Practical-4

AIM: Understand and identify Layer-2 functionality.

Tools required:
1. Desktop Computer
2. Cisco Packet Tracer

Simulate different scenarios given below in Cisco packet tracker and fill up table.

Note: While applying IP address, student need to allocate IP address as per his/her student ID. For Example, if student ID is 20ce005 then IP address allocation for first network should start with 5.0.0.0. For subsequent network, it should start with ID+1 i.e. 6.0.0.0, 7.0.0.0. and so on.

Submission: After writing answer into this word document, Student need to change name to his ID followed by practical number. Ex 20ce005_Pr1.docx. Upload on assignment segment.
Rubrics: Nicely drafted document with clarity in answers leads to full marks. Otherwise, submission carries proportional mark.
Recommended to type, avoid copy-past to increase your typing skill.

Scenario 1: Let's assume that there are three PCs in network. All are connected with layer 2 device switch. All are assigned with the shown IP Addresses in the figure 4.1. All ARP tables are empty. Now, PC0 wants to send some data to PC2. How communication will do?

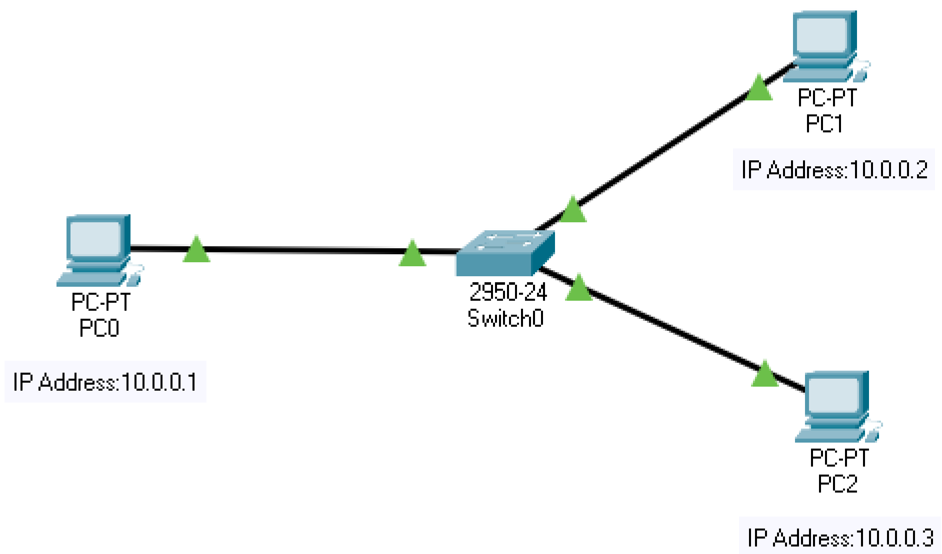


Figure 4.1 Scenario 1

Conclusion: The sender is a host and wants to send a packet to another host on the same network.

- Uses ARP to find another host's physical address

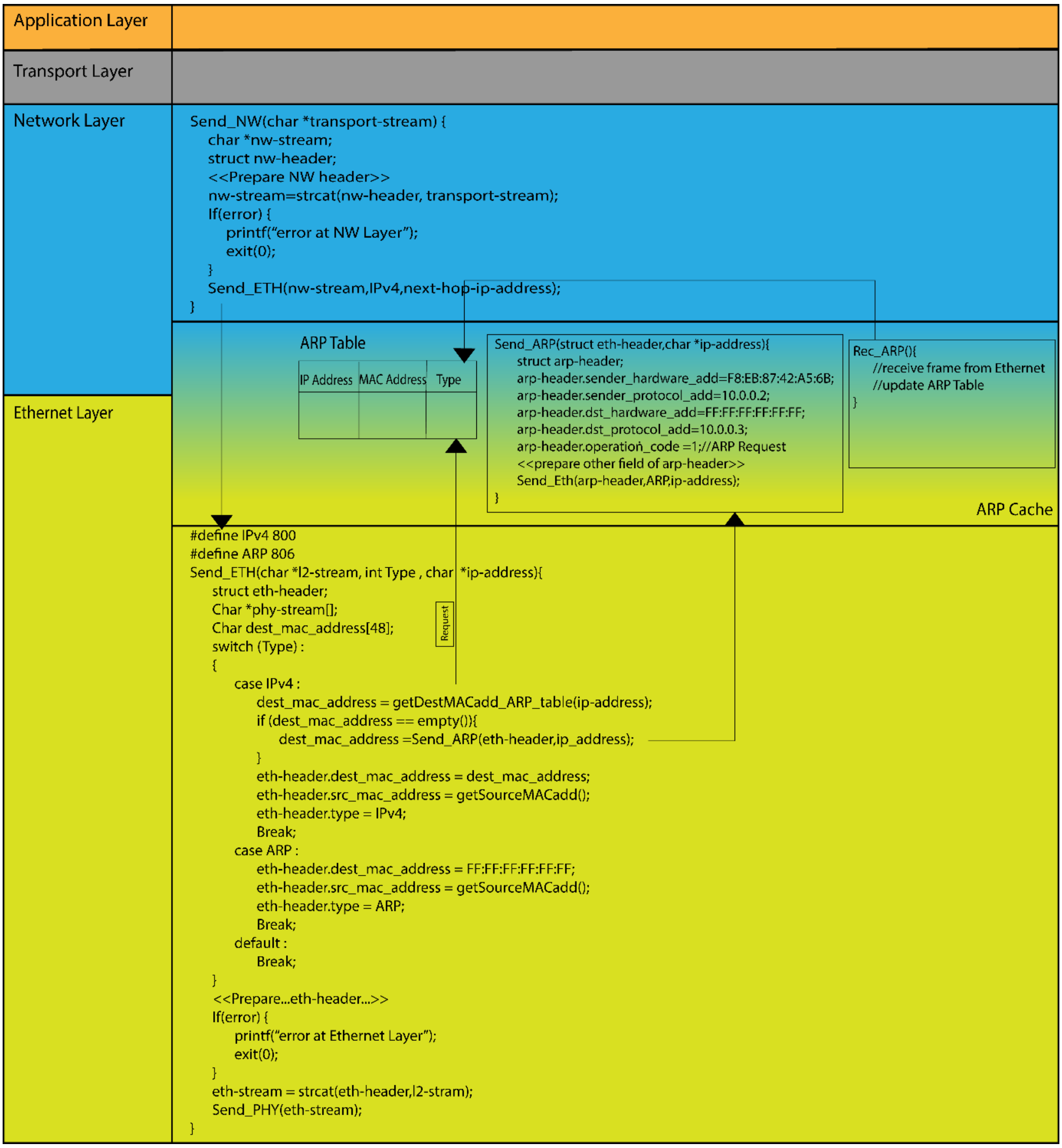


Figure 4.2 TCP/IP stack

Address Resolution Protocol (ARP):

Most of the computer programs/applications use logical address (IP address) to send/receive messages, however, the actual communication happens over the physical address (MAC address) i.e., from layer 2 of the TCP/IP model. So, our mission is to get the destination MAC address which helps in communicating with other devices. This is where ARP comes into the picture, its functionality is to translate IP address to physical addresses.

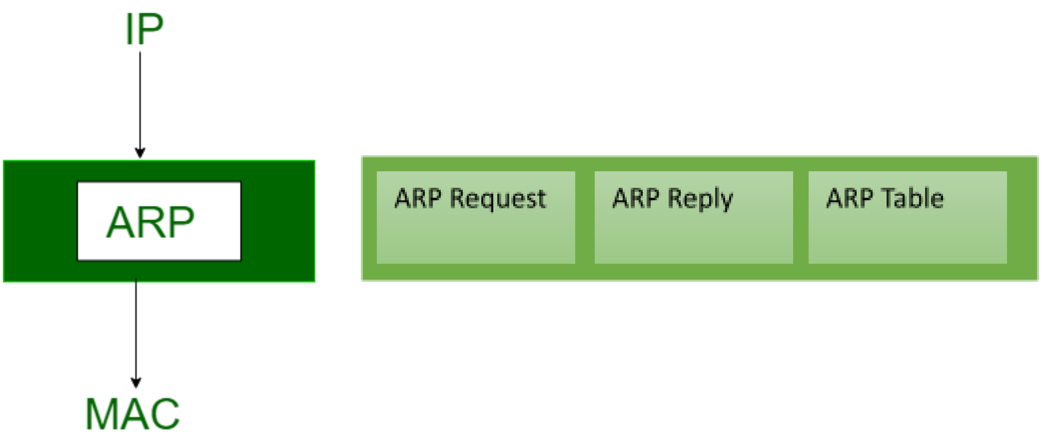


Figure 4.3 ARP Maps IP address to MAC address

The acronym ARP stands for Address Resolution Protocol which is one of the most important protocols of the Network layer in the OSI model.
ARP finds the hardware address, also known as Media Access Control (MAC) address, of a host from its known IP address.

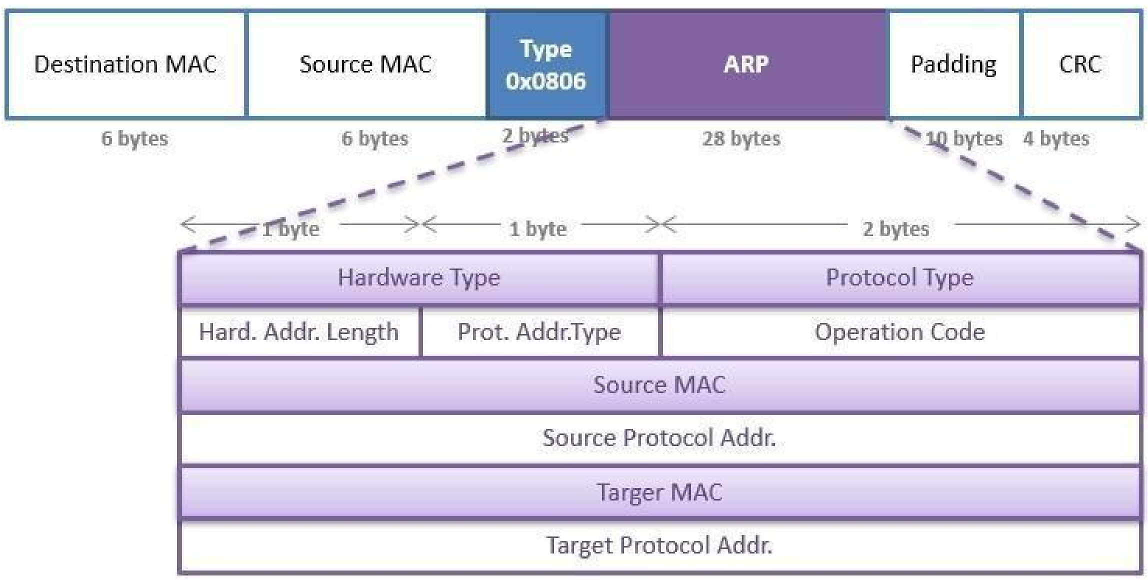


Figure 4.4 ARP header

Scenario 2: With respect to given topology shown in figure 4.5, IP addresses are assigned to all the PCs. Initial ARP tables are empty. Now, PC0 wants to send data to PC3. Write down the step how communication will take place.

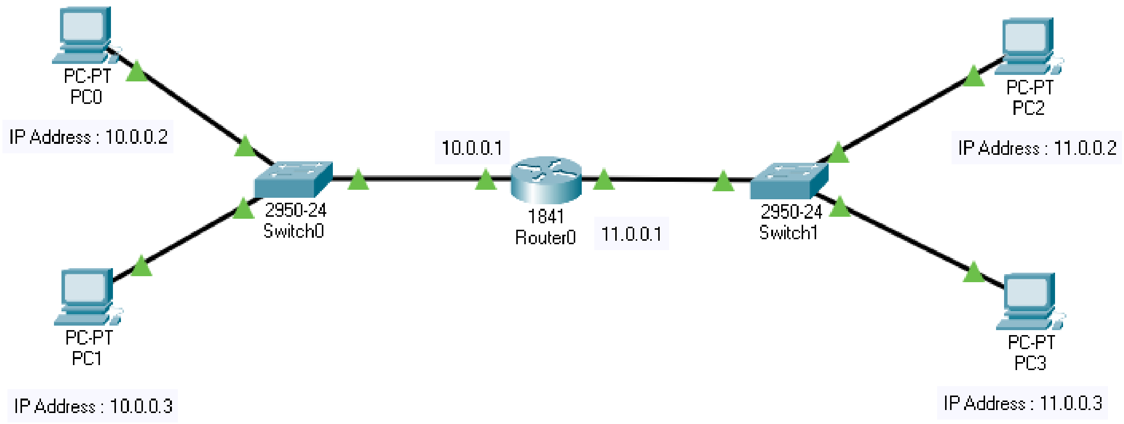


Figure 4.5 Scenario 2

Conclusion: The sender is a host and wants to send a packet to another host on another network.

- The sender looks at its routing table.
- Find the IP address of the next-hop (router) for this destination.
- Use ARP to find the router's physical address

Scenario 3: With respect to given topology shown in figure 4.6, IP addresses are assigned to all the PCs. Initial ARP tables are empty. Now, Packet is received by router 1 with destination IP address 12.0.0.2. Write down the step how communication will take place.

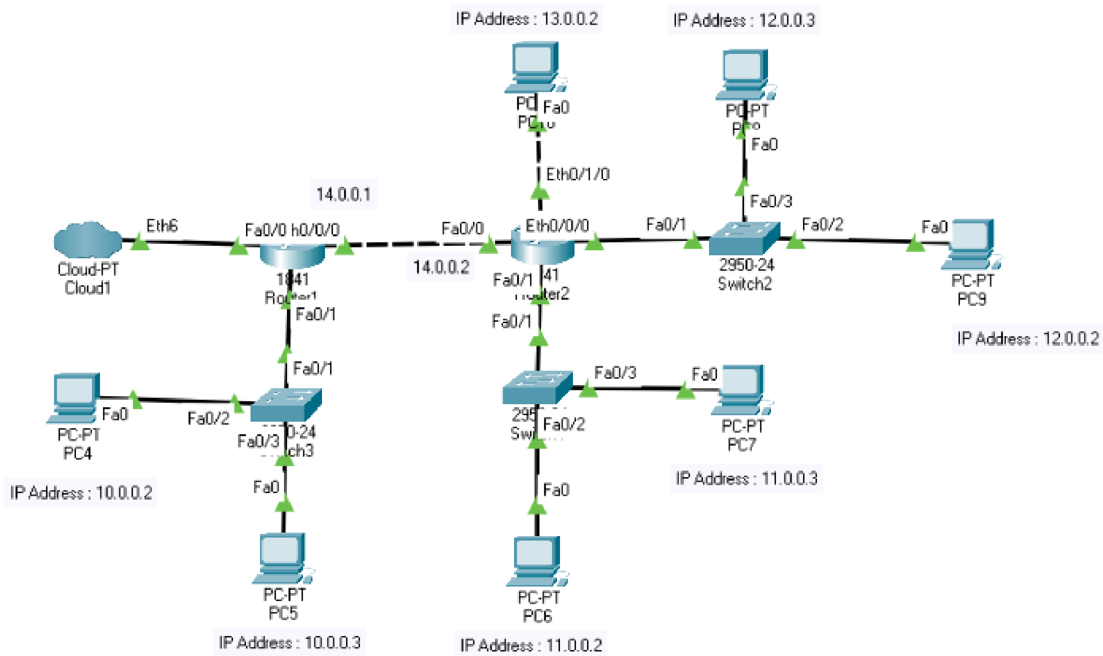


Figure 4.6 Scenario 3

Conclusion: the sender is a router and received a datagram destined for a host on another network.

- The router checks its routing table.
- Find the IP address of the next router.
- Use ARP to find the next router's physical address.

Question 1: List out Different network devices and give a small introduction of the same.

Ans : Routers allow packets to be transmitted to their destinations by monitoring the sea of networking devices interconnected with different network topologies.

The **hub** link various networking devices. A network also functions as amplification by amplifying signals that deteriorate over cables after long distances

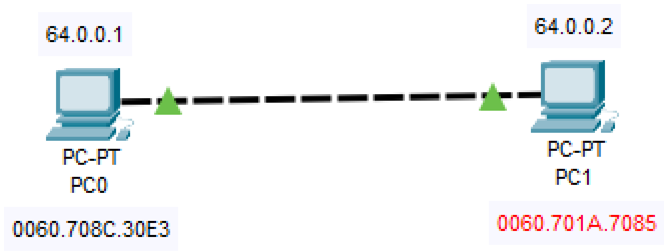
Question 2: write down difference between HUB and SWITCH.

Ans.

Hub	Switch
To connect a network of personal computers together	**To connect multiple devices and ports.**
Electrical signal or bits	**Frames or Packets**

Exercise-1

Redraw above diagram which includes IP address and MAC address. Take IP address and MAC address as per your knowledge.



Ping from PC0 to PC1 and vice versa and insert snap of output here.

```
C:\>ping 64.0.0.2

Pinging 64.0.0.2 with 32 bytes of data:

Reply from 64.0.0.2: bytes=32 time<1ms TTL=128
Reply from 64.0.0.2: bytes=32 time<1ms TTL=128
Reply from 64.0.0.2: bytes=32 time<1ms TTL=128
Reply from 64.0.0.2: bytes=32 time<1ms TTL=128

Ping statistics for 64.0.0.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

```
C:\>ping 64.0.0.1

Pinging 64.0.0.1 with 32 bytes of data:

Reply from 64.0.0.1: bytes=32 time<1ms TTL=128
Reply from 64.0.0.1: bytes=32 time<1ms TTL=128
Reply from 64.0.0.1: bytes=32 time<1ms TTL=128
Reply from 64.0.0.1: bytes=32 time<1ms TTL=128

Ping statistics for 64.0.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

Write down ARP table of PC0 and PC1.

```
C:\>arp -a
  Internet Address      Physical Address      Type
  64.0.0.2              0060.701a.7085        dynamic
```

```
C:\>arp -a
  Internet Address      Physical Address      Type
  64.0.0.1              0060.708c.30e3        dynamic
```

Fill ARP table entry of PC0

IP address	MAC Address
Internet Address Physical	64.0.0.1
Address	0060.708c.30e3
Type	dynamic

Fill ARP table entry of PC1

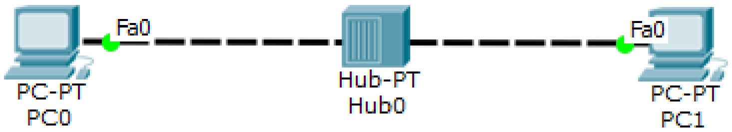
IP address	MAC Address
Internet Address Physical	64.0.0.2
Address	0060.701a.7085
Type	dynamic

Questions:

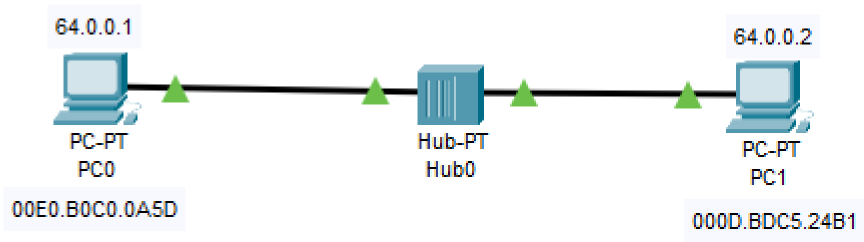
1. What does ARP table contain?
   Ans : IP address and MAC address
2. Why there is need of ARP table?
   Ans : To check that the device are ping.
3. What is topology name of exercise-1?
   Ans : Point to Point Topology.
4. What is relation of IP address with MAC address?
   Ans : Yes, there is relation between IP and MAC address they are given to a specific machine and are used to ping or search of the machines.
5. Can we change MAC Address of machine?
   Ans : No.
6. Can we change IP address of machine?

   Ans : Yes

Exercise-2 :



Redraw above diagram which includes IP address and MAC address. Take IP address and MAC address as per your knowledge.



Ping from PC0 to PC1 and vice versa and insert snap of output here.

```
C:\>ping 64.0.0.2

Pinging 64.0.0.2 with 32 bytes of data:

Reply from 64.0.0.2: bytes=32 time<1ms TTL=128
Reply from 64.0.0.2: bytes=32 time<1ms TTL=128
Reply from 64.0.0.2: bytes=32 time<1ms TTL=128
Reply from 64.0.0.2: bytes=32 time<1ms TTL=128

Ping statistics for 64.0.0.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

```
C:\>ping 64.0.0.1

Pinging 64.0.0.1 with 32 bytes of data:

Reply from 64.0.0.1: bytes=32 time<1ms TTL=128
Reply from 64.0.0.1: bytes=32 time<1ms TTL=128
Reply from 64.0.0.1: bytes=32 time<1ms TTL=128
Reply from 64.0.0.1: bytes=32 time<1ms TTL=128

Ping statistics for 64.0.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

Write down ARP table of PC0 and PC1. Write down switch table of Hub0.
Fill ARP table entry of PC0

```
C:\>arp -a
  Internet Address      Physical Address      Type
  64.0.0.2              000d.bdc5.24b1        dynamic
```

```
C:\>arp -a
  Internet Address      Physical Address      Type
  64.0.0.1              00e0.b0c0.0a5d        dynamic
```

IP address	MAC Address
Internet Address	64.0.0.2
Physical Address	000d.bdc5.24b1
Type	Dynamic

Fill ARP table entry of PC1

IP address	MAC Address
Internet Address	64.0.0.1
Physical Address	00e0.b0c0.0a5d
Type	Dynamic

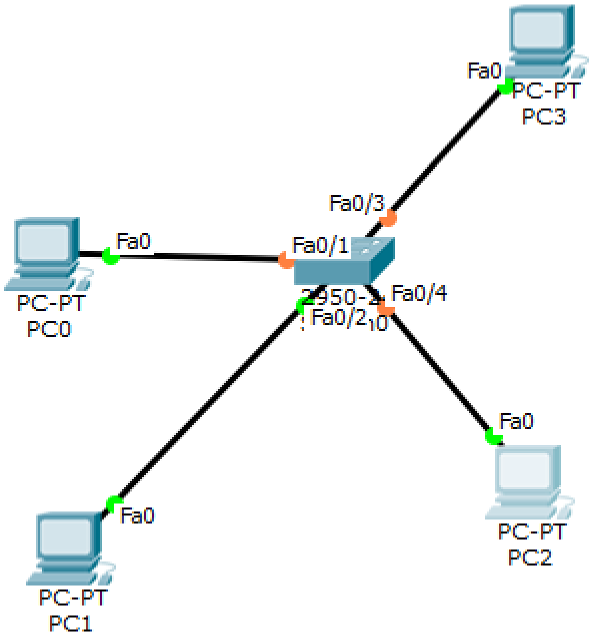
Fill Switch table entry of Hub0

Ans : There is no MAC Address for Hub0.

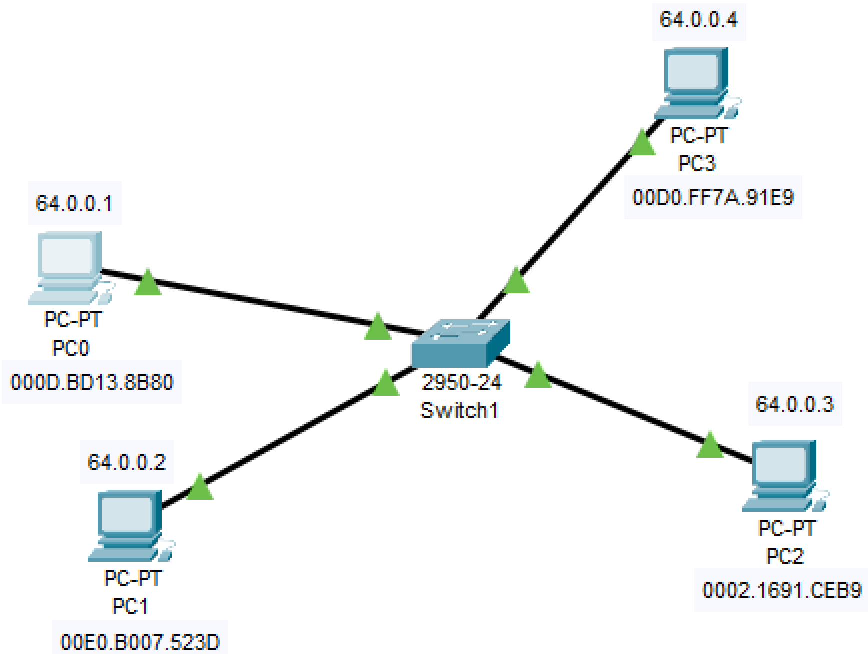
Questions:

1. What is functionality of Hub?
   Ans : Hubs connect multiple computer networking devices together
2. Does hub have IP address?
   Ans : No
3. Does hub have switch (ARP) table?
   Ans : Yes.
4. What is topology of exercise-2?
   Ans : Linear topology.

Exercise-3



Redraw above diagram which includes IP address and MAC address. Take IP address and MAC address as per your knowledge.



Ping from PC0,PC1,PC2 and PC3 respectively and insert snap of output here.

From PC0

```
C:\>ping 64.0.0.2

Pinging 64.0.0.2 with 32 bytes of data:

Reply from 64.0.0.2: bytes=32 time<1ms TTL=128
Reply from 64.0.0.2: bytes=32 time<1ms TTL=128
Reply from 64.0.0.2: bytes=32 time<1ms TTL=128
Reply from 64.0.0.2: bytes=32 time<1ms TTL=128

Ping statistics for 64.0.0.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

```
C:\>ping 64.0.0.3

Pinging 64.0.0.3 with 32 bytes of data:

Reply from 64.0.0.3: bytes=32 time<1ms TTL=128
Reply from 64.0.0.3: bytes=32 time<1ms TTL=128
Reply from 64.0.0.3: bytes=32 time<1ms TTL=128
Reply from 64.0.0.3: bytes=32 time=1ms TTL=128

Ping statistics for 64.0.0.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 1ms, Average = 0ms
```

```
C:\>ping 64.0.0.4

Pinging 64.0.0.4 with 32 bytes of data:

Reply from 64.0.0.4: bytes=32 time<1ms TTL=128
Reply from 64.0.0.4: bytes=32 time<1ms TTL=128
Reply from 64.0.0.4: bytes=32 time<1ms TTL=128
Reply from 64.0.0.4: bytes=32 time<1ms TTL=128

Ping statistics for 64.0.0.4:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

From PC1

```
C:\>ping 64.0.0.1

Pinging 64.0.0.1 with 32 bytes of data:

Reply from 64.0.0.1: bytes=32 time<1ms TTL=128
Reply from 64.0.0.1: bytes=32 time<1ms TTL=128
Reply from 64.0.0.1: bytes=32 time<1ms TTL=128
Reply from 64.0.0.1: bytes=32 time=7ms TTL=128

Ping statistics for 64.0.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 7ms, Average = 1ms
```

```
C:\>ping 64.0.0.3

Pinging 64.0.0.3 with 32 bytes of data:

Reply from 64.0.0.3: bytes=32 time<1ms TTL=128
Reply from 64.0.0.3: bytes=32 time<1ms TTL=128
Reply from 64.0.0.3: bytes=32 time<1ms TTL=128
Reply from 64.0.0.3: bytes=32 time=1ms TTL=128

Ping statistics for 64.0.0.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 1ms, Average = 0ms
```

```
C:\>ping 64.0.0.4

Pinging 64.0.0.4 with 32 bytes of data:

Reply from 64.0.0.4: bytes=32 time<1ms TTL=128
Reply from 64.0.0.4: bytes=32 time<1ms TTL=128
Reply from 64.0.0.4: bytes=32 time<1ms TTL=128
Reply from 64.0.0.4: bytes=32 time<1ms TTL=128

Ping statistics for 64.0.0.4:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

From PC2

```
C:\>ping 64.0.0.1

Pinging 64.0.0.1 with 32 bytes of data:

Reply from 64.0.0.1: bytes=32 time<1ms TTL=128
Reply from 64.0.0.1: bytes=32 time<1ms TTL=128
Reply from 64.0.0.1: bytes=32 time<1ms TTL=128
Reply from 64.0.0.1: bytes=32 time<1ms TTL=128

Ping statistics for 64.0.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

```
C:\>ping 64.0.0.2

Pinging 64.0.0.2 with 32 bytes of data:

Reply from 64.0.0.2: bytes=32 time<1ms TTL=128
Reply from 64.0.0.2: bytes=32 time=5ms TTL=128
Reply from 64.0.0.2: bytes=32 time<1ms TTL=128
Reply from 64.0.0.2: bytes=32 time<1ms TTL=128

Ping statistics for 64.0.0.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 5ms, Average = 1ms
```

```
C:\>ping 64.0.0.4

Pinging 64.0.0.4 with 32 bytes of data:

Reply from 64.0.0.4: bytes=32 time<1ms TTL=128
Reply from 64.0.0.4: bytes=32 time<1ms TTL=128
Reply from 64.0.0.4: bytes=32 time<1ms TTL=128
Reply from 64.0.0.4: bytes=32 time<1ms TTL=128

Ping statistics for 64.0.0.4:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

From PC3

```
C:\>ping 64.0.0.1

Pinging 64.0.0.1 with 32 bytes of data:

Reply from 64.0.0.1: bytes=32 time<1ms TTL=128
Reply from 64.0.0.1: bytes=32 time<1ms TTL=128
Reply from 64.0.0.1: bytes=32 time<1ms TTL=128
Reply from 64.0.0.1: bytes=32 time<1ms TTL=128

Ping statistics for 64.0.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

```
C:\>ping 64.0.0.2

Pinging 64.0.0.2 with 32 bytes of data:

Reply from 64.0.0.2: bytes=32 time=1ms TTL=128
Reply from 64.0.0.2: bytes=32 time<1ms TTL=128
Reply from 64.0.0.2: bytes=32 time<1ms TTL=128
Reply from 64.0.0.2: bytes=32 time<1ms TTL=128

Ping statistics for 64.0.0.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 1ms, Average = 0ms
```

```
C:\>ping 64.0.0.3

Pinging 64.0.0.3 with 32 bytes of data:

Reply from 64.0.0.3: bytes=32 time<1ms TTL=128
Reply from 64.0.0.3: bytes=32 time<1ms TTL=128
Reply from 64.0.0.3: bytes=32 time<1ms TTL=128
Reply from 64.0.0.3: bytes=32 time<1ms TTL=128

Ping statistics for 64.0.0.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

Write down ARP tables of PC0,PC1,PC2 and PC3. Write down switch table of Switch0.

PC0

```
C:\>arp -a
  Internet Address      Physical Address      Type
  64.0.0.2              00e0.b007.523d        dynamic
  64.0.0.3              0002.1691.ceb9        dynamic
  64.0.0.4              00d0.ff7a.91e9        dynamic
```

PC1

```
C:\>arp -a
  Internet Address      Physical Address      Type
  64.0.0.1              000d.bd13.8b80        dynamic
  64.0.0.3              0002.1691.ceb9        dynamic
  64.0.0.4              00d0.ff7a.91e9        dynamic
```

PC2

```
C:\>arp -a
  Internet Address      Physical Address      Type
  64.0.0.1              000d.bd13.8b80        dynamic
  64.0.0.2              00e0.b007.523d        dynamic
  64.0.0.4              00d0.ff7a.91e9        dynamic
```

PC3

```
C:\>arp -a
  Internet Address      Physical Address      Type
  64.0.0.1              000d.bd13.8b80        dynamic
  64.0.0.2              00e0.b007.523d        dynamic
  64.0.0.3              0002.1691.ceb9        dynamic
```

Fill ARP table entry of PC0

IP address	MAC Address
64.0.0.2	00e0.b007.523d
64.0.0.3	0002.1691.ceb9
64.0.0.4	00d0.ff7a.91e9

Fill ARP table entry of PC1

IP address	MAC Address
64.0.0.1	000d.bd13.8b80
64.0.0.3	0002.1691.ceb9
64.0.0.4	00d0.ff7a.91e9

Fill ARP table entry of PC2

IP address	MAC Address
64.0.0.1	000d.bd13.8b80
64.0.0.2	00e0.b007.523d
64.0.0.4	00d0.ff7a.91e9

Fill ARP table entry of PC3

IP address	MAC Address
64.0.0.1	000d.bd13.8b80

64.0.0.2	00e0.b007.523d
64.0.0.3	0002.1691.ceb9

Fill Switch table entry of Switch0

1	0002.1691.CEB9	FastEthernet0/3
1	000D.BD13.8B80	FastEthernet0/1
1	00D0.FF7A.91E9	FastEthernet0/4
1	00E0.B007.523D	FastEthernet0/2

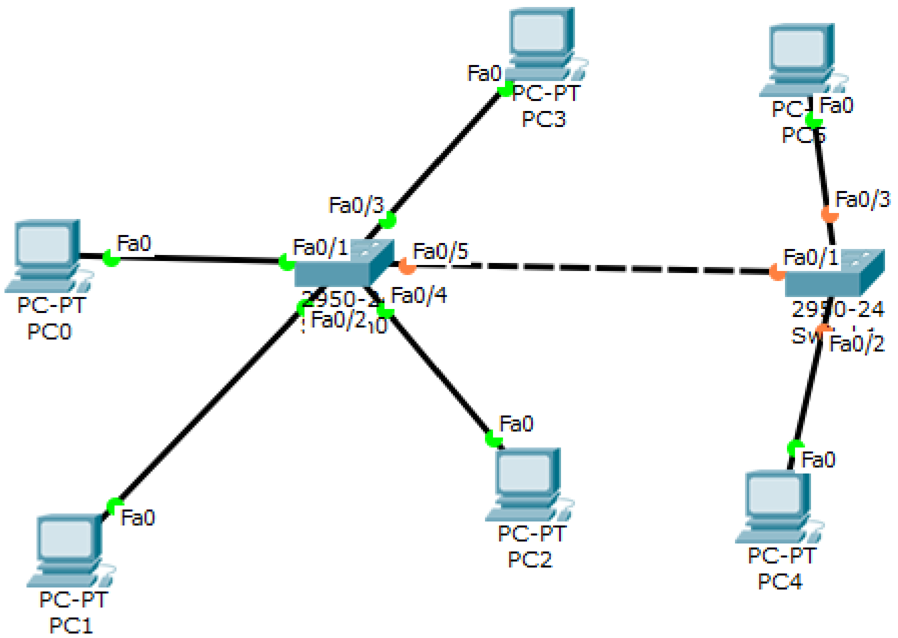
MAC Address	Ethernet port no
0002.1691.CEB9	FastEthernet0/3
000D.BD13.8D80	FastEthernet0/1
00D0.FF74.91E9	FastEthernet0/4
00E0.B007.523D	FastEthernet0/2

Questions:

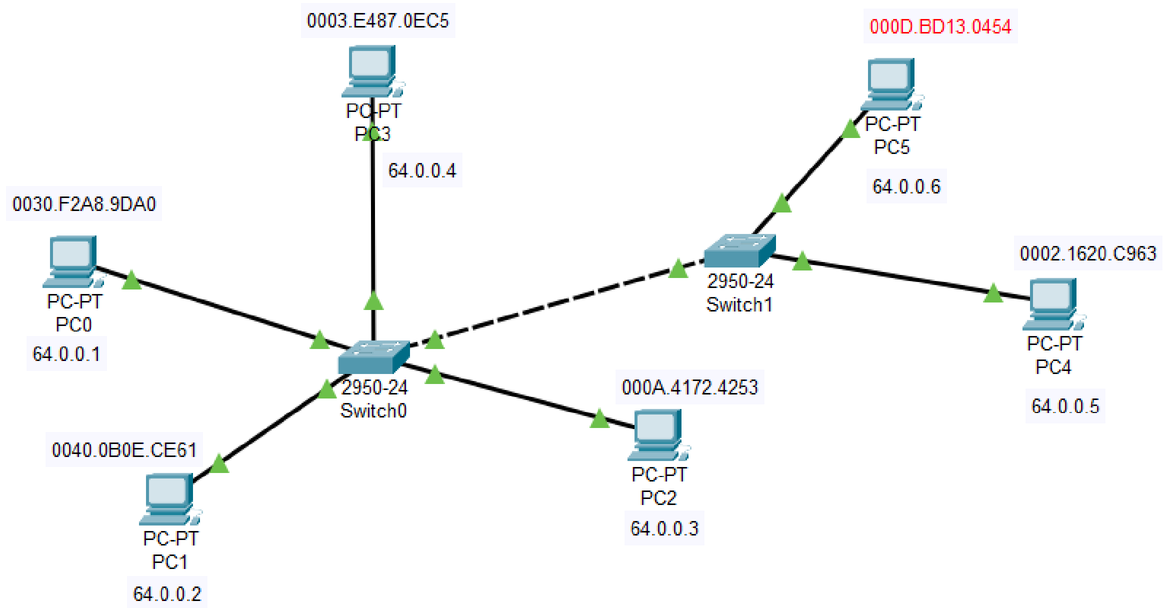
1. What is functionality of switch?
   Ans : network switch connects devices in a network to each other, and allows them to 'talk' by exchanging data packets.
2. Does switch have IP address?
   Ans : No.
3. Does switch have switch (ARP) table?
   Ans : Yes.
4. What is the topology name of exercise-3?

   Ans : It is star topology.

Exercise-4



Redraw above diagram which includes IP address and MAC address. Take IP address and MAC address as per your knowledge.



Ping from all given PCs respectively and insert snap of output here.

From PC0 :

```
C:\>ping 64.0.0.2

Pinging 64.0.0.2 with 32 bytes of data:

Reply from 64.0.0.2: bytes=32 time<1ms TTL=128
Reply from 64.0.0.2: bytes=32 time<1ms TTL=128
Reply from 64.0.0.2: bytes=32 time<1ms TTL=128
Reply from 64.0.0.2: bytes=32 time<1ms TTL=128

Ping statistics for 64.0.0.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>ping 64.0.0.3

Pinging 64.0.0.3 with 32 bytes of data:

Reply from 64.0.0.3: bytes=32 time<1ms TTL=128
Reply from 64.0.0.3: bytes=32 time<1ms TTL=128
Reply from 64.0.0.3: bytes=32 time<1ms TTL=128
Reply from 64.0.0.3: bytes=32 time<1ms TTL=128

Ping statistics for 64.0.0.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>ping 64.0.0.4

Pinging 64.0.0.4 with 32 bytes of data:

Reply from 64.0.0.4: bytes=32 time<1ms TTL=128
Reply from 64.0.0.4: bytes=32 time<1ms TTL=128
Reply from 64.0.0.4: bytes=32 time<1ms TTL=128
Reply from 64.0.0.4: bytes=32 time<1ms TTL=128

Ping statistics for 64.0.0.4:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

```
C:\>ping 64.0.0.5

Pinging 64.0.0.5 with 32 bytes of data:

Reply from 64.0.0.5: bytes=32 time=1ms TTL=128
Reply from 64.0.0.5: bytes=32 time<1ms TTL=128
Reply from 64.0.0.5: bytes=32 time<1ms TTL=128
Reply from 64.0.0.5: bytes=32 time<1ms TTL=128

Ping statistics for 64.0.0.5:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 1ms, Average = 0ms

C:\>ping 64.0.0.6

Pinging 64.0.0.6 with 32 bytes of data:

Reply from 64.0.0.6: bytes=32 time<1ms TTL=128
Reply from 64.0.0.6: bytes=32 time<1ms TTL=128
Reply from 64.0.0.6: bytes=32 time<1ms TTL=128
Reply from 64.0.0.6: bytes=32 time<1ms TTL=128

Ping statistics for 64.0.0.6:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

From PC1 :

```
C:\>ping 64.0.0.1

Pinging 64.0.0.1 with 32 bytes of data:

Reply from 64.0.0.1: bytes=32 time<1ms TTL=128
Reply from 64.0.0.1: bytes=32 time<1ms TTL=128
Reply from 64.0.0.1: bytes=32 time=7ms TTL=128
Reply from 64.0.0.1: bytes=32 time<1ms TTL=128

Ping statistics for 64.0.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 7ms, Average = 1ms

C:\>ping 64.0.0.3

Pinging 64.0.0.3 with 32 bytes of data:

Reply from 64.0.0.3: bytes=32 time=1ms TTL=128
Reply from 64.0.0.3: bytes=32 time<1ms TTL=128
Reply from 64.0.0.3: bytes=32 time<1ms TTL=128
Reply from 64.0.0.3: bytes=32 time<1ms TTL=128

Ping statistics for 64.0.0.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 1ms, Average = 0ms

C:\>ping 64.0.0.4

Pinging 64.0.0.4 with 32 bytes of data:

Reply from 64.0.0.4: bytes=32 time<1ms TTL=128
Reply from 64.0.0.4: bytes=32 time<1ms TTL=128
Reply from 64.0.0.4: bytes=32 time<1ms TTL=128
Reply from 64.0.0.4: bytes=32 time<1ms TTL=128

Ping statistics for 64.0.0.4:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

```
C:\>ping 64.0.0.5

Pinging 64.0.0.5 with 32 bytes of data:

Reply from 64.0.0.5: bytes=32 time=1ms TTL=128
Reply from 64.0.0.5: bytes=32 time=1ms TTL=128
Reply from 64.0.0.5: bytes=32 time=1ms TTL=128
Reply from 64.0.0.5: bytes=32 time<1ms TTL=128

Ping statistics for 64.0.0.5:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 1ms, Average = 0ms

C:\>ping 64.0.0.6

Pinging 64.0.0.6 with 32 bytes of data:

Reply from 64.0.0.6: bytes=32 time<1ms TTL=128
Reply from 64.0.0.6: bytes=32 time<1ms TTL=128
Reply from 64.0.0.6: bytes=32 time<1ms TTL=128
Reply from 64.0.0.6: bytes=32 time<1ms TTL=128

Ping statistics for 64.0.0.6:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

From PC2 :

```
C:\>ping 64.0.0.1

Pinging 64.0.0.1 with 32 bytes of data:

Reply from 64.0.0.1: bytes=32 time<1ms TTL=128
Reply from 64.0.0.1: bytes=32 time<1ms TTL=128
Reply from 64.0.0.1: bytes=32 time<1ms TTL=128
Reply from 64.0.0.1: bytes=32 time<1ms TTL=128

Ping statistics for 64.0.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>ping 64.0.0.2

Pinging 64.0.0.2 with 32 bytes of data:

Reply from 64.0.0.2: bytes=32 time<1ms TTL=128
Reply from 64.0.0.2: bytes=32 time<1ms TTL=128
Reply from 64.0.0.2: bytes=32 time<1ms TTL=128
Reply from 64.0.0.2: bytes=32 time=1ms TTL=128

Ping statistics for 64.0.0.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 1ms, Average = 0ms

C:\>ping 64.0.0.4

Pinging 64.0.0.4 with 32 bytes of data:

Reply from 64.0.0.4: bytes=32 time<1ms TTL=128
Reply from 64.0.0.4: bytes=32 time<1ms TTL=128
Reply from 64.0.0.4: bytes=32 time<1ms TTL=128
Reply from 64.0.0.4: bytes=32 time<1ms TTL=128

Ping statistics for 64.0.0.4:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

```
C:\>ping 64.0.0.5

Pinging 64.0.0.5 with 32 bytes of data:

Reply from 64.0.0.5: bytes=32 time<1ms TTL=128
Reply from 64.0.0.5: bytes=32 time<1ms TTL=128
Reply from 64.0.0.5: bytes=32 time<1ms TTL=128
Reply from 64.0.0.5: bytes=32 time<1ms TTL=128

Ping statistics for 64.0.0.5:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>ping 64.0.0.6

Pinging 64.0.0.6 with 32 bytes of data:

Reply from 64.0.0.6: bytes=32 time<1ms TTL=128
Reply from 64.0.0.6: bytes=32 time<1ms TTL=128
Reply from 64.0.0.6: bytes=32 time<1ms TTL=128
Reply from 64.0.0.6: bytes=32 time<1ms TTL=128

Ping statistics for 64.0.0.6:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

From PC3 :

```
C:\>ping 64.0.0.1

Pinging 64.0.0.1 with 32 bytes of data:

Reply from 64.0.0.1: bytes=32 time<1ms TTL=128
Reply from 64.0.0.1: bytes=32 time<1ms TTL=128
Reply from 64.0.0.1: bytes=32 time<1ms TTL=128
Reply from 64.0.0.1: bytes=32 time<1ms TTL=128

Ping statistics for 64.0.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>ping 64.0.0.2

Pinging 64.0.0.2 with 32 bytes of data:

Reply from 64.0.0.2: bytes=32 time<1ms TTL=128
Reply from 64.0.0.2: bytes=32 time<1ms TTL=128
Reply from 64.0.0.2: bytes=32 time<1ms TTL=128
Reply from 64.0.0.2: bytes=32 time<1ms TTL=128

Ping statistics for 64.0.0.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>ping 64.0.0.3

Pinging 64.0.0.3 with 32 bytes of data:

Reply from 64.0.0.3: bytes=32 time<1ms TTL=128
Reply from 64.0.0.3: bytes=32 time<1ms TTL=128
Reply from 64.0.0.3: bytes=32 time<1ms TTL=128
Reply from 64.0.0.3: bytes=32 time<1ms TTL=128

Ping statistics for 64.0.0.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

```
C:\>ping 64.0.0.5

Pinging 64.0.0.5 with 32 bytes of data:

Reply from 64.0.0.5: bytes=32 time<1ms TTL=128
Reply from 64.0.0.5: bytes=32 time<1ms TTL=128
Reply from 64.0.0.5: bytes=32 time<1ms TTL=128
Reply from 64.0.0.5: bytes=32 time<1ms TTL=128

Ping statistics for 64.0.0.5:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>ping 64.0.0.6

Pinging 64.0.0.6 with 32 bytes of data:

Reply from 64.0.0.6: bytes=32 time<1ms TTL=128
Reply from 64.0.0.6: bytes=32 time<1ms TTL=128
Reply from 64.0.0.6: bytes=32 time<1ms TTL=128
Reply from 64.0.0.6: bytes=32 time<1ms TTL=128

Ping statistics for 64.0.0.6:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

From PC4 :

```
C:\>ping 64.0.0.1

Pinging 64.0.0.1 with 32 bytes of data:

Reply from 64.0.0.1: bytes=32 time<1ms TTL=128
Reply from 64.0.0.1: bytes=32 time<1ms TTL=128
Reply from 64.0.0.1: bytes=32 time<1ms TTL=128
Reply from 64.0.0.1: bytes=32 time<1ms TTL=128

Ping statistics for 64.0.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>ping 64.0.0.2

Pinging 64.0.0.2 with 32 bytes of data:

Reply from 64.0.0.2: bytes=32 time<1ms TTL=128
Reply from 64.0.0.2: bytes=32 time<1ms TTL=128
Reply from 64.0.0.2: bytes=32 time<1ms TTL=128
Reply from 64.0.0.2: bytes=32 time<1ms TTL=128

Ping statistics for 64.0.0.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>ping 64.0.0.3

Pinging 64.0.0.3 with 32 bytes of data:

Reply from 64.0.0.3: bytes=32 time<1ms TTL=128
Reply from 64.0.0.3: bytes=32 time<1ms TTL=128
Reply from 64.0.0.3: bytes=32 time<1ms TTL=128
Reply from 64.0.0.3: bytes=32 time<1ms TTL=128

Ping statistics for 64.0.0.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

```
C:\>ping 64.0.0.4

Pinging 64.0.0.4 with 32 bytes of data:

Reply from 64.0.0.4: bytes=32 time<1ms TTL=128
Reply from 64.0.0.4: bytes=32 time<1ms TTL=128
Reply from 64.0.0.4: bytes=32 time<1ms TTL=128
Reply from 64.0.0.4: bytes=32 time<1ms TTL=128

Ping statistics for 64.0.0.4:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>ping 64.0.0.6

Pinging 64.0.0.6 with 32 bytes of data:

Reply from 64.0.0.6: bytes=32 time<1ms TTL=128
Reply from 64.0.0.6: bytes=32 time<1ms TTL=128
Reply from 64.0.0.6: bytes=32 time<1ms TTL=128
Reply from 64.0.0.6: bytes=32 time=4ms TTL=128

Ping statistics for 64.0.0.6:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 4ms, Average = 1ms
```

From PC5 :

```
C:\>ping 64.0.0.1

Pinging 64.0.0.1 with 32 bytes of data:

Reply from 64.0.0.1: bytes=32 time<1ms TTL=128
Reply from 64.0.0.1: bytes=32 time<1ms TTL=128
Reply from 64.0.0.1: bytes=32 time<1ms TTL=128
Reply from 64.0.0.1: bytes=32 time<1ms TTL=128

Ping statistics for 64.0.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>ping 64.0.0.2

Pinging 64.0.0.2 with 32 bytes of data:

Reply from 64.0.0.2: bytes=32 time<1ms TTL=128
Reply from 64.0.0.2: bytes=32 time<1ms TTL=128
Reply from 64.0.0.2: bytes=32 time<1ms TTL=128
Reply from 64.0.0.2: bytes=32 time<1ms TTL=128

Ping statistics for 64.0.0.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>ping 64.0.0.3

Pinging 64.0.0.3 with 32 bytes of data:

Reply from 64.0.0.3: bytes=32 time<1ms TTL=128
Reply from 64.0.0.3: bytes=32 time<1ms TTL=128
Reply from 64.0.0.3: bytes=32 time<1ms TTL=128
Reply from 64.0.0.3: bytes=32 time<1ms TTL=128

Ping statistics for 64.0.0.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

```
C:\>ping 64.0.0.4

Pinging 64.0.0.4 with 32 bytes of data:

Reply from 64.0.0.4: bytes=32 time<1ms TTL=128
Reply from 64.0.0.4: bytes=32 time<1ms TTL=128
Reply from 64.0.0.4: bytes=32 time<1ms TTL=128
Reply from 64.0.0.4: bytes=32 time<1ms TTL=128

Ping statistics for 64.0.0.4:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>ping 64.0.0.5

Pinging 64.0.0.5 with 32 bytes of data:

Reply from 64.0.0.5: bytes=32 time<1ms TTL=128
Reply from 64.0.0.5: bytes=32 time=7ms TTL=128
Reply from 64.0.0.5: bytes=32 time<1ms TTL=128
Reply from 64.0.0.5: bytes=32 time<1ms TTL=128

Ping statistics for 64.0.0.5:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 7ms, Average = 1ms
```

Write down ARP table of all given PCs. Write down switch table of Switch0 and Switch1.
ARP table entry of PC0

```
C:\>arp -a
  Internet Address      Physical Address      Type
  64.0.0.2              0040.0b0e.ce61        dynamic
  64.0.0.3              000a.4172.4253        dynamic
  64.0.0.4              0003.e487.0ec5        dynamic
  64.0.0.5              0002.1620.c963        dynamic
  64.0.0.6              000d.bd13.0454        dynamic
```

IP address	MAC Address
64.0.0.2	0040.0b0e.ce61
64.0.0.3	000a.4172.4253
64.0.0.4	0003.e487.0ec5
64.0.0.5	0002.1620.c963
64.0.0.6	000d.bd13.0454

ARP table entry of PC1

```
C:\>arp -a
  Internet Address      Physical Address      Type
  64.0.0.1              0030.f2a8.9da0        dynamic
  64.0.0.3              000a.4172.4253        dynamic
  64.0.0.4              0003.e487.0ec5        dynamic
  64.0.0.5              0002.1620.c963        dynamic
  64.0.0.6              000d.bd13.0454        dynamic
```

IP address	MAC Address
64.0.0.1	0030.f2a8.9da0
64.0.0.3	000a.4172.4253
64.0.0.4	0003.e487.0ec5
64.0.0.5	0002.1620.c963
64.0.0.6	000d.bd13.0454

ARP table entry of PC2

```
C:\>arp -a
  Internet Address      Physical Address      Type
  64.0.0.1              0030.f2a8.9da0        dynamic
  64.0.0.2              0040.0b0e.ce61        dynamic
  64.0.0.4              0003.e487.0ec5        dynamic
  64.0.0.5              0002.1620.c963        dynamic
  64.0.0.6              000d.bd13.0454        dynamic
```

IP address	MAC Address
64.0.0.1	0030.f2a8.9da0
64.0.0.2	000a.4172.4253
64.0.0.4	0003.e487.0ec5
64.0.0.5	0002.1620.c963
64.0.0.6	000d.bd13.0454

ARP table entry of PC3

```
C:\>arp -a
  Internet Address      Physical Address      Type
  64.0.0.1              0030.f2a8.9da0        dynamic
  64.0.0.2              0040.0b0e.ce61        dynamic
  64.0.0.3              000a.4172.4253        dynamic
  64.0.0.5              0002.1620.c963        dynamic
  64.0.0.6              000d.bd13.0454        dynamic
```

IP address	MAC Address
64.0.0.1	0030.f2a8.9da0
64.0.0.2	000a.4172.4253

64.0.0.3	0003.e487.0ec5
64.0.0.5	0002.1620.c963
64.0.0.6	000d.bd13.0454

ARP table entry of PC4

```
C:\>arp -a
  Internet Address      Physical Address      Type
  64.0.0.1              0030.f2a8.9da0        dynamic
  64.0.0.2              0040.0b0e.ce61        dynamic
  64.0.0.3              000a.4172.4253        dynamic
  64.0.0.4              0003.e487.0ec5        dynamic
  64.0.0.6              000d.bd13.0454        dynamic
```

IP address	**MAC Address**
64.0.0.1	0030.f2a8.9da0
64.0.0.2	000a.4172.4253
64.0.0.3	0003.e487.0ec5
64.0.0.4	0003.e487.0ec5
64.0.0.6	000d.bd13.0454

ARP table entry of PC5

```
C:\>arp -a
  Internet Address      Physical Address      Type
  64.0.0.1              0030.f2a8.9da0        dynamic
  64.0.0.2              0040.0b0e.ce61        dynamic
  64.0.0.3              000a.4172.4253        dynamic
  64.0.0.4              0003.e487.0ec5        dynamic
  64.0.0.5              0002.1620.c963        dynamic
```

IP address	**MAC Address**
64.0.0.1	0030.f2a8.9da0
64.0.0.2	000a.4172.4253
64.0.0.3	0003.e487.0ec5
64.0.0.4	0003.e487.0ec5
64.0.0.5	0002.1620.c963

1	0001.97EC.E001	FastEthernet0/5
1	0002.1620.C963	FastEthernet0/5
1	0003.E487.0EC5	FastEthernet0/4
1	000A.4172.4253	FastEthernet0/3
1	000D.BD13.0454	FastEthernet0/5

VLAN	Mac Address	Port
1	0003.E487.0EC5	FastEthernet0/4
1	000A.4172.4253	FastEthernet0/3
1	000D.BD13.0454	FastEthernet0/5
1	0030.F2A8.9DA0	FastEthernet0/1
1	0040.0B0E.CE61	FastEthernet0/2

Switch table entry of Switch0

MAC Address	Ethernet port no
0001.97EC.E001	FastEthernet 0/5
0002.1620.C963	FastEthernet 0/5
0003.E487.0EC5	FastEthernet 0/4
000A.4172.4253	FastEthernet 0/3
000D.BD13.0454	FastEthernet 0/5
0030.F2A8.9DA0	FastEthernet 0/1
0040.0B0E.CE61	FastEthernet 0/2

Switch table entry of Switch1

VLAN	Mac Address	Port
1	000A.4172.4253	FastEthernet0/1
1	000C.CFAB.2805	FastEthernet0/1
1	000D.BD13.0454	FastEthernet0/3
1	0030.F2A8.9DA0	FastEthernet0/1
1	0040.0B0E.CE61	FastEthernet0/1

VLAN	Mac Address	Port
1	0002.1620.C963	FastEthernet0/2
1	0003.E487.0EC5	FastEthernet0/1
1	000A.4172.4253	FastEthernet0/1
1	000C.CFAB.2805	FastEthernet0/1
1	000D.BD13.0454	FastEthernet0/3

64.(

MAC Address	**Ethernet port no**
0002.1620.C963	FastEthernet 0/2
0003.E487.0EC5	FastEthernet 0/1
000A.4172.4253	FastEthernet 0/1
000C.CFAB.2805	FastEthernet 0/1
000D.BD13.0454	FastEthernet 0/3
0030.F2A8.9DA0	FastEthernet 0/1
0040.0B0E.CE61	FastEthernet 0/1

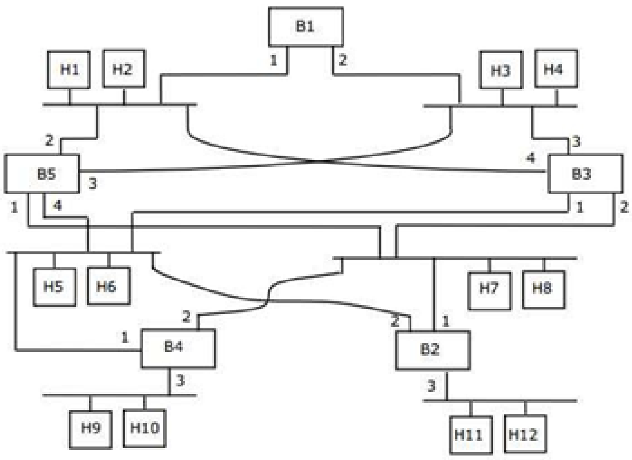
Questions

1. Does both the switch will have identical switching table?
   Ans : No, they do not have identical switching table

2. In Exercise-4, why does PC0 contain IP MAC address of PC4?
   Ans : We have ping the PC4 w.r.t PC0.
3. What is topology name of exercise-4?
   Ans : Its a **mesh topology**.

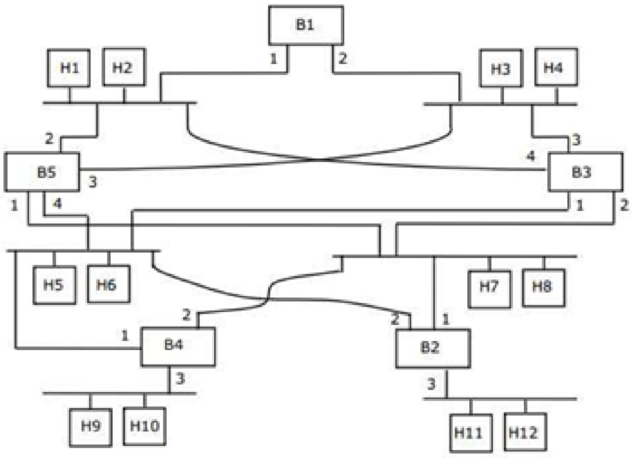
GATE Questions:

1. **Consider the diagram shown below where a number of LANs are connected by (transparent) bridges. In order to avoid packets looping through circuits in the graph, the bridges organize themselves in a spanning tree. First, the root bridge is identified as the bridge with the least serial number. Next, the root sends out (one or more) data units to enable the setting up of the spanning tree of shortest paths from the root bridge to each bridge.**
   Each bridge identifies a port (the root port) through which it will forward frames to the root bridge. Port conflicts are always resolved in favour of the port with the lower index value. When there is a possibility of multiple bridges forwarding to the same LAN (but not through the root port), ties are broken as follows: bridges closest to the root get preference and between such bridges, the one with the lowest serial number is preferred.



   For the given connection of LANs by bridges, which one of the following choices represents the depth first traversal of the spanning tree of bridges?
   A) B1, B5, B3, B4, B2
   B) B1, B3, B5, B2, B4
   C) B1, B5, B2, B3, B4
   D) B1, B3, B4, B5, B2

2. **Consider the diagram shown below where a number of LANs are connected by (transparent) bridges. In order to avoid packets looping through circuits in the graph, the bridges organize themselves in a spanning tree. First, the root bridge is identified as the bridge with the least serial number. Next, the root sends out (one or more) data units to enable the setting up of the spanning tree of shortest paths from the root bridge to each bridge. Each bridge identifies a port (the root port) through which it will forward frames to the root bridge. Port conflicts are always resolved in favour of the port with the lower index value. When there is a possibility of multiple bridges forwarding to the same LAN (but not through the root port), ties are broken as follows: bridges closest to the root get preference and between such bridges, the one with the lowest serial number is preferred.**



Consider the spanning tree B1, B5, B3, B4, B2 for the given connection of LANs by bridges that represents the depth first traversal of the spanning tree of bridges. Let host H1 send out a broadcast ping packet. Which of the following options represents the correct forwarding table on B3?

A)	**Hosts**	**Port**
	H1, H2, H3, H4	3
	H5, H6, H9, H10	1
	H7, H8, H11, H12	2
B)	**Hosts**	**Port**
	H1, H2	4
	H3, H4	3
	H5, H6	1
	H7, H8, H9, H10, H11, H12	2
C)	**Hosts**	**Port**
	H3, H4	3
	H5, H6, H9, H10	1
	H1, H2	4
	H7, H8, H11, H12	2
D)	**Hosts**	**Port**
	H1, H2, H3, H4	3
	H5, H7, H9, H10	1
	H7, H8, H11, H12	4

A) A
B) B
C) C
D) D

3. **Consider a simple communication system where multiple nodes are connected by a shared broadcast medium (like Ethernet or wireless). The nodes in the system use the following carrier-sense based medium access protocol. A node that receives a packet to transmit will carrier-sense the medium for 5 units of time. If the node does not detect any other transmission, it starts transmitting its packet in the next time unit. If the node detects another transmission, it waits until this other transmission finishes, and then begins to carrier-sense for 5 time units again. Once they start to transmit, nodes do not perform any collision detection and continue transmission even if a collision occurs. All transmissions last for 20 units of time. Assume that the transmission signal travels at the speed of 10 meters per unit time in the medium.**

   Assume that the system has two nodes P and Q, located at a distance d meters from each other. P start transmitting a packet at time t = 0 after successfully completing its carrier-sense phase. Node Q has a packet to transmit at time t = 0 and begins to carrier- sense the medium.

   The maximum distance d (in meters, rounded to the closest integer) that allows Q to successfully avoid a collision between its proposed transmission and P 's ongoing transmission is 50.

4. **A and B are the only two stations on an Ethernet. Each has a steady queue of frames to send. Both A and B attempt to transmit a frame, collide, and A wins the first backoff race. At the end of this successful transmission by A, both A and B attempt to transmit and collide. The probability that A wins the second backoff race is:**
   A) 0.5
   B) 0.625
   C) 0.75
   D) 1.0

5. **Which of the following statements is TRUE?**
   A) Both Ethernet frame and IP packet include checksum fields
   B) Ethernet frame includes a checksum field and IP packet includes a CRC field
   C) Ethernet frame includes a CRC field and IP packet includes a checksum field
   D) Both Ethernet frame and IP packet include CRC fields

6. **Consider three IP networks A, B and C. Host HA in network A sends messages each containing 180 bytes of application data to a host HC in network C. The TCP layer prefixes 20 byte header to the message.**
   This passes through an intermediate network B.The maximum packet size, including 20 byte IP header, in each network is:
   A: 1000 bytes
   B: 100 bytes
   C: 1000 bytes

The network A and B are connected through a 1 Mbps link, while B and C are connected by a 512 Kbps link (bps = bits per second).

Assuming that the packets are correctly delivered, how many bytes, including headers, are delivered to the IP layer at the destination for one application message, in the best case? Consider only data packets.

A) 200
B) 220
C) 240
D) 260

7. **Consider three IP networks A, B and C. Host HA in network A sends messages each containing 180 bytes of application data to a host HC in network C. The TCP layer prefixes 20 byte header to the message. This passes through an intermediate network B. The maximum packet size, including 20 byte IP header, in each network, is:**

    A : 1000 bytes
    B : 100 bytes
    C : 1000 bytes

    The network A and B are connected through a 1 Mbps link, while B and C are connected by a 512 Kbps link (bps = bits per second).

What is the rate at which application data is transferred to host HC? Ignore errors, acknowledgments, and other overheads.

A) 325.5 Kbps
B) 354.5 Kbps
C) 409.6 Kbps
D) 512.0 Kbps

8. **A 2 km long broadcast LAN has 107 bps bandwidth and uses CSMA/CD. The signal travels along the wire at 2 × 108 m/s. What is the minimum packet size that can be used on this network?**

A) 50 bytes
B) 100 bytes
C) 200 bytes
D) None of the above

9. **Consider that 15 machines need to be connected in a LAN using 8-port Ethernet switches. Assume that these switches do not have any separate uplink ports. The minimum number of switches needed is 3.**

10. A host is connected to a Department network which is part of a University network. The University network, in turn, is part of the Internet. The largest network in which the Ethernet address of the host is unique is:

A) the subnet to which the host belongs
B) the Department network
C) the University network
D) the Internet

11. Which of the following statements is FALSE regarding a bridge?

A) Bridge is a layer 2 device
B) Bridge reduces collision domain
C) Bridge is used to connect two or more LAN segments
D) Bridge reduces broadcast domain

12. Suppose the round trip propagation delay for a 10 Mbps Ethernet having 48-bit jamming signal is 46.4 µs. The minimum frame size is:

A) 94
B) 416
C) 464
D) 512

13. A link has transmission speed of 106 bits/sec. It uses data packets of size 1000 bytes each. Assume that the acknowledgment has negligible transmission delay and that its propagation delay is the same as the data propagation delay. Also, assume that the processing delays at nodes are negligible. The efficiency of the stop-and-wait protocol in this setup is exactly 25%. The value of the one way propagation delay (in milliseconds) is 12 .

14. Consider a simplified time slotted MAC protocol, where each host always has data to send and transmits with probability p = 0.2 in every slot. There is no backoff and one frame can be transmitted in one slot. If more than one host transmits in the same slot, then the transmissions are unsuccessful due to collision. What is the maximum number of hosts which this protocol can support if each host has to be provided a minimum throughput of 0.16 frames per time slot?

A) 1
B) 2
C) 3
D) 4

15. In a TDM medium access control bus LAN, each station is assigned one time slot per cycle for transmission. Assume that the length of each time slot is the time to transmit 100 bitsplus the end-to-end propagation delay. Assume a propagation speed of 2 × 108m/sec. The length of the LAN is 1 km with a bandwidth of 10 Mbps. The maximum number of stations that can be allowed in the LAN so that the throughput of each station can be 2/3 Mbpsis

A) 3
B) 5
C) 10
D) 20

16. The address resolution protocol (ARP) is used for:
A) Finding the IP address from the DNS
B) Finding the IP address of the default gateway
C) Finding the IP address that corresponds to a MAC address
D) Finding the MAC address that corresponds to an IP address

17. Suppose that in an IP-over-Ethernet network, a machine X wishes to find the MAC address of another machine Y in its subnet. Which one of the following techniques can be used for this?
A) X sends an ARP request packet to the local gateway's IP address which then finds the MAC address of Y and sends to X
B) X sends an ARP request packet to the local gateway's MAC address which then finds the MAC address of Y and sends to X
C) X sends an ARP request packet with broadcast MAC address in its local subnet
D) X sends an ARP request packet with broadcast IP address in its local subnet

18. **Consider the following two statements.**

S1: Destination MAC address of an ARP reply is a broadcast address.
S2: Destination MAC address of an ARP request is a broadcast address.

Which one of the following choices is correct?
A) Both S1 and S2 are true
B) S1 is true and S2 is false
C) S1 is false and S2 is true
D) Both S1 and S2 are false

19. Consider the following clauses:
i. Not inherently suitable for client authentication.
ii. Not a state sensitive protocol.
iii. Must be operated with more than one server.
iv. Suitable for structured message organization.
v. May need two ports on the serve side for proper operation.
The option that has the maximum number of correct matches is
A) IMAP-i; FTP-ii; HTTP-iii; DNS-iv; POP3-v
B) FTP-i; POP3-ii; SMTP-iii; HTTP-iv; IMAP-v
C) POP3-i; SMTP-ii; DNS-iii; IMAP-iv; HTTP-v
D) SMTP-i; HTTP-ii; IMAP-iii; DNS-iv; FTP-v

20. In a packet switching network, packets are routed from source to destination along a single path having two intermediate nodes. If the message size is 24 bytes and each packet contains a header of 3 bytes, then the optimum packet size is:

A) 4
B) 6
C) 7
D) 9

21. Which one of the following statements is FALSE?

A) Packet switching leads to better utilization of bandwidth resources than circuit switching
B) Packet switching results in less variation in delay than circuit switching
C) Packet switching requires more per-packet processing than circuit switching
D) Packet switching can lead to reordering unlike in circuit switching

22. Consider a network using the pure ALOHA medium access control protocol, where each frame is of length 1, 000 bits. The channel transmission rate is 1 Mbps (= 106 bits per second). The aggregate number of transmissions across all the nodes (including new frame transmissions and retransmitted frames due to collisions) is modelled as a Poisson process with a rate of 1, 000 frames per second. Throughput is defined as the average number of frames successfully transmitted per second. The throughput of the network (rounded to the nearest integer) is 135 .

23. For the IEEE 802.11 MAC protocol for wireless communication, which of the following statements is/are TRUE?

i. At least three non-overlapping channels are available for transmissions.
ii. The RTS-CTS mechanism is used for collision detection.
iii. Unicast frames are ACKed.

A) I, II, and III
B) I and III only
C) II and III only
D) II only

24. The minimum frame size required for a CSMA/CD based computer network running at 1Gbps on a 200m cable with a link speed of 2 × 108m/sec is:

A) 125 bytes
B) 250bytes
C) 500bytes
D) None of the above