

CSCI 4171/CSCI 6704

NETWORKS

ASSIGNMENT NO.3

Date Given: Saturday, October 27, 2018

Date Due: Saturday, November 10, 2018

In this assignment, you will write two programs for simulation - one for bridge processing and one for routing. You may use Java, C, C++ or Python as the language for building your programs. Download the associated text files (given next to the Assignment link).

Exercise 1: In this exercise, you will write a simple program to simulate the bridge-processing flowchart for a bridge that we discussed in the lectures. You are given a text file (BridgeFDB.txt) that contains the forwarding database of a bridge with four ports. The entries in the text file are arranged as shown below:

```
00-00-00-11-0b-0d    → MAC address of the destination host
1                    → Port number
00-13-46-c6-a5-35
2
00-13-8f-cd-44-58
3
00-0c-29-51-33-c1
4
etc.
```

You are given another text file (RandomFrames.txt) that contains the source MAC address, the destination MAC address of a random frame on the network that the bridge is connected to, and the port number of the bridge that it arrives on. The entries in this text file are arranged as shown below:

```
00-00-00-11-0b-0d    → Source MAC address
00-13-46-c6-a5-35    → Destination MAC address
1                    → Arrival port number
00-0c-29-51-33-c1
01-00-5e-7f-ff-64
4
01-00-5e-7f-ff-64
00-00-4f-31-fa-fb
3
etc.
```

Your program should read the two text files, and for each random frame, it should make a decision either to forward/discard/broadcast the frame. If the source MAC address is not in the database, then it should update it.

Ignore the CRC error detection part (that is, assume that the frames are error free). Store the resulting output in another text file (BridgeOutput.txt).

Your output would look something like this (for the above three sample frames).

```
00-00-00-11-0b-0d    00-13-46-c6-a5-35    1    Forwarded on port 2
00-0c-29-51-33-c1    01-00-5e-7f-ff-64    4    Discarded
01-00-5e-7f-ff-64    00-00-4f-31-fa-fb    3    Broadcast
```

Note: The reason the last frame is broadcast is because the entry 00-00-4f-31-fa-fb is not in the FDB.

You should keep track of the updated FDB and also submit that as part of your output. What to submit: Source code files and BridgeOutput.txt. Your program will be tested using the sample BridgeFDB.txt and RandomFrames.txt.

Exercise 2: In this exercise, you will write a simple program to simulate the routing decision taken at a router that we discussed in the lectures. You are given a text file (RoutingTable.txt) containing a routing table for a router (the five column generic routing table with mask, destination address, next hop, flag and interface). You will notice that this is almost the same routing table as the one that we discussed in the lectures. The first few lines of the text file are shown below:

```
192.168.1.0/24  → Destination address/Number of ones in the mask
-              → - means no next hop
E0            → Outgoing interface
192.168.2.0/24  → Destination address/Number of ones in the mask
192.168.4.2     → Next hop
S0            → Outgoing interface
Etc.
```

You are given another text file (RandomPackets.txt) that contains a list of packets with random destination addresses that arrive at this router. The first few lines of the text file are shown below:

```
192.168.1.1
192.168.3.2
200.56.45.89
255.0.0.0
127.0.0.1
etc.
```

If you check the above packets with the given routing table, you will notice that packet #2 is a host specific packet, packet #3 should take the default route, packet #4 does not belong to class A, B or C and should be discarded, packet #5 has a loopback address and should be discarded.

Your program should read the two text files and decide what to do with each packet, displayed as follows:

```
192.168.1.1 will be forwarded to 192.168.5.2 out on interface S1 on the
directly connected network on interface E0.
192.168.3.2 will be forwarded to 192.168.4.2 out on interface
S0 192.168.5.2 out on interface S1
200.56.45.89 will be forwarded to 192.168.4.2 out on interface S0
255.0.0.0 is malformed; discarded
127.0.0.1 is loopback; discarded
```

Store the answers in a text file (RoutingOutput.txt).

Write your program in a **general** manner, in the sense, it should work for any routing table and any packet. You may assume that all the masks in the table are the default ones for Class A (255.0.0.0), Class B (255.255.0.0) and Class C (255.255.255.0)..

What to submit: Source codes and RoutingOutput.txt. Your program will be tested against the given Routing Table and the given set of packets.

Final submission will be one zip file.