Lab06 by Katrina

Exercise 1

Q1.

Both Node 0 and Node 2 conmunicates with Node 5.
Node 0: 0 -> 1 -> 4 -> 5
Node 2: 2 -> 3 -> 5
It doesn't change over time.

Q2.

At time 1.0, the route between Node 1 and Node 4 is down, and it recovers at time 1.2. During this time, the routes are not changed, but Node 5 cannot receive the packet sent from Node 1 as Node 1 is not able to send packet to Node 4.

Q3.

When the route is down, the route between Node 0 and Node 5 is changed. Instead of keeping sending to Node 4, Node 1 send the pkt to Node 2, and Node 2 send Node 1's pkt to Node 3 and finally Node 5.
On the other hand, compared to Step 3, Step 4 is establishing connection (Nodes are sending SYN and ACK to each other).

Q4.

The route between Node 0 and Node 5 changes from the beginning.
Before: 0 -> 1 -> 4 -> 5
Current: 0 -> 1 -> 2 -> 3 -> 5
This is because the added script `ns cost $n1 $n4 3` increases the cost between Node 1 and Node 4. As those Nodes use Distance-Vector routing protocol, Node 1 finds that sending to Node 4 costs more than sending pkt to Node 2. Hence it decides to send pkt to Node 2.

Q5.

In this case, Node 2 uses multiple routes.
Route 1: 2 -> 3 -> 5
Route 2: 2 -> 1 -> -> 4 -> 5

It's because the script sets the cost between Node 1 and Node 4 to 2, and cost between Node 3 and Node 5 to 3. So now, the costs of the two paths are equal, which are 4. Also, multiple path is now allowed by the script, hence Node 2 switches between Route 1 and Route 2.

However, this causes a packet drop at time 1.0 as Node 4 is overload.

Exercise 2

Q1.

Because during that time, tcp1 is sharing bandwidth with tcp2 and tcp4 but tcp2 is only sharing with tcp3. According to tcp fairness, tcp2 occupies more bandwidth than tcp1, therefore it has higher throughput.

Q2.

tcp1 is using Slow Start, so at the beginning it's looking for bandwitch with maximum utilization.

Q3.

Because the maximum bandwidth is 2.5Mbps and it has to be distrbuted to at least two tcp connection. From the graph, can see that while other tcp occupies 1Mbps, another sharing tcp can only occupies up to `2.5 - 1 = 1.5Mbps`.

Exercise 3

Q1.

Data size 2000 and 3500 caused fragmentation. 192.168.1.103 has fragmented the original datagram. 2 fragments have been created when data size is 2000.For example, for the first ping with data size 2000, the first fragment 1500 bytes, the second one is 548 bytes.

Q2.

Yes. It can be seen from No.42, No.43 and No.44. Yes, the reply also get fragmented as it's on ICMP protocol and the incoming packets are fragmented.

Q3.

The first fragment:
ID: 0x7a7b
length: 1500
flag: 0x01 (More Fragmetns)

offset: 0

The second fragment:
ID: 0x7a7b
length: 1500
flag: 0x01 (More Fragments)
offset: 1480

The third fragment:
ID: 0x7a7b
length: 568
flag: 0x00
offset: 2960

Q4.

No. Because during the transmittion of 3500 bytes data, can only see three fragments per packet both in request and reply, so there is no fragmentation of fragments occured.

Q5.

192.168.103 has to retransmit all of the fragments.