Cloud Computing

HomeWork-3

W1650226-Snehal Vijay Nikam

Task 1 : Defining custom topologies

1. What is the output of "nodes" and "net"

Output of nodes :

```
mininet> nodes
available nodes are:
c0 h1 h2 h3 h4 h5 h6 h7 h8 s1 s2 s3 s4 s5 s6 s7
mininet>
```

Output of net :

```
mininet> net
h1 h1-eth0:s3-eth1
h2 h2-eth0:s3-eth2
h3 h3-eth0:s4-eth1
h4 h4-eth0:s4-eth2
h5 h5-eth0:s6-eth1
h6 h6-eth0:s6-eth2
h7 h7-eth0:s7-eth1
h8 h8-eth0:s7-eth2
s1 lo:  s1-eth1:s2-eth3 s1-eth2:s5-eth3
s2 lo:  s2-eth1:s3-eth3 s2-eth2:s4-eth3 s2-eth3:s1-eth1
s3 lo:  s3-eth1:h1-eth0 s3-eth2:h2-eth0 s3-eth3:s2-eth1
s4 lo:  s4-eth1:h3-eth0 s4-eth2:h4-eth0 s4-eth3:s2-eth2
s5 lo:  s5-eth1:s6-eth3 s5-eth2:s7-eth3 s5-eth3:s1-eth2
s6 lo:  s6-eth1:h5-eth0 s6-eth2:h6-eth0 s6-eth3:s5-eth1
s7 lo:  s7-eth1:h7-eth0 s7-eth2:h8-eth0 s7-eth3:s5-eth2
c0
mininet>
```

2. What is the output of "h7 ifconfig"

```
mininet> h7 ifconfig
h7-eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.0.7  netmask 255.0.0.0  broadcast 10.255.255.255
        inet6 fe80::d4b4:ff:fe69:7a13  prefixlen 64  scopeid 0x20<link
        ether d6:b4:00:69:7a:13  txqueuelen 1000  (Ethernet)
        RX packets 61  bytes 4802 (4.8 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 9  bytes 726 (726.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

Task 2 :Analyze the "of_tutorial' controller

1. Draw the function call graph of this controller. For example, once a packet comes to the controller, which function is the first to be called, which one is the second, and so forth?

1. Starting the POX Listener: To initiate the POX listener, execute the following command: ./pox.py log.level --DEBUG misc.of_tutorial
2. Function Call Graph: The sequence of function calls is as follows:
    - Start Switch: This command triggers the method _handle_PacketIn(). This method handles packets received from the switch in the message.

- _handle_PacketIn(): Upon receiving a packet, this method calls the act_like_hub() method.
- act_like_hub(): Simulating a hub environment, this method sends the packets to all ports except the input port.
- resend_packet(): Following act_like_hub(), this method adds a packet to the message data and performs an action on it.
- The switch is then instructed to resend the packet to a specific port using this message.

2. Have h1 ping h2, and h1 ping h8 for 100 times (e.g., h1 ping -c100 p2).

a. How long does it take (on average) to ping for each case?

b. What is the minimum and maximum ping you have observed?

c. What is the difference, and why?

→

a. For h1 ping h2, 100 times, on average it takes about 1.524 ms to ping.
For h1 ping h8, 100 times, it takes about an average of 4.880 ms to ping.

b. For h1 ping h2 100 times, the min ping takes 1.269 ms and max ping takes 3.647 ms.
For h1 ping h8 100 times, the min ping takes 4.546 ms and max ping takes 10.393 ms.

c. h1 can communicate with h8 slower than h2 because there are more connections (hops) to travel through. h1 only needs to go through one switch (s3) to reach h2. On the other hand, h1 needs to travel through several switches (s3, s2, s1, s5, and s7) to reach h8, adding extra time to the journey.

Cloud Computing

HomeWork-3

W1650226-Snehal Vijay Nikam

Screenshots for h1 ping -c100 h2

```
mininet> h1 ping -c100 h2
PING 10.0.0.2 (10.0.0.2) 56(84) bytes of data.
64 bytes from 10.0.0.2: icmp_seq=1 ttl=64 time=3.65 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=1.44 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=1.46 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=1.34 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=1.89 ms
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=1.40 ms
64 bytes from 10.0.0.2: icmp_seq=7 ttl=64 time=1.54 ms
64 bytes from 10.0.0.2: icmp_seq=8 ttl=64 time=1.53 ms
64 bytes from 10.0.0.2: icmp_seq=9 ttl=64 time=1.44 ms
64 bytes from 10.0.0.2: icmp_seq=10 ttl=64 time=1.31 ms
64 bytes from 10.0.0.2: icmp_seq=11 ttl=64 time=1.39 ms
64 bytes from 10.0.0.2: icmp_seq=12 ttl=64 time=1.37 ms
64 bytes from 10.0.0.2: icmp_seq=13 ttl=64 time=1.59 ms
64 bytes from 10.0.0.2: icmp_seq=14 ttl=64 time=1.47 ms
64 bytes from 10.0.0.2: icmp_seq=15 ttl=64 time=1.43 ms
64 bytes from 10.0.0.2: icmp_seq=16 ttl=64 time=1.41 ms
64 bytes from 10.0.0.2: icmp_seq=17 ttl=64 time=1.56 ms
64 bytes from 10.0.0.2: icmp_seq=18 ttl=64 time=1.45 ms
64 bytes from 10.0.0.2: icmp_seq=19 ttl=64 time=1.43 ms
64 bytes from 10.0.0.2: icmp_seq=20 ttl=64 time=1.60 ms
64 bytes from 10.0.0.2: icmp_seq=21 ttl=64 time=1.78 ms
64 bytes from 10.0.0.2: icmp_seq=22 ttl=64 time=1.66 ms
64 bytes from 10.0.0.2: icmp_seq=23 ttl=64 time=1.58 ms
64 bytes from 10.0.0.2: icmp_seq=24 ttl=64 time=1.61 ms
64 bytes from 10.0.0.2: icmp_seq=25 ttl=64 time=1.44 ms
64 bytes from 10.0.0.2: icmp_seq=26 ttl=64 time=1.56 ms
64 bytes from 10.0.0.2: icmp_seq=27 ttl=64 time=1.50 ms
64 bytes from 10.0.0.2: icmp_seq=28 ttl=64 time=1.43 ms
64 bytes from 10.0.0.2: icmp_seq=29 ttl=64 time=1.40 ms
64 bytes from 10.0.0.2: icmp_seq=30 ttl=64 time=1.46 ms
64 bytes from 10.0.0.2: icmp_seq=31 ttl=64 time=1.64 ms
64 bytes from 10.0.0.2: icmp_seq=32 ttl=64 time=1.65 ms
64 bytes from 10.0.0.2: icmp_seq=33 ttl=64 time=1.58 ms
64 bytes from 10.0.0.2: icmp_seq=34 ttl=64 time=1.35 ms
64 bytes from 10.0.0.2: icmp_seq=35 ttl=64 time=1.54 ms
64 bytes from 10.0.0.2: icmp_seq=36 ttl=64 time=1.60 ms
64 bytes from 10.0.0.2: icmp_seq=37 ttl=64 time=1.45 ms
64 bytes from 10.0.0.2: icmp_seq=38 ttl=64 time=1.36 ms
64 bytes from 10.0.0.2: icmp_seq=39 ttl=64 time=1.36 ms
64 bytes from 10.0.0.2: icmp_seq=40 ttl=64 time=1.27 ms
```

Cloud Computing

HomeWork-3

W1650226-Snehal Vijay Nikam

```
64 bytes from 10.0.0.2: icmp_seq=41 ttl=64 time=1.49 ms
64 bytes from 10.0.0.2: icmp_seq=42 ttl=64 time=1.39 ms
64 bytes from 10.0.0.2: icmp_seq=43 ttl=64 time=1.30 ms
64 bytes from 10.0.0.2: icmp_seq=44 ttl=64 time=1.69 ms
64 bytes from 10.0.0.2: icmp_seq=45 ttl=64 time=1.34 ms
64 bytes from 10.0.0.2: icmp_seq=46 ttl=64 time=1.60 ms
64 bytes from 10.0.0.2: icmp_seq=47 ttl=64 time=1.50 ms
64 bytes from 10.0.0.2: icmp_seq=48 ttl=64 time=1.38 ms
64 bytes from 10.0.0.2: icmp_seq=49 ttl=64 time=1.42 ms
64 bytes from 10.0.0.2: icmp_seq=50 ttl=64 time=1.54 ms
64 bytes from 10.0.0.2: icmp_seq=51 ttl=64 time=1.43 ms
64 bytes from 10.0.0.2: icmp_seq=52 ttl=64 time=1.51 ms
64 bytes from 10.0.0.2: icmp_seq=53 ttl=64 time=1.56 ms
64 bytes from 10.0.0.2: icmp_seq=54 ttl=64 time=1.59 ms
64 bytes from 10.0.0.2: icmp_seq=55 ttl=64 time=1.60 ms
64 bytes from 10.0.0.2: icmp_seq=56 ttl=64 time=1.61 ms
64 bytes from 10.0.0.2: icmp_seq=57 ttl=64 time=1.50 ms
64 bytes from 10.0.0.2: icmp_seq=58 ttl=64 time=1.50 ms
64 bytes from 10.0.0.2: icmp_seq=59 ttl=64 time=1.37 ms
64 bytes from 10.0.0.2: icmp_seq=60 ttl=64 time=1.44 ms
64 bytes from 10.0.0.2: icmp_seq=61 ttl=64 time=1.48 ms
64 bytes from 10.0.0.2: icmp_seq=62 ttl=64 time=1.43 ms
64 bytes from 10.0.0.2: icmp_seq=63 ttl=64 time=1.49 ms
64 bytes from 10.0.0.2: icmp_seq=64 ttl=64 time=1.48 ms
64 bytes from 10.0.0.2: icmp_seq=65 ttl=64 time=1.72 ms
64 bytes from 10.0.0.2: icmp_seq=66 ttl=64 time=1.76 ms
64 bytes from 10.0.0.2: icmp_seq=67 ttl=64 time=1.58 ms
64 bytes from 10.0.0.2: icmp_seq=68 ttl=64 time=1.65 ms
64 bytes from 10.0.0.2: icmp_seq=69 ttl=64 time=1.35 ms
64 bytes from 10.0.0.2: icmp_seq=70 ttl=64 time=1.45 ms
64 bytes from 10.0.0.2: icmp_seq=71 ttl=64 time=1.56 ms
64 bytes from 10.0.0.2: icmp_seq=72 ttl=64 time=1.49 ms
64 bytes from 10.0.0.2: icmp_seq=73 ttl=64 time=1.69 ms
64 bytes from 10.0.0.2: icmp_seq=74 ttl=64 time=1.60 ms
64 bytes from 10.0.0.2: icmp_seq=75 ttl=64 time=1.62 ms
64 bytes from 10.0.0.2: icmp_seq=76 ttl=64 time=1.53 ms
64 bytes from 10.0.0.2: icmp_seq=77 ttl=64 time=1.36 ms
64 bytes from 10.0.0.2: icmp_seq=78 ttl=64 time=1.52 ms
64 bytes from 10.0.0.2: icmp_seq=79 ttl=64 time=1.49 ms
64 bytes from 10.0.0.2: icmp_seq=80 ttl=64 time=1.51 ms
64 bytes from 10.0.0.2: icmp_seq=81 ttl=64 time=1.49 ms
64 bytes from 10.0.0.2: icmp_seq=82 ttl=64 time=1.42 ms
64 bytes from 10.0.0.2: icmp_seq=83 ttl=64 time=1.46 ms
```

```
64 bytes from 10.0.0.2: icmp_seq=84 ttl=64 time=1.53 ms
64 bytes from 10.0.0.2: icmp_seq=85 ttl=64 time=1.34 ms
64 bytes from 10.0.0.2: icmp_seq=86 ttl=64 time=1.52 ms
64 bytes from 10.0.0.2: icmp_seq=87 ttl=64 time=1.60 ms
64 bytes from 10.0.0.2: icmp_seq=88 ttl=64 time=1.46 ms
64 bytes from 10.0.0.2: icmp_seq=89 ttl=64 time=1.40 ms
64 bytes from 10.0.0.2: icmp_seq=90 ttl=64 time=1.52 ms
64 bytes from 10.0.0.2: icmp_seq=91 ttl=64 time=1.64 ms
64 bytes from 10.0.0.2: icmp_seq=92 ttl=64 time=1.30 ms
64 bytes from 10.0.0.2: icmp_seq=93 ttl=64 time=1.41 ms
64 bytes from 10.0.0.2: icmp_seq=94 ttl=64 time=1.52 ms
64 bytes from 10.0.0.2: icmp_seq=95 ttl=64 time=1.68 ms
64 bytes from 10.0.0.2: icmp_seq=96 ttl=64 time=1.56 ms
64 bytes from 10.0.0.2: icmp_seq=97 ttl=64 time=1.51 ms
64 bytes from 10.0.0.2: icmp_seq=98 ttl=64 time=1.51 ms
64 bytes from 10.0.0.2: icmp_seq=99 ttl=64 time=1.60 ms
64 bytes from 10.0.0.2: icmp_seq=100 ttl=64 time=1.51 ms

--- 10.0.0.2 ping statistics ---
100 packets transmitted, 100 received, 0% packet loss, time
rtt min/avg/max/mdev = 1.269/1.524/3.647/0.241 ms
mininet>
```

Cloud Computing

HomeWork-3

W1650226-Snehal Vijay Nikam

Screenshot for h1 ping -c100 h8

```
mininet> h1 ping -c100 h8
PING 10.0.0.8 (10.0.0.8) 56(84) bytes of data.
64 bytes from 10.0.0.8: icmp_seq=1 ttl=64 time=10.4 ms
64 bytes from 10.0.0.8: icmp_seq=2 ttl=64 time=4.88 ms
64 bytes from 10.0.0.8: icmp_seq=3 ttl=64 time=4.69 ms
64 bytes from 10.0.0.8: icmp_seq=4 ttl=64 time=4.76 ms
64 bytes from 10.0.0.8: icmp_seq=5 ttl=64 time=4.85 ms
64 bytes from 10.0.0.8: icmp_seq=6 ttl=64 time=4.73 ms
64 bytes from 10.0.0.8: icmp_seq=7 ttl=64 time=4.78 ms
64 bytes from 10.0.0.8: icmp_seq=8 ttl=64 time=4.87 ms
64 bytes from 10.0.0.8: icmp_seq=9 ttl=64 time=4.77 ms
64 bytes from 10.0.0.8: icmp_seq=10 ttl=64 time=4.79 ms
64 bytes from 10.0.0.8: icmp_seq=11 ttl=64 time=4.86 ms
64 bytes from 10.0.0.8: icmp_seq=12 ttl=64 time=4.80 ms
64 bytes from 10.0.0.8: icmp_seq=13 ttl=64 time=4.89 ms
64 bytes from 10.0.0.8: icmp_seq=14 ttl=64 time=4.66 ms
64 bytes from 10.0.0.8: icmp_seq=15 ttl=64 time=4.88 ms
64 bytes from 10.0.0.8: icmp_seq=16 ttl=64 time=4.79 ms
64 bytes from 10.0.0.8: icmp_seq=17 ttl=64 time=4.76 ms
64 bytes from 10.0.0.8: icmp_seq=18 ttl=64 time=4.84 ms
64 bytes from 10.0.0.8: icmp_seq=19 ttl=64 time=4.72 ms
64 bytes from 10.0.0.8: icmp_seq=20 ttl=64 time=4.84 ms
64 bytes from 10.0.0.8: icmp_seq=21 ttl=64 time=4.82 ms
64 bytes from 10.0.0.8: icmp_seq=22 ttl=64 time=4.77 ms
64 bytes from 10.0.0.8: icmp_seq=23 ttl=64 time=4.67 ms
64 bytes from 10.0.0.8: icmp_seq=24 ttl=64 time=4.95 ms
64 bytes from 10.0.0.8: icmp_seq=25 ttl=64 time=4.95 ms
64 bytes from 10.0.0.8: icmp_seq=26 ttl=64 time=4.93 ms
64 bytes from 10.0.0.8: icmp_seq=27 ttl=64 time=4.94 ms
64 bytes from 10.0.0.8: icmp_seq=28 ttl=64 time=4.84 ms
64 bytes from 10.0.0.8: icmp_seq=29 ttl=64 time=4.81 ms
64 bytes from 10.0.0.8: icmp_seq=30 ttl=64 time=4.75 ms
64 bytes from 10.0.0.8: icmp_seq=31 ttl=64 time=4.86 ms
64 bytes from 10.0.0.8: icmp_seq=32 ttl=64 time=4.90 ms
64 bytes from 10.0.0.8: icmp_seq=33 ttl=64 time=5.00 ms
64 bytes from 10.0.0.8: icmp_seq=34 ttl=64 time=4.81 ms
64 bytes from 10.0.0.8: icmp_seq=35 ttl=64 time=4.64 ms
64 bytes from 10.0.0.8: icmp_seq=36 ttl=64 time=4.76 ms
64 bytes from 10.0.0.8: icmp_seq=37 ttl=64 time=4.98 ms
64 bytes from 10.0.0.8: icmp_seq=38 ttl=64 time=4.80 ms
64 bytes from 10.0.0.8: icmp_seq=39 ttl=64 time=4.76 ms
64 bytes from 10.0.0.8: icmp_seq=40 ttl=64 time=4.90 ms
64 bytes from 10.0.0.8: icmp_seq=41 ttl=64 time=4.71 ms
64 bytes from 10.0.0.8: icmp_seq=42 ttl=64 time=4.73 ms
64 bytes from 10.0.0.8: icmp_seq=43 ttl=64 time=4.85 ms
64 bytes from 10.0.0.8: icmp_seq=44 ttl=64 time=4.75 ms
64 bytes from 10.0.0.8: icmp_seq=45 ttl=64 time=4.77 ms
64 bytes from 10.0.0.8: icmp_seq=46 ttl=64 time=4.76 ms
64 bytes from 10.0.0.8: icmp_seq=47 ttl=64 time=4.92 ms
64 bytes from 10.0.0.8: icmp_seq=48 ttl=64 time=4.86 ms
64 bytes from 10.0.0.8: icmp_seq=49 ttl=64 time=4.84 ms
64 bytes from 10.0.0.8: icmp_seq=50 ttl=64 time=4.86 ms
64 bytes from 10.0.0.8: icmp_seq=51 ttl=64 time=4.77 ms
64 bytes from 10.0.0.8: icmp_seq=52 ttl=64 time=4.78 ms
64 bytes from 10.0.0.8: icmp_seq=53 ttl=64 time=4.67 ms
64 bytes from 10.0.0.8: icmp_seq=54 ttl=64 time=4.98 ms
64 bytes from 10.0.0.8: icmp_seq=55 ttl=64 time=4.81 ms
```

Cloud Computing

HomeWork-3

W1650226-Snehal Vijay Nikam

```
64 bytes from 10.0.0.8: icmp_seq=56 ttl=64 time=4.93 ms
64 bytes from 10.0.0.8: icmp_seq=57 ttl=64 time=4.79 ms
64 bytes from 10.0.0.8: icmp_seq=58 ttl=64 time=4.84 ms
64 bytes from 10.0.0.8: icmp_seq=59 ttl=64 time=4.92 ms
64 bytes from 10.0.0.8: icmp_seq=60 ttl=64 time=5.71 ms
64 bytes from 10.0.0.8: icmp_seq=61 ttl=64 time=4.65 ms
64 bytes from 10.0.0.8: icmp_seq=62 ttl=64 time=4.80 ms
64 bytes from 10.0.0.8: icmp_seq=63 ttl=64 time=4.93 ms
64 bytes from 10.0.0.8: icmp_seq=64 ttl=64 time=4.70 ms
64 bytes from 10.0.0.8: icmp_seq=65 ttl=64 time=4.67 ms
64 bytes from 10.0.0.8: icmp_seq=66 ttl=64 time=5.67 ms
64 bytes from 10.0.0.8: icmp_seq=67 ttl=64 time=4.90 ms
64 bytes from 10.0.0.8: icmp_seq=68 ttl=64 time=4.71 ms
64 bytes from 10.0.0.8: icmp_seq=69 ttl=64 time=5.03 ms
64 bytes from 10.0.0.8: icmp_seq=70 ttl=64 time=4.90 ms
64 bytes from 10.0.0.8: icmp_seq=71 ttl=64 time=5.02 ms
64 bytes from 10.0.0.8: icmp_seq=72 ttl=64 time=4.97 ms
64 bytes from 10.0.0.8: icmp_seq=73 ttl=64 time=4.73 ms
```

```
64 bytes from 10.0.0.8: icmp_seq=74 ttl=64 time=4.69 ms
64 bytes from 10.0.0.8: icmp_seq=75 ttl=64 time=4.67 ms
64 bytes from 10.0.0.8: icmp_seq=76 ttl=64 time=4.74 ms
64 bytes from 10.0.0.8: icmp_seq=77 ttl=64 time=4.78 ms
64 bytes from 10.0.0.8: icmp_seq=78 ttl=64 time=4.71 ms
64 bytes from 10.0.0.8: icmp_seq=79 ttl=64 time=4.65 ms
64 bytes from 10.0.0.8: icmp_seq=80 ttl=64 time=4.71 ms
64 bytes from 10.0.0.8: icmp_seq=81 ttl=64 time=4.91 ms
64 bytes from 10.0.0.8: icmp_seq=82 ttl=64 time=4.76 ms
64 bytes from 10.0.0.8: icmp_seq=83 ttl=64 time=4.90 ms
64 bytes from 10.0.0.8: icmp_seq=84 ttl=64 time=4.71 ms
64 bytes from 10.0.0.8: icmp_seq=85 ttl=64 time=4.85 ms
64 bytes from 10.0.0.8: icmp_seq=86 ttl=64 time=4.69 ms
64 bytes from 10.0.0.8: icmp_seq=87 ttl=64 time=4.55 ms
64 bytes from 10.0.0.8: icmp_seq=88 ttl=64 time=4.80 ms
64 bytes from 10.0.0.8: icmp_seq=89 ttl=64 time=4.63 ms
64 bytes from 10.0.0.8: icmp_seq=90 ttl=64 time=4.76 ms
64 bytes from 10.0.0.8: icmp_seq=91 ttl=64 time=4.84 ms
64 bytes from 10.0.0.8: icmp_seq=92 ttl=64 time=4.94 ms
64 bytes from 10.0.0.8: icmp_seq=93 ttl=64 time=4.82 ms
64 bytes from 10.0.0.8: icmp_seq=94 ttl=64 time=4.94 ms
64 bytes from 10.0.0.8: icmp_seq=95 ttl=64 time=4.61 ms
64 bytes from 10.0.0.8: icmp_seq=96 ttl=64 time=4.86 ms
64 bytes from 10.0.0.8: icmp_seq=97 ttl=64 time=4.86 ms
64 bytes from 10.0.0.8: icmp_seq=98 ttl=64 time=4.72 ms
64 bytes from 10.0.0.8: icmp_seq=99 ttl=64 time=5.01 ms
64 bytes from 10.0.0.8: icmp_seq=100 ttl=64 time=4.72 ms

--- 10.0.0.8 ping statistics ---
100 packets transmitted, 100 received, 0% packet loss, time 99171ms
rtt min/avg/max/mdev = 4.546/4.880/10.393/0.576 ms
mininet>
```

3. Run “iperf h1 h2” and “iperf h1 h8”

a. What is “iperf” used for?

b. What is the throughput for each case?

c. What is the difference, and explain the reasons for the difference.

a. Iperf is an open-source tool designed to assist administrators in gauging the bandwidth and assessing the performance and quality of a network connection between two hosts. It enables the measurement of throughput between any two nodes on a network line.

---

b. The throughput in first case h1 to h2 is 3.78 Mbits/Sec and 2.44 Mbits/sec in second case.

```
mininet> iperf h1 h8
*** Iperf: testing TCP bandwidth between h1 and h8
*** Results: ['2.44 Mbits/sec', '2.38 Mbits/sec']
```

```
mininet> iperf h1 h2
*** Iperf: testing TCP bandwidth between h1 and h2
*** Results: ['3.78 Mbits/sec', '3.75 Mbits/sec']
```

c. h1 can move data to h2 faster than h8 because h2 is a more direct neighbor. There's just one stop (switch s3) along the way for data packets to reach h2. On the other hand, getting to h8 requires traveling through several junctions (switches s3, s2, s1, s5, and s7). Each junction introduces a slight wait, slowing down the overall flow of data (throughput) to h8.

4. Which of the switches observe traffic? Please describe your way for observing such traffic on switches (e.g., adding some functions in the “of_tutorial” controller).

The _handle_PacketIn function in of_tutorial.py acts like a traffic listener for the switches. By adding a line with log.info, we can see which switch receives packets. This is helpful because normally switches flood all ports initially, so every switch would see the traffic at least once in this scenario. Remember, flooding means sending the packet out on all ports except the one it came in on.

Task 3: MAC Learning Controller

1. Describe how the above code works, such as how the "MAC to Port" map is established. You could use a ‘ping’ example to describe the establishment process (e.g., h1 ping h2).

A key function in network management, the MAC Learning Controller, optimizes network efficiency in two ways. Firstly, it leverages a function called act_like_switch to identify the physical location (port) of network devices. This is achieved by learning their Media Access Control (MAC) addresses. With this knowledge, the controller can significantly accelerate packet delivery for known devices. Instead of flooding the network with data packets, it can direct them specifically to the port where the intended recipient resides.

Secondly, the MAC Learning Controller minimizes network congestion. By learning device locations, it reduces the need for flooding, a process that broadcasts packets to all ports on the network. This targeted approach not only improves overall network performance but also translates to faster response times (ping times) and increased data transfer rates (throughput). Even when encountering an unknown device, the controller can still deliver packets using temporary flooding until the device's location is learned.

Screenshots can be found below :

Cloud Computing

HomeWork-3

W1650226-Snehal Vijay Nikam

```
root@b242d800f2d0:~/pox# ./pox.py log.level --DEBUG misc.of_tutorial
POX 0.7.0 (gar) / Copyright 2011-2020 James McCauley, et al.
DEBUG:core:POX 0.7.0 (gar) going up...
DEBUG:core:Running on CPython (3.10.6/Mar 10 2023 10:55:28)
DEBUG:core:Platform is Linux-6.5.0-1016-azure-x86_64-with-glibc2.35
WARNING:version:POX requires one of the following versions of Python: 3.6 3.7 3.8 3.9
WARNING:version:You're running Python 3.10.
WARNING:version:If you run into problems, try using a supported version.
INFO:core:POX 0.7.0 (gar) is up.
DEBUG:openflow.of_01:Listening on 0.0.0.0:6633
INFO:openflow.of_01:[00-00-00-00-00-07 2] connected
DEBUG:misc.of_tutorial:Controlling [00-00-00-00-00-07 2]
INFO:openflow.of_01:[00-00-00-00-00-06 3] connected
DEBUG:misc.of_tutorial:Controlling [00-00-00-00-00-06 3]
INFO:openflow.of_01:[00-00-00-00-00-04 4] connected
DEBUG:misc.of_tutorial:Controlling [00-00-00-00-00-04 4]
INFO:openflow.of_01:[00-00-00-00-00-01 5] connected
DEBUG:misc.of_tutorial:Controlling [00-00-00-00-00-01 5]
INFO:openflow.of_01:[00-00-00-00-00-03 6] connected
DEBUG:misc.of_tutorial:Controlling [00-00-00-00-00-03 6]
INFO:openflow.of_01:[00-00-00-00-00-02 7] connected
DEBUG:misc.of_tutorial:Controlling [00-00-00-00-00-02 7]
INFO:openflow.of_01:[00-00-00-00-00-05 8] connected
DEBUG:misc.of_tutorial:Controlling [00-00-00-00-00-05 8]
Learning that f6:8f:99:bf:2f:aa is attached at port 2
33:33:00:00:00:16 not known, resend to everybody
Learning that f6:8f:99:bf:2f:aa is attached at port 2
33:33:00:00:00:16 not known, resend to everybody
Learning that f6:8f:99:bf:2f:aa is attached at port 3
33:33:00:00:00:16 not known, resend to everybody
Learning that f6:8f:99:bf:2f:aa is attached at port 2
33:33:00:00:00:16 not known, resend to everybody
Learning that f6:8f:99:bf:2f:aa is attached at port 3
33:33:00:00:00:16 not known, resend to everybody
Learning that f6:8f:99:bf:2f:aa is attached at port 3
33:33:00:00:00:16 not known, resend to everybody
Learning that f6:8f:99:bf:2f:aa is attached at port 3
33:33:00:00:00:16 not known, resend to everybody
Learning that 4e:9e:86:16:fa:b5 is attached at port 2
33:33:00:00:00:16 not known, resend to everybody
Learning that 4e:9e:86:16:fa:b5 is attached at port 2
33:33:00:00:00:16 not known, resend to everybody
Learning that 4e:9e:86:16:fa:b5 is attached at port 1
```

```
33:33:00:00:00:16 not known, resend to everybody
Learning that 9e:d5:5d:c7:50:a4 is attached at port 3
33:33:00:00:00:16 not known, resend to everybody
Learning that 9e:d5:5d:c7:50:a4 is attached at port 1
33:33:00:00:00:16 not known, resend to everybody
Learning that 9e:d5:5d:c7:50:a4 is attached at port 3
33:33:00:00:00:16 not known, resend to everybody
Learning that 9e:d5:5d:c7:50:a4 is attached at port 3
33:33:00:00:00:16 not known, resend to everybody
Learning that 9e:d5:5d:c7:50:a4 is attached at port 3
33:33:00:00:00:16 not known, resend to everybody
33:33:ff:bf:2f:aa not known, resend to everybody
33:33:ff:bf:2f:aa not known, resend to everybody
33:33:ff:bf:2f:aa not known, resend to everybody
33:33:ff:bf:2f:aa not known, resend to everybody
33:33:ff:bf:2f:aa not known, resend to everybody
33:33:ff:bf:2f:aa not known, resend to everybody
33:33:ff:bf:2f:aa not known, resend to everybody
Learning that be:98:c6:8c:2f:12 is attached at port 1
33:33:ff:8c:2f:12 not known, resend to everybody
Learning that ea:62:01:9d:27:26 is attached at port 2
```

Cloud Computing

HomeWork-3

W1650226-Snehal Vijay Nikam

```
33:33:ff:9d:27:26 not known, resend to everybody
Learning that be:98:c6:8c:2f:12 is attached at port 2
33:33:ff:8c:2f:12 not known, resend to everybody
Learning that ea:62:01:9d:27:26 is attached at port 1
33:33:ff:9d:27:26 not known, resend to everybody
Learning that be:98:c6:8c:2f:12 is attached at port 1
33:33:ff:8c:2f:12 not known, resend to everybody
Learning that be:98:c6:8c:2f:12 is attached at port 3
33:33:ff:8c:2f:12 not known, resend to everybody
Learning that be:98:c6:8c:2f:12 is attached at port 3
33:33:ff:8c:2f:12 not known, resend to everybody
Learning that be:98:c6:8c:2f:12 is attached at port 3
33:33:ff:8c:2f:12 not known, resend to everybody
Learning that be:98:c6:8c:2f:12 is attached at port 3
33:33:ff:8c:2f:12 not known, resend to everybody
Learning that ea:62:01:9d:27:26 is attached at port 3
33:33:ff:9d:27:26 not known, resend to everybody
Learning that ea:62:01:9d:27:26 is attached at port 1
33:33:ff:9d:27:26 not known, resend to everybody
Learning that ea:62:01:9d:27:26 is attached at port 3
33:33:ff:9d:27:26 not known, resend to everybody
Learning that ea:62:01:9d:27:26 is attached at port 3
```

```
33:33:00:00:00:16 not known, resend to everybody
Learning that 9e:d5:5d:c7:50:a4 is attached at port 3
33:33:00:00:00:16 not known, resend to everybody
Learning that 9e:d5:5d:c7:50:a4 is attached at port 1
33:33:00:00:00:16 not known, resend to everybody
Learning that 9e:d5:5d:c7:50:a4 is attached at port 3
33:33:00:00:00:16 not known, resend to everybody
Learning that 9e:d5:5d:c7:50:a4 is attached at port 3
33:33:00:00:00:16 not known, resend to everybody
Learning that 9e:d5:5d:c7:50:a4 is attached at port 3
33:33:00:00:00:16 not known, resend to everybody
33:33:ff:bf:2f:aa not known, resend to everybody
33:33:ff:bf:2f:aa not known, resend to everybody
33:33:ff:bf:2f:aa not known, resend to everybody
33:33:ff:bf:2f:aa not known, resend to everybody
33:33:ff:bf:2f:aa not known, resend to everybody
33:33:ff:bf:2f:aa not known, resend to everybody
33:33:ff:bf:2f:aa not known, resend to everybody
Learning that be:98:c6:8c:2f:12 is attached at port 1
33:33:ff:8c:2f:12 not known, resend to everybody
Learning that ea:62:01:9d:27:26 is attached at port 2
33:33:ff:9d:27:26 not known, resend to everybody
Learning that be:98:c6:8c:2f:12 is attached at port 2
33:33:ff:8c:2f:12 not known, resend to everybody
Learning that ea:62:01:9d:27:26 is attached at port 1
33:33:ff:9d:27:26 not known, resend to everybody
Learning that be:98:c6:8c:2f:12 is attached at port 1
33:33:ff:8c:2f:12 not known, resend to everybody
Learning that be:98:c6:8c:2f:12 is attached at port 3
33:33:ff:8c:2f:12 not known, resend to everybody
Learning that be:98:c6:8c:2f:12 is attached at port 3
33:33:ff:8c:2f:12 not known, resend to everybody
Learning that be:98:c6:8c:2f:12 is attached at port 3
33:33:ff:8c:2f:12 not known, resend to everybody
Learning that be:98:c6:8c:2f:12 is attached at port 3
33:33:ff:8c:2f:12 not known, resend to everybody
Learning that ea:62:01:9d:27:26 is attached at port 3
33:33:ff:9d:27:26 not known, resend to everybody
Learning that ea:62:01:9d:27:26 is attached at port 1
33:33:ff:9d:27:26 not known, resend to everybody
Learning that ea:62:01:9d:27:26 is attached at port 3
33:33:ff:9d:27:26 not known, resend to everybody
Learning that ea:62:01:9d:27:26 is attached at port 3
```

Cloud Computing

HomeWork-3

W1650226-Snehal Vijay Nikam

```
33:33:00:00:00:16 not known, resend to everybody
33:33:00:00:00:16 not known, resend to everybody
33:33:00:00:00:16 not known, resend to everybody
Learning that 36:6b:96:82:d0:4d is attached at port 1
33:33:00:00:00:16 not known, resend to everybody
33:33:ff:82:d0:4d not known, resend to everybody
Learning that 36:6b:96:82:d0:4d is attached at port 1
33:33:00:00:00:16 not known, resend to everybody
Learning that 36:6b:96:82:d0:4d is attached at port 3
33:33:00:00:00:16 not known, resend to everybody
Learning that 36:6b:96:82:d0:4d is attached at port 2
33:33:00:00:00:16 not known, resend to everybody
Learning that 36:6b:96:82:d0:4d is attached at port 3
33:33:00:00:00:16 not known, resend to everybody
Learning that 36:6b:96:82:d0:4d is attached at port 3
33:33:00:00:00:16 not known, resend to everybody
Learning that 36:6b:96:82:d0:4d is attached at port 3
33:33:00:00:00:16 not known, resend to everybody
33:33:ff:82:d0:4d not known, resend to everybody
33:33:ff:82:d0:4d not known, resend to everybody
33:33:ff:82:d0:4d not known, resend to everybody
33:33:ff:82:d0:4d not known, resend to everybody
33:33:ff:82:d0:4d not known, resend to everybody
33:33:ff:82:d0:4d not known, resend to everybody
33:33:00:00:00:16 not known, resend to everybody
33:33:00:00:00:16 not known, resend to everybody
33:33:00:00:00:16 not known, resend to everybody
33:33:00:00:00:16 not known, resend to everybody
33:33:00:00:00:16 not known, resend to everybody
33:33:00:00:00:16 not known, resend to everybody
33:33:00:00:00:16 not known, resend to everybody
```

```
9e:d5:5d:c7:50:a4 destination known. only send message to it
ea:62:01:9d:27:26 destination known. only send message to it
9e:d5:5d:c7:50:a4 destination known. only send message to it
ea:62:01:9d:27:26 destination known. only send message to it
9e:d5:5d:c7:50:a4 destination known. only send message to it
ea:62:01:9d:27:26 destination known. only send message to it
9e:d5:5d:c7:50:a4 destination known. only send message to it
ea:62:01:9d:27:26 destination known. only send message to it
9e:d5:5d:c7:50:a4 destination known. only send message to it
ea:62:01:9d:27:26 destination known. only send message to it
9e:d5:5d:c7:50:a4 destination known. only send message to it
ea:62:01:9d:27:26 destination known. only send message to it
9e:d5:5d:c7:50:a4 destination known. only send message to it
ea:62:01:9d:27:26 destination known. only send message to it
9e:d5:5d:c7:50:a4 destination known. only send message to it
ea:62:01:9d:27:26 destination known. only send message to it
9e:d5:5d:c7:50:a4 destination known. only send message to it
ea:62:01:9d:27:26 destination known. only send message to it
9e:d5:5d:c7:50:a4 destination known. only send message to it
33:33:00:00:00:02 not known, resend to everybody
33:33:00:00:00:02 not known, resend to everybody
33:33:00:00:00:02 not known, resend to everybody
33:33:00:00:00:02 not known, resend to everybody
33:33:00:00:00:02 not known, resend to everybody
33:33:00:00:00:02 not known, resend to everybody
33:33:00:00:00:02 not known, resend to everybody
ea:62:01:9d:27:26 destination known. only send message to it
9e:d5:5d:c7:50:a4 destination known. only send message to it
33:33:00:00:00:02 not known, resend to everybody
33:33:00:00:00:02 not known, resend to everybody
33:33:00:00:00:02 not known, resend to everybody
33:33:00:00:00:02 not known, resend to everybody
33:33:00:00:00:02 not known, resend to everybody
33:33:00:00:00:02 not known, resend to everybody
33:33:00:00:00:02 not known, resend to everybody
33:33:00:00:00:02 not known, resend to everybody
33:33:00:00:00:02 not known, resend to everybody
33:33:00:00:00:02 not known, resend to everybody
33:33:00:00:00:02 not known, resend to everybody
33:33:00:00:00:02 not known, resend to everybody
33:33:00:00:00:02 not known, resend to everybody
33:33:00:00:00:02 not known, resend to everybody
ea:62:01:9d:27:26 destination known. only send message to it
```

Cloud Computing

HomeWork-3

W1650226-Snehal Vijay Nikam

---

2. (Comment out all prints before doing this experiment) Have h1 ping h2, and h1 ping h8 for 100 times (e.g., h1 ping -c100 p2).
a. How long did it take (on average) to ping for each case? b. What is the minimum and maximum ping you have observed? c. Any difference from Task 2 and why do you think there is a change if there is?

a. For h1 ping h2, 100 times, on average it takes about 1.380 ms to ping.
   For h1 ping h8, 100 times, it takes about an average of 4.527 ms to ping.
b. For h1 ping h2 100 times, the min ping takes 1.115 ms and max ping takes 1.669 ms.
   For h1 ping h8 100 times, the min ping takes 4.194 ms and max ping takes 6.032 ms.
c. Task 3 demonstrates a clear advantage in ping times compared to task 2. Pings between h1 and h2 show a modest improvement, likely due to factors beyond the number of switches involved. However, pings from h1 to h8 exhibit a significant speed increase in task 3. This is primarily because task 3 utilizes a MAC Learning Controller. The MAC Learning Controller works by initially flooding the network with a few packets for unknown destinations. Once the destination's MAC address is learned, the controller directs subsequent packets directly to the appropriate port. This targeted approach reduces network congestion, especially for frequently accessed devices like h8. As a result, subsequent pings experience a noticeable improvement in speed.

h1 ping -c100 h2

```
mininet> h1 ping -c100 h2
PING 10.0.0.2 (10.0.0.2) 56(84) bytes of data.
64 bytes from 10.0.0.2: icmp_seq=1 ttl=64 time=1.52 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=1.40 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=1.32 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=1.24 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=1.38 ms
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=1.26 ms
64 bytes from 10.0.0.2: icmp_seq=7 ttl=64 time=1.45 ms
64 bytes from 10.0.0.2: icmp_seq=8 ttl=64 time=1.52 ms
64 bytes from 10.0.0.2: icmp_seq=9 ttl=64 time=1.20 ms
64 bytes from 10.0.0.2: icmp_seq=10 ttl=64 time=1.51 ms
64 bytes from 10.0.0.2: icmp_seq=11 ttl=64 time=1.22 ms
64 bytes from 10.0.0.2: icmp_seq=12 ttl=64 time=1.43 ms
64 bytes from 10.0.0.2: icmp_seq=13 ttl=64 time=1.42 ms
64 bytes from 10.0.0.2: icmp_seq=14 ttl=64 time=1.32 ms
64 bytes from 10.0.0.2: icmp_seq=15 ttl=64 time=1.37 ms
64 bytes from 10.0.0.2: icmp_seq=16 ttl=64 time=1.12 ms
64 bytes from 10.0.0.2: icmp_seq=17 ttl=64 time=1.45 ms
64 bytes from 10.0.0.2: icmp_seq=18 ttl=64 time=1.31 ms
64 bytes from 10.0.0.2: icmp_seq=19 ttl=64 time=1.32 ms
64 bytes from 10.0.0.2: icmp_seq=20 ttl=64 time=1.55 ms
64 bytes from 10.0.0.2: icmp_seq=21 ttl=64 time=1.12 ms
64 bytes from 10.0.0.2: icmp_seq=22 ttl=64 time=1.47 ms
64 bytes from 10.0.0.2: icmp_seq=23 ttl=64 time=1.50 ms
64 bytes from 10.0.0.2: icmp_seq=24 ttl=64 time=1.31 ms
64 bytes from 10.0.0.2: icmp_seq=25 ttl=64 time=1.54 ms
64 bytes from 10.0.0.2: icmp_seq=26 ttl=64 time=1.48 ms
64 bytes from 10.0.0.2: icmp_seq=27 ttl=64 time=1.32 ms
64 bytes from 10.0.0.2: icmp_seq=28 ttl=64 time=1.46 ms
64 bytes from 10.0.0.2: icmp_seq=29 ttl=64 time=1.24 ms
64 bytes from 10.0.0.2: icmp_seq=30 ttl=64 time=1.30 ms
64 bytes from 10.0.0.2: icmp_seq=31 ttl=64 time=1.12 ms
64 bytes from 10.0.0.2: icmp_seq=32 ttl=64 time=1.34 ms
64 bytes from 10.0.0.2: icmp_seq=33 ttl=64 time=1.46 ms
64 bytes from 10.0.0.2: icmp_seq=34 ttl=64 time=1.30 ms
64 bytes from 10.0.0.2: icmp_seq=35 ttl=64 time=1.52 ms
64 bytes from 10.0.0.2: icmp_seq=36 ttl=64 time=1.26 ms
64 bytes from 10.0.0.2: icmp_seq=37 ttl=64 time=1.43 ms
64 bytes from 10.0.0.2: icmp_seq=38 ttl=64 time=1.48 ms
64 bytes from 10.0.0.2: icmp_seq=39 ttl=64 time=1.29 ms
64 bytes from 10.0.0.2: icmp_seq=40 ttl=64 time=1.45 ms
64 bytes from 10.0.0.2: icmp_seq=41 ttl=64 time=1.45 ms
```

Cloud Computing

HomeWork-3

W1650226-Snehal Vijay Nikam

```
64 bytes from 10.0.0.2: icmp_seq=42 ttl=64 time=1.29 ms
64 bytes from 10.0.0.2: icmp_seq=43 ttl=64 time=1.42 ms
64 bytes from 10.0.0.2: icmp_seq=44 ttl=64 time=1.28 ms
64 bytes from 10.0.0.2: icmp_seq=45 ttl=64 time=1.24 ms
64 bytes from 10.0.0.2: icmp_seq=46 ttl=64 time=1.41 ms
64 bytes from 10.0.0.2: icmp_seq=47 ttl=64 time=1.29 ms
64 bytes from 10.0.0.2: icmp_seq=48 ttl=64 time=1.67 ms
64 bytes from 10.0.0.2: icmp_seq=49 ttl=64 time=1.49 ms
64 bytes from 10.0.0.2: icmp_seq=50 ttl=64 time=1.29 ms
64 bytes from 10.0.0.2: icmp_seq=51 ttl=64 time=1.31 ms
64 bytes from 10.0.0.2: icmp_seq=52 ttl=64 time=1.55 ms
64 bytes from 10.0.0.2: icmp_seq=53 ttl=64 time=1.37 ms
64 bytes from 10.0.0.2: icmp_seq=54 ttl=64 time=1.31 ms
64 bytes from 10.0.0.2: icmp_seq=55 ttl=64 time=1.41 ms
64 bytes from 10.0.0.2: icmp_seq=56 ttl=64 time=1.37 ms
64 bytes from 10.0.0.2: icmp_seq=57 ttl=64 time=1.47 ms
64 bytes from 10.0.0.2: icmp_seq=58 ttl=64 time=1.24 ms
64 bytes from 10.0.0.2: icmp_seq=59 ttl=64 time=1.26 ms
64 bytes from 10.0.0.2: icmp_seq=60 ttl=64 time=1.54 ms
64 bytes from 10.0.0.2: icmp_seq=61 ttl=64 time=1.42 ms
64 bytes from 10.0.0.2: icmp_seq=62 ttl=64 time=1.33 ms
64 bytes from 10.0.0.2: icmp_seq=63 ttl=64 time=1.43 ms
64 bytes from 10.0.0.2: icmp_seq=64 ttl=64 time=1.43 ms
64 bytes from 10.0.0.2: icmp_seq=65 ttl=64 time=1.40 ms
64 bytes from 10.0.0.2: icmp_seq=66 ttl=64 time=1.38 ms
64 bytes from 10.0.0.2: icmp_seq=67 ttl=64 time=1.52 ms
64 bytes from 10.0.0.2: icmp_seq=68 ttl=64 time=1.34 ms
64 bytes from 10.0.0.2: icmp_seq=69 ttl=64 time=1.39 ms
64 bytes from 10.0.0.2: icmp_seq=70 ttl=64 time=1.37 ms
64 bytes from 10.0.0.2: icmp_seq=71 ttl=64 time=1.41 ms
64 bytes from 10.0.0.2: icmp_seq=72 ttl=64 time=1.48 ms
64 bytes from 10.0.0.2: icmp_seq=73 ttl=64 time=1.36 ms
64 bytes from 10.0.0.2: icmp_seq=74 ttl=64 time=1.21 ms
64 bytes from 10.0.0.2: icmp_seq=75 ttl=64 time=1.31 ms
64 bytes from 10.0.0.2: icmp_seq=76 ttl=64 time=1.35 ms
64 bytes from 10.0.0.2: icmp_seq=77 ttl=64 time=1.49 ms
64 bytes from 10.0.0.2: icmp_seq=78 ttl=64 time=1.41 ms
64 bytes from 10.0.0.2: icmp_seq=79 ttl=64 time=1.32 ms
64 bytes from 10.0.0.2: icmp_seq=80 ttl=64 time=1.49 ms
64 bytes from 10.0.0.2: icmp_seq=81 ttl=64 time=1.43 ms
64 bytes from 10.0.0.2: icmp_seq=82 ttl=64 time=1.38 ms
64 bytes from 10.0.0.2: icmp_seq=83 ttl=64 time=1.40 ms
64 bytes from 10.0.0.2: icmp_seq=84 ttl=64 time=1.40 ms
```

```
64 bytes from 10.0.0.2: icmp_seq=85 ttl=64 time=1.35 ms
64 bytes from 10.0.0.2: icmp_seq=86 ttl=64 time=1.32 ms
64 bytes from 10.0.0.2: icmp_seq=87 ttl=64 time=1.47 ms
64 bytes from 10.0.0.2: icmp_seq=88 ttl=64 time=1.22 ms
64 bytes from 10.0.0.2: icmp_seq=89 ttl=64 time=1.45 ms
64 bytes from 10.0.0.2: icmp_seq=90 ttl=64 time=1.49 ms
64 bytes from 10.0.0.2: icmp_seq=91 ttl=64 time=1.31 ms
64 bytes from 10.0.0.2: icmp_seq=92 ttl=64 time=1.32 ms
64 bytes from 10.0.0.2: icmp_seq=93 ttl=64 time=1.49 ms
64 bytes from 10.0.0.2: icmp_seq=94 ttl=64 time=1.33 ms
64 bytes from 10.0.0.2: icmp_seq=95 ttl=64 time=1.48 ms
64 bytes from 10.0.0.2: icmp_seq=96 ttl=64 time=1.39 ms
64 bytes from 10.0.0.2: icmp_seq=97 ttl=64 time=1.36 ms
64 bytes from 10.0.0.2: icmp_seq=98 ttl=64 time=1.40 ms
64 bytes from 10.0.0.2: icmp_seq=99 ttl=64 time=1.50 ms
64 bytes from 10.0.0.2: icmp_seq=100 ttl=64 time=1.42 ms

--- 10.0.0.2 ping statistics ---
100 packets transmitted, 100 received, 0% packet loss, t
rtt min/avg/max/mdev = 1.115/1.380/1.669/0.102 ms
mininet>
```

Cloud Computing

HomeWork-3

W1650226-Snehal Vijay Nikam

---

h1 ping -c100 h8 :

```
mininet> h1 ping -c100 h8
PING 10.0.0.8 (10.0.0.8) 56(84) bytes of data.
64 bytes from 10.0.0.8: icmp_seq=1 ttl=64 time=5.51 ms
64 bytes from 10.0.0.8: icmp_seq=2 ttl=64 time=4.54 ms
64 bytes from 10.0.0.8: icmp_seq=3 ttl=64 time=4.44 ms
64 bytes from 10.0.0.8: icmp_seq=4 ttl=64 time=4.42 ms
64 bytes from 10.0.0.8: icmp_seq=5 ttl=64 time=4.63 ms
64 bytes from 10.0.0.8: icmp_seq=6 ttl=64 time=4.52 ms
64 bytes from 10.0.0.8: icmp_seq=7 ttl=64 time=4.70 ms
64 bytes from 10.0.0.8: icmp_seq=8 ttl=64 time=4.43 ms
64 bytes from 10.0.0.8: icmp_seq=9 ttl=64 time=4.53 ms
64 bytes from 10.0.0.8: icmp_seq=10 ttl=64 time=4.52 ms
64 bytes from 10.0.0.8: icmp_seq=11 ttl=64 time=4.54 ms
64 bytes from 10.0.0.8: icmp_seq=12 ttl=64 time=4.66 ms
64 bytes from 10.0.0.8: icmp_seq=13 ttl=64 time=4.55 ms
64 bytes from 10.0.0.8: icmp_seq=14 ttl=64 time=4.33 ms
64 bytes from 10.0.0.8: icmp_seq=15 ttl=64 time=4.48 ms
64 bytes from 10.0.0.8: icmp_seq=16 ttl=64 time=4.45 ms
64 bytes from 10.0.0.8: icmp_seq=17 ttl=64 time=4.48 ms
64 bytes from 10.0.0.8: icmp_seq=18 ttl=64 time=4.45 ms
64 bytes from 10.0.0.8: icmp_seq=19 ttl=64 time=4.50 ms
64 bytes from 10.0.0.8: icmp_seq=20 ttl=64 time=4.32 ms
64 bytes from 10.0.0.8: icmp_seq=21 ttl=64 time=4.36 ms
64 bytes from 10.0.0.8: icmp_seq=22 ttl=64 time=4.54 ms
64 bytes from 10.0.0.8: icmp_seq=23 ttl=64 time=4.53 ms
64 bytes from 10.0.0.8: icmp_seq=24 ttl=64 time=4.40 ms
64 bytes from 10.0.0.8: icmp_seq=25 ttl=64 time=4.64 ms
64 bytes from 10.0.0.8: icmp_seq=26 ttl=64 time=4.50 ms
64 bytes from 10.0.0.8: icmp_seq=27 ttl=64 time=4.49 ms
64 bytes from 10.0.0.8: icmp_seq=28 ttl=64 time=4.43 ms
64 bytes from 10.0.0.8: icmp_seq=29 ttl=64 time=4.38 ms
64 bytes from 10.0.0.8: icmp_seq=30 ttl=64 time=4.56 ms
64 bytes from 10.0.0.8: icmp_seq=31 ttl=64 time=4.19 ms
64 bytes from 10.0.0.8: icmp_seq=32 ttl=64 time=4.50 ms
64 bytes from 10.0.0.8: icmp_seq=33 ttl=64 time=4.37 ms
64 bytes from 10.0.0.8: icmp_seq=34 ttl=64 time=4.40 ms
64 bytes from 10.0.0.8: icmp_seq=35 ttl=64 time=4.54 ms
64 bytes from 10.0.0.8: icmp_seq=36 ttl=64 time=4.39 ms
64 bytes from 10.0.0.8: icmp_seq=37 ttl=64 time=4.45 ms
64 bytes from 10.0.0.8: icmp_seq=38 ttl=64 time=4.54 ms
64 bytes from 10.0.0.8: icmp_seq=39 ttl=64 time=4.20 ms
64 bytes from 10.0.0.8: icmp_seq=40 ttl=64 time=4.56 ms
64 bytes from 10.0.0.8: icmp_seq=41 ttl=64 time=4.47 ms
```

Cloud Computing

HomeWork-3

W1650226-Snehal Vijay Nikam

```
64 bytes from 10.0.0.8: icmp_seq=42 ttl=64 time=4.56 ms
64 bytes from 10.0.0.8: icmp_seq=43 ttl=64 time=4.75 ms
64 bytes from 10.0.0.8: icmp_seq=44 ttl=64 time=4.42 ms
64 bytes from 10.0.0.8: icmp_seq=45 ttl=64 time=4.40 ms
64 bytes from 10.0.0.8: icmp_seq=46 ttl=64 time=4.49 ms
64 bytes from 10.0.0.8: icmp_seq=47 ttl=64 time=4.22 ms
64 bytes from 10.0.0.8: icmp_seq=48 ttl=64 time=4.44 ms
64 bytes from 10.0.0.8: icmp_seq=49 ttl=64 time=4.52 ms
64 bytes from 10.0.0.8: icmp_seq=50 ttl=64 time=4.63 ms
64 bytes from 10.0.0.8: icmp_seq=51 ttl=64 time=4.60 ms
64 bytes from 10.0.0.8: icmp_seq=52 ttl=64 time=4.42 ms
64 bytes from 10.0.0.8: icmp_seq=53 ttl=64 time=4.53 ms
64 bytes from 10.0.0.8: icmp_seq=54 ttl=64 time=4.51 ms
64 bytes from 10.0.0.8: icmp_seq=55 ttl=64 time=4.69 ms
64 bytes from 10.0.0.8: icmp_seq=56 ttl=64 time=4.49 ms
64 bytes from 10.0.0.8: icmp_seq=57 ttl=64 time=4.51 ms
64 bytes from 10.0.0.8: icmp_seq=58 ttl=64 time=4.48 ms
64 bytes from 10.0.0.8: icmp_seq=59 ttl=64 time=4.58 ms
64 bytes from 10.0.0.8: icmp_seq=60 ttl=64 time=4.56 ms
64 bytes from 10.0.0.8: icmp_seq=61 ttl=64 time=4.45 ms
64 bytes from 10.0.0.8: icmp_seq=62 ttl=64 time=4.30 ms
64 bytes from 10.0.0.8: icmp_seq=63 ttl=64 time=4.76 ms
64 bytes from 10.0.0.8: icmp_seq=64 ttl=64 time=4.52 ms
64 bytes from 10.0.0.8: icmp_seq=65 ttl=64 time=4.39 ms
64 bytes from 10.0.0.8: icmp_seq=66 ttl=64 time=4.44 ms
64 bytes from 10.0.0.8: icmp_seq=67 ttl=64 time=4.47 ms
64 bytes from 10.0.0.8: icmp_seq=68 ttl=64 time=4.67 ms
64 bytes from 10.0.0.8: icmp_seq=69 ttl=64 time=4.58 ms
64 bytes from 10.0.0.8: icmp_seq=70 ttl=64 time=6.03 ms
64 bytes from 10.0.0.8: icmp_seq=71 ttl=64 time=4.65 ms
64 bytes from 10.0.0.8: icmp_seq=72 ttl=64 time=4.33 ms
64 bytes from 10.0.0.8: icmp_seq=73 ttl=64 time=4.59 ms
64 bytes from 10.0.0.8: icmp_seq=74 ttl=64 time=4.59 ms
64 bytes from 10.0.0.8: icmp_seq=75 ttl=64 time=4.60 ms
64 bytes from 10.0.0.8: icmp_seq=76 ttl=64 time=4.55 ms
64 bytes from 10.0.0.8: icmp_seq=77 ttl=64 time=4.48 ms
64 bytes from 10.0.0.8: icmp_seq=78 ttl=64 time=4.48 ms
64 bytes from 10.0.0.8: icmp_seq=79 ttl=64 time=4.49 ms
64 bytes from 10.0.0.8: icmp_seq=80 ttl=64 time=4.55 ms
64 bytes from 10.0.0.8: icmp_seq=81 ttl=64 time=4.44 ms
64 bytes from 10.0.0.8: icmp_seq=82 ttl=64 time=4.27 ms
64 bytes from 10.0.0.8: icmp_seq=83 ttl=64 time=4.55 ms
64 bytes from 10.0.0.8: icmp_seq=84 ttl=64 time=4.58 ms
```

```
64 bytes from 10.0.0.8: icmp_seq=85 ttl=64 time=4.50 ms
64 bytes from 10.0.0.8: icmp_seq=86 ttl=64 time=4.52 ms
64 bytes from 10.0.0.8: icmp_seq=87 ttl=64 time=4.57 ms
64 bytes from 10.0.0.8: icmp_seq=88 ttl=64 time=4.50 ms
64 bytes from 10.0.0.8: icmp_seq=89 ttl=64 time=4.54 ms
64 bytes from 10.0.0.8: icmp_seq=90 ttl=64 time=4.45 ms
64 bytes from 10.0.0.8: icmp_seq=91 ttl=64 time=4.64 ms
64 bytes from 10.0.0.8: icmp_seq=92 ttl=64 time=4.46 ms
64 bytes from 10.0.0.8: icmp_seq=93 ttl=64 time=4.41 ms
64 bytes from 10.0.0.8: icmp_seq=94 ttl=64 time=4.49 ms
64 bytes from 10.0.0.8: icmp_seq=95 ttl=64 time=5.35 ms
64 bytes from 10.0.0.8: icmp_seq=96 ttl=64 time=4.64 ms
64 bytes from 10.0.0.8: icmp_seq=97 ttl=64 time=4.51 ms
64 bytes from 10.0.0.8: icmp_seq=98 ttl=64 time=4.62 ms
64 bytes from 10.0.0.8: icmp_seq=99 ttl=64 time=4.26 ms
64 bytes from 10.0.0.8: icmp_seq=100 ttl=64 time=4.34 ms

--- 10.0.0.8 ping statistics ---
100 packets transmitted, 100 received, 0% packet loss, t
rtt min/avg/max/mdev = 4.194/4.527/6.032/0.228 ms
mininet>
```

Cloud Computing

HomeWork-3

W1650226-Snehal Vijay Nikam

---

3. Q.3 Run “iperf h1 h2” and “iperf h1 h8”.

a. What is the throughput for each case?

b. What is the difference from Task 2 and why do you think there is a change if there is?

a. The throughput in first case h1 to h2 is 50.0 Mbits/Sec and 3.72 Mbits/sec in second case.

```
mininet> iperf h1 h2
*** Iperf: testing TCP bandwidth between h1 and h2
*** Results: ['50.0 Mbits/sec', '49.4 Mbits/sec']
mininet> iperf h1 h8
*** Iperf: testing TCP bandwidth between h1 and h8
*** Results: ['3.72 Mbits/sec', '3.69 Mbits/sec']
mininet>
```

b. Task 3 consistently delivers higher throughput compared to Task 2. This is primarily due to reduced network congestion. In Task 3, the MAC Learning Controller eliminates unnecessary flooding after initial discovery. This prevents switches from becoming overloaded with packets destined for unknown locations. However, the improvement in throughput isn't uniform across all connections. While the overall network benefits from pre-planned and dynamically adjusted routes based on learned MAC-to-port mappings (controller changes), connections like h1 to h8 might see a smaller improvement. This is because these paths likely involve more hops (intermediate devices) where delays can occur, and even occasional packet drops can impact overall performance.
