Practical 6
Configuring Simple and multi-area OSPF

Aim: Configuring Simple and multi-area OSPF

Theory:

Open shortest path first (OSPF) is a link-state routing protocol that is used to find the best path between the source and the destination router using its own shortest path first (SPF) algorithm. A link-state routing protocol is a protocol that uses the concept of triggered updates, i.e., if there is a change observed in the learned routing table then the updates are triggered only, not like the distance-vector routing protocol where the routing table is exchanged at a period of time.

Open shortest path first (OSPF) is developed by Internet Engineering Task Force (IETF) as one of the Interior Gateway Protocol (IGP), i.e., the protocol which aims at moving the packet within a large autonomous system or routing domain.

OSPF advantages –

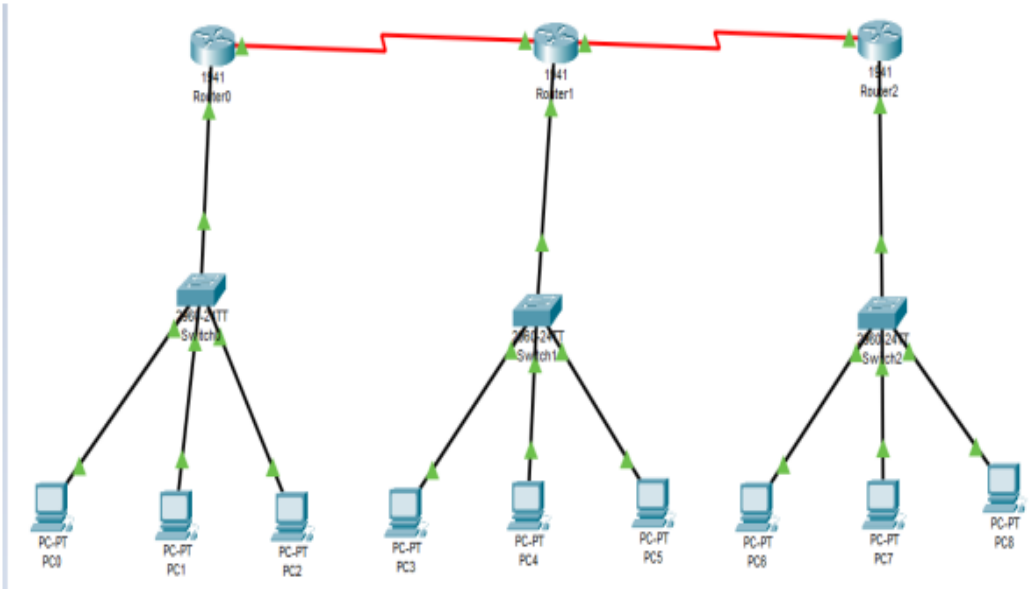
1) Both IPv4 and IPv6 routed protocols
2) Load balancing with equal-cost routes for the same destination
3) Unlimited hop counts
4) Trigger updates for fast convergence
5) A loop-free topology using SPF algorithm
6) Run-on most routers
7) Classless protocol

There are some disadvantages of OSPF

1) It requires an extra CPU process to run the SPF algorithm
2) Requiring more RAM to store adjacency topology, and
3) Being more complex to set up and hard to troubleshoot

Part a) Simple OSPF

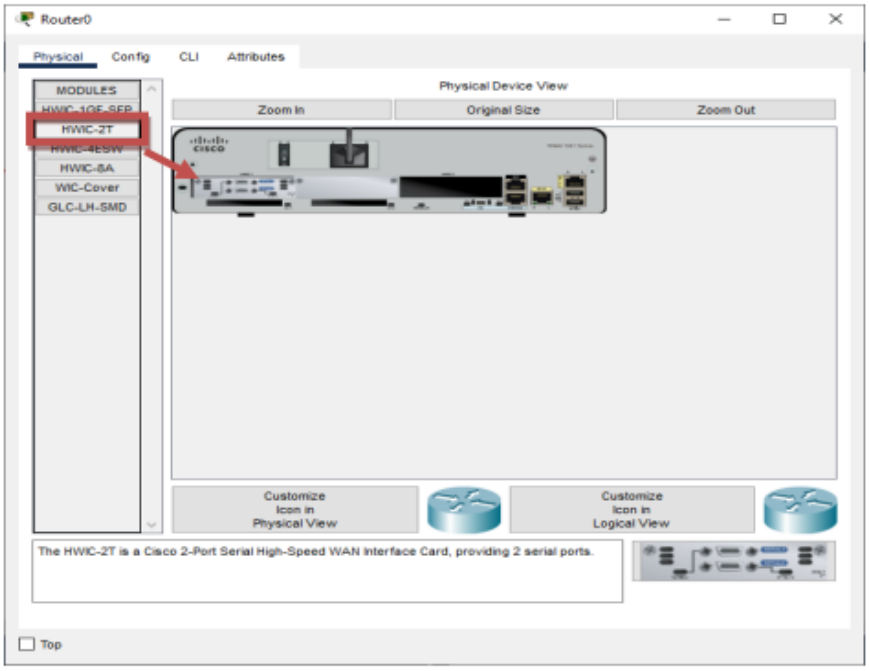
We use the following topology for the present case



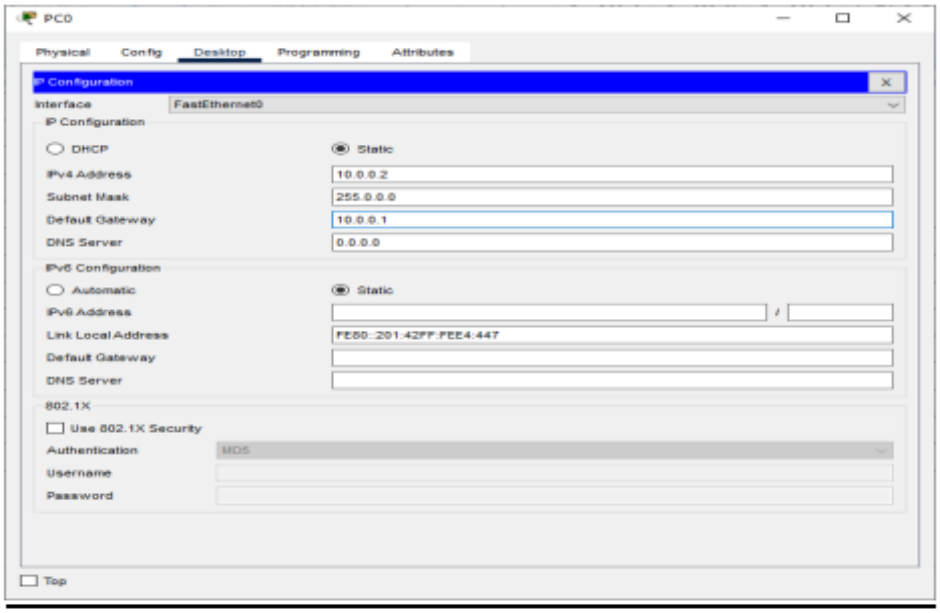
We configure the above network using the following IP addresses

Host	Interface	IP address	Default Gateway	Subnet Mask	Wildcard Mask
Router 0	G0/0	10.0.0.1		255.0.0.0	0.255.255.255
	S0/1/0	40.0.0.1			
Router 1	G0/0	20.0.0.1			
	S0/1/0	40.0.0.2			
	S0/1/1	50.0.0.1			
Router 2	G0/0	30.0.0.1			
	S0/1/1	50.0.0.2			
PC0	FastEthernet0	10.0.0.2	10.0.0.1		
PC1	FastEthernet0	10.0.0.3			
PC2	FastEthernet0	10.0.0.4			
PC3	FastEthernet0	20.0.0.2	20.0.0.1		
PC4	FastEthernet0	20.0.0.3			
PC5	FastEthernet0	20.0.0.4			
PC6	FastEthernet0	30.0.0.2	30.0.0.1		
PC7	FastEthernet0	30.0.0.3			
PC8	FastEthernet0	30.0.0.4			

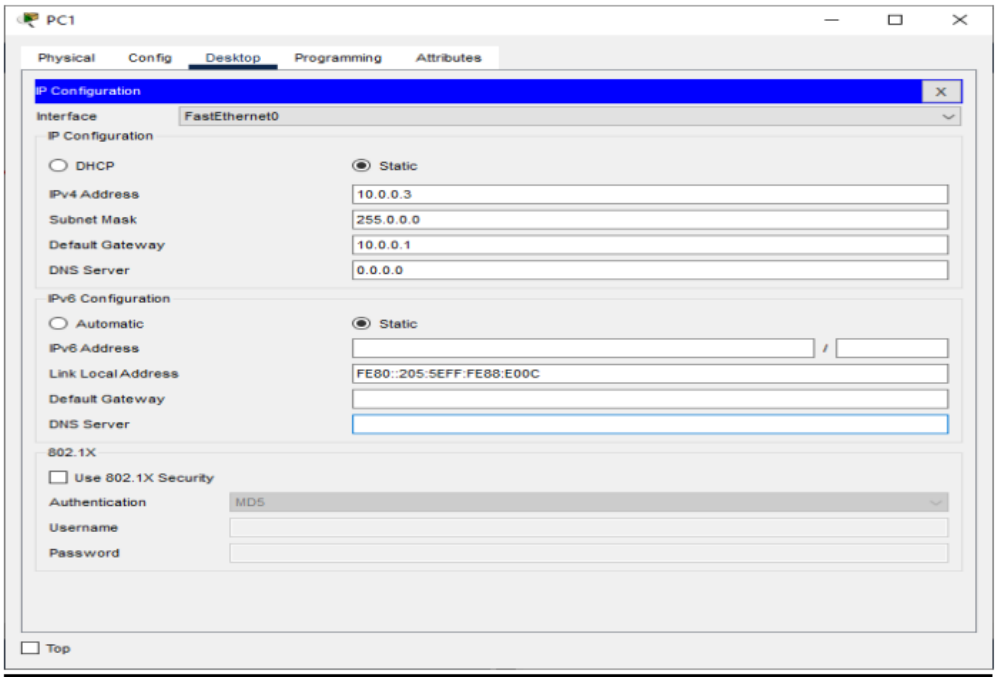
We need to add a Serial Interface in each Router, it is done as follows



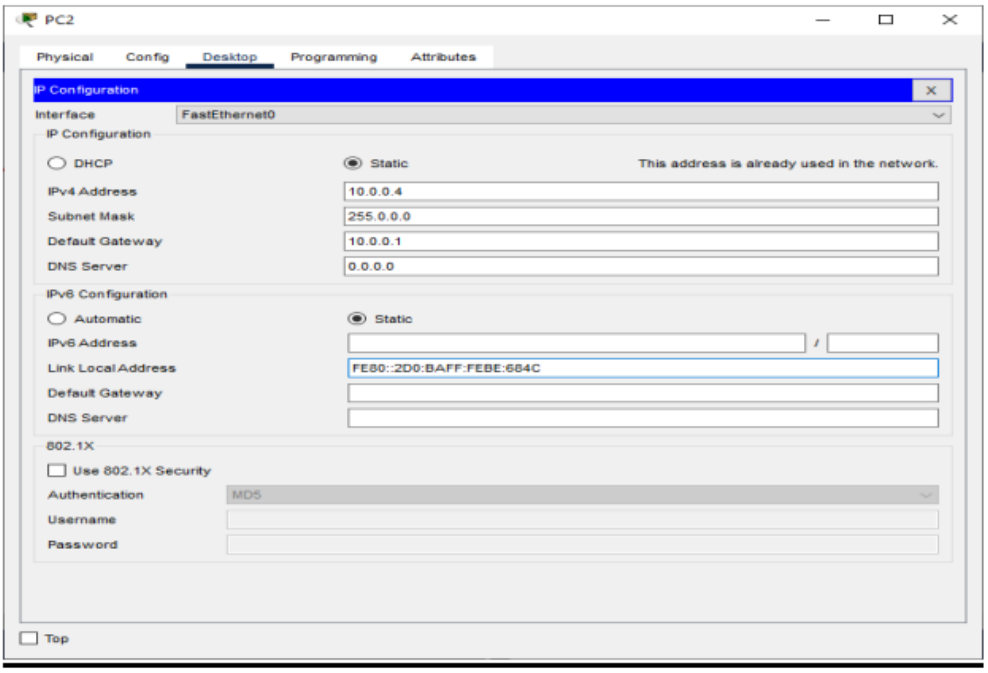
Configuring PC0:



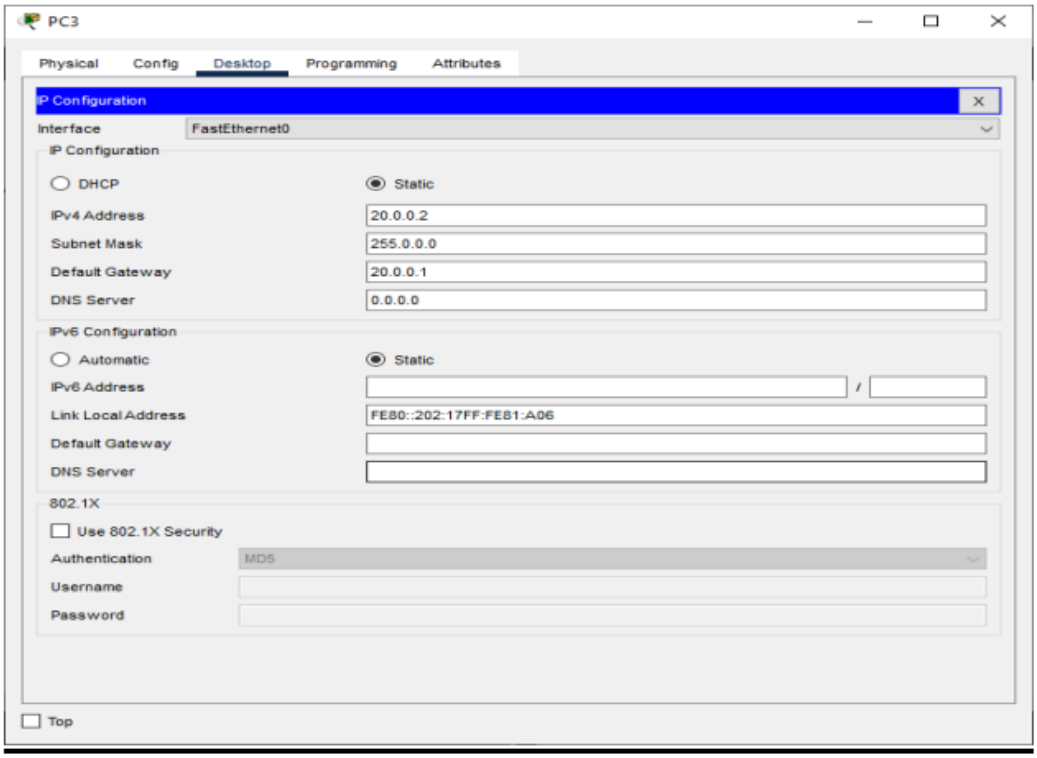
Configuring PC1:



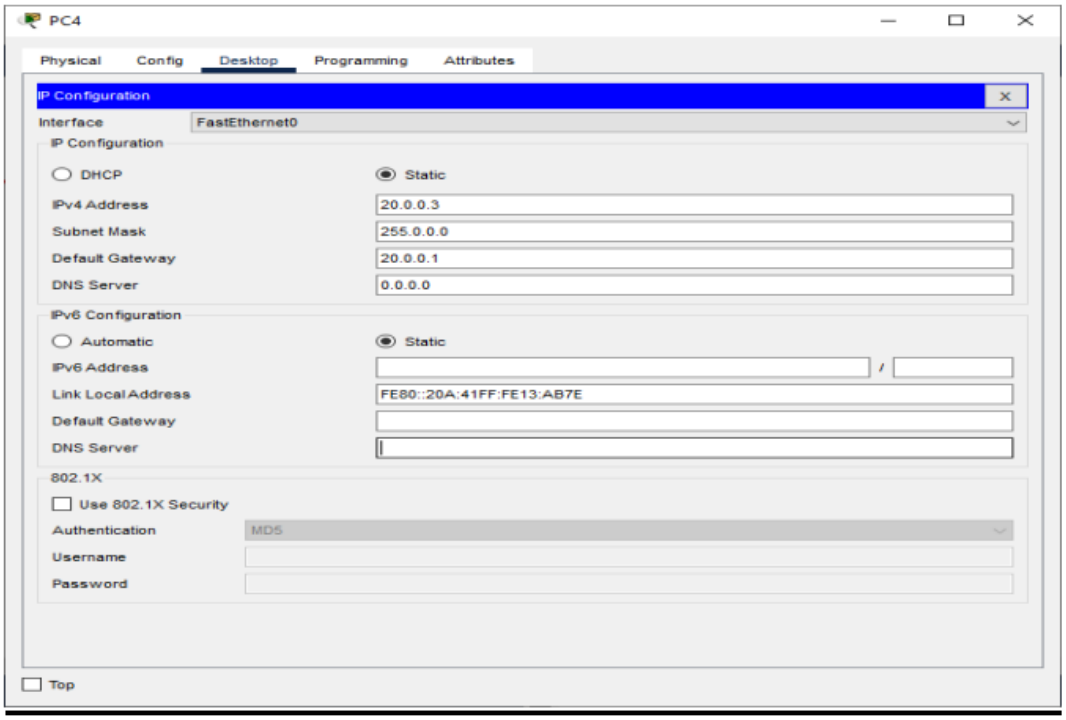
Configuring PC2:



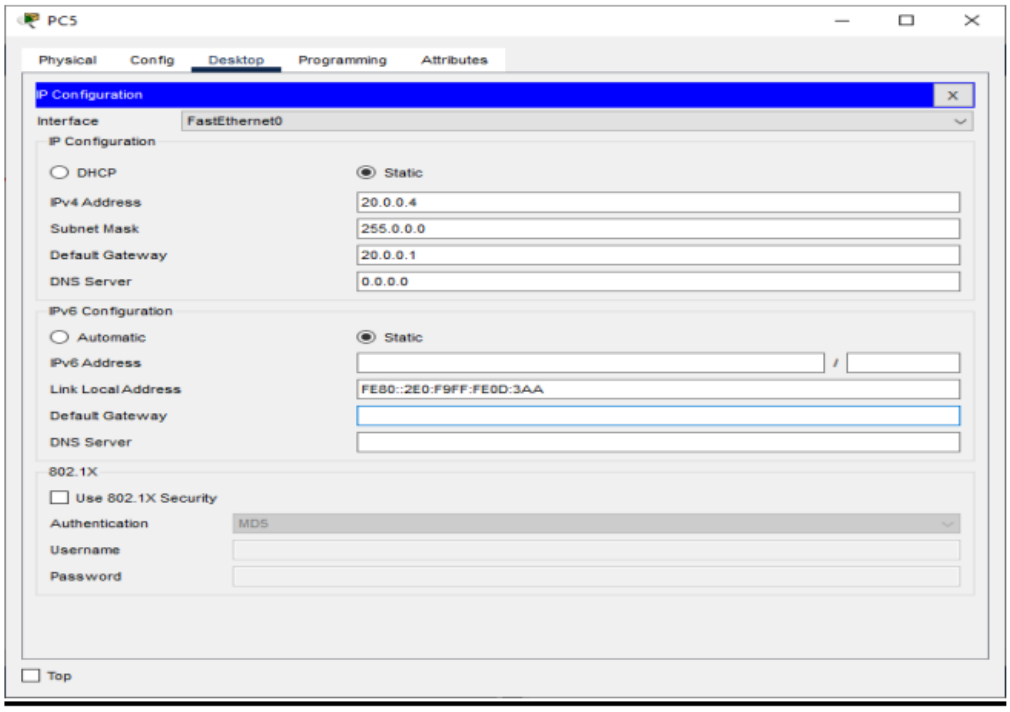
Configuring PC3:



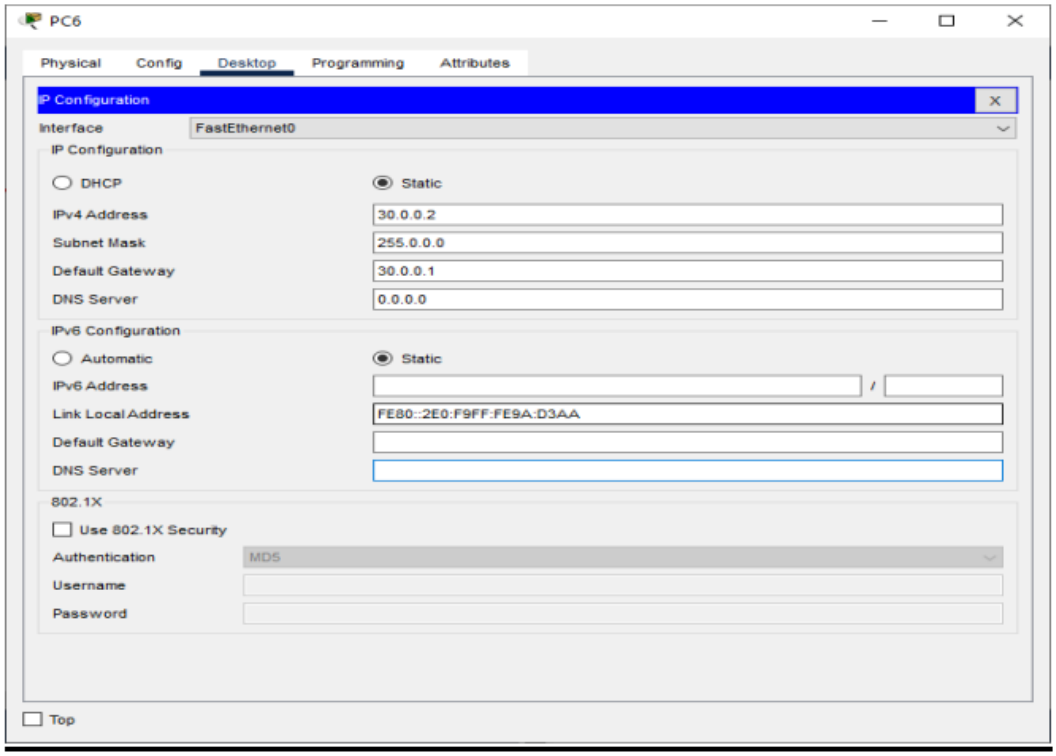
Configuring PC4:



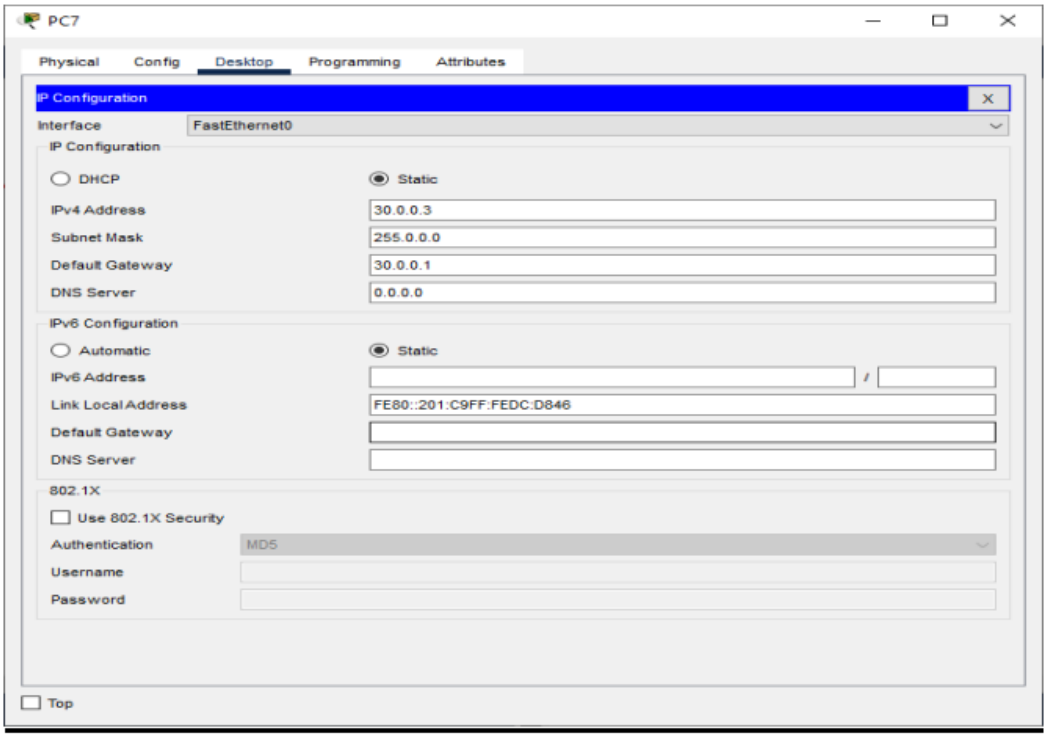
Configuring PC5:



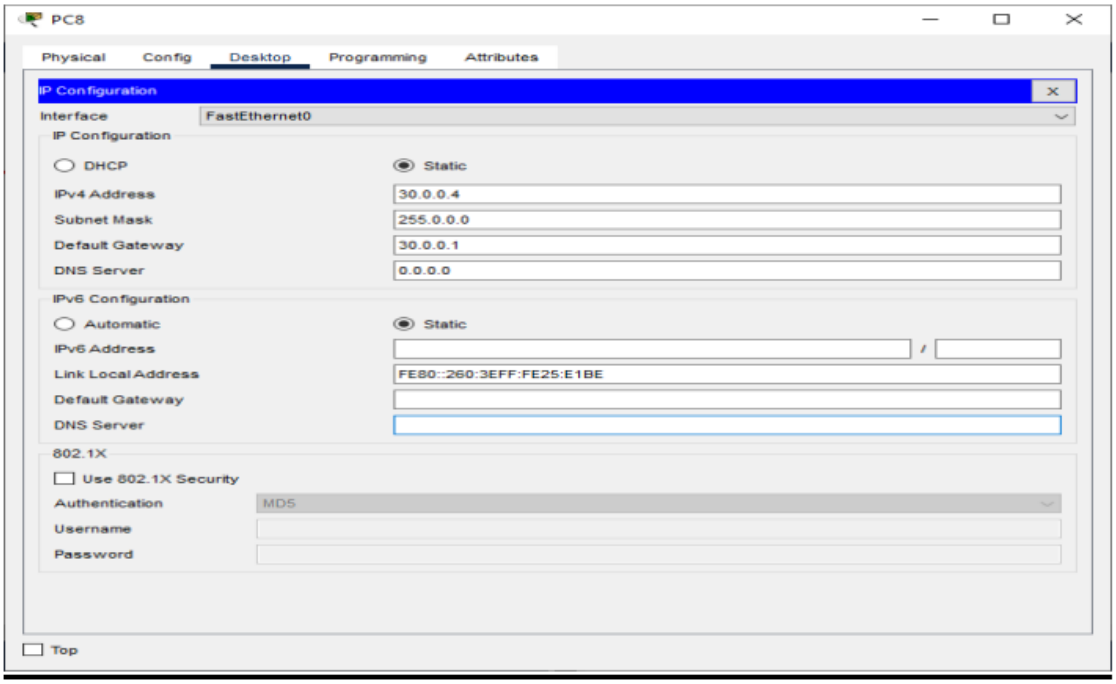
Configuring PC6:



Configuring PC7:

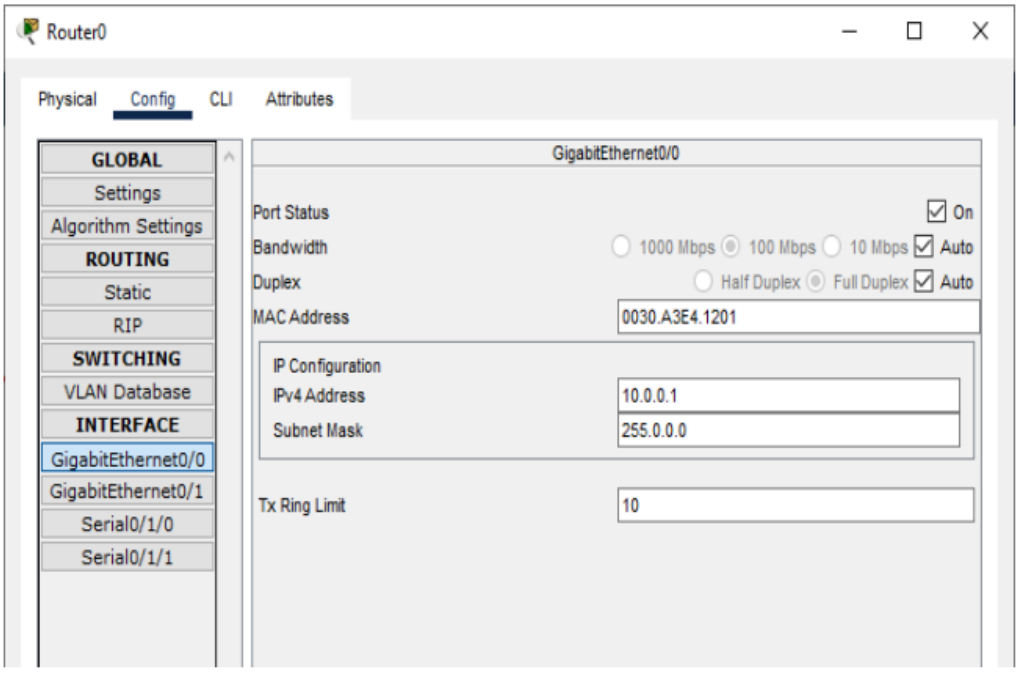


Configuring PC8:

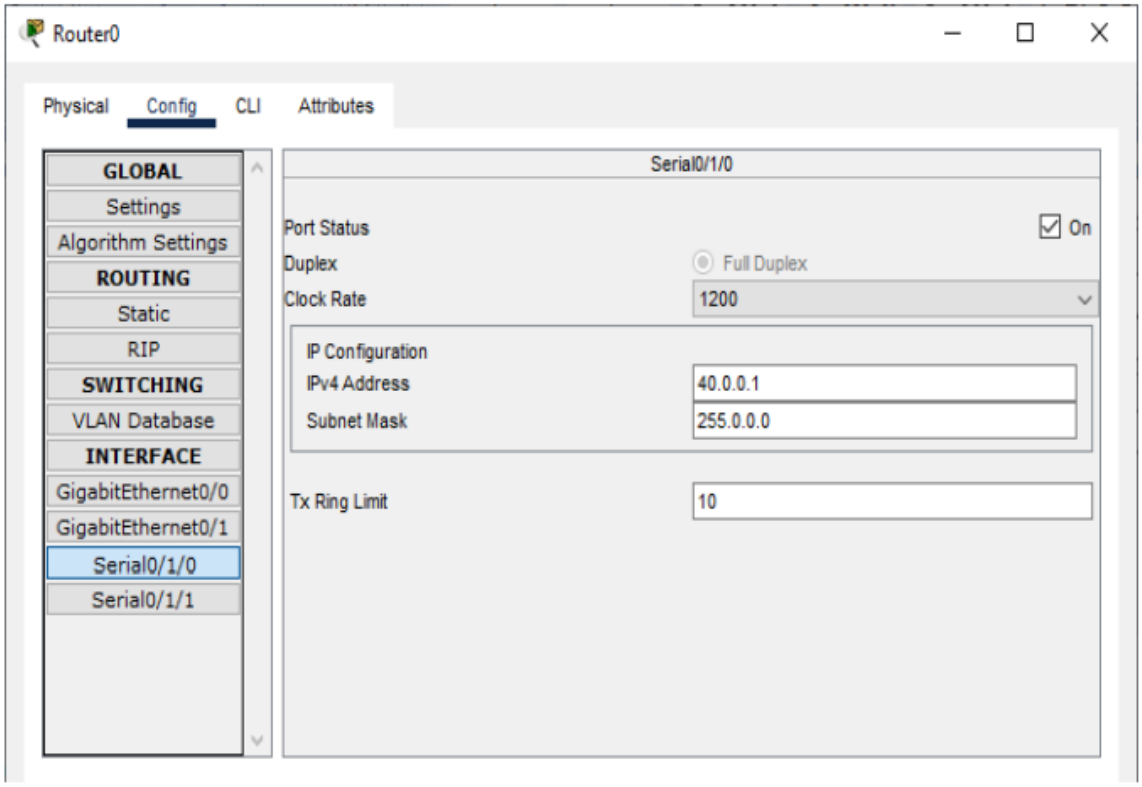


Configuring IP addresses on Router 0

i) Interface G0/0

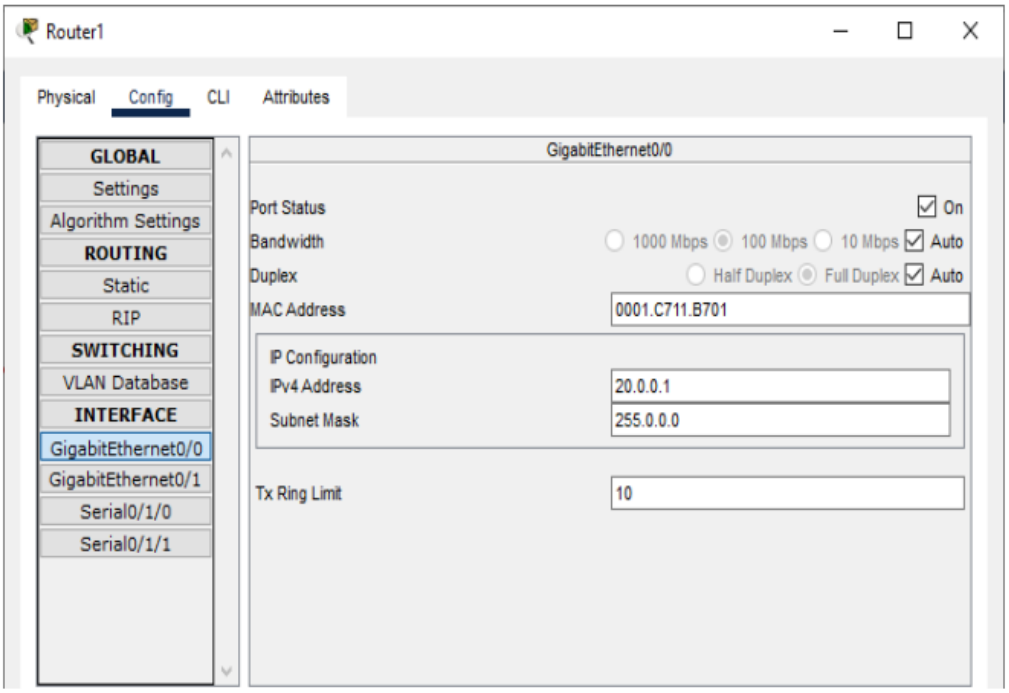


ii) Interface S0/1/0

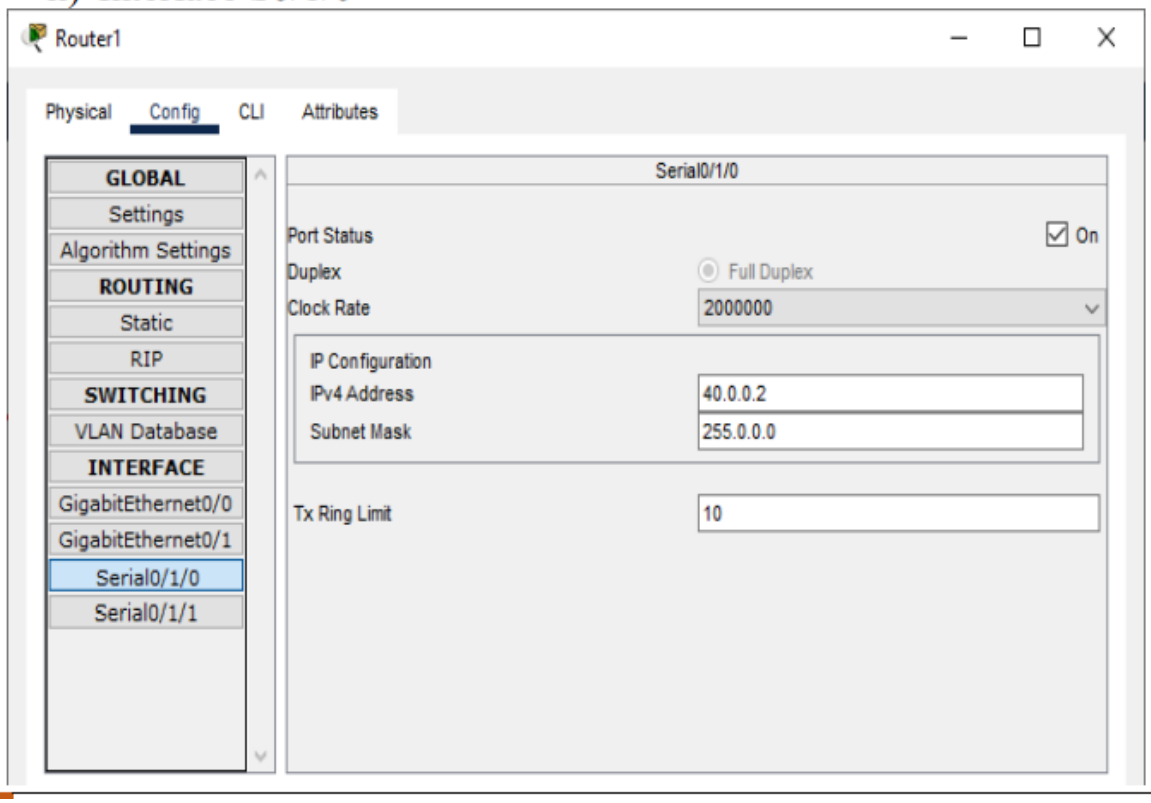


Configuring IP addresses on Router 1

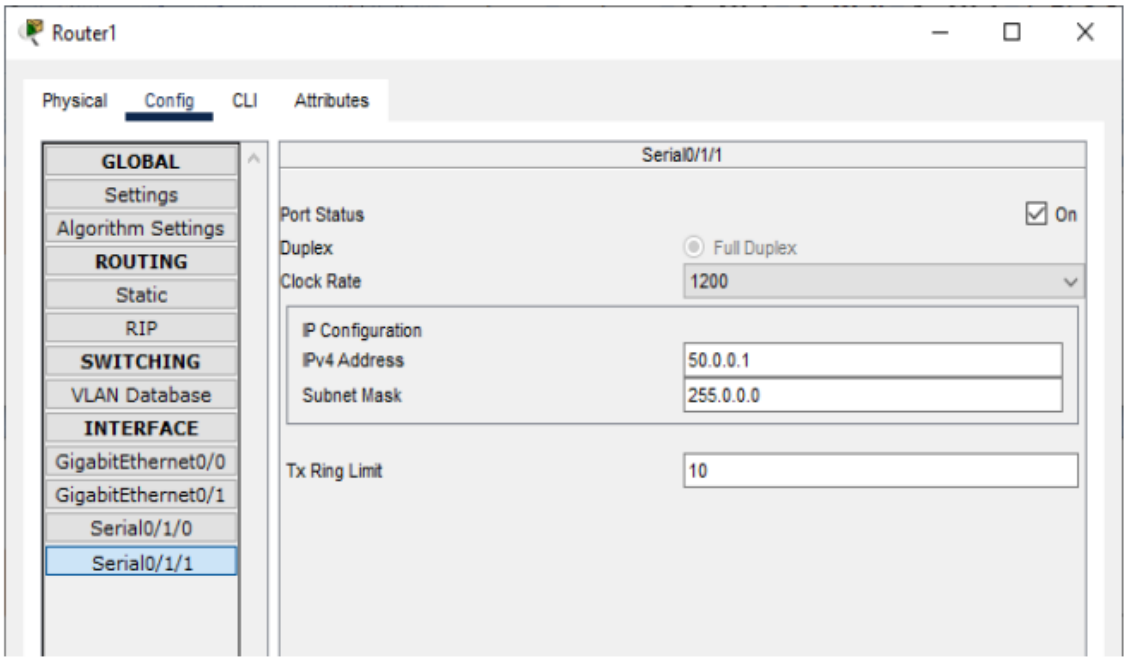
i) Interface G0/0



ii) Interface S0/1/0

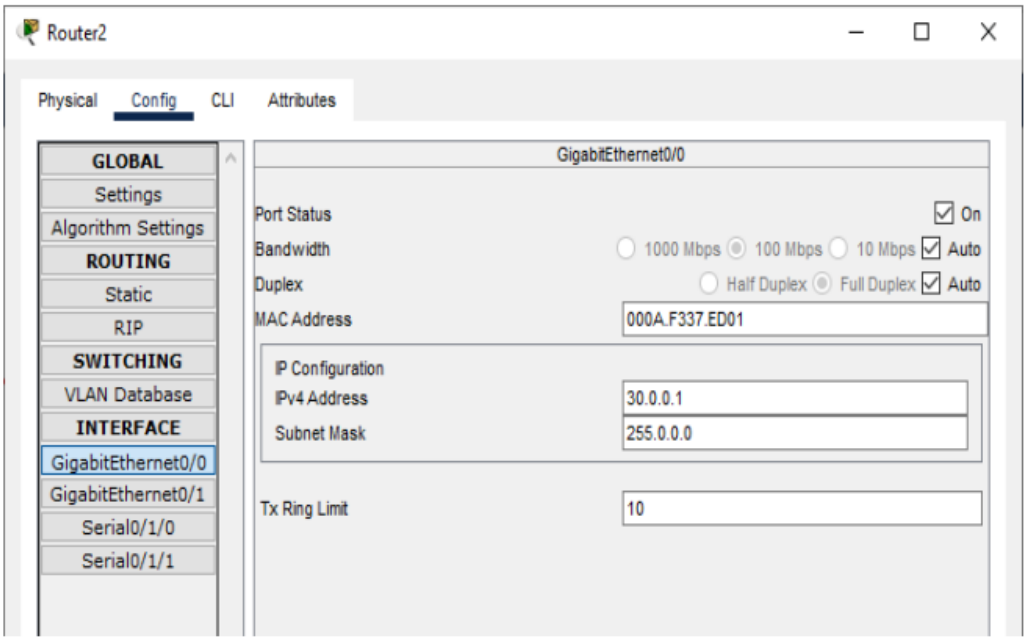


iii) Interface S0/1/1

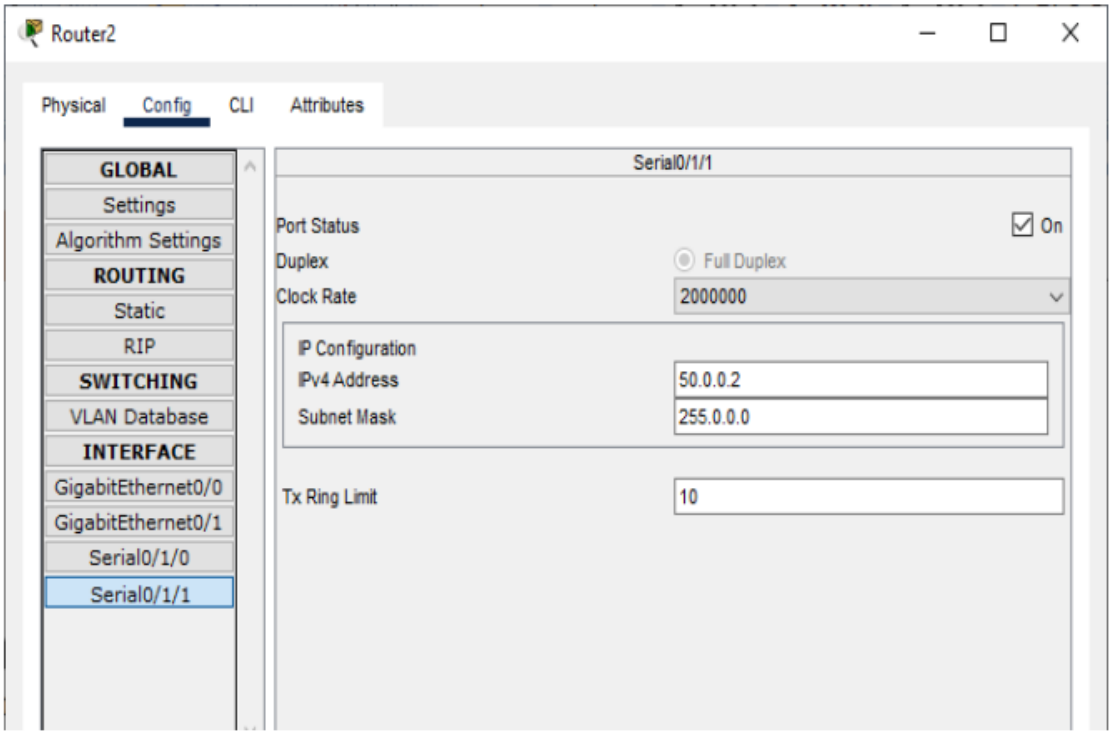


Configuring IP addresses on Router 2

i) Interface G0/0



ii) Interface S0/1/1



Configuring Router 0 for OSPF (using the CLI mode)

```
Router(config)#
Router(config)#router ospf 1
Router(config-router)#network 10.0.0.0 0.0.0.255 area 1
Router(config-router)#network 40.0.0.0 0.0.0.255 area 1
Router(config-router)#exit
Router(config)#
```

Configuring Router 1 for OSPF (using the CLI mode)

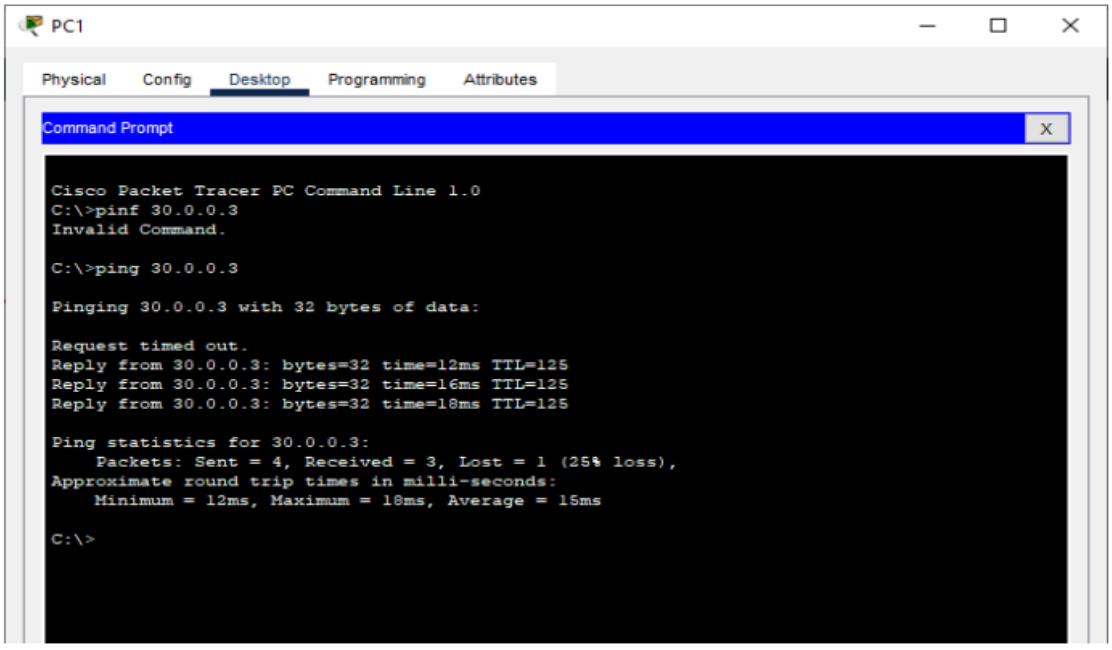
```
Router(config)#
Router(config)#router ospf 1
Router(config-router)#
Router(config-router)#network 20.0.0.0 0.0.0.255 area 1
Router(config-router)#network 40.0.0.0 0.0.0.255 area 1
Router(config-router)#network 50.0.0.0 0.0.0.255 area 1
Router(config-router)#exit
Router(config)#
```

Configuring Router 2 for OSPF (using the CLI mode)

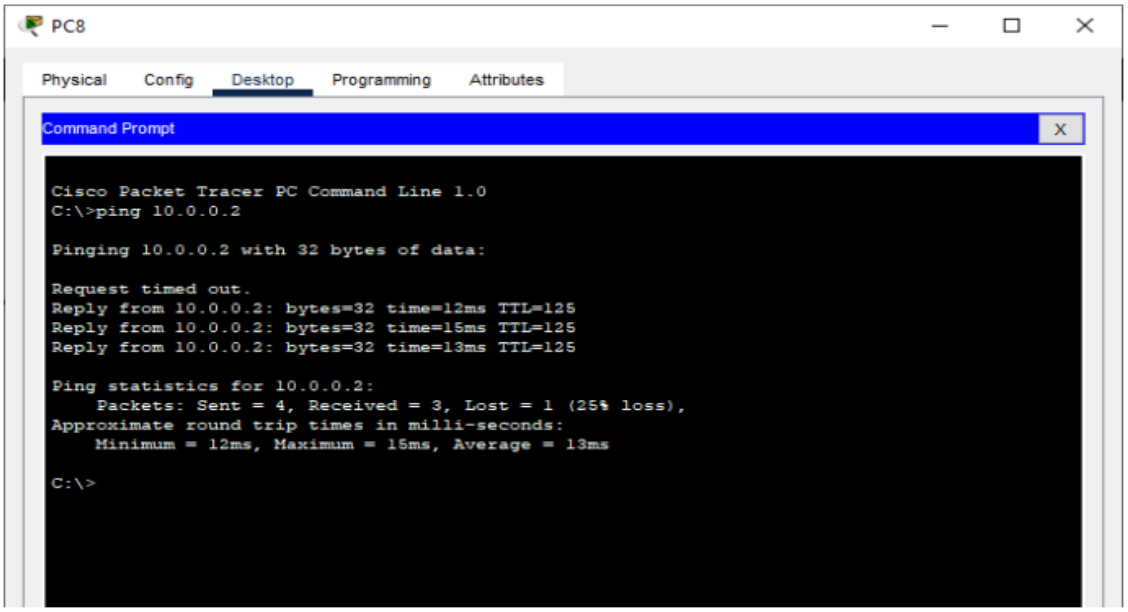
```
Router(config)#
Router(config)#router ospf 1
Router(config-router)#
Router(config-router)#network 30.0.0.0 0.0.0.255 area 1
Router(config-router)#network 50.0.0.0 0.0.0.255 area 1
Router(config-router)# exit
Router(config)#
```

Checking the connectivity by using the ping command

i) Pinging PC8 (ip address 10.30.0.4) from PC1



ii) Pinging PC0 (ip address 10.10.0.2) from PC8



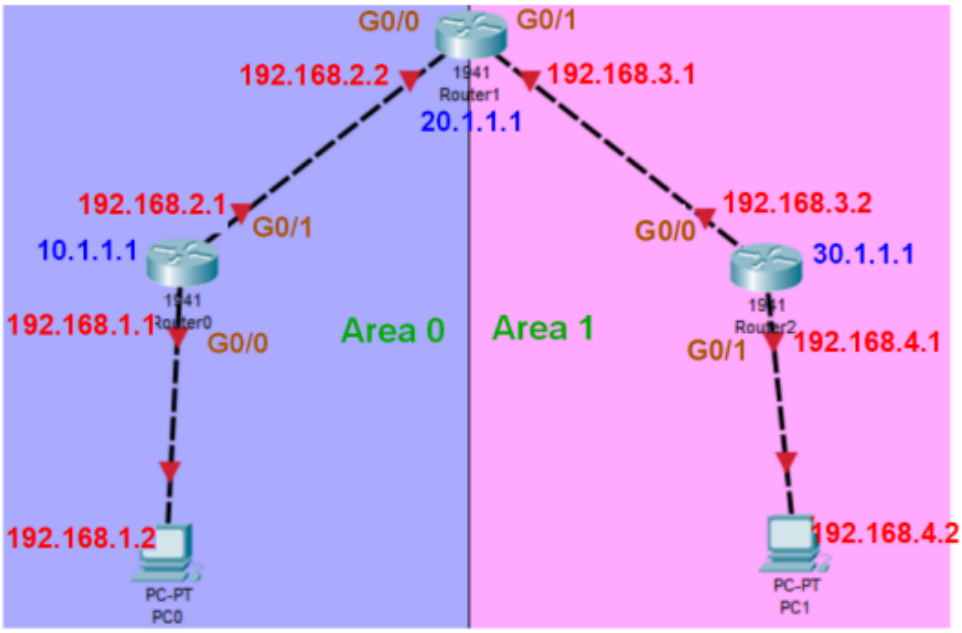
Result:

Hence the OSPF has been studied and verified through the given network

Scan the following QR-code for the video demostaration of the practical Simple OSPF

Part b) Multi-area OSPF

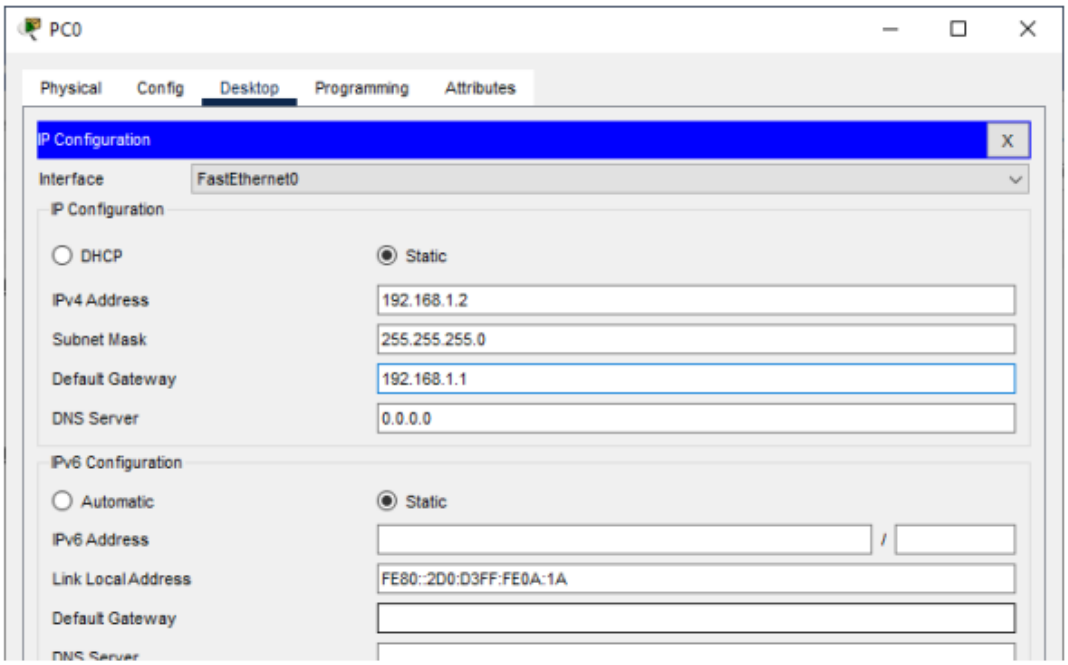
Consider the following topology



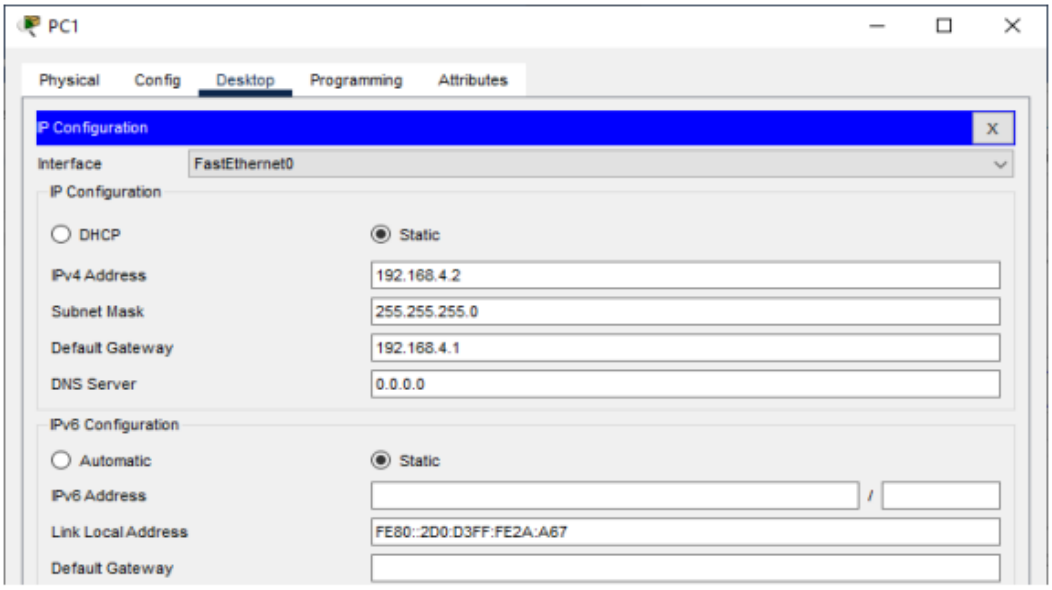
We use the following IP addresses for the given topology

Host	IP Address	Network Address	Network Mask	Wild Card Mask
PC 0	192.168.1.2	192.168.1.0	255.255.255.0	0.0.0.255
PC 1	192.168.4.2	192.168.4.0	255.255.255.0	0.0.0.255
ROUTER 0	G0/0 192.168.1.1	192.168.1.0	255.255.255.0	0.0.0.255
	G0/1 192.168.2.1	192.168.2.0	255.255.255.0	0.0.0.255
	LOOPBACK 10.1.1.1	10.0.0.0	255.0.0.0	0.255.255.255
ROUTER 1	G0/0 192.168.2.2	192.168.2.0	255.255.255.0	0.0.0.255
	G0/1 192.168.3.1	192.168.3.0	255.255.255.0	0.0.0.255
	LOOPBACK 20.1.1.1	20.0.0.0	255.0.0.0	0.255.255.255
ROUTER 2	G0/0 192.168.3.2	192.168.3.0	255.255.255.0	0.0.0.255
	G0/1 192.168.4.1	192.168.4.0	255.255.255.0	0.0.0.255
	LOOPBACK 30.1.1.1	30.0.0.0	255.0.0.0	0.255.255.255

Configuring PC0



Configuring PC1



Configuring Router0

```
Router>enable
Router#
Router#configure terminal
Router(config)#
Router(config)#interface gigabitEthernet 0/0
Router(config-if)#
Router(config-if)#ip address 192.168.1.1 255.255.255.0
Router(config-if)#no shutdown
Router(config-if)#exit
Router(config)#
Router(config)#interface gigabitEthernet 0/1
Router(config-if)#
Router(config-if)#ip address 192.168.2.1 255.255.255.0
Router(config-if)#no shutdown
Router(config-if)#exit
Router(config)#
Router(config)#interface loopback 0
Router(config-if)#
Router(config-if)#ip address 10.1.1.1 255.0.0.0
Router(config-if)#no shutdown
Router(config-if)#exit
Router(config)#
```

Configuring Router1

```
Router>enable
Router#
Router#configure terminal
Router(config)#
Router(config)#interface gigabitEthernet 0/0
Router(config-if)#
Router(config-if)#ip address 192.168.2.2 255.255.255.0
Router(config-if)#no shutdown
Router(config-if)#exit
Router(config)#
Router(config)#interface gigabitEthernet 0/1
Router(config-if)#
Router(config-if)#ip address 192.168.3.1 255.255.255.0
Router(config-if)#no shutdown
Router(config-if)#exit
Router(config)#
Router(config)#interface loopback 0
Router(config-if)#
Router(config-if)#ip address 10.1.1.1 255.0.0.0
Router(config-if)#no shutdown
Router(config-if)#exit
Router(config)#
Router(config)#interface loopback 0
Router(config-if)#
Router(config-if)#ip address 20.1.1.1 255.0.0.0
Router(config-if)#no shutdown
Router(config-if)#exit
Router(config)#
```

Configuring Router2

```
Router>enable
Router#
Router#configure terminal
Router(config)#
Router(config)#interface gigabitEthernet 0/0
Router(config-if)#
Router(config-if)#ip address 192.168.3.2 255.255.255.0
Router(config-if)#no shutdown
Router(config-if)#
Router(config-if)#exit
Router(config)#
Router(config)#interface gigabitEthernet 0/1
Router(config-if)#
Router(config-if)#ip address 192.168.4.1 255.255.255.0
Router(config-if)#no shutdown
Router(config-if)#exit
Router(config)#
```

```
Router(config)#interface loopback 0
Router(config-if)#
Router(config-if)#ip address 30.1.1.1 255.0.0.0
Router(config-if)#no shutdown
Router(config-if)#exit
Router(config)#
```

Configuring Router0 for OSPF

```
Router(config)#
Router(config)#router ospf 1
Router(config-router)#
Router(config-router)#network 192.168.1.0 255.255.255.0 area 0
Router(config-router)#network 192.168.2.0 255.255.255.0 area 0
Router(config-router)#exit
```

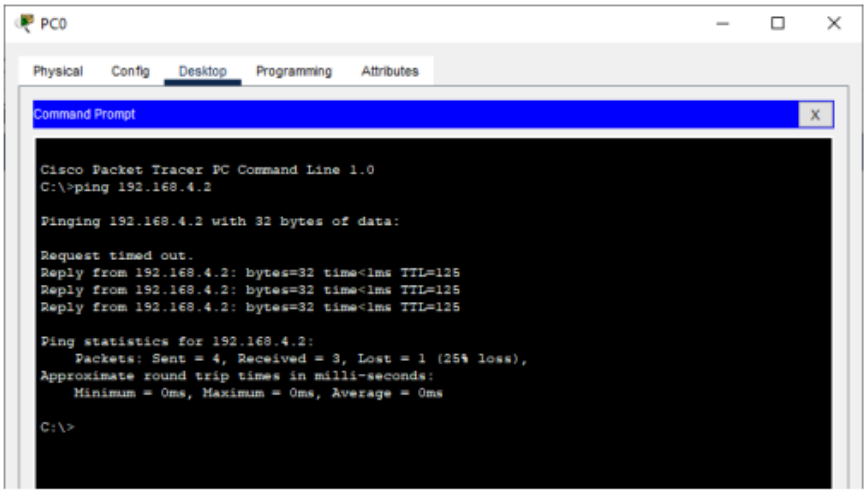
Configuring Router1 for OSPF

```
Router(config)#
Router(config)#router ospf 1
Router(config-router)#
Router(config-router)#network 192.168.2.0 255.255.255.0 area 0
Router(config-router)#network 192.168.3.0 255.255.255.0 area 1
Router(config-router)#exit
```

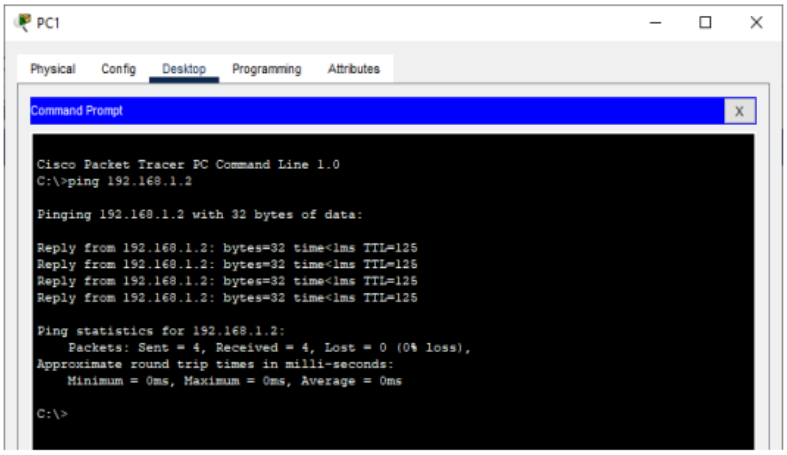
Configuring Router2 for OSPF

```
Router(config)#
Router(config)#router ospf 1
Router(config-router)#
Router(config-router)#network 192.168.3.0 255.255.255.0 area 1
Router(config-router)#network 192.168.4.0 255.255.255.0 area 1
Router(config-router)#exit
```

Verify the connectivity (ping PC1 from PC0)



Verify the connectivity (ping PC0 from PC1)



Hence OSPF with Multiple Areas (area 0 and area 1) was successfully configured and verified

Scan the following QR-code for the video demostaration of the practical OSPF with Multiple Areas