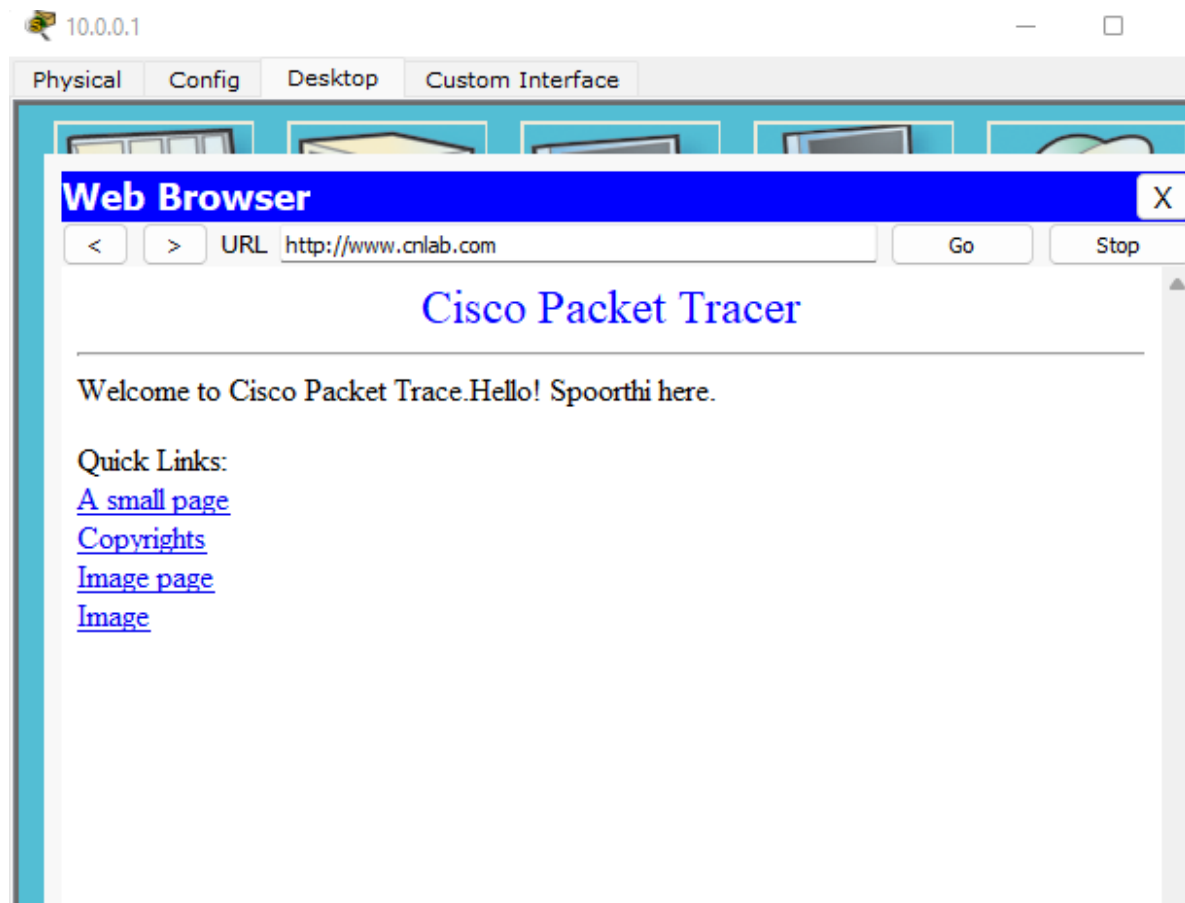
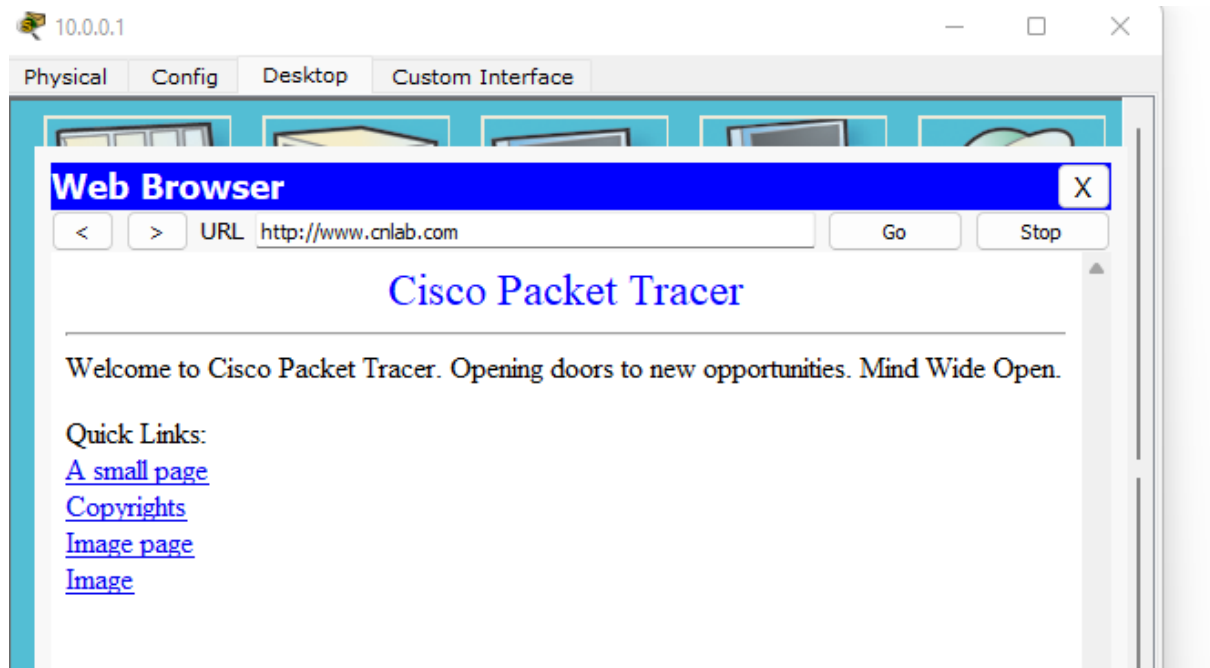


Configure ip address of PC-10.0.0.1

Server-10.0.0.2

Go to services in server select dns and add a new name as cnlab.com and address as 10.0.0.2 address of your server.



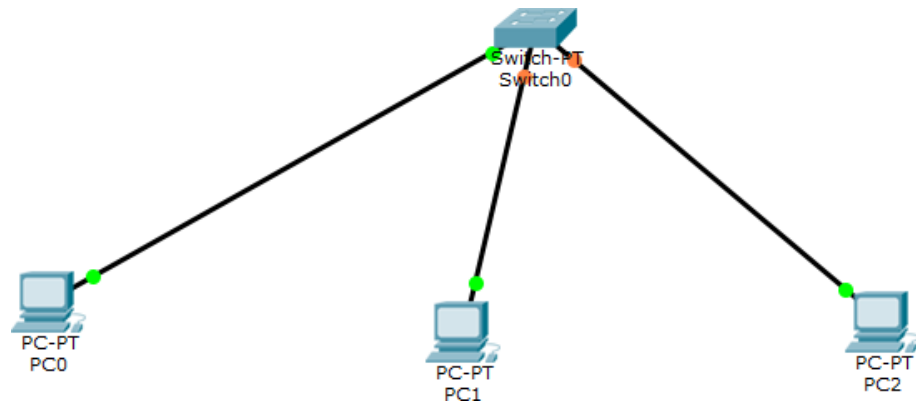


In PC go to desktop - Web browser and try to connect to server using server ip address. Output is shown (type 10.0.0.2)

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

ARP

Topology



Configure ip address for pc. No default gateway for switches.

Go to any pc cmd prompt and type as below

Command Prompt

```
Packet Tracer PC Command Line 1.0
PC>arp -a
No ARP Entries Found
PC>ping 10.0.0.2

Pinging 10.0.0.2 with 32 bytes of data:

Reply from 10.0.0.2: bytes=32 time=1ms TTL=128
Reply from 10.0.0.2: bytes=32 time=0ms TTL=128
Reply from 10.0.0.2: bytes=32 time=0ms TTL=128
Reply from 10.0.0.2: bytes=32 time=0ms TTL=128

Ping statistics for 10.0.0.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 1ms, Average = 0ms

PC>arp -a
    Internet Address      Physical Address      Type
    10.0.0.2              00d0.ffb9.3792       dynamic

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=1ms 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
```

Command Prompt



Minimum = 0ms, Maximum = 1ms, Average = 0ms

PC>arp -a

Internet Address	Physical Address	Type
10.0.0.2	00d0.ffb9.3792	dynamic

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=1ms TTL=128

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

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

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

Ping statistics for 10.0.0.3:

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

Approximate round trip times in milli-seconds:

Minimum = 0ms, Maximum = 1ms, Average = 0ms

PC>arp -a

Internet Address	Physical Address	Type
10.0.0.2	00d0.ffb9.3792	dynamic
10.0.0.3	0000.0c56.799a	dynamic

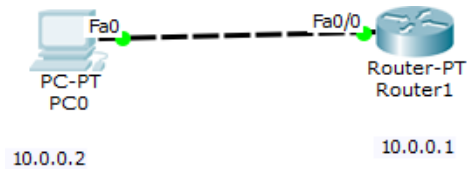
PC>arp -d

PC>arp -a

No ARP Entries Found

PC>

10) To understand the operation of TELNET by accessing the router in server room from a PC in IT off



```
Router1
Physical Config CLI
IOS Command Line Interface
32K bytes of non-volatile configuration memory.
63488K bytes of ATA CompactFlash (Read/Write)

--- System Configuration Dialog ---
Continue with configuration dialog? [yes/no]: n

Press RETURN to get started!

Router>enable
Router#config
Configuring from terminal, memory, or network [terminal]? t
Enter configuration commands, one per line. End with CNTL/Z.
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)#
%LINK-S-CHANGED: Interface FastEthernet0/0, changed state to up

%LINEPROTO-S-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up

r1(config-if)#line vty 0 5
r1(config-line)#login
% Login disabled on line 132, until 'password' is set
% Login disabled on line 133, until 'password' is set
% Login disabled on line 134, until 'password' is set
% Login disabled on line 135, until 'password' is set
% Login disabled on line 136, until 'password' is set
% Login disabled on line 137, until 'password' is set
r1(config-line)#password p0
r1(config-line)#
r1(config-line)#exit
r1(config)#exit
r1#
%SYS-5-CONFIG_I: Configured from console by console

r1#wr
Building configuration...
[OK]
r1#
```

hostname R1

enable secret p1

interface fastethernet 0/0

ip address 10.0.0.1 255.0.0.0

no shut

line vty 0 5 --to allow virtual terminal access for 6 users

login

password p0

exit

exit

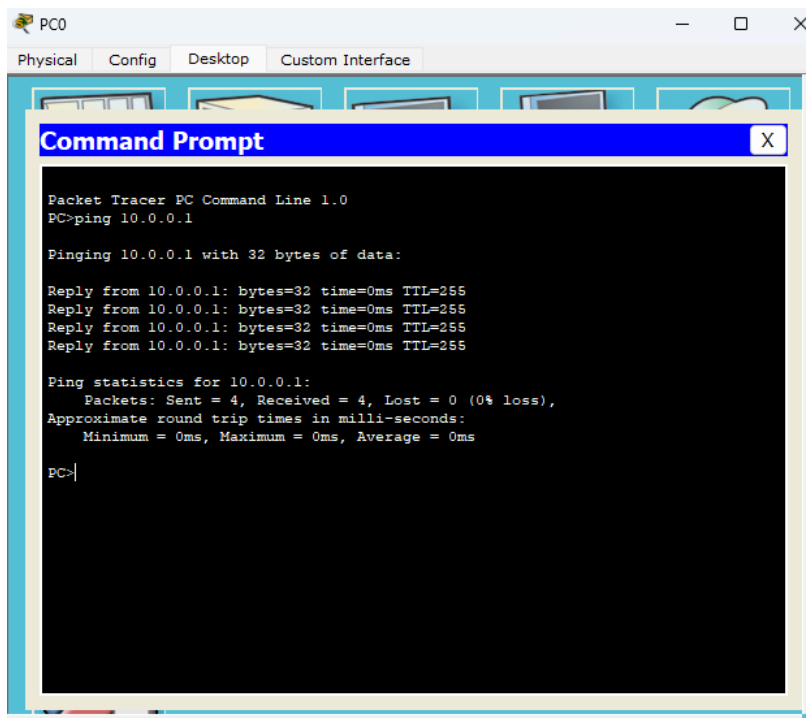
wr – to save changes in router

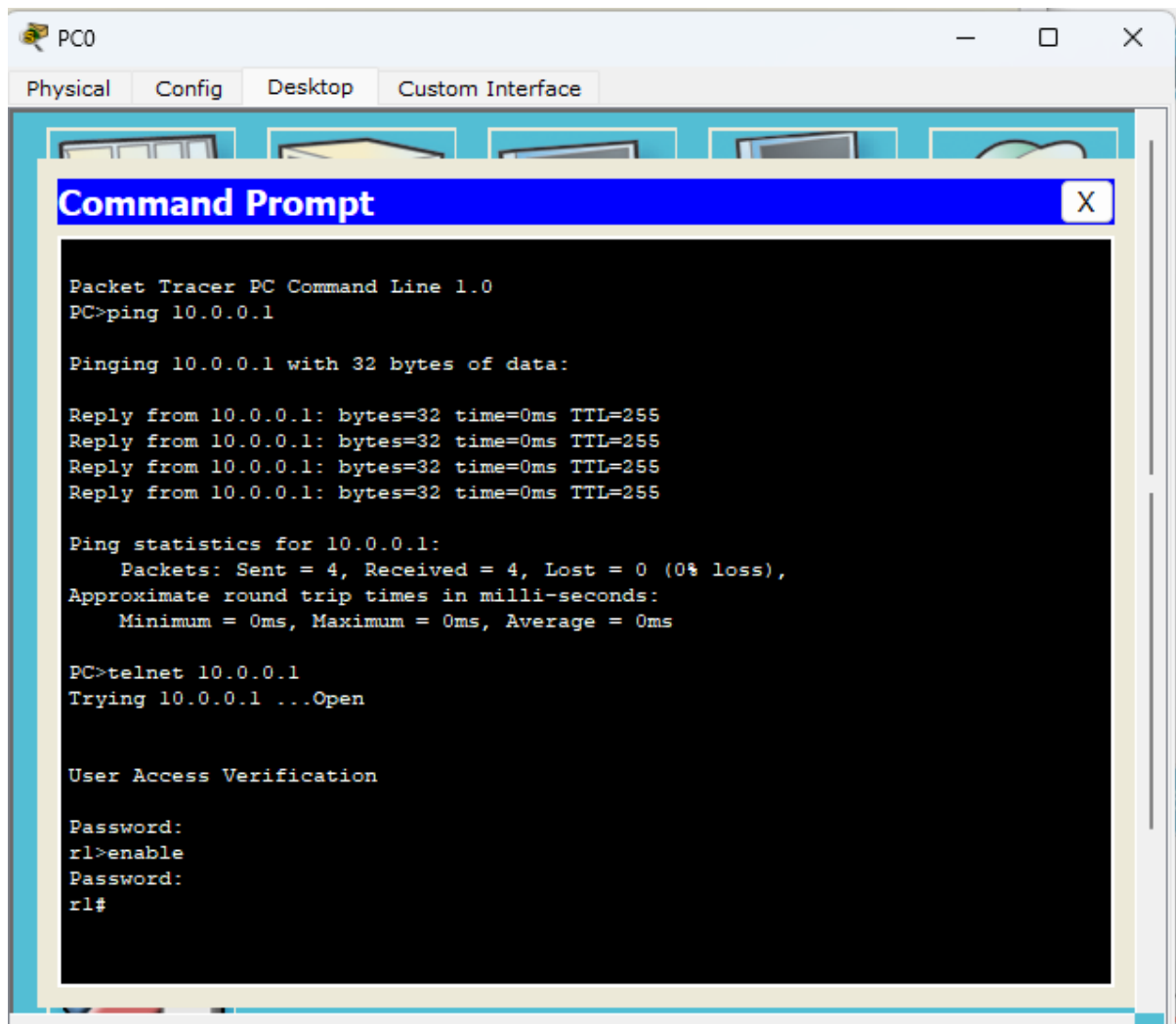
Commands in PC

In command prompt,

Ping 10.0.0.1

Ping results seen

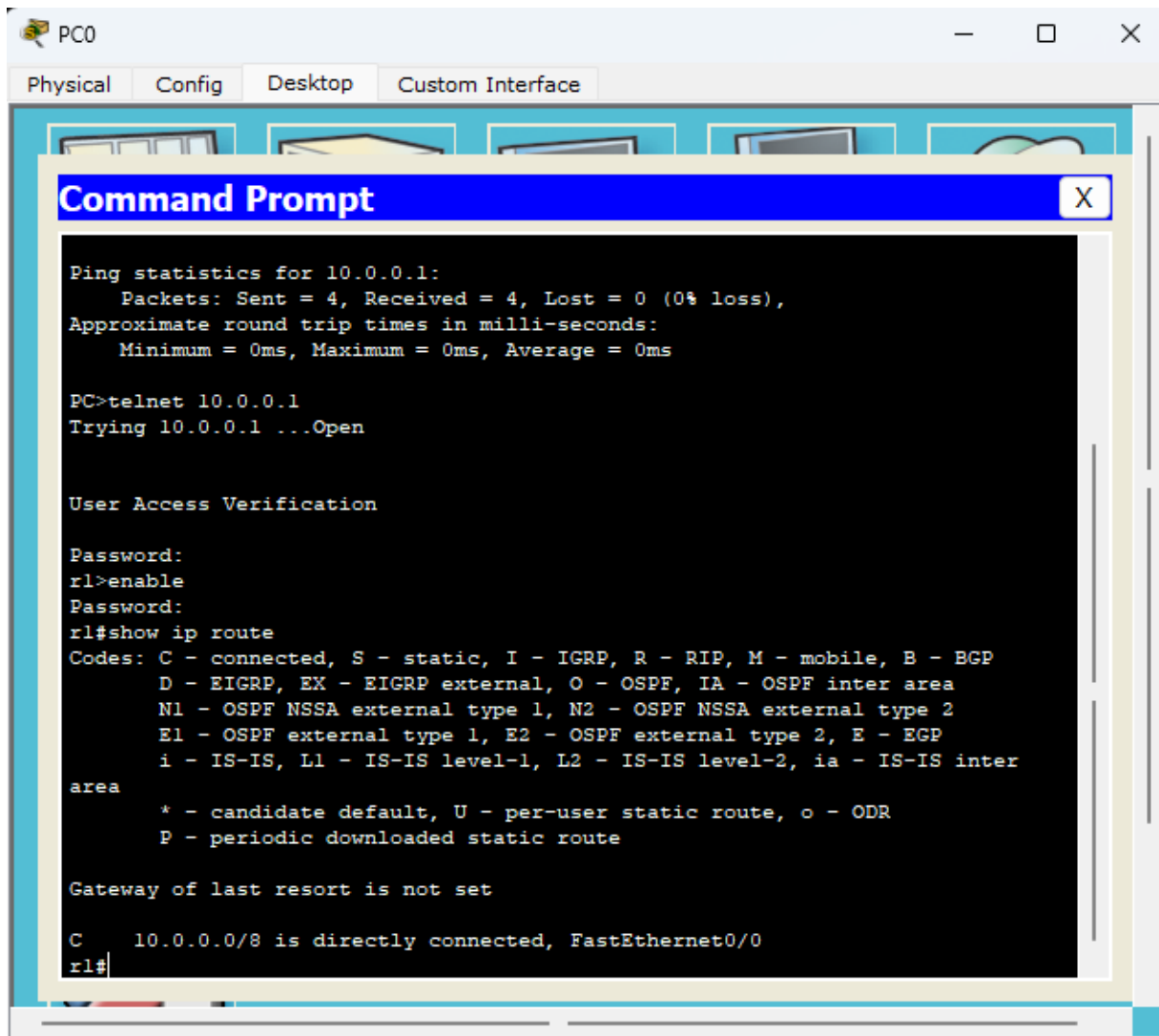




Password for User Access Verification is p0

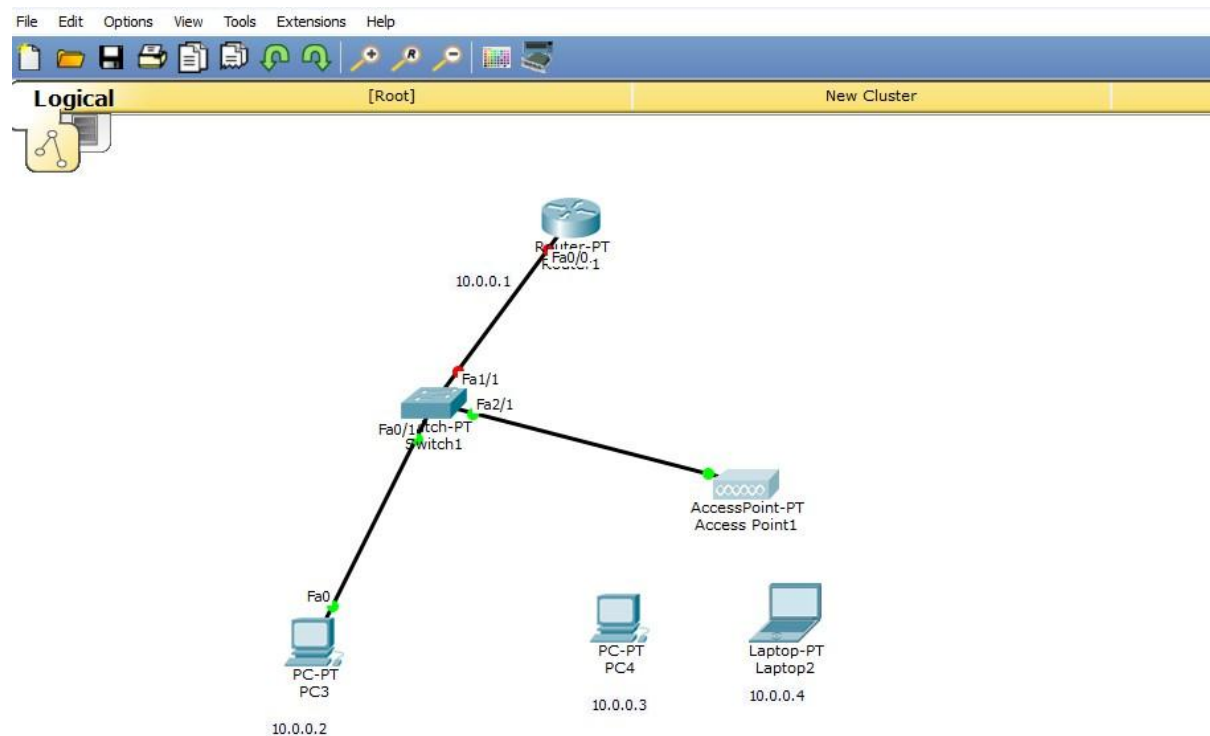
Password for enable is p1

Accessing router CLI from PC



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

11) To construct a WLAN and make the nodes communicate wirelessly.

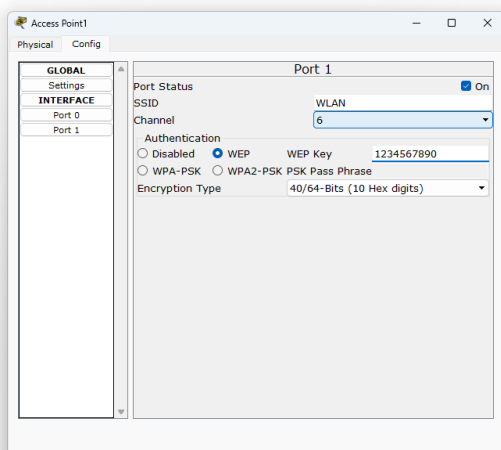


Construct the above topology

Configure PC3 and the Router1 as is normally done

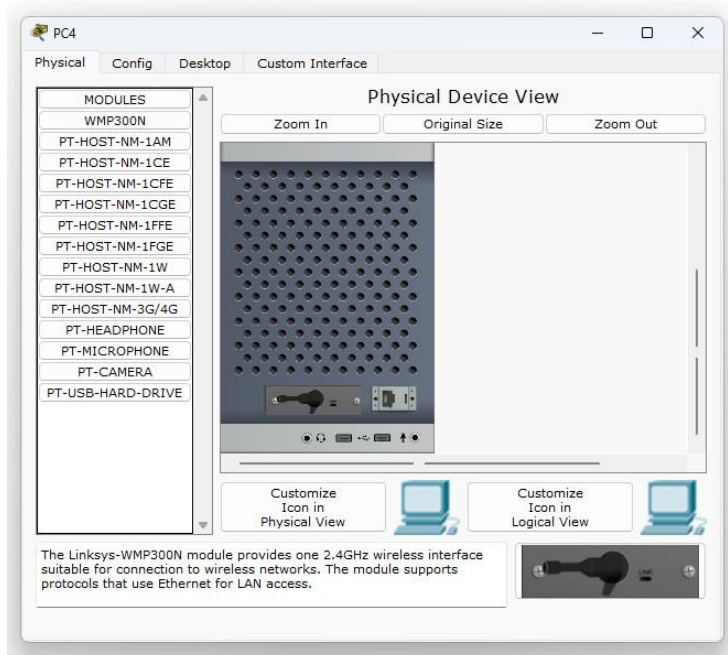
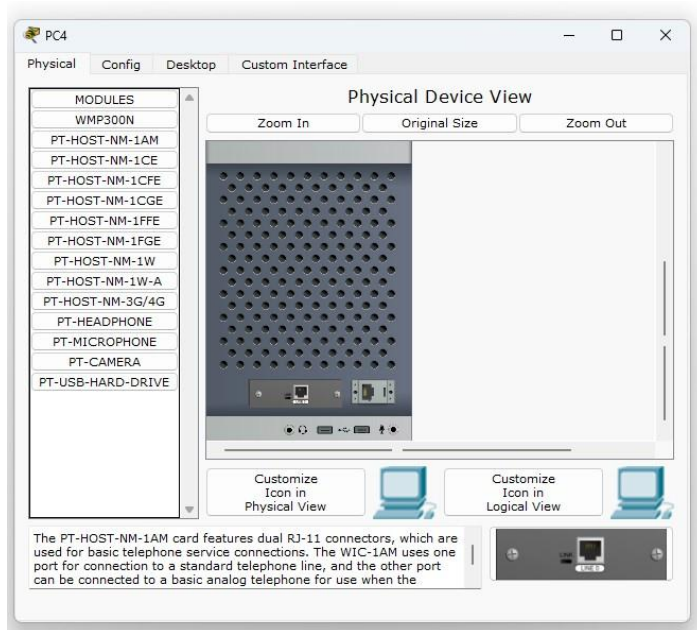
Configure Access Point1- Port1 -> SSID Name- any name(WLAN here)

Select WEP and give any 10 digit hex key – 1234567890 here



Configuring 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** (as normally done) to the device.

PC4

PhysicalConfigDesktopCustom Interface

GLOBAL

Settings

Algorithm Settings

INTERFACE

Wireless0

Wireless0

Port Status

On

Bandwidth

1 Mbps

MAC Address

0002.166C.E6CB

SSID

WLAN

Authentication

Disabled

WEP

WPA-PSK

WPA2-PSK

WPA

WPA2

WEP Key

1234567890

PSK Pass Phrase

User ID

Password

Encryption Type

40/64-Bits (10 Hex digits)

IP Configuration

DHCP

Static

IP Address

10.0.0.3

Subnet Mask

255.0.0.0

IPv6 Configuration

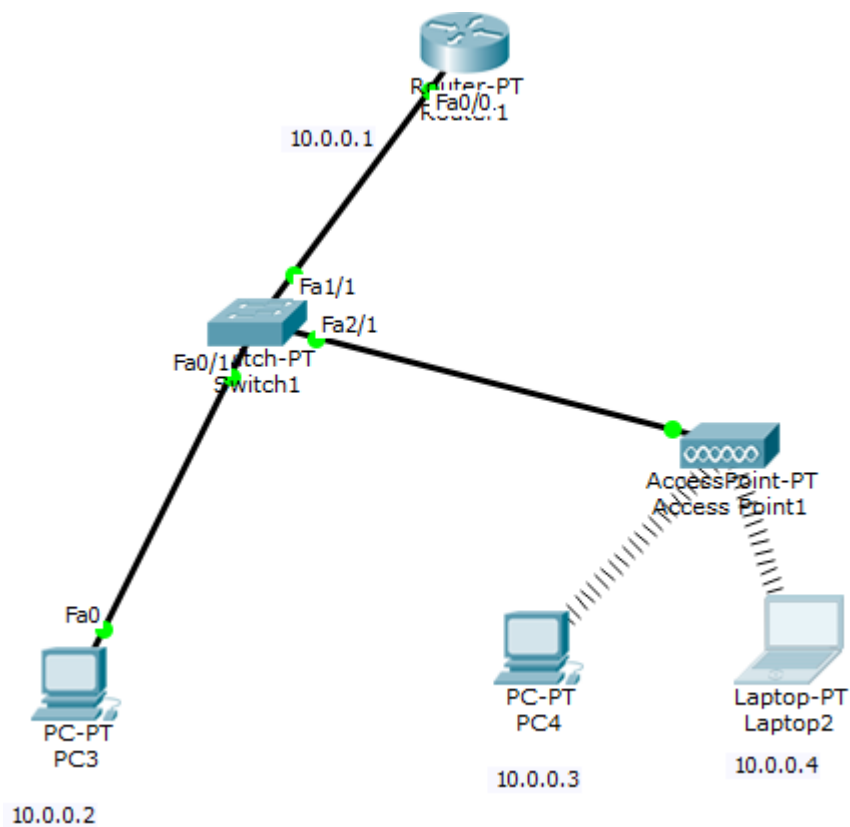
DHCP

Auto Config

Static

67

Final topology on screen

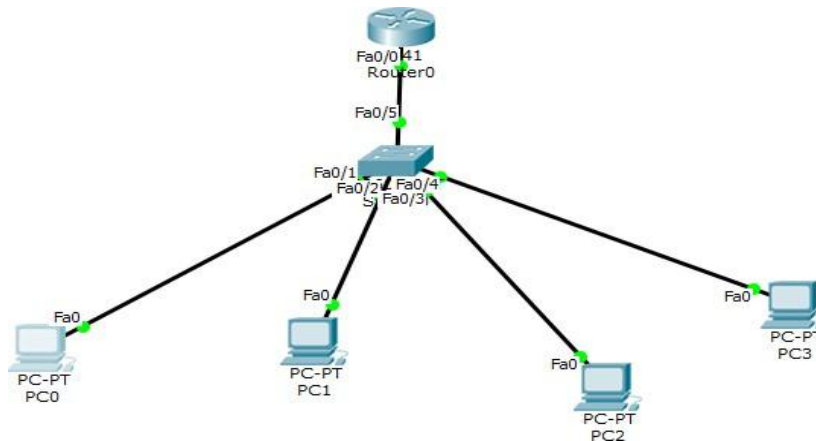


Ping from every device to every other device and see the results

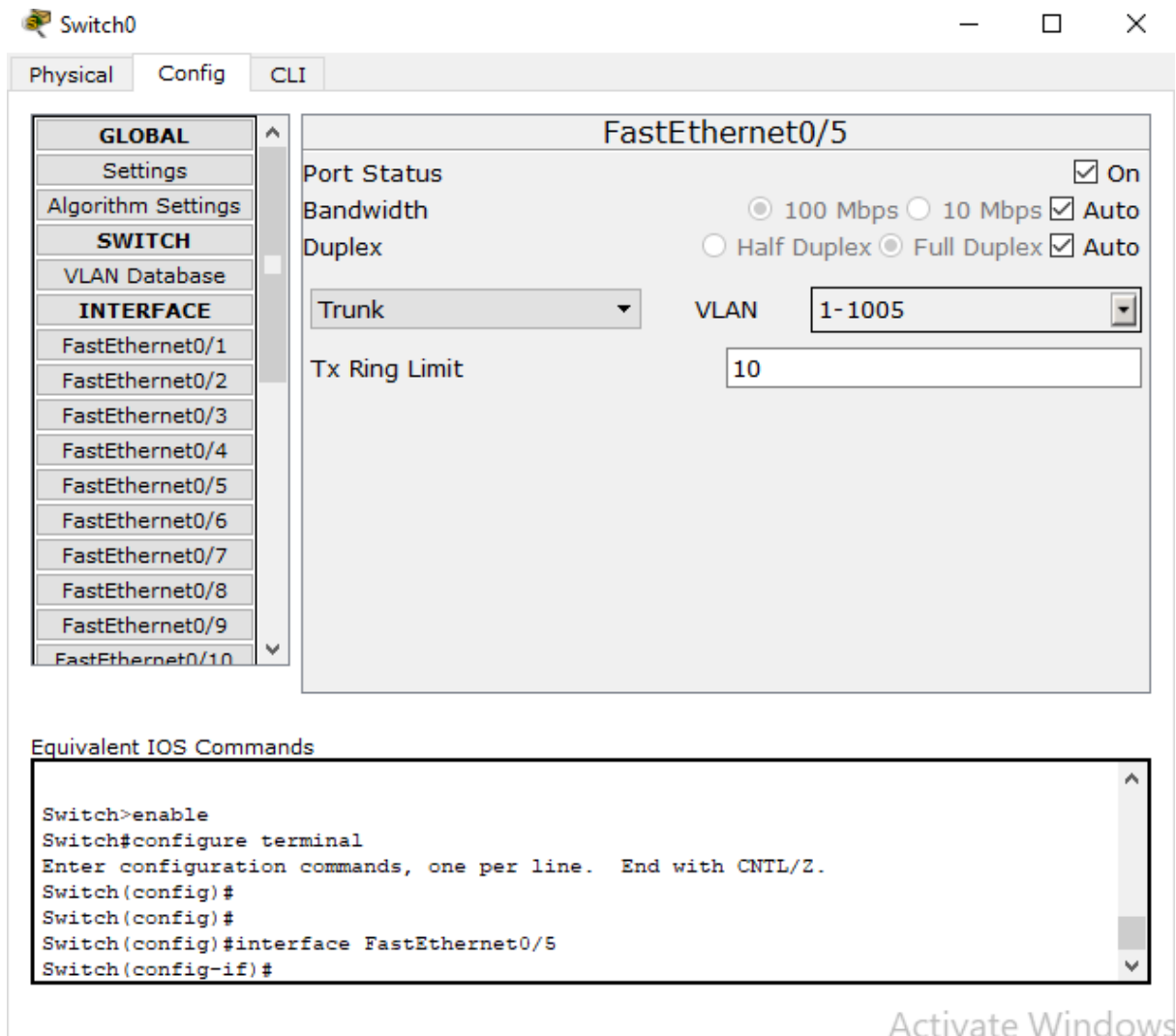
12) To construct a VLAN and make the PC's communicate among a VLAN

VLAN

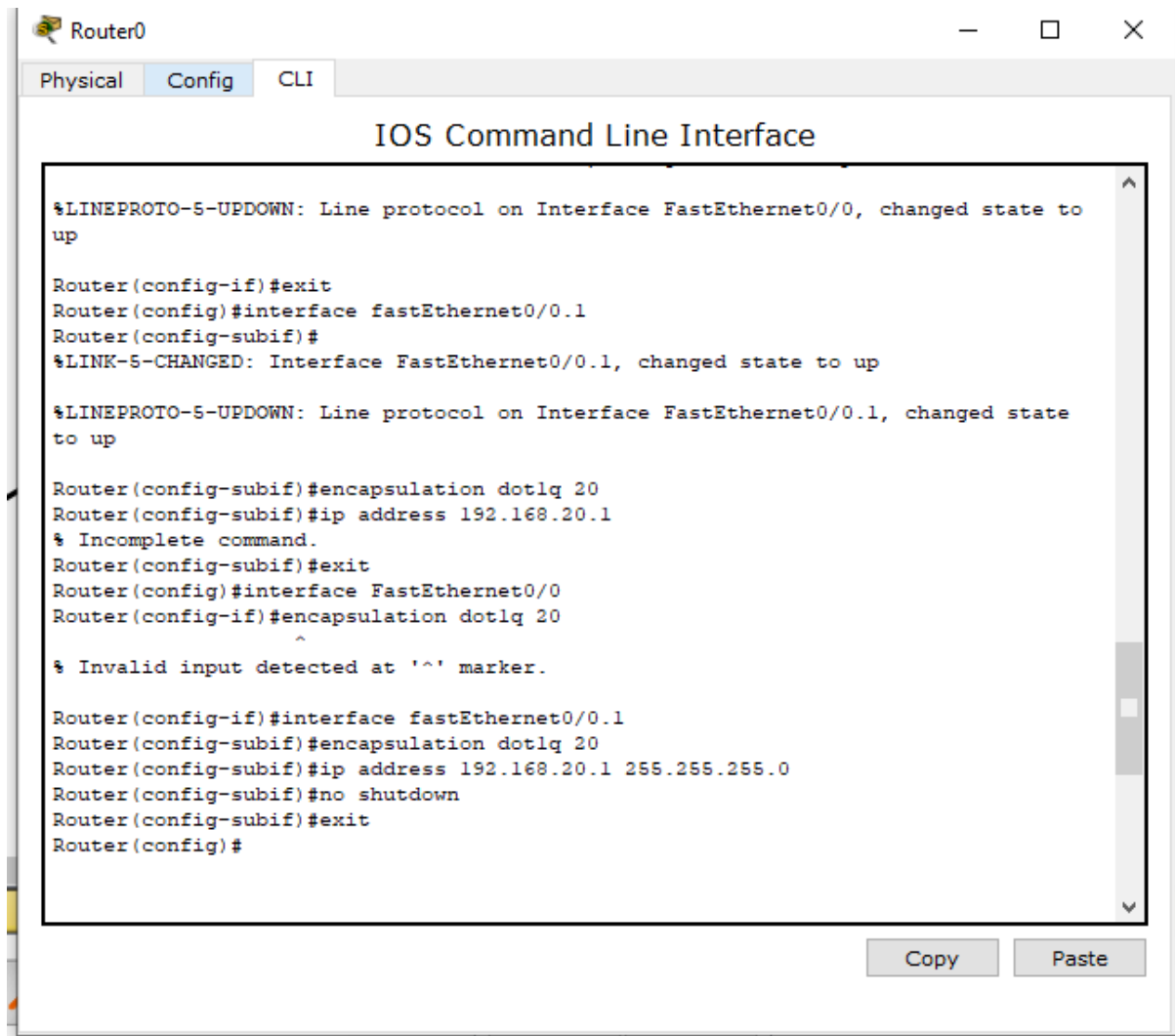
TOPOLOGY



1. Connect pc's as shown. Switch-2960 Router-1841
2. Configure IP address and gateway to pc.
3. Go to switch ->config->VLAN database set any VLAN name. But vlan number must be equal to the last but one number of the ip address (this must not be initially configured as the interface address of router) if we have 2 g/w as 192.168.1.1 and 192.168.20.1 and you have configured basically 192.168.1.1 for router interface then vlan number is 20.
4. In switch select the interface which goes to router and set dropdown to trunk



5.The right side sys conn to switch must be selected as vlan 20.
Go to router and foll commands.



Command Prompt

```
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=0ms 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 = 0ms, Average = 0ms
```

```
PC>ping 192.168.20.2
```

```
Pinging 192.168.20.2 with 32 bytes of data:
```

```
Reply from 192.168.20.2: bytes=32 time=0ms TTL=127
```

```
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 = 4, Lost = 0 (0% loss),
```

```
Approximate round trip times in milli-seconds:
```

```
    Minimum = 0ms, Maximum = 1ms, Average = 0ms
```

```
PC>
```

CYCLE-II

1) Write a program for congestion control using Leaky bucket algorithm.

```
#include<stdio.h>
#include<stdlib.h>
#include<unistd.h>
#include<time.h>
#define bkt 512

void bktinp(int a,int b){
    if(a>bkt){
        printf("Bucket overflow");
    }

    else{
        usleep(500000);
        while(a>b){
            printf("Outputted %d \n",b);
            a-=b;
            usleep(500000);
        }

        if(a>0){
            printf("Outputted %d \n",a);
        }
    }
}

void main(){
    int pkt,i,op;
    srand(time(NULL));
```

```

printf("Enter op rate \n");
scanf("%d",&op);
for(i=1;i<=5;i++){
    usleep(rand()%1000000);
    pkt=rand()%1000;
    printf("Pkt size= %d ",pkt);
    bktinp(pkt,op);

}

}

```

```

Enter op rate
30
Pkt size= 975 Bucket overflowPkt size= 181 Outputted 30
Outputted 30
Outputted 30
Outputted 30
Outputted 30
Outputted 30
Outputted 1
Pkt size= 575 Bucket overflowPkt size= 207 Outputted 30
Outputted 30
Outputted 30
Outputted 30
Outputted 30
Outputted 30
Outputted 27
Pkt size= 768 Bucket overflow

...Program finished with exit code 0
Press ENTER to exit console.

```


2) 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.

```
1  from socket import *
2  serverName="127.0.0.1"
3  serverPort=12000
4
5  clientSocket=socket(AF_INET,SOCK_STREAM)
6  clientSocket.connect((serverName,serverPort))
7  sentence=input("Enter file name")
8  clientSocket.send(sentence.encode())
9  filecontents=clientSocket.recv(1024).decode()
10 print("From server \n")
11 print(filecontents)
12 clientSocket.close()
```

```
1  from socket import *
2  serverName="127.0.0.1"
3  serverPort=12000
4  serverSocket=socket(AF_INET,SOCK_STREAM)
5  serverSocket.bind((serverName,serverPort))
6  serverSocket.listen(1)
7  while(1):
8      print("Server is ready to receive")
9      connectionSocket,addr=serverSocket.accept()
10     sentence=connectionSocket.recv(1024).decode()
11     file=open(sentence,'r')
12     l=file.read(1024)
13     connectionSocket.send(l.encode())
14     print("\n Sent contents of "+sentence)
15     file.close()
16     connectionSocket.close()
```

```

PS C:\Users\Vaishnavi Kamath\Desktop\propy> python clientTCP.py
Traceback (most recent call last):
  File "C:\Users\Vaishnavi Kamath\Desktop\propy\clientTCP.py", line 6, in <module>
    clientSocket.connect((serverName,serverPort))
ConnectionRefusedError: [WinError 10061] No connection could be made because the target machine actively refused it
PS C:\Users\Vaishnavi Kamath\Desktop\propy> python ServerTCP.py
Server is ready to receive

Sent contents of ServerTCP.py
Server is ready to receive

```

```

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\Vaishnavi Kamath> cd Desktop/propy
PS C:\Users\Vaishnavi Kamath\Desktop\propy> python clientTCP.py
Enter file nameServerTCP.py
From server

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

PS C:\Users\Vaishnavi Kamath\Desktop\propy>

```

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

```
clientUDP.py  serverUDP.py X
C: > Users > yasha > OneDrive > Desktop > python > serverUDP.py > ...
1  from socket import *
2  serverPort = 12000
3  serverSocket = socket(AF_INET, SOCK_DGRAM)
4  serverSocket.bind(("127.0.0.1", serverPort))
5  print ("The server is ready to receive")
6  while 1:
7      sentence,clientAddress = serverSocket.recvfrom(2048)
8
9      file=open(sentence,"r")
10     l=file.read(2048)
11
12     serverSocket.sendto(bytes(l,"utf-8"),clientAddress)
13     print("sent back to client",l)
14     file.close()
```

```
clientUDP.py X
C: > Users > yasha > OneDrive > Desktop > python > clientUDP.py > ...
1  from socket import *
2  serverName = "127.0.0.1"
3  serverPort = 12000
4  clientSocket = socket(AF_INET, SOCK_DGRAM)
5
6  sentence = input("Enter file name")
7  clientSocket.sendto(bytes(sentence,"utf-8"),(serverName, serverPort))
8  filecontents,serverAddress = clientSocket.recvfrom(2048)
9  print ('From Server\n')
10  print(filecontents)
11
12  clientSocket.close()
13
14
```

OUTPUT

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\yasha> cd C:\Users\yasha\OneDrive\Desktop\python
PS C:\Users\yasha\OneDrive\Desktop\python> python clientUDP.py
Enter file nameserverUDP.py
From Server

b'from socket import *\nserverPort = 12000\nserverSocket = socket(AF_INET, SOCK_DGRAM)\nserverSocket.bind(("127.0.0.1",
serverPort))\nprint ("The server is ready to receive")\nwhile 1:\n    sentence,clientAddress = serverSocket.recvfrom(20
48)\n    \n    file=open(sentence,"r")\n    l=file.read(2048)\n    \n    serverSocket.sendto(bytes(l,"utf-8"),clientAdd
ress)\n    print("sent back to client",l)\n    file.close()\n'
```

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\yasha> cd C:\Users\yasha\OneDrive\Desktop\python
PS C:\Users\yasha\OneDrive\Desktop\python> python serverUDP.py
The server is ready to receive
sent back to client 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)

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

    serverSocket.sendto(bytes(l,"utf-8"),clientAddress)
    print("sent back to client",l)
    file.close()
```

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

```
#include <stdio.h>
#include <string.h>

// length of the generator polynomial
#define N strlen(gen_poly)

// data to be transmitted and received
char data[28];

// CRC value
char check_value[28];

// generator polynomial
char gen_poly[10];

// variables
int data_length, i, j;

// function that performs XOR operation
void XOR()
{
    // if both bits are the same, the output is 0
    // if the bits are different the output is 1
    for (j = 1; j < N; j++)
        check_value[j] = ((check_value[j] == gen_poly[j]) ? '0' : '1');
}

// Function to check for errors on the receiver side
void receiver()
{
    // get the received data
    printf("Enter the received data: ");
    scanf("%s", data);
    printf("\n.....\n");
    printf("Data received: %s", data);

    // Cyclic Redundancy Check
```

```

    crc();

    // Check if the remainder is zero to find the error
    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()
{
    // initializing check_value
    for (i = 0; i < N; i++)
        check_value[i] = data[i];
    do
    {
        // check if the first bit is 1 and calls XOR function
        if (check_value[0] == '1')
            XOR();
        // Move the bits by 1 position for the next computation
        for (j = 0; j < N - 1; j++)
            check_value[j] = check_value[j + 1];
        // appending a bit from data
        check_value[j] = data[i++];
    } while (i <= data_length + N - 1);
    // loop until the data ends
}

int main()

```

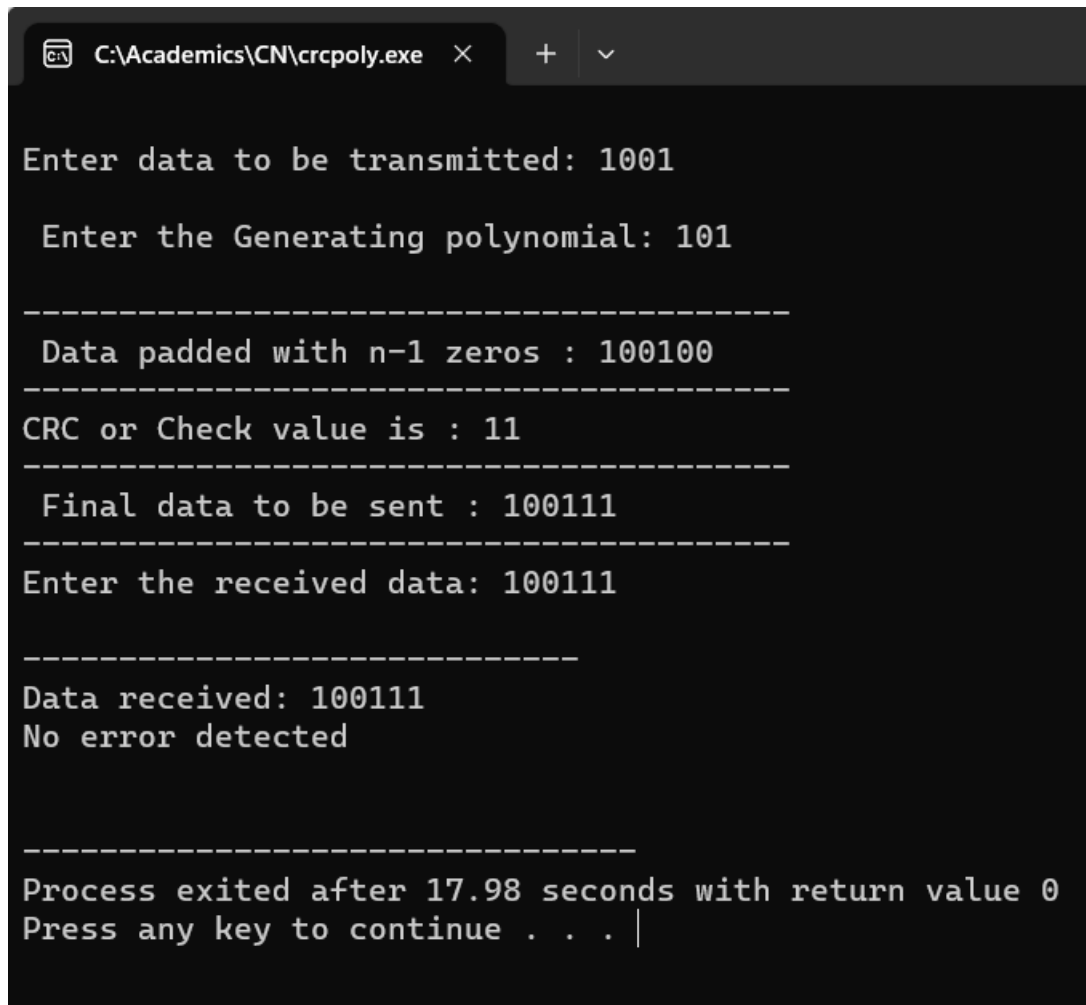
```

{
    // get the data to be transmitted
    printf("\nEnter data to be transmitted: ");
    scanf("%s", data);
    printf("\n Enter the Generating polynomial: ");
    // get the generator polynomial
    scanf("%s", gen_poly);
    // find the length of data
    data_length = strlen(data);
    // appending n-1 zeros to the data
    for (i = data_length; i < data_length + N - 1; i++)
        data[i] = '0';
    printf("\n.....");
    // print the data with padded zeros
    printf("\n Data padded with n-1 zeros : %s", data);
    printf("\n.....");
    // Cyclic Redundancy Check
    crc();
    // print the computed check value
    printf("\nCRC or Check value is : %s", check_value);
    // Append data with check_value(CRC)
    for (i = data_length; i < data_length + N - 1; i++)
        data[i] = check_value[i - data_length];
    printf("\n.....");
    // printing the final data to be sent
    printf("\n Final data to be sent : %s", data);
    printf("\n.....\n");
    // Calling the receiver function to check errors
    receiver();
    return 0;
}

```

}

OUTPUT



```
C:\Academics\CN\crcpoly.exe  ×  +  v

Enter data to be transmitted: 1001

Enter the Generating polynomial: 101

-----
Data padded with n-1 zeros : 100100
-----
CRC or Check value is : 11
-----
Final data to be sent : 100111
-----
Enter the received data: 100111

-----
Data received: 100111
No error detected

-----
Process exited after 17.98 seconds with return value 0
Press any key to continue . . . |
```



```
C:\Academics\CN\crcpoly.exe  ×  +  ∨

Enter data to be transmitted: 1001

Enter the Generating polynomial: 101

-----
Data padded with n-1 zeros : 100100
-----
CRC or Check value is : 11
-----
Final data to be sent : 100111
-----
Enter the received data: 100101

-----
Data received: 100101
Error detected

-----
Process exited after 8.587 seconds with return value 0
Press any key to continue . . . |
```

WIRESHARK

Observation

Tool exploration

WIRESHARK:

→ Wireshark is an open source Packet Analyzer tool, which is used for education, analysis, software development, communication protocol debug and network troubleshooting. It is used to filter packets so that each one is filtered to meet our specific needs. It is commonly called as sniffer, network protocol analyzer and network analyzer. It is mostly used by network security engineers to examine security problems.

→ Wireshark is a free to use Application which is used to apprehend the data back & forth. It is often called as free packet sniffer computer application. It puts the network card into an unselective mode i.e. to accept all the packets which it receives.

→ uses of wireshark -

- It allows users to watch all the traffic being passed over the network.
- It is used by network security Engineers to examine all the security problems.
- used in troubleshooting network issues.
- used in troubleshooting latency issues & malicious activities on a network.
- used to analyze dropped packets.
- we can see how devices communicate in a local network or the rest of the world.

→ Functionality -

- Wireshark is similar to tcpdump in networking. TCP dump is a common packet analyzer which allows the user to display other packets & TCP/IP packets.
- Wireshark allows user to capture & related data packets as they are transmitted over a network.
- The Wireshark is useful for monitoring network performance, identifying anomalies & troubleshooting issues.
- Wireshark provides powerful filtering & search capabilities, allowing users to narrow down the displayed packets based on specific criteria.
- Wireshark can decode & display the contents of various network protocols.
- The tool provides various statistics & graphical representation of network traffic.
- Wireshark provides path and things which made easier for security auditors.
- Wireshark is available for multiple platforms, including Windows, macOS and other Linux distributions.

Overall, Wireshark is a versatile and powerful network analysis tool that provides deep insights into network traffic, making it an indispensable tool for network administrators, security experts and anyone interested in understanding and troubleshooting network communications.