CSE 434 Computer Networks (Fall 2019) Assignment 2

Duo Lu <duolu@asu.edu>

This assignment is due on 11:59 pm on October 2, 2019. It requires hands-on experiences about IP routing with the GNS3 simulator, which can easily take more than 10 hours even if you have some knowledge of IP network. So please start early and search online from time to time to make sure you understand what you are doing. The grading is effort-based.

Note that you can comment on this document and ask questions in the comment. You are welcome to discuss with your classmates on the assignment. However, for every student, please go through the lab at least once on your own to understand what you are doing at every step. Please do not resolve existing comments. If you resolve it, others cannot see it.

Task 1: Install and configure GNS3.

Download the GNS3 network simulator / emulator and install it on your machine. It is NOT recommended to install the simulator inside a virtual machine. Detailed installation instructions can be found by the following link and video.

Windows:

https://docs.gns3.com/11YYG4NQlPSl31YwvVvBS9RAsOLSYv0Ocy-uG2K8ytIY/index.html

Mac OS:

https://docs.gns3.com/1MlG-VjkfQVEDVwGMxE3sJ15eU2KTDsktnZZH8HSR-IQ/index.html

Linux:

https://docs.gns3.com/1QXVIihk7dsOL7Xr7Bmz4zRzTsJ02wklfImGuHwTlaA4/index.html

You can watch this video for a walkthrough on Windows:

https://www.youtube.com/watch?v=APpcufvoCZQ

or this video on Mac OS:

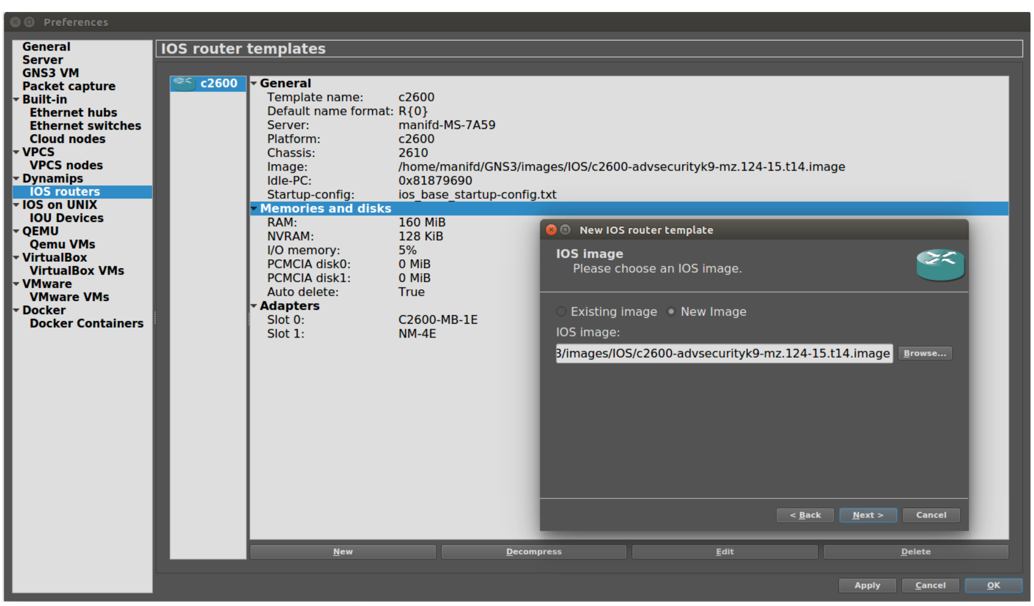
https://www.youtube.com/watch?v=OFjt3jS3Mhg

Note that during the installation, you can choose to ignore the installation of GNS3 VM. As far as I know, for our lab, even without installing any GNS3 VM, GNS3 will still work fine. We use the Dynamips emulator inside the GNS3 to run the Cisco IOS image. Those virtual machines are required if you want to run more advanced routers.

After that, go through this video to understand how to set up a basic network.

https://www.youtube.com/watch?v=ueNACHY10OM&t=1s

Please use the 2600 Cisco IOS router image (posted on Canvas). Our IT department has purchased training services and Cisco VIRL platform so we are OK to use the image for instructional purpose. You can add this image by clicking "Edit" in the menu bar, then click "Preference", then in the left column of the dialog, choose "IOS routers" under "Dynamips, and click the button "new" at the bottom of the right column (scroll down or resize the window if you can not see the button). Next, choose "new image" in the new dialog, find the image you downloaded from the blackboard, as shown in the following figure. Ignore the warnings. After that, you will be able to drag and drop a Cisco 2600 router in your project.



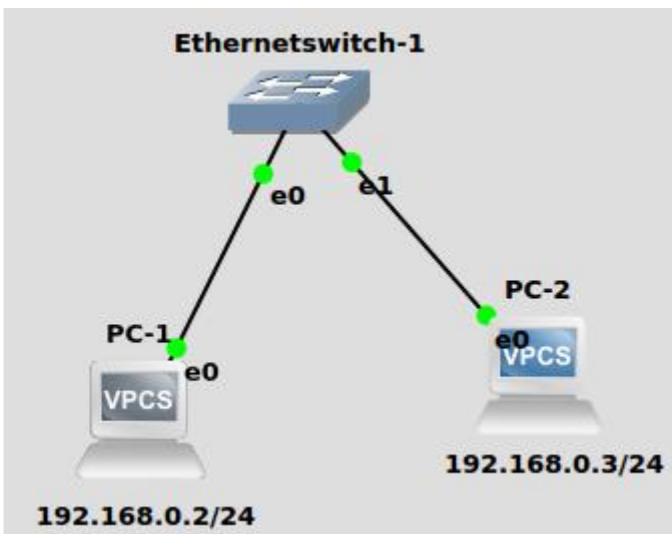
It should be noted that GNS3 may eat a lot of disk space. If you're mysteriously losing a bunch of disk space, please check your GNS3 project folder. Specifically, it is the dynamips program creating massive log files.

Also, if you're having issues dragging the router into the work space try adding GNS3 to the exceptions in your antivirus and your firewall software. If you still have trouble running GNS3 properly after trying every possible means and configurations, the best practice is just borrow another computer to finish the assignment. Based on previous experience, there are always a few students with Mac machines that can not make GNS run as expected. If you can not run it on a physical machine with Mac, you can try to install it in a virtual machine or another physical machine with a different OS, especially a 32-bit operating system. However, I have encountered issues of finding IDLEPC inside a virtual machine, and the CPU usage can soar up to 100% all the time without a correct IDLEPC.

Task 2: Configure Ethernet switching network.

Start a new project in GNS3. Set up a network like the following figure. Use the "Ethernet switch" in the switch device column and "VPCS" in the end device column (scroll down to find it). Optionally, you can click the button "add a note" on the top to put some text in the topology.

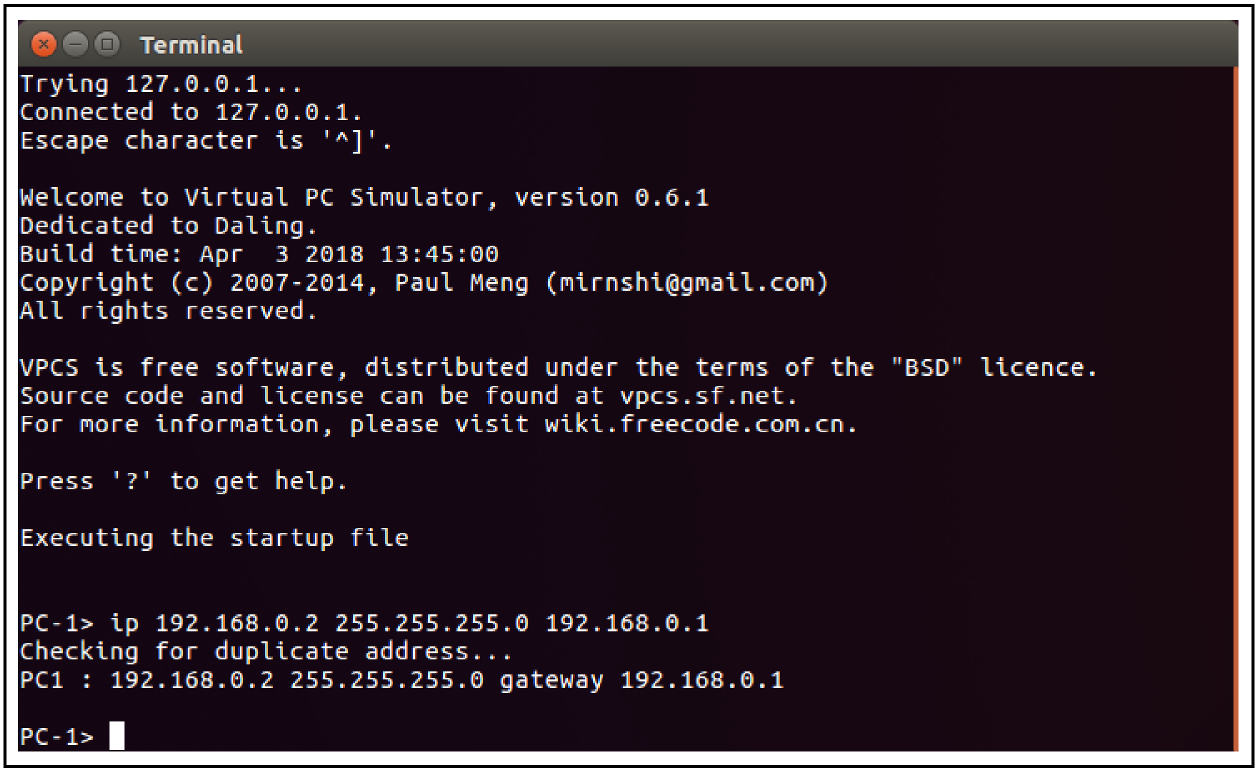
Please change the name of the Ethernet switch with a suffix of your name or your ASU email address such as "switch-duolu". This is intended to make your work unique to you, which is partially a proof of work.



Then configure the designated IP address on the two hosts as follows.

1) Click the "start all devices" button. Or you can right click the VPCS host and then click "start".
2) Double click the VPCS host to show the console of the host. Or you can right click the VPCS host and then click "console".
3) Type the command as follows to configure IP address for the host PC-1. The first part, "192.168.0.2", is the IP address of the interface of the host; the second part, "255.255.255.0" is the subnet mask (i.e., the number of consecutive "1" bits indicates

which part is the prefix of the subnet); the third part, "192.168.0.1" is the gateway, which means by default all outgoing IP packets are sent to the gateway, usually the IP address of the router interface directly connected to that subnet. Gateway is not used in this lab, but in the next lab on IP routing it needs to be configured. Note that these IP addresses text shown in the figure are added manually. It does not show up automatically.



4) Configure the PC-2 similarly. Note that PC-2 has a different IP address from PC-1 but they are in the same IP subnet.
5) The switch does not need any configuration.

The VPCS host has only one ethernet interface. Type "show" on each host of PC-1 and PC-2, take a screenshot, and show what is the MAC address of each host.

After that, try to "ping" PC-2 from PC-1 by typing "ping 192.168.0.3" on PC-1. Make sure that they can reach each other. Take a screenshot showing the result of the ping command, and briefly explain what packets are sent and received in this network. Optionally, you can also right click a wire, and click "start capture" to sniff the packets on that wire with Wireshark (assume Wireshark is installed on your machine). You can use this method to verify your explanation.

Next, type "show arp" on PC-1 and PC-2, take a screenshot, and explain the meaning of each entry of result. It should not be empty. If it is empty, ping the host again and check the ARP entries. What does ARP do? (Hint: just search it online.) Note that ARP entries are cached and it expires in about 2 minutes. Hence, you need to run "ping" first and immediately check the ARP table.

At last, open the console of the Ethernet switch, type "mac" to show the forwarding table. Take a screenshot and explain each entry of the forwarding table. You may need to have a look at slide set 5 on Ethernet.

show on PC-1

```
PC-1> ip 192.168.0.2 255.255.255.0 192.168.0.1
Checking for duplicate address...
PC1 : 192.168.0.2 255.255.255.0 gateway 192.168.0.1

PC-1> show

NAME   IP/MASK              GATEWAY           MAC                LPORT  RHOST:PORT
PC-1   192.168.0.2/24       192.168.0.1       00:50:79:66:68:00  10002  127.0.0.1:10003
       fe80::250:79ff:fe66:6800/64
```

MAC address of PC-1 = 00:50:79:66:68:00

show on PC-2

```
PC-2> ip 192.168.0.3 255.255.255.0 192.168.0.1
Checking for duplicate address...
PC1 : 192.168.0.3 255.255.255.0 gateway 192.168.0.1

PC-2> show

NAME   IP/MASK              GATEWAY           MAC                LPORT  RHOST:PORT
PC-2   192.168.0.3/24       192.168.0.1       00:50:79:66:68:01  10000  127.0.0.1:10001
       fe80::250:79ff:fe66:6801/64
```

MAC address of PC-2 = 00:50:79:66:68:01

Ping PC-2 from PC-1:

```
PC-1> ping 192.168.0.3
84 bytes from 192.168.0.3 icmp_seq=1 ttl=64 time=0.000 ms
84 bytes from 192.168.0.3 icmp_seq=2 ttl=64 time=0.964 ms
84 bytes from 192.168.0.3 icmp_seq=3 ttl=64 time=0.000 ms
84 bytes from 192.168.0.3 icmp_seq=4 ttl=64 time=0.000 ms
84 bytes from 192.168.0.3 icmp_seq=5 ttl=64 time=0.996 ms
```

Ping operates by sending Internet Control Message Protocol (ICMP) echo request packets to the target host and waiting for an ICMP echo reply.

show arp on PC-1:

```
PC-1> show arp

00:50:79:66:68:01  192.168.0.3 expires in 112 seconds
```

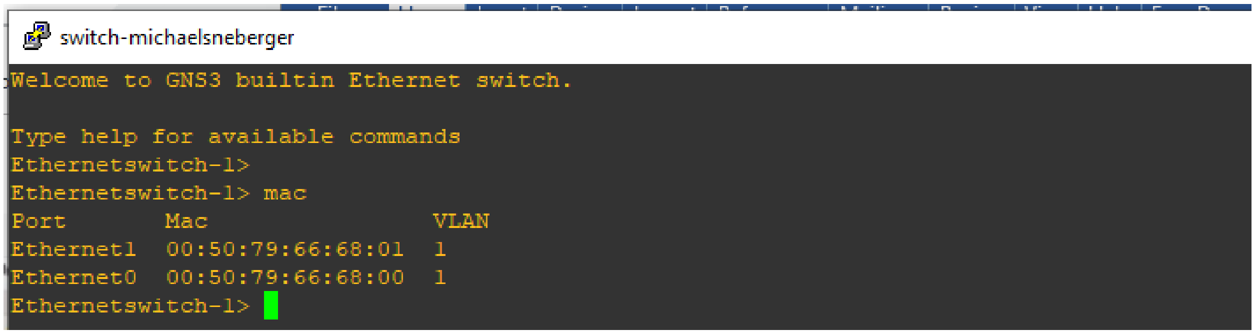
The show arp command shows the MAC address and IP address of other hosts on network. Here, running show arp from PC-1 shows the MAC and IP addresses of PC-2.

show arp on PC-2:

```
PC-2> show arp

00:50:79:66:68:00  192.168.0.2 expires in 83 seconds
```

The show arp command shows the MAC address and IP address of other hosts on network. Here, running show arp from PC-2 shows the MAC and IP addresses of PC-1.

Forwarding table of Ethernet switch:

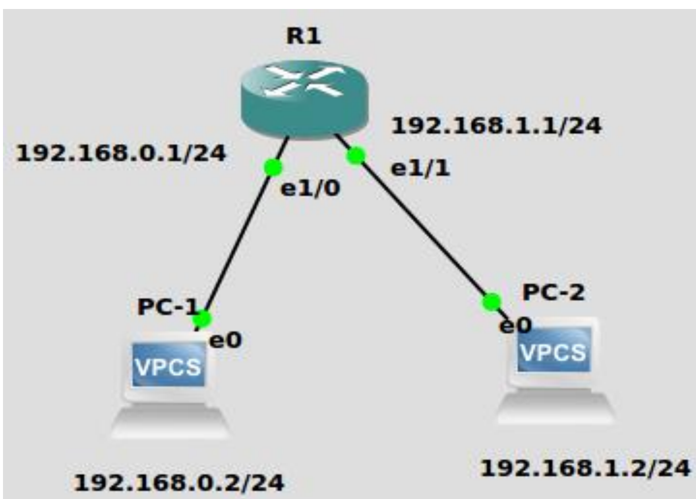


The forwarding table shows the ports of the switch and the MAC addresses of each host on the those ports.

Task 3: Configure a router to connect two subnets.

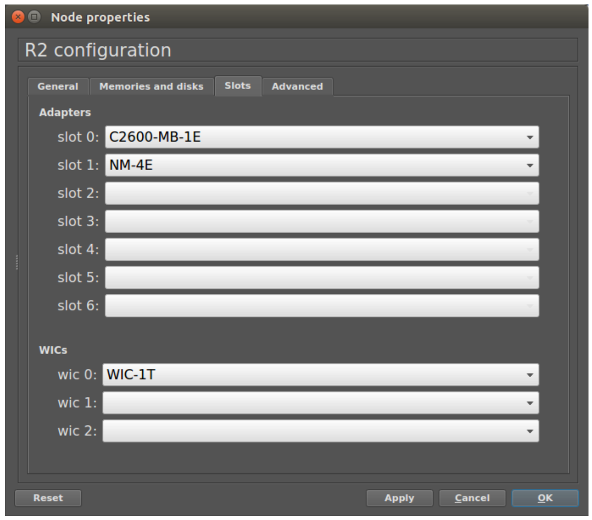
Start a new project in GNS3. Set up a network like the following figure. Use the "c2600" in the router device column and "VPCS" in the end device column (scroll down to find it).

Please change the name of the router switch with a suffix of your name or your ASU email address such as "R1-duolu".



Then configure the designated IP address on the two hosts. Note that the IP address of PC-2 is changed. PC-1 and PC-2 are not in the same IP subnet anymore. As we know, an IP router forwards IP packets among different IP network. Thus, each interface of the router is in a different IP subnet. This time, configure the gateway of PC-1 as "192.168.0.1" and configure the gateway of PC-2 as "192.168.1.1". Please do remember to configure the correct default gateway for PC-1 and PC-2. If you do not configure the default gateway correctly, packets from a PC will just be dropped when looking up forwarding table on that PC (similar to the situation in quiz 3 discussed in class). Use the "show" command to make sure the address, subnet mask, and gateway are correctly configured.

You need to configure the router, add a "NM-4E" module on the router, by right clicking the router, clicking the "configure", and then clicking the "slots" tab, as shown in the following figure. This module provides you four additional ethernet ports on the router. Otherwise, there is not enough ports on the router to connect the two hosts. Similarly, also add a "WIC-1T" module on the router, which provides a serial port for the next lab.



Next, open the console of the router. Remember to click "start all devices" button. You will see a lot of startup information in the console of the router. Just type "Enter" a few times to see the command line prompt. Then type the following commands (bold part is the command you need to type, others are what you should see when the command is executed).168.1

R1#**configure terminal**
Enter configuration commands, one per line. End with CNTL/Z.
R1(config)#**interface e1/0**
R1(config-if)#**ip address 192.168.0.1 255.255.255.0**
R1(config-if)#**no shutdown**
R1(config-if)#
*Mar 1 00:19:55.639: %LINK-3-UPDOWN: Interface Ethernet1/0, changed state to up
*Mar 1 00:19:56.641: %LINEPROTO-5-UPDOWN: Line protocol on Interface Ethernet1/0, changed state to up
R1(config-if)#**exit**
R1(config)#**interface e1/1**
R1(config-if)#**ip address 192.168.1.1 255.255.255.0**
R1(config-if)#**no shutdown**
R1(config-if)#**exit**
R1(config)#
*Mar 1 00:21:13.527: %LINK-3-UPDOWN: Interface Ethernet1/1, changed state to up
*Mar 1 00:21:14.529: %LINEPROTO-5-UPDOWN: Line protocol on Interface Ethernet1/1, changed state to up
R1(config)#

These lines configure the IP addresses on the router interfaces, such that the IP address of each interface is in the same IP subnet as the host the router interface connects. First we type "configure terminal" to enter the configuration mode. Then we type "interface e1/0" to configure a specific interface. Be careful in my topology e1/0 is connected to PC-1 and e1/1 is connected to PC-2. If you connect the hosts using different interfaces such as e1/2 or in different order, modify the command above to make them agree with each other. Also, notice that there are messages about link up and down shown in the terminal. If you did not see these messages, the interfaces of the router might not be activated and the "ping" in later steps would probably fail.

Finally, try to "ping" PC-2 from PC-1 by typing "ping 192.168.1.2" on PC-1. Make sure that they can reach each other. Take a screenshot showing the result of the ping command, and **1)** briefly explain what happens behind the curtain. Note that these two hosts are not directly connected to each other by a wire. **2)** How can they talk to each other? **3)** What does the router do? Please briefly explain the job of this router. You can use the command "show ip route" to **4)** show the forwarding table of the router.

PC-1: the instructions were insufficient as they did not explain how to set up the gateway (Mr. Lu later added a comment about needing to set the gateway, but with no instructions how to do so).

```
PC-1> ip 192.168.0.2/24 192.168.0.1
Checking for duplicate address...
PC1 : 192.168.0.2 255.255.255.0 gateway 192.168.0.1

PC-1> ping 192.168.1.2
192.168.1.2 icmp_seq=1 timeout
84 bytes from 192.168.1.2 icmp_seq=2 ttl=63 time=19.000 ms
84 bytes from 192.168.1.2 icmp_seq=3 ttl=63 time=13.963 ms
84 bytes from 192.168.1.2 icmp_seq=4 ttl=63 time=13.966 ms
84 bytes from 192.168.1.2 icmp_seq=5 ttl=63 time=14.961 ms

PC-1>
```

1)2)3) When PC-1 pings PC-2 PC-1 sends packets to PC-2. These packets are received by the router's gateway then the router looks at the packet header to determine the intended destination of the packet. The router looks at its forwarding table to find the path to the destination and forwards the packet to PC-2 with the return address of PC-1 replaced with the return address of the router. Upon receiving the packet, PC-2 returns a packet to the return address which is the router, and the router undertakes the same process in reverse, forwarding the packet back to PC-1. The routers job is to connect subnets. Each of the VPCS is on a different subnet, thus they cannot connect to each other directly, they need the router in the middle of their communications.

PC-2 the instructions were insufficient as they did not mention setting up the gateway

```
PC-2> ip 192.168.1.2/24 192.168.1.1
Checking for duplicate address...
PC1 : 192.168.1.2 255.255.255.0 gateway 192.168.1.1

PC-2>
```

Here there are two things to notice: 1) writing out the 255.255.255.0 mask is not needed as the GNS3 image of the c2600 router recognizes the /24 notation; and 2) the command for setting both the ip address of PC-2 and the gateway that allows PC-2 to connect with the router is as shown.

ROUTER

R1-michaelsneberger

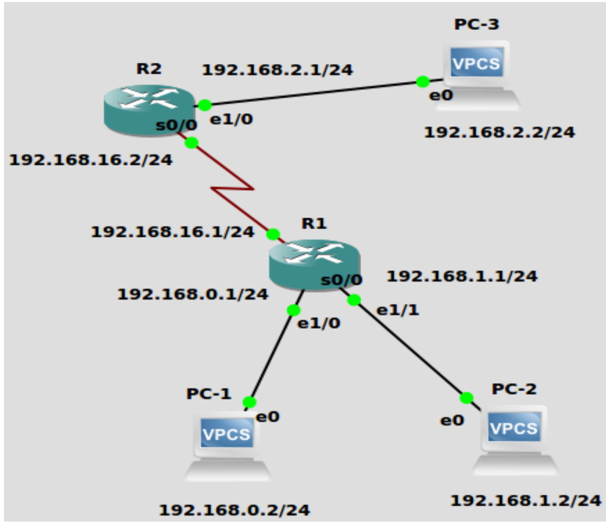
```
R1-michaelsneberger#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R1-michaelsneberger(config)#interface e1/0
R1-michaelsneberger(config-if)#ip address 192.168.0.1 255.255.255.0
R1-michaelsneberger(config-if)#no shutdown
R1-michaelsneberger(config-if)#
*Mar  1 00:02:13.338: %LINK-3-UPDOWN: Interface Ethernet1/0, changed state to up
*Mar  1 00:02:14.340: %LINEPROTO-5-UPDOWN: Line protocol on Interface Ethernet1/0, changed state to up
R1-michaelsneberger(config-if)#exit
R1-michaelsneberger(config)#interface e1/1
R1-michaelsneberger(config-if)#ip address 192.168.1.1 255.255.255.0
R1-michaelsneberger(config-if)#no shutdown
R1-michaelsneberger(config-if)#
*Mar  1 00:03:16.125: %LINK-3-UPDOWN: Interface Ethernet1/1, changed state to up
*Mar  1 00:03:17.127: %LINEPROTO-5-UPDOWN: Line protocol on Interface Ethernet1/1, changed state to up
R1-michaelsneberger(config-if)#exit
R1-michaelsneberger(config)#^Z
R1-michaelsneberger#
```

4) show ip route command:

```
R1-michaelsneberger#show ip route
Codes: C - connected, S - static, 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
       i - IS-IS, su - IS-IS summary, 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    192.168.0.0/24 is directly connected, Ethernet1/0
C    192.168.1.0/24 is directly connected, Ethernet1/1
R1-michaelsneberger#
```

Task 4: IP static routing.

Continue using the network in the previous lab. Add a router R2 and a host PC-3. The second router should have the same module as the first router. Configure the IP address of PC-3 and the interfaces of R2 as shown in the following figure. Be careful that the link between the two routers is a serial link (assume this is a leased line from some ISP that spans several miles).



The serial link is a point-to-point link. By default it uses HDLC encapsulation (i.e., the link layer protocol). HDLC is the predecessor of PPP. We can configure PPP on this link. Type the following command on this interface. Be Careful these commands are typed on the second router. If you are configuring the first router, change the IP address accordingly. Do not forget to enter configuration mode by typing "configure terminal". Also, do not forget to type "no shutdown" on each interface.

```
R2#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#interface s0/0
R2(config-if)#encapsulation ppp
*Mar  1 00:29:41.267: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0,
changed state to down
*Mar  1 00:30:33.469: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0,
changed state to up
R2(config-if)#ip address 192.168.16.2 255.255.255.0
R2(config-if)#no shutdown
R2(config-if)#exit
R2(config)#exit
R2#
```

You can show the interface status by typing "show ip interface brief" if you are not in configuration mode (see the "exit"). You can also ping another interface if you are not in configuration mode. Routers are just computers with multiple interfaces. Check each interface and verify that it is OK. If you forget to type "no shutdown" when configure an interface, the interface can stay down forever. Be careful on this serial link. It might go down without any reason. Please check the interface table and make sure it is up. If it is not "up", just enter the interface and type "no shutdown" again. See the highlighted row in the following table. If it does not show the information as expected, e.g., sometimes "status" is up but "protocol" is down, you need to type "shutdown" in the interface configuration mode and type "no shutdown" again.

```
R1#show ip interface brief
Interface              IP-Address      OK? Method Status                Protocol
Ethernet0/0              unassigned      YES unset  administratively down down
Serial0/0              192.168.16.1    YES manual up                    up
Ethernet1/0              192.168.0.1     YES manual up                    up
Ethernet1/1              192.168.1.1     YES manual up                    up
Ethernet1/2              unassigned      YES unset  administratively down down
Ethernet1/3              unassigned      YES unset  administratively down down
R1#
```

Try to ping PC-3 from PC-1 and take a screenshot. The ping will not be successful. Why

```
PC-1> ping 192.168.2.2
*192.168.0.1 icmp_seq=1 ttl=255 time=19.906 ms (ICMP type:3, code:1, Destination host unreachable)
*192.168.0.1 icmp_seq=2 ttl=255 time=15.621 ms (ICMP type:3, code:1, Destination host unreachable)
*192.168.0.1 icmp_seq=3 ttl=255 time=15.661 ms (ICMP type:3, code:1, Destination host unreachable)
*192.168.0.1 icmp_seq=4 ttl=255 time=15.621 ms (ICMP type:3, code:1, Destination host unreachable)
*192.168.0.1 icmp_seq=5 ttl=255 time=15.661 ms (ICMP type:3, code:1, Destination host unreachable)

PC-1>
```

Answer: there is a serial connection between the routers, but no identified pathway in Router 1's route table to subnet 192.168.2.0 which contains PC-3.

Now type "show ip route" (outside the configuration mode) on each router to show the IP routing table, and take a screenshot, like the following.

```
R1#show ip route
(IP route information omitted...)
```

Explain each entry of the IP routing table. You will be able to observe that the network 192.168.2.0/24 can not be seen on the routing table of R1. Similarly, 192.168.0.0/24 and 192.168.1.0/24 can not be seen on the routing table of R2.

Router 1 before static route commands:

R1-michaelsneberger

```
R1-michaelsneberger#show ip route
Codes: C - connected, S - static, 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
       i - IS-IS, su - IS-IS summary, 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    192.168.0.0/24 is directly connected, Ethernet1/0
     192.168.16.0/24 is variably subnetted, 2 subnets, 2 masks
C       192.168.16.0/24 is directly connected, Serial0/0
C       192.168.16.2/32 is directly connected, Serial0/0
C    192.168.1.0/24 is directly connected, Ethernet1/1
R1-michaelsneberger#
```

The C flag indicates and entry is connected. The following are connected:

Subnet 192.168.0.0/24 which is the subnet containing PC-1

Subnet 192.168.1.0/24 which is the subnet containing PC-2

Subnet 192.168.16.0/24 which is serially connected to the serial port on Router 2. Since this serial connection only needs space for two addresses, it is subnetted by a 32-bit mask. Note that the other end of the serial connection on Router 2 192.168.16.2 is listed.

Router 2 before static route command:

R2-michaelsneberger

```
R2-michaelsneberger#show ip route
Codes: C - connected, S - static, 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
       i - IS-IS, su - IS-IS summary, 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

     192.168.16.0/24 is variably subnetted, 2 subnets, 2 masks
C       192.168.16.0/24 is directly connected, Serial0/0
C       192.168.16.1/32 is directly connected, Serial0/0
C    192.168.2.0/24 is directly connected, Ethernet1/0
R2-michaelsneberger#
```

The C flag indicates and entry is connected. The following are connected:

Subnet 192.168.2.0/24 which is the subnet containing PC-3

Subnet 192.168.16.0/24 which is serially connected to the serial port on Router 1. Since this serial connection only needs space for two addresses, it is subnetted by a 32-bit mask. Note that the other end of the serial connection on Router 1 192.168.16.1 is listed.

Now type the following command on the first router (in configuration mode), which configure a static route.

R1(config)#**ip route 192.168.2.0 255.255.255.0 192.168.16.2**

And the following on the second router.

R2(config)#**ip route 192.168.0.0 255.255.255.0 192.168.16.1** R2(config)#**ip route 192.168.1.0 255.255.255.0 192.168.16.1**

These static route points the destination network to a next hop address directly connected to an interface of the router. Similarly, you can also change this next hop to an outgoing interface.

Finally, **0)** try to ping PC-3 from PC-1. Try multiple times, and take a screenshot. Make sure these two hosts are reachable. Also **1)** show the IP routing table of each router and take screenshots. Point out the static route you configured, and **2)** explain why the ping packets are successfully delivered. **3)** Can you aggregate the two static routes on the second router? Why? If you can, what is the aggregated static route?

Deliverables: Please show four screenshots of PC-3, PC-1, router R1 and router R2. In these screenshots, please include all typed commands and command line outputs, especially the results of the "ping" before and after configuration of the routers. Please also answer related questions below the screenshots.

0) First ping after static configuration: successfully delivered as Router 1 now has an entry in its route table telling it how to get to PC-3.

```
PC-1> ping 192.168.2.2
192.168.2.2 icmp_seq=1 timeout
84 bytes from 192.168.2.2 icmp_seq=2 ttl=62 time=12.965 ms
84 bytes from 192.168.2.2 icmp_seq=3 ttl=62 time=14.958 ms
84 bytes from 192.168.2.2 icmp_seq=4 ttl=62 time=21.942 ms
84 bytes from 192.168.2.2 icmp_seq=5 ttl=62 time=22.898 ms

PC-1>
```

0) Second ping after static configuration:

```
PC-1> ping 192.168.2.2
84 bytes from 192.168.2.2 icmp_seq=1 ttl=62 time=31.288 ms
84 bytes from 192.168.2.2 icmp_seq=2 ttl=62 time=31.283 ms
84 bytes from 192.168.2.2 icmp_seq=3 ttl=62 time=31.202 ms
84 bytes from 192.168.2.2 icmp_seq=4 ttl=62 time=31.283 ms
84 bytes from 192.168.2.2 icmp_seq=5 ttl=62 time=31.242 ms

PC-1>
```

A screenshot of PC-3 would show nothing as pinging PC-3 from somewhere else does not result in any output on PC-3's console.

1) Router 1 after static route commands:

R1-michaelsneberger

```
R1-michaelsneberger#
*Mar  1 00:16:12.365: %SYS-5-CONFIG_I: Configured from console by console
R1-michaelsneberger#show ip route
Codes: C - connected, S - static, 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
       i - IS-IS, su - IS-IS summary, 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    192.168.0.0/24 is directly connected, Ethernet1/0
     192.168.16.0/24 is variably subnetted, 2 subnets, 2 masks
C       192.168.16.0/24 is directly connected, Serial0/0
C       192.168.16.2/32 is directly connected, Serial0/0
C    192.168.1.0/24 is directly connected, Ethernet1/1
S    192.168.2.0/24 [1/0] via 192.168.16.2
R1-michaelsneberger#
```

Compared to before the static route commands were run you can see that a static connection has been added that indicates that Router 1 can get subnet 192.168.2.0 by way of Router 2's 192.168.16.2 address so Router 1 now knows a path to devices on Router 2's subnets.

1) Router 2 after static route command:

R2-michaelsneberger

```
R2-michaelsneberger#show ip route
Codes: C - connected, S - static, 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
       i - IS-IS, su - IS-IS summary, 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    192.168.0.0/24 [1/0] via 192.168.16.1
     192.168.16.0/24 is variably subnetted, 2 subnets, 2 masks
C       192.168.16.0/24 is directly connected, Serial0/0
C       192.168.16.1/32 is directly connected, Serial0/0
S    192.168.1.0/24 [1/0] via 192.168.16.1
C    192.168.2.0/24 is directly connected, Ethernet1/0
R2-michaelsneberger#
```

Compared to before the static route commands were run you can see that static connections have been added that indicates that Router 2 can get subnets 192.168.0.0 and 192.168.1.0 by way of Router 1's 192.168.16.1 address so Router 2 now knows a path to devices on Router 1's subnets.

3) You can aggregate the static routes on Router 2 by supernetting which is the opposite of subnetting and which consolidates selected multiple routes into a single route advertisement, in contrast to the current flat routing in which every routing table contains a unique entry for each route. To implement route summarization in IP Version 4 (IPv4), Classless Inter-Domain Routing (CIDR) must be used. All IP addresses in the route advertisement must share identical high-order bits. The length of the prefix must not exceed 32 bits which is what we have here.

If you compare the binary representation of the two subnets you will see that they are the same until the last bit:

11000000.10101000.00010000.00000000

11000000.10101000.00010000.00000001

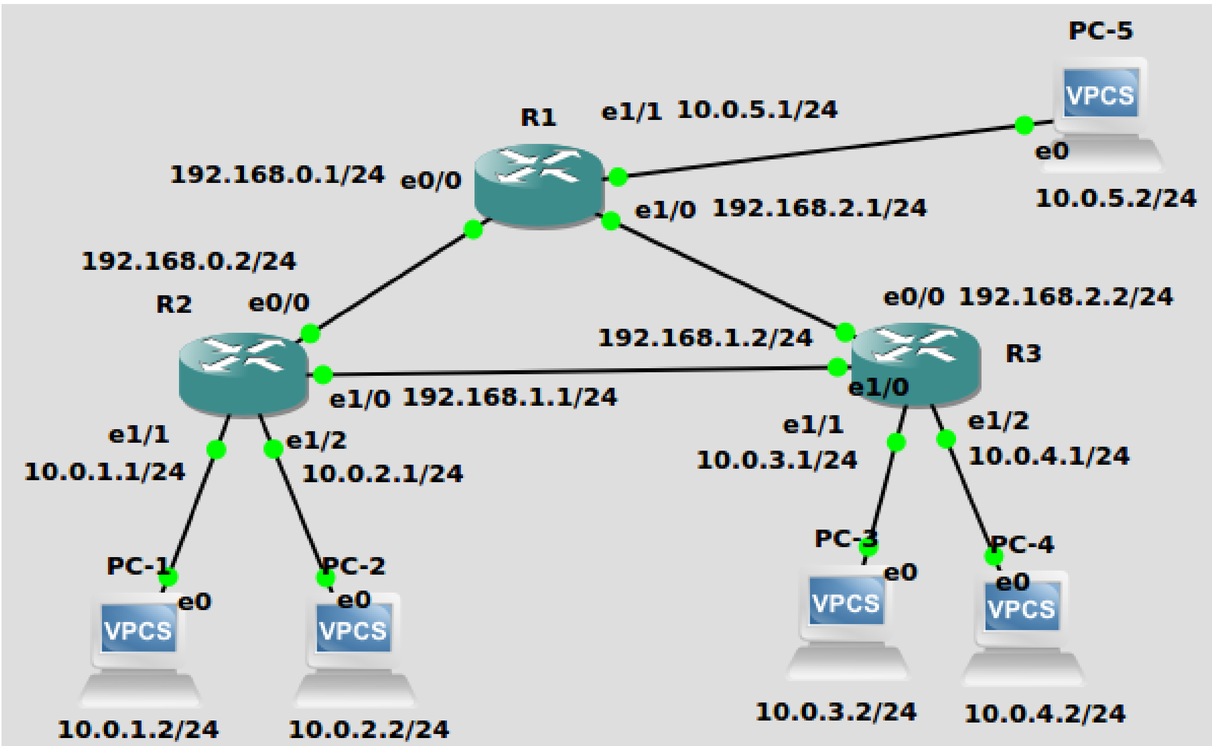
So if you aggregated these two subnets into a single one, borrowing bit from the network part of the address to the host part you would have both represented by the address 192.168.16.0/31 because with only two networks to aggregate we only need to use one bit from the network address.

Route summarization offers advantages over flat routing: minimize the latency in a complex network, especially when many routers are involved; reduce the overhead for routing protocols is minimized; improvement in network stability; reduction in processor workloads, memory requirements and bandwidth demand. But for a tiny network like this aggregation does not appear to be worth the trouble.

Task 5: Dynamic routing with RIP.

Start a new project in GNS3. Set up a network like the following figure. Use the "c2600" in the router device column and "VPCS" in the end device column (scroll down to find it). Please add necessary modules on the router to provide enough interfaces.

Similar as previous labs, please rename the routers with a suffix of your name, such as "R1-duolu", "R2-duolu", "R3-duolu".



Configure IP addresses of all the PCs and interfaces of routers as shown in the above image. A summary of IP addresses is shown in the following table.

Host / Router	Interface	Peer	IP Address	Subnet Mask	Gateway
PC-1	e0	R2 e1/1	10.0.1.2	255.255.255.0	10.0.1.1
PC-2	e0	R2 e1/2	10.0.2.2	255.255.255.0	10.0.2.1
PC-3	e0	R3 e1/1	10.0.3.2	255.255.255.0	10.0.3.1
PC-4	e0	R3 e1/2	10.0.4.2	255.255.255.0	10.0.4.1
PC-5	e0	R1 e1/1	10.0.5.2	255.255.255.0	10.0.5.1

R1	e0/0	R2 e0/0	192.168.0.1	255.255.255.0	N/A
	e1/0	R3 e0/0	192.168.2.1	255.255.255.0	N/A
	e1/1	PC-5 e0	10.0.5.1	255.255.255.0	N/A
R2	e0/0	R1 e0/0	192.168.0.2	255.255.255.0	N/A
	e1/0	R3 e1/0	192.168.1.1	255.255.255.0	N/A
	e1/1	PC-1 e0	10.0.1.1	255.255.255.0	N/A
	e1/2	PC-2 e0	10.0.2.1	255.255.255.0	N/A
R3	e0/0	R1 e1/0	192.168.2.2	255.255.255.0	N/A
	e1/0	R2 e1/0	192.168.1.2	255.255.255.0	N/A
	e1/1	PC-3 e0	10.0.3.1	255.255.255.0	N/A
	e1/2	PC-4 e0	10.0.4.1	255.255.255.0	N/A

Once you finish configuration, verify that every interface can be reached by the other end, i.e., ping the IP address of the other end. For example, open the terminal of R1 and ping 192.168.0.2, 192.168.2.2, and 10.0.5.2. They should be all reachable. If there is any issue, please check the configuration of the interfaces. Similarly ping the other end on each router and host. Note that on a router you can run ping command if you are not inside the configuration mode, like the following. On the contrary, inside the configuration mode you can not do that. You may type "do ping 192.168.0.1" to call the ping command. Note that if you forget to type "no shutdown" when configure an interface, the interface can stay down forever and unreachable from the other end.

```
R1#ping 192.168.0.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.0.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 4/10/16 ms
R1#ping 192.168.2.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.2.2, timeout is 2 seconds:
```

```
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 4/9/12 ms
R1#show ip interface brief
Interface            IP-Address      OK? Method Status                Protocol
Ethernet0/0            192.168.0.1     YES manual up                    up
Ethernet1/0            192.168.2.1     YES manual up                    up
Ethernet1/1            10.0.5.1      YES manual up                  up
Ethernet1/2            unassigned      YES unset  administratively down down
Ethernet1/3            unassigned      YES unset  administratively down down
R1#
```

Please finish the verification before proceeding. Otherwise, if there are some mistakes, esp., some interface is down, the routing protocol will ignore learning the IP subnet on that interface.

R1-michaelsneberger

```
R1#ping 192.168.0.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.0.2, timeout is 2 seconds:
.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 20/34/44 ms
R1#ping 192.168.2.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.2.2, timeout is 2 seconds:
.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 24/35/44 ms
R1#ping 10.0.5.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.0.5.2, timeout is 2 seconds:
.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 1/7/12 ms
R1#show ip interface brief
Interface                  IP-Address      OK? Method Status                Protocol
Ethernet0/0                192.168.0.1     YES manual up                    up
Serial0/0                  unassigned      YES unset  administratively down down
Ethernet1/0                192.168.2.1     YES manual up                    up
Ethernet1/1                10.0.5.1        YES manual up                    up
Ethernet1/2                unassigned      YES unset  administratively down down
Ethernet1/3                unassigned      YES unset  administratively down down
R1#
```

Try to ping PC-5 from PC-1 and take a screenshot. Apparently they are not reachable from each other. Why? Can you show a proof?

```
PC-1> ping 10.0.5.2
*10.0.1.1 icmp_seq=1 ttl=255 time=15.623 ms (ICMP type:3, code:1, Destination host unreachable)
*10.0.1.1 icmp_seq=2 ttl=255 time=15.622 ms (ICMP type:3, code:1, Destination host unreachable)
*10.0.1.1 icmp_seq=3 ttl=255 time=15.622 ms (ICMP type:3, code:1, Destination host unreachable)
*10.0.1.1 icmp_seq=4 ttl=255 time=15.622 ms (ICMP type:3, code:1, Destination host unreachable)
*10.0.1.1 icmp_seq=5 ttl=255 time=15.622 ms (ICMP type:3, code:1, Destination host unreachable)

PC-1>
```

PC-5 cannot reach PC-1 because Router 1 to which PC-5 connects does not yet have a pathway to Router 2 to which PC-1 connect in its ip route table.

Now we configure routing protocols on the three routers so that they can communicate with each other and build routing tables to allow every host to reach every other host. Type the following on each router.

On router 1 (in configuration mode):

```
R1(config)#router rip
R1(config-router)#version 2
R1(config-router)#no auto-summary
R1(config-router)#network 10.0.5.0
R1(config-router)#network 192.168.0.0
R1(config-router)#network 192.168.2.0
```

On router 2 (in configuration mode):

```
R2(config)#router rip
R2(config-router)#version 2
R2(config-router)#no auto-summary
R2(config-router)#network 10.0.2.0
R2(config-router)#network 10.0.1.0
R2(config-router)#network 192.168.0.0
R2(config-router)#network 192.168.1.0
```

On router 3 (in configuration mode):

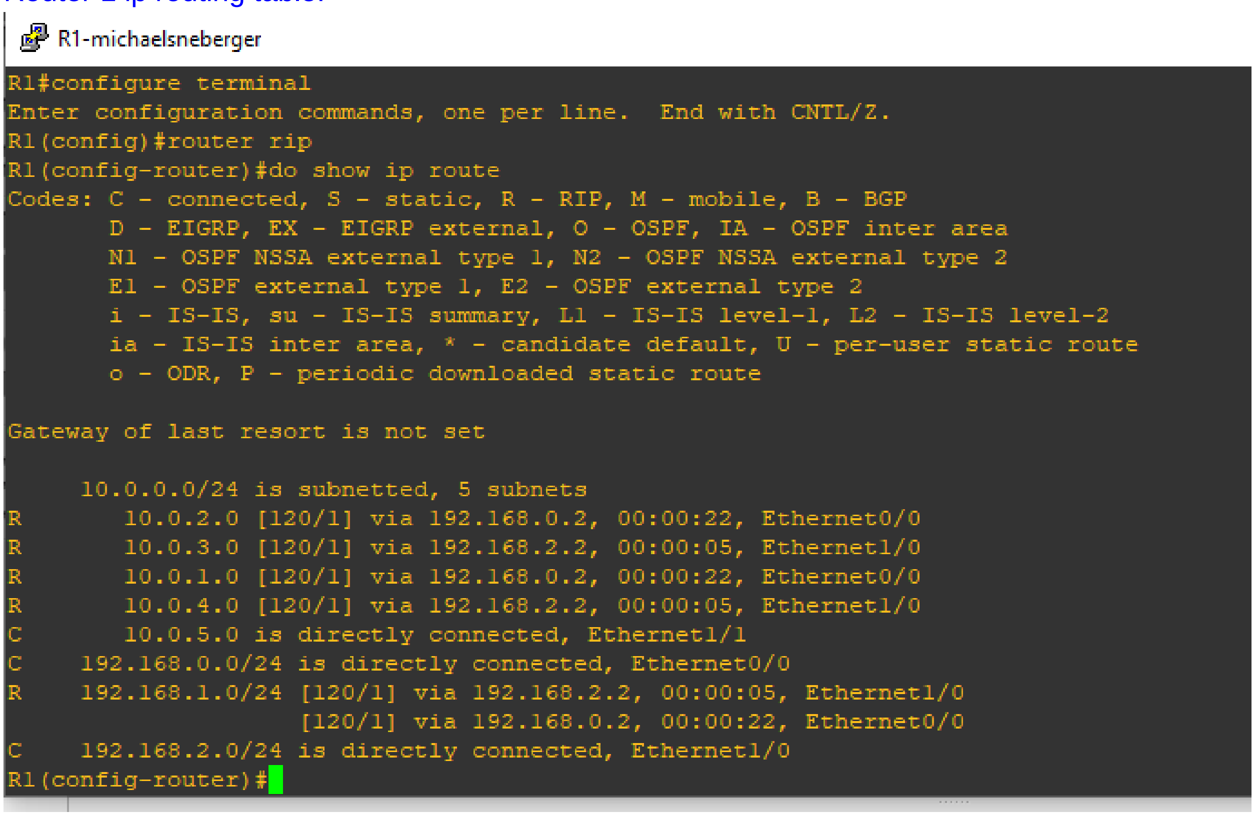
```
R3(config)#router rip
R3(config-router)#version 2
R3(config-router)#no auto-summary
R3(config-router)#network 10.0.3.0
R3(config-router)#network 10.0.4.0
R3(config-router)#network 192.168.1.0
R3(config-router)#network 192.168.2.0
```

The "router rip" and "version 2" command enables routing using RIP version 2. The "no auto-summary" command disable automatic combination of multiple destination network into a supernet. The "network xxx.xxx.xxx.xxx" command enables routing on that network. Typically we enable the directly connected network of the router.

After that, show the IP routing table of each router and take a screenshot. You will be able to see the learned entries with a preceding "R". Note that you can type "show ip route" inside the configuration mode by putting a prefix of "do" like the following.

```
R1(config-router)#do show ip route
Codes: C - connected, S - static, 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
       i - IS-IS, su - IS-IS summary, 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

     10.0.0.0/24 is subnetted, 3 subnets
R       10.0.2.0 [120/1] via 192.168.0.2, 00:00:01, Ethernet0/0
R       10.0.1.0 [120/1] via 192.168.0.2, 00:00:01, Ethernet0/0
C       10.0.5.0 is directly connected, Ethernet1/1
C    192.168.0.0/24 is directly connected, Ethernet0/0
R    192.168.1.0/24 [120/1] via 192.168.0.2, 00:00:01, Ethernet0/0
C    192.168.2.0/24 is directly connected, Ethernet1/0
R1(config-router)#
```

Router 1 ip routing table:



```
R1-michaelsneberger
R1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#router rip
R1(config-router)#do show ip route
Codes: C - connected, S - static, 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
       i - IS-IS, su - IS-IS summary, 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

     10.0.0.0/24 is subnetted, 5 subnets
R       10.0.2.0 [120/1] via 192.168.0.2, 00:00:22, Ethernet0/0
R       10.0.3.0 [120/1] via 192.168.2.2, 00:00:05, Ethernet1/0
R       10.0.1.0 [120/1] via 192.168.0.2, 00:00:22, Ethernet0/0
R       10.0.4.0 [120/1] via 192.168.2.2, 00:00:05, Ethernet1/0
C       10.0.5.0 is directly connected, Ethernet1/1
C    192.168.0.0/24 is directly connected, Ethernet0/0
R    192.168.1.0/24 [120/1] via 192.168.2.2, 00:00:05, Ethernet1/0
                    [120/1] via 192.168.0.2, 00:00:22, Ethernet0/0
C    192.168.2.0/24 is directly connected, Ethernet1/0
R1(config-router)#
```

The rip procedure provides the necessary pathways for the router to reach devices on the network

Router 2 ip routing table:

R2-michaelsneberger

```
R2-michaelsneberger#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R2-michaelsneberger(config)#router rip
R2-michaelsneberger(config-router)#do show ip route
Codes: C - connected, S - static, 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
       i - IS-IS, su - IS-IS summary, 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

     10.0.0.0/24 is subnetted, 5 subnets
C       10.0.2.0 is directly connected, Ethernet1/2
R       10.0.3.0 [120/2] via 192.168.0.1, 00:00:11, Ethernet0/0
C       10.0.1.0 is directly connected, Ethernet1/1
R       10.0.4.0 [120/2] via 192.168.0.1, 00:00:11, Ethernet0/0
R       10.0.5.0 [120/1] via 192.168.0.1, 00:00:11, Ethernet0/0
C    192.168.0.0/24 is directly connected, Ethernet0/0
C    192.168.1.0/24 is directly connected, Ethernet1/0
R    192.168.2.0/24 [120/1] via 192.168.0.1, 00:00:11, Ethernet0/0
R2-michaelsneberger(config-router)#
```

The rip procedure provides the necessary pathways for the router to reach devices on the network. Here it even doubled up due to the three routers being in a triangular arrangement.

Router 3 ip routing table:

R3-michaelsneberger

```
R3-michaelsneberger#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R3-michaelsneberger(config)#router rip
R3-michaelsneberger(config-router)#do show ip route
Codes: C - connected, S - static, 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
       i - IS-IS, su - IS-IS summary, 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

     10.0.0.0/24 is subnetted, 5 subnets
R       10.0.2.0 [120/2] via 192.168.2.1, 00:00:00, Ethernet0/0
C       10.0.3.0 is directly connected, Ethernet1/1
R       10.0.1.0 [120/2] via 192.168.2.1, 00:00:00, Ethernet0/0
C       10.0.4.0 is directly connected, Ethernet1/2
R       10.0.5.0 [120/1] via 192.168.2.1, 00:00:00, Ethernet0/0
R    192.168.0.0/24 [120/1] via 192.168.2.1, 00:00:00, Ethernet0/0
C    192.168.1.0/24 is directly connected, Ethernet1/0
C    192.168.2.0/24 is directly connected, Ethernet0/0
R3-michaelsneberger(config-router)#
```

The rip procedure provides the necessary pathways for the router to reach devices on the network. Here it even doubled up due to the three routers being in a triangular arrangement.

Finally, try to ping PC-2 from PC-5. Try multiple times if the first ping is not successful, and take a screenshot. Make sure every host is reachable from every other host. Also show the IP routing table of each router and take screenshots **(see above)** Briefly explain what happened when we configured each router using RIP and explain why the ping packets are successfully delivered.

PC-5 can reach PC-2, PC-1, PC-3, and PC4 respectively:

```
PC-5> ping 10.0.2.2
10.0.2.2 icmp_seq=1 timeout
84 bytes from 10.0.2.2 icmp_seq=2 ttl=62 time=31.914 ms
84 bytes from 10.0.2.2 icmp_seq=3 ttl=62 time=33.910 ms
84 bytes from 10.0.2.2 icmp_seq=4 ttl=62 time=62.487 ms
84 bytes from 10.0.2.2 icmp_seq=5 ttl=62 time=78.106 ms

PC-5> ping 10.0.1.2
10.0.1.2 icmp_seq=1 timeout
10.0.1.2 icmp_seq=2 timeout
84 bytes from 10.0.1.2 icmp_seq=3 ttl=62 time=57.301 ms
84 bytes from 10.0.1.2 icmp_seq=4 ttl=62 time=22.939 ms
84 bytes from 10.0.1.2 icmp_seq=5 ttl=62 time=43.883 ms

PC-5> ping 10.0.3.2
10.0.3.2 icmp_seq=1 timeout
84 bytes from 10.0.3.2 icmp_seq=2 ttl=62 time=46.864 ms
84 bytes from 10.0.3.2 icmp_seq=3 ttl=62 time=46.865 ms
84 bytes from 10.0.3.2 icmp_seq=4 ttl=62 time=62.485 ms
84 bytes from 10.0.3.2 icmp_seq=5 ttl=62 time=62.489 ms

PC-5> ping 10.0.4.2
10.0.4.2 icmp_seq=1 timeout
84 bytes from 10.0.4.2 icmp_seq=2 ttl=62 time=14.960 ms
84 bytes from 10.0.4.2 icmp_seq=3 ttl=62 time=24.934 ms
84 bytes from 10.0.4.2 icmp_seq=4 ttl=62 time=44.880 ms
84 bytes from 10.0.4.2 icmp_seq=5 ttl=62 time=12.965 ms

PC-5>
```

Subsequent testing showed that each VPCS can reach each other VPCS.

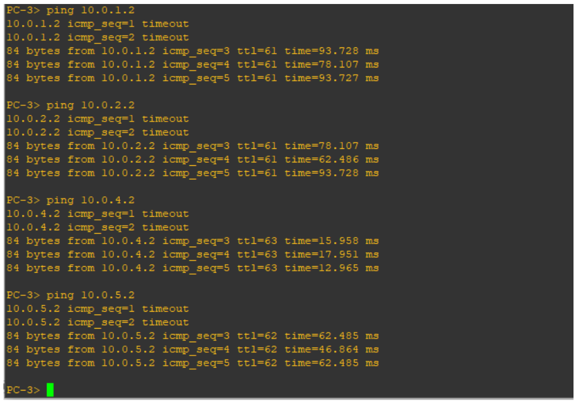
PC-1 can reach all others:

```
PC-1> ping 10.0.2.2
10.0.2.2 icmp_seq=1 timeout
10.0.2.2 icmp_seq=2 timeout
84 bytes from 10.0.2.2 icmp_seq=3 ttl=63 time=31.242 ms
84 bytes from 10.0.2.2 icmp_seq=4 ttl=63 time=31.244 ms
84 bytes from 10.0.2.2 icmp_seq=5 ttl=63 time=31.242 ms

PC-1> ping 10.0.3.2
10.0.3.2 icmp_seq=1 timeout
10.0.3.2 icmp_seq=2 timeout
84 bytes from 10.0.3.2 icmp_seq=3 ttl=61 time=93.729 ms
84 bytes from 10.0.3.2 icmp_seq=4 ttl=61 time=62.485 ms
84 bytes from 10.0.3.2 icmp_seq=5 ttl=61 time=78.107 ms

PC-1> ping 10.0.5.2
10.0.5.2 icmp_seq=1 timeout
10.0.5.2 icmp_seq=2 timeout
84 bytes from 10.0.5.2 icmp_seq=3 ttl=62 time=46.864 ms
84 bytes from 10.0.5.2 icmp_seq=4 ttl=62 time=62.485 ms
84 bytes from 10.0.5.2 icmp_seq=5 ttl=62 time=46.864 ms

PC-1> ping 10.0.4.2
10.0.4.2 icmp_seq=1 timeout
10.0.4.2 icmp_seq=2 timeout
84 bytes from 10.0.4.2 icmp_seq=3 ttl=61 time=78.106 ms
84 bytes from 10.0.4.2 icmp_seq=4 ttl=61 time=63.888 ms
84 bytes from 10.0.4.2 icmp_seq=5 ttl=61 time=20.943 ms

PC-1>
```

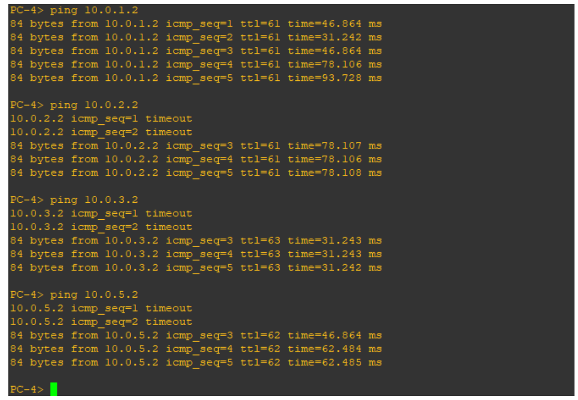
PC-2 can reach all other:

```
PC-2> ping 10.0.1.2
10.0.1.2 icmp_seq=1 timeout
84 bytes from 10.0.1.2 icmp_seq=2 ttl=63 time=1025.070 ms
84 bytes from 10.0.1.2 icmp_seq=3 ttl=63 time=26.414 ms
84 bytes from 10.0.1.2 icmp_seq=4 ttl=63 time=17.951 ms
84 bytes from 10.0.1.2 icmp_seq=5 ttl=63 time=12.965 ms

PC-2> ping 10.0.3.2
10.0.3.2 icmp_seq=1 timeout
10.0.3.2 icmp_seq=2 timeout
84 bytes from 10.0.3.2 icmp_seq=3 ttl=61 time=78.106 ms
84 bytes from 10.0.3.2 icmp_seq=4 ttl=61 time=62.490 ms
84 bytes from 10.0.3.2 icmp_seq=5 ttl=61 time=78.107 ms

PC-2> ping 10.0.4.2
10.0.4.2 icmp_seq=1 timeout
10.0.4.2 icmp_seq=2 timeout
84 bytes from 10.0.4.2 icmp_seq=3 ttl=61 time=93.729 ms
84 bytes from 10.0.4.2 icmp_seq=4 ttl=61 time=62.485 ms
84 bytes from 10.0.4.2 icmp_seq=5 ttl=61 time=78.106 ms

PC-2> ping 10.0.5.2
10.0.5.2 icmp_seq=1 timeout
10.0.5.2 icmp_seq=2 timeout
84 bytes from 10.0.5.2 icmp_seq=3 ttl=62 time=32.985 ms
84 bytes from 10.0.5.2 icmp_seq=4 ttl=62 time=62.486 ms
84 bytes from 10.0.5.2 icmp_seq=5 ttl=62 time=46.864 ms

PC-2>
```

PC-3 can reach all others:

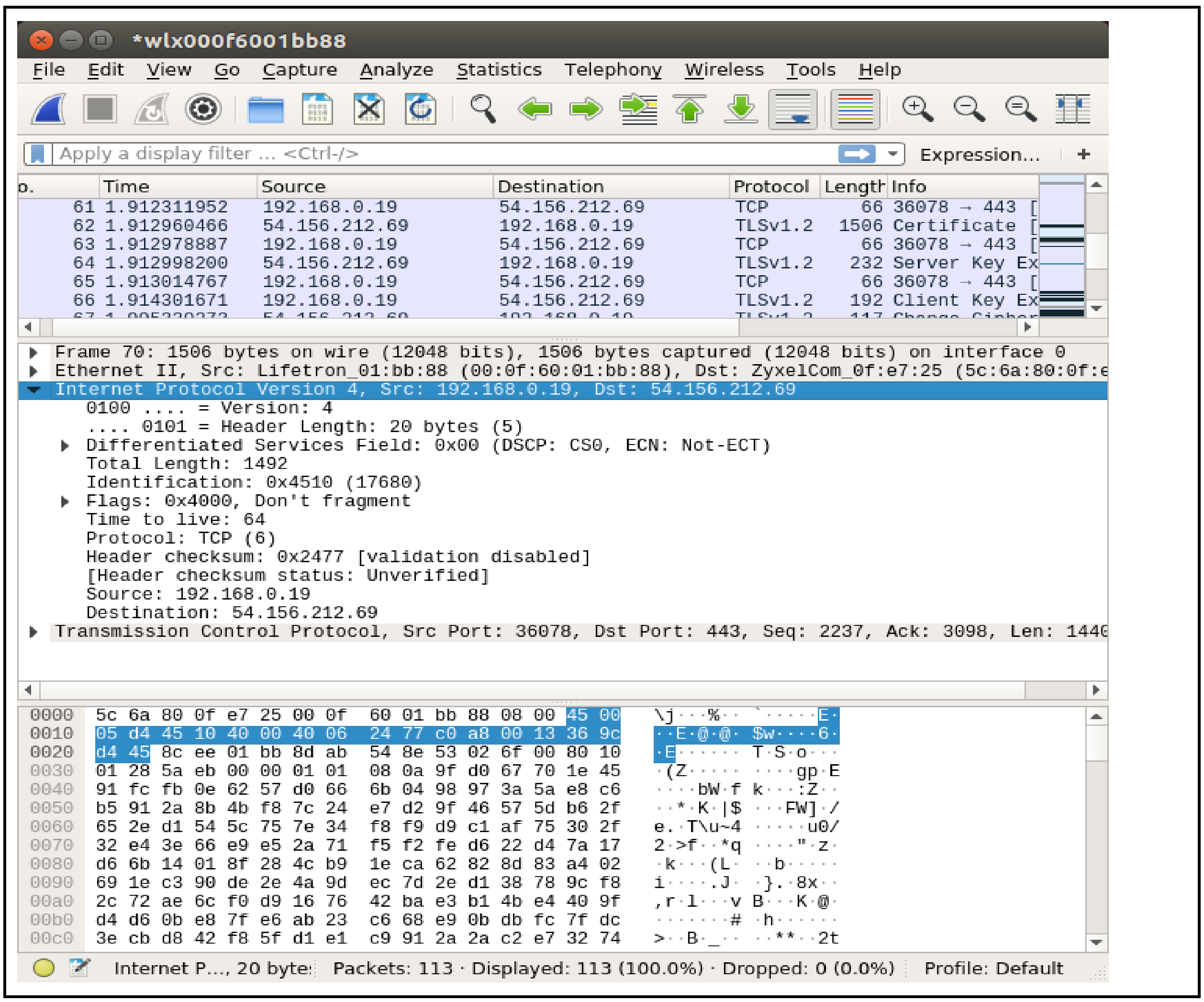


PC-4 can reach all others:



Task 6: Answer the following questions (no hands-on required):

(1) Here is an IP packet I captured.

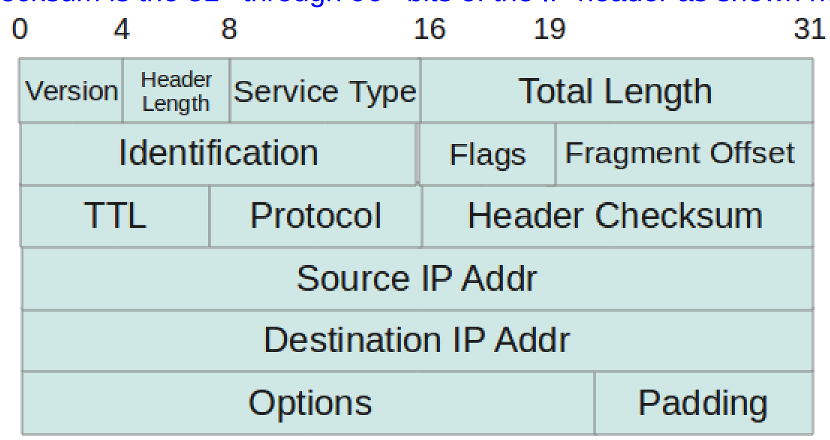


Note that the IP header contains the following.

```
45 00 05 d4 45 10 40 00 40 06 24 77 c0 a8 00 13 36 9c d4 45
```

The header checksum is 0x2477. Wireshark does not verify the checksum. Please show the steps on how the checksum is derived and verify whether this IP header is corrupted based on this checksum.

The checksum is the 81st through 96th bits of the IP header as shown here:



The checksum is calculated by using 1's compliment addition to sum all the 16-bit words in the header other than the 16-bit word containing the checksum. In our example we have the following 16-bit words (note the checksum word is switched to all 0s by the receiver after it saves the value) and we sum those up by 1's complement addition to get the checksum. In 1's complement addition if there is a carryover bit or bits on the most-significant end, you remove them and add their stand-alone value to the binary number that is left:

1)	4500 = 0100010100000000 = 17664		
2)	05d4 = 0000010111010100 = 1492	1 + 2	= 0100101011010100
3)	4510 = 0100010100010000 = 17680	plus 3	= 1000111111100100
4)	4000 = 0100000000000000 = 16384	plus 4	= 1100111111100100
5)	4006 = 0100000000000110 = 16390	plus 5	= 0000111111101011
6)	0000 =		
7)	c0a8 = 1100000010101000 = 49320	plus 7	= 1101000010010011
8)	0013 = 0000000000010011 = 19	plus 8	= 1101000010100110
9)	369c = 0011011010011100 = 13980	plus 9	= 0000011101000011
10)	d445 = 1101010001000101 = 54341	plus 10	= 1101101110001000
		1's	= **0010010001110111 = 0x2477**
		Checksum shown on Wireshark	**= 0010010001110111 = 0x2477**

The checksum DOES match so the header is NOT corrupted.

(2) Consider the Hamming (7, 4) error correction code.

If the sender wants to send four data bits (0, 1, 0, 1), what are the seven bits that actually sent on the wire?

ANSWER: Three parity bits and the four data bits.

Consider a receiver receives the following bit sequence, please answer whether there is an error, which bit is the error bit, what is the corrected bit sequence by the decoder, and what is the data bits transmitted by the sender.

Position 1 checks bits 3,5,7
Position 2 checks bits 3,6,7
Position 4 checks bits 5,6,7

A. (0, 1, 0, 0, 1, 0, **0**) = error 0@1 = even parity for bits 3, 5, 7 and 010 = odd parity
= error 1@2 = odd parity for bits 3, 6, 7 and 000 = even parity
= error 0@4 = even parity for bits 5, 6, 7 and 100 = odd parity

Here parity bits in positions 1, 2, and 4 indicate errors so 1 + 2 + 4 or **position 7 is incorrect. The data that was transmitted by the sender was (0, 1, 0, 1)** not (0, 1, 0, 0)

B. (0, 1, 0, 0, 1, **1**, 1) = OK 0@1 = even parity for bits 3, 5, 7 and 0011 = even parity
= error 1@2 = even odd bits for 3, 6, 7 and 011 = even parity
= error 0@4 = even parity for bits5, 6,7 and 111 = odd parity

Here parity bits in positions 2, and 4 indicates errors so 2 + 4 or **position 6 is incorrect. The data that was transmitted by the sender was (0, 1, 0, 1)** not (0, 1, 1, 1)

C. (0, 1, 0, 0, **0**, 0, 1) = error 0@1 = even parity for bits 3, 5, 7 and 001 = odd parity
= OK 1@2 = odd parity for bits 3, 6, 7 and 001 = odd parity
= error 0@4 = even parity for bits 5, 6, 7 and 001 = odd parity

Here parity bits in positions 1, and 4 indicate errors so 1 + 4 or **position 5 is incorrect. The data that was transmitted by the sender was (0, 1, 0, 1)** not (0, 0, 0, 1)

D. (0, 1, 0, **1**, 1, 0, 1) = OK 0@1 = even parity for bits for 3, 5, 7 and 011 = even parity
= OK 1@2 = odd parity for bits 3, 6, 7 and 001 = odd parity
= error 1@4 = odd parity for bits 5, 6, 7 and 101 = even parity

Here only parity bit in position 4 indicates error so **position 4 is incorrect. The data that was transmitted by the sender was (0, 1, 0, 1)**

E. (0, 1, **1**, 0, 1, 0, 1) = error 0@1 = even parity for bits 1, 3, 5, 7 and 111 = odd parity
= error 1@2 = odd parity for bits 3, 6, 7 and 101 = even parity
= OK 0@4 = even parity for bits 5, 6, 7 and 101 = even parity

Here parity bits in positions 1, and 2 indicate errors so 1 + 2 or **position 3 is incorrect. The data that was transmitted by the sender was (0, 1, 0, 1)** not (1, 1, 0, 1)

F. (0, **0**, 0, 0, 1, 0, 1) = OK 0@1 = even parity bits for 3, 5, 7 and 011 = even parity
= error 0@2 = even parity for bits 3, 6, 7 and 001 = odd parity
= OK 0@4 = even parity for bits 5, 6, 7 and 101 – even parity

Here only parity bit in position 2 indicates error so **position 2 is incorrect. The data that was transmitted by the sender was (0, 1, 0, 1)**

G. (**1**, 1, 0, 0, 1, 0, 1) = error 1@1 = odd parity for bits 3, 5, 7 and 011 = even parity
= OK 1@2 = odd parity for bits 3, 6, 7 and 001 = odd parity
= OK 0@4 = even parity for bits 5, 6, 7 and 101 = even parity

Here only parity bit in position 1 indicates error so **position 1 is incorrect. The data that was transmitted by the sender was (0, 1, 0, 1)**

What if the sender wants to send four data bits (0, 1, 0, 1), but the receiver receives (1, 1, 0, 0, 1, 0, 0)? **0)** Does the Hamming decoder report an error bit? **1)** Which bit is the error bit? **2)** What is the corrected bit sequence by the decoder? **3)** What is the decoded data bits? **4)** Are they the same as the bits sent by the sender? Why?

0) Yes the Hamming decoder would report an error
1) The bit in the 7^{th} position is the error
2) the corrected bit sequence is (0, 1, 0, 0, 1, 0, 1)
3) the decoded data bits after correction are (0, 1, 0, 1)
4) the decoded data bits are the same as the bits sent by the sender. That is the purpose of the Hamming error correction code and its decoder.

_, _, 0, _ 1, 0, 1 should be **0, 1, 0, 0 ,1, 0, 1** derived from error checking on 1, 1, 0, 0, 1, 0, 0 which shows that each parity bit 1, 2, and 4 show an error so the bit 0 in the 7^{th} position is in error.

If you feel confused with Hamming code, just read the Hamming (7, 4) code web page on wikipedia.

(3) Why the classical Ethernet has a minimum frame size of 64 bytes? (Hint: have a look at the book "Computer Networks (5th Edition)" written by Tanenbaum, page 284 to 285.)

To make it easier to distinguish valid frames from garbage. When a transceiver detects a collision it truncates the current frame so that stray bits and pieces of frames appear all the time. Requiring that a frame be at least 64 bytes long from destination address to checksum.