DCCN LAB 7-8

Name: **Maloth Aditya**

Roll No.: **120CS0124**

Q1) **Write Tcl script to create scenario and study the performance of token ring protocols through simulation. Create 6 nodes that forms a network numbered from 1 to 6. Create duplex links between the nodes to form a Ring Topology with bandwidth of 100 Mbps and delay of 2ms. Setup TCP Connection between node 1 and node 4. Apply FTP Traffic over TCP. Finish the transmission at 100 sec.**

Code :

q1.tcl

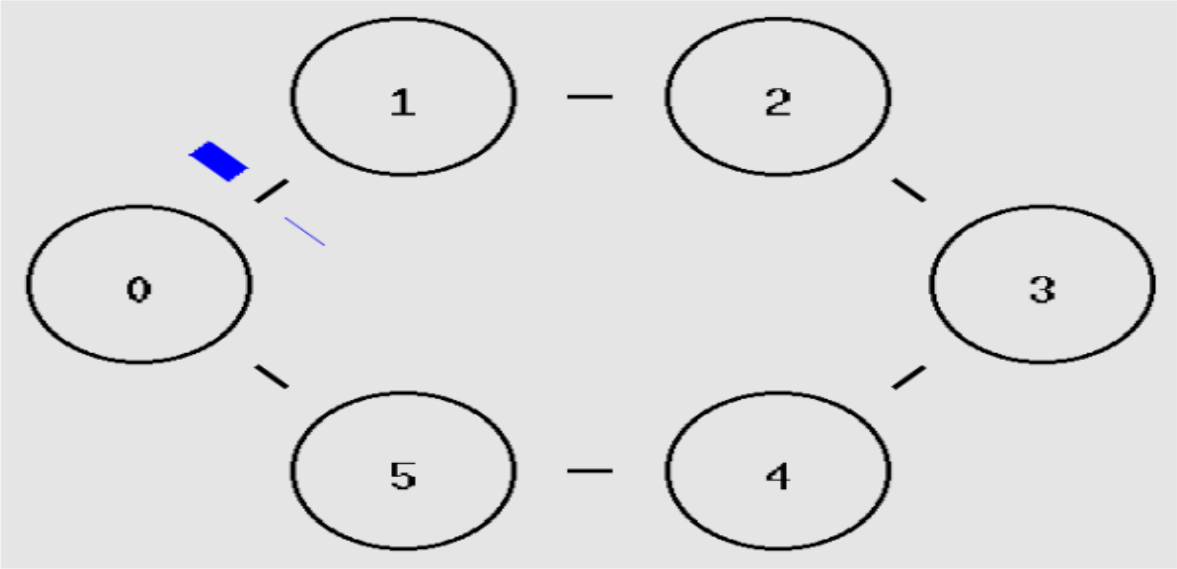
```
#Create a simulator object
set ns [new Simulator]

#Define different colors for data flows
$ns color 1 Blue

#Open the nam trace file
set nf [open q1.nam w]
$ns namtrace-all $nf

#Define a 'finish' procedure
proc finish {} {
    global ns nf
    $ns flush-trace

    #Close the trace file
    close $nf

    #Execute nam on the trace file
    exec nam q1.nam &

    exit 0
}

#Create six nodes
set n1 [$ns node]
set n2 [$ns node]
set n3 [$ns node]
set n4 [$ns node]
set n5 [$ns node]
set n6 [$ns node]

#Create links between the nodes
$ns duplex-link $n1 $n2 100Mb 2ms DropTail
$ns duplex-link $n2 $n3 100Mb 2ms DropTail
$ns duplex-link $n3 $n4 100Mb 2ms DropTail
$ns duplex-link $n4 $n5 100Mb 2ms DropTail
$ns duplex-link $n5 $n6 100Mb 2ms DropTail
$ns duplex-link $n6 $n1 100Mb 2ms DropTail

```

DCCN LAB 7-8

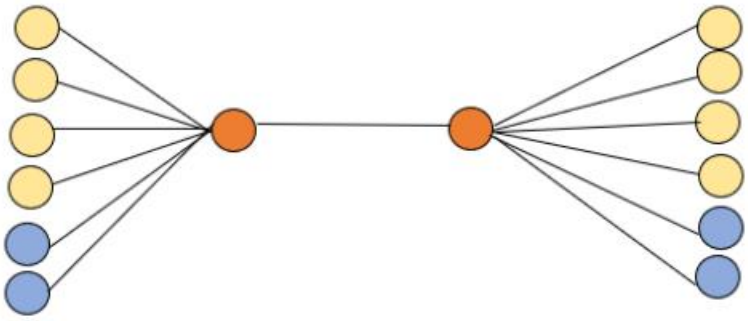
```
#Give node position (for NAM)
$ns duplex-link-op $n1 $n2 orient right-up
$ns duplex-link-op $n2 $n3 orient right
$ns duplex-link-op $n3 $n4 orient right-down
$ns duplex-link-op $n4 $n5 orient left-down
$ns duplex-link-op $n5 $n6 orient left
$ns duplex-link-op $n6 $n1 orient left-up

#Setup a TCP connection
set tcp [new Agent/TCP]
$tcp set class_ 2
$ns attach-agent $n1 $tcp

set sink [new Agent/TCPSink]
$ns attach-agent $n4 $sink
$ns connect $tcp $sink
$tcp set fid_ 1

#Setup a FTP over TCP connection
set ftp [new Application/FTP]
$ftp attach-agent $tcp
$ftp set type_ FTP

#Schedule events for FTP agent
$ns at 0.0 "$ftp start"
$ns at 95.0 "$ftp stop"

#Call the finish procedure after 5 seconds of simulation time
$ns at 100.0 "finish"

#Run the simulation
$ns run
```

Output :

DCCN LAB 7-8



2) Write a Tcl script that forms a network consisting of 6 nodes, numbered from 1 to 6. Each of source and destination has bandwidth of 300 Mbps and delay of 20 ms. Set the bottleneck link bandwidth as 500 sec and delay 10ms. Set the routing protocol to Droptail. Define different colors for different data flows. Send TCP packet from node 1 to node 4 and UDP packet from node 5 to 6. Start the TCP data transmission at 1 sec and UDP at 15 sec. Finish the transmission at 100 sec. Then run nam to view the results.



Calculate the following performance metrics using awk script:

a) Throughput
b) Delay
c) Packet loss ratio
d) Jain Fairness index.
e) Plot throughput graph using gnuplot (Tahoe vs Reno)
f) Plot Jain Fairness index graph using gnuplot

DCCN LAB 7-8

a) Throughput .

--tahoe.tcl code ---

```
set ns [new Simulator]

$ns color 1 Blue
$ns color 2 Red
$ns color 3 Yellow
$ns color 4 Pink
$ns color 5 Black
$ns color 6 Green

set tracefile [open tahoe.tr w]
$ns trace-all $tracefile

set namfile [open tahoe.nam w]
$ns namtrace-all $namfile


for {set i 0} {$i < 6} {incr i} {
 set n($i) [$ns node]
}

for {set i 0} { $i < 6} {incr i} {
set r($i) [$ns node]
}

set b(0) [$ns node]
set b(1) [$ns node]
```

DCCN LAB 7-8

```
for {set i 0} { $i < 6} {incr i} {
$ns duplex-link $n($i) $b(0) 300Mb 20ms DropTail
}


$ns duplex-link $b(0) $b(1) 500Mb 10ms DropTail

for {set i 0} { $i < 6} {incr i} {
$ns duplex-link $r($i) $b(1) 300Mb 20ms DropTail
}




#tcp setup

for {set i 0} { $i < 4} {incr i} {
  set tcp($i) [new Agent/TCP]
set sink($i) [new Agent/TCPSink]
$ns attach-agent $n($i) $tcp($i)
$ns attach-agent $r($i) $sink($i)
set ftp($i) [new Application/FTP]
$ftp($i) attach-agent $tcp($i)

$ns connect $tcp($i) $sink($i)
$tcp($i) set fid_ $i+1

}

#udp setup
```

DCCN LAB 7-8

```
set udp(0) [new Agent/UDP]
set udp(1) [new Agent/UDP]
set null(0) [new Agent/Null]
set null(1) [new Agent/Null]

$ns attach-agent $n(4) $udp(0)
$ns attach-agent $r(4) $null(0)
set cbr(0) [new Application/Traffic/CBR]

$cbr(0) attach-agent $udp(0)

$ns connect $udp(0) $null(0)
$udp(0) set fid_ 5

$ns attach-agent $n(5) $udp(1)
$ns attach-agent $r(5) $null(1)
set cbr(1) [new Application/Traffic/CBR]

$cbr(1) attach-agent $udp(1)

$ns connect $udp(1) $null(1)
$udp(1) set fid_ 6

 $ns at 1.0 "$ftp(0) start"
 $ns at 1.0 "$ftp(1) start"
 $ns at 1.0 "$ftp(2) start"
 $ns at 1.0 "$ftp(3) start"

 $ns at 15.0 "$cbr(0) start"
 $ns at 15.0 "$cbr(1) start"
```

DCCN LAB 7-8

```
$ns at 100.0 "finish"

proc finish {} {
  global ns tracefile namfile
  $ns flush-trace
  close $tracefile
  close $namfile
  exit 0
}


$ns run
```

--reno.tcl—

```
#Reno
set ns [new Simulator]

$ns color 1 Blue
$ns color 2 Red
$ns color 3 Yellow
$ns color 4 Pink
$ns color 5 Black
$ns color 6 Green

set tracefile [open reno.tr w]
$ns trace-all $tracefile

set namfile [open reno.nam w]
$ns namtrace-all $namfile


for {set i 0} {$i < 6} {incr i} {
 set n($i) [$ns node]
```

DCCN LAB 7-8

```
}

for {set i 0} { $i < 6} {incr i} {
set r($i) [$ns node]
}

set b(0) [$ns node]
set b(1) [$ns node]


for {set i 0} { $i < 6} {incr i} {
$ns duplex-link $n($i) $b(0) 300Kb 20ms DropTail
}


$ns duplex-link $b(0) $b(1) 150Kb 10ms DropTail

for {set i 0} { $i < 6} {incr i} {
$ns duplex-link $r($i) $b(1) 300Kb 20ms DropTail
}




#tcp setup

for {set i 0} { $i < 4} {incr i} {
  set tcp($i) [new Agent/TCP/Reno]
set sink($i) [new Agent/TCPSink]
$ns attach-agent $n($i) $tcp($i)
$ns attach-agent $r($i) $sink($i)
```

DCCN LAB 7-8

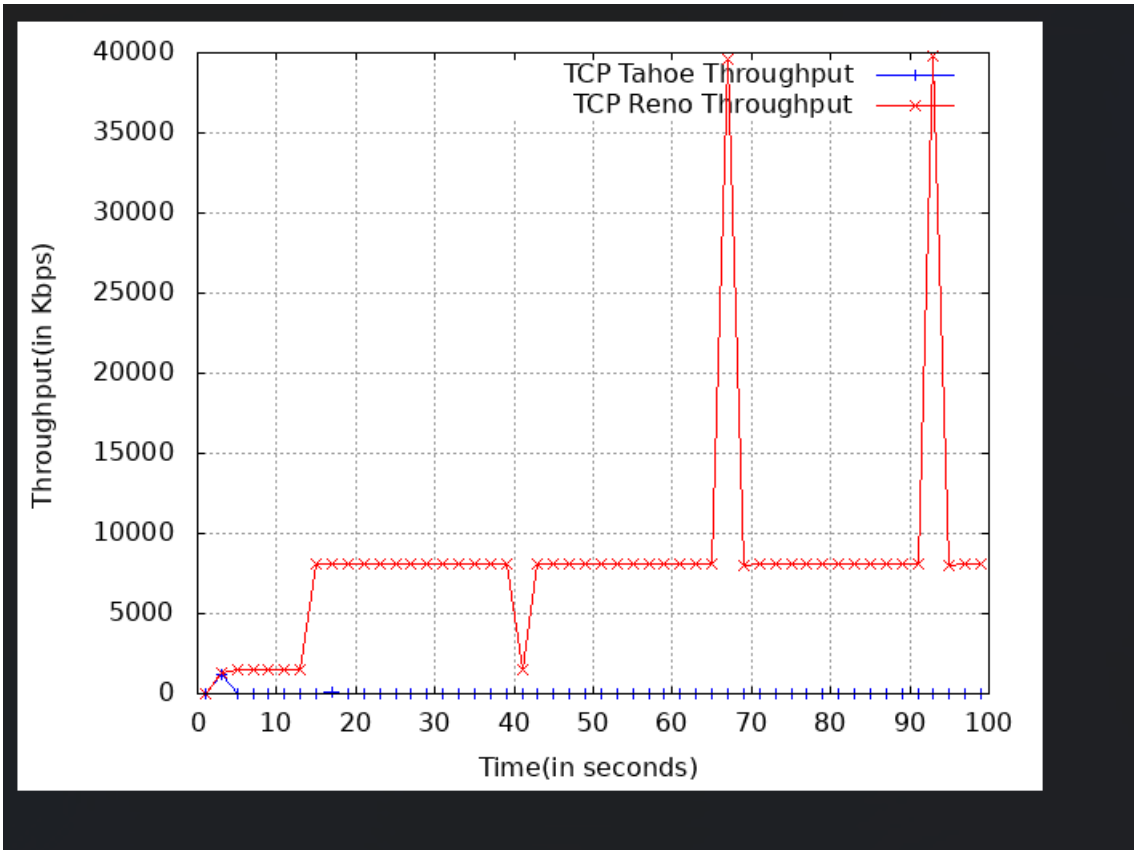
```
set ftp($i) [new Application/FTP]
$ftp($i) attach-agent $tcp($i)

$ns connect $tcp($i) $sink($i)
$tcp($i) set fid_ $i+1

}

#udp setup

set udp(0) [new Agent/UDP]
set udp(1) [new Agent/UDP]
set null(0) [new Agent/Null]
set null(1) [new Agent/Null]

$ns attach-agent $n(4) $udp(0)
$ns attach-agent $r(4) $null(0)
set cbr(0) [new Application/Traffic/CBR]

$cbr(0) attach-agent $udp(0)

$ns connect $udp(0) $null(0)
$udp(0) set fid_ 5

$ns attach-agent $n(5) $udp(1)
$ns attach-agent $r(5) $null(1)
set cbr(1) [new Application/Traffic/CBR]

$cbr(1) attach-agent $udp(1)

$ns connect $udp(1) $null(1)
$udp(1) set fid_ 6
```

DCCN LAB 7-8

```
$ns at 1.0 "$ftp(0) start"
$ns at 1.0 "$ftp(1) start"
$ns at 1.0 "$ftp(2) start"
$ns at 1.0 "$ftp(3) start"

$ns at 15.0 "$cbr(0) start"
$ns at 15.0 "$cbr(1) start"

$ns at 100.0 "finish"

proc finish {} {
  global ns tracefile namfile
  $ns flush-trace
  close $tracefile
  close $namfile
  exit 0
}


$ns run
```

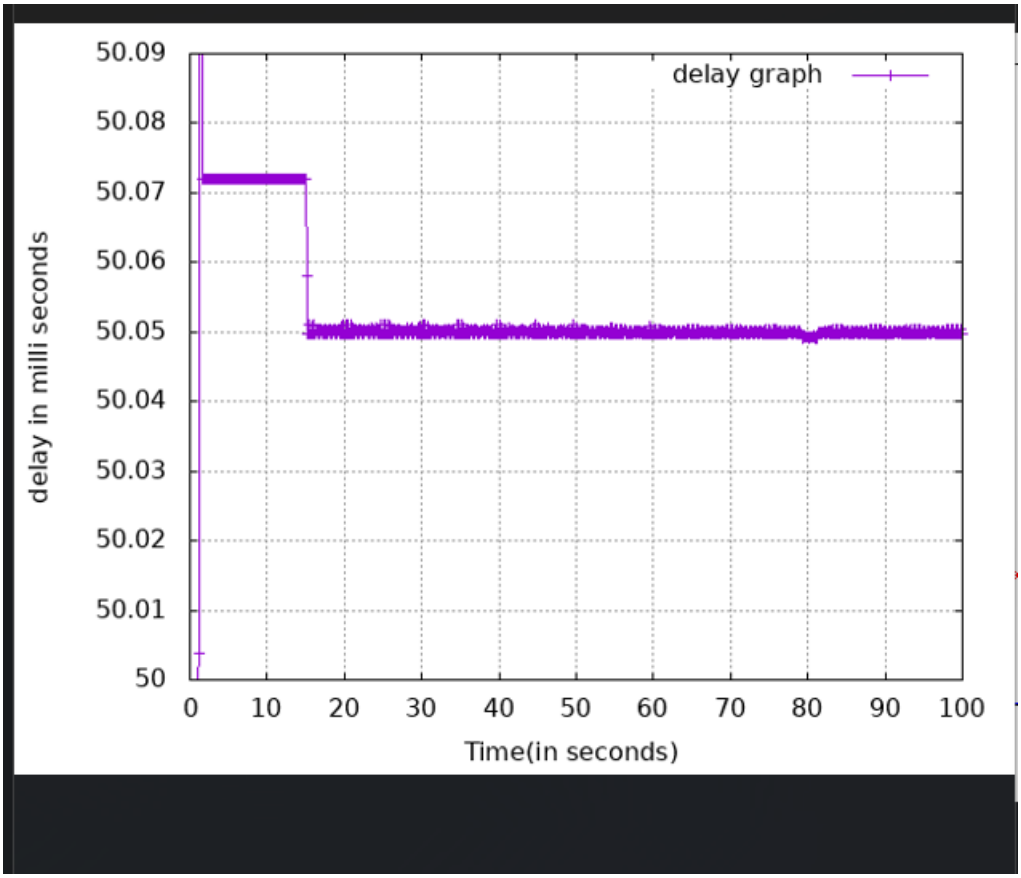
Throughput Graph:

DCCN LAB 7-8



b) Delay graph (Tahoe vs reno)

DCCN LAB 7-8

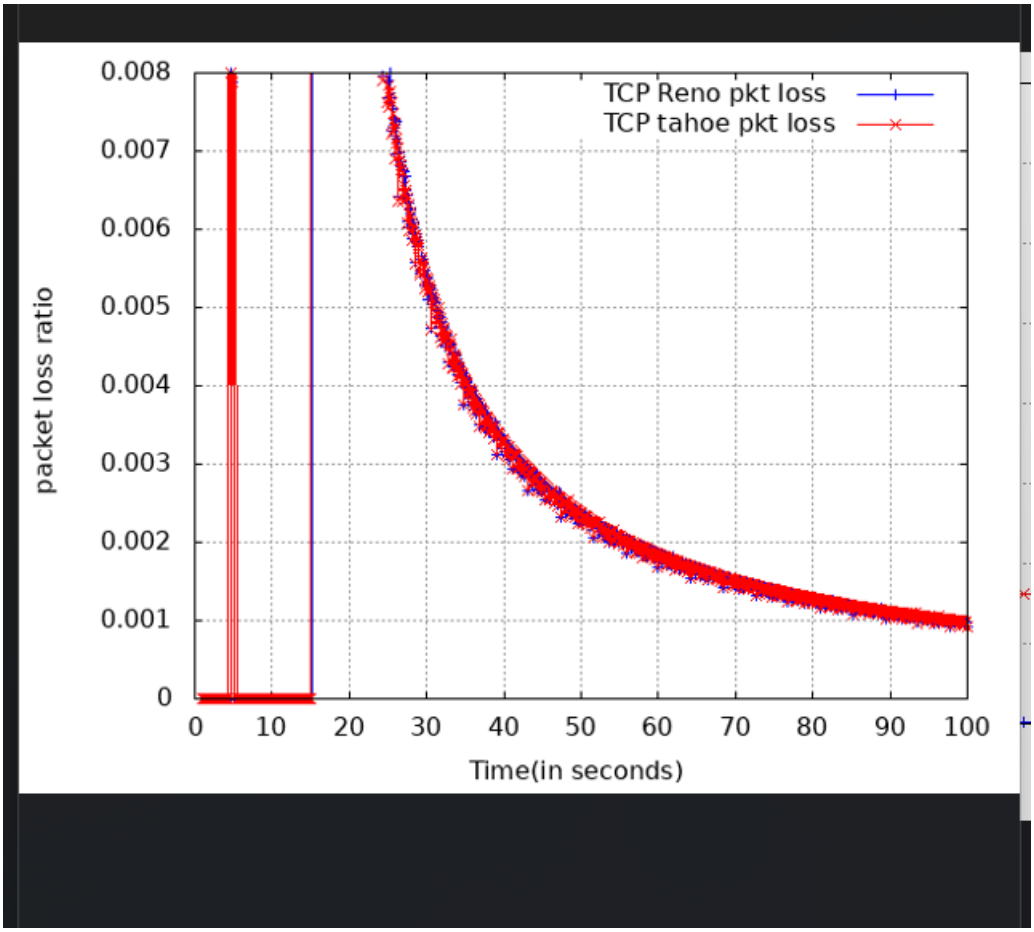


c) Pkt loss ratio (tahoe Vs reno)

Here, sender's bandwidth is reduced to 300Kbps from 300Mbps and Bottleneck bandwidth is reduced to 50Kbps from 500Mbps, in order to

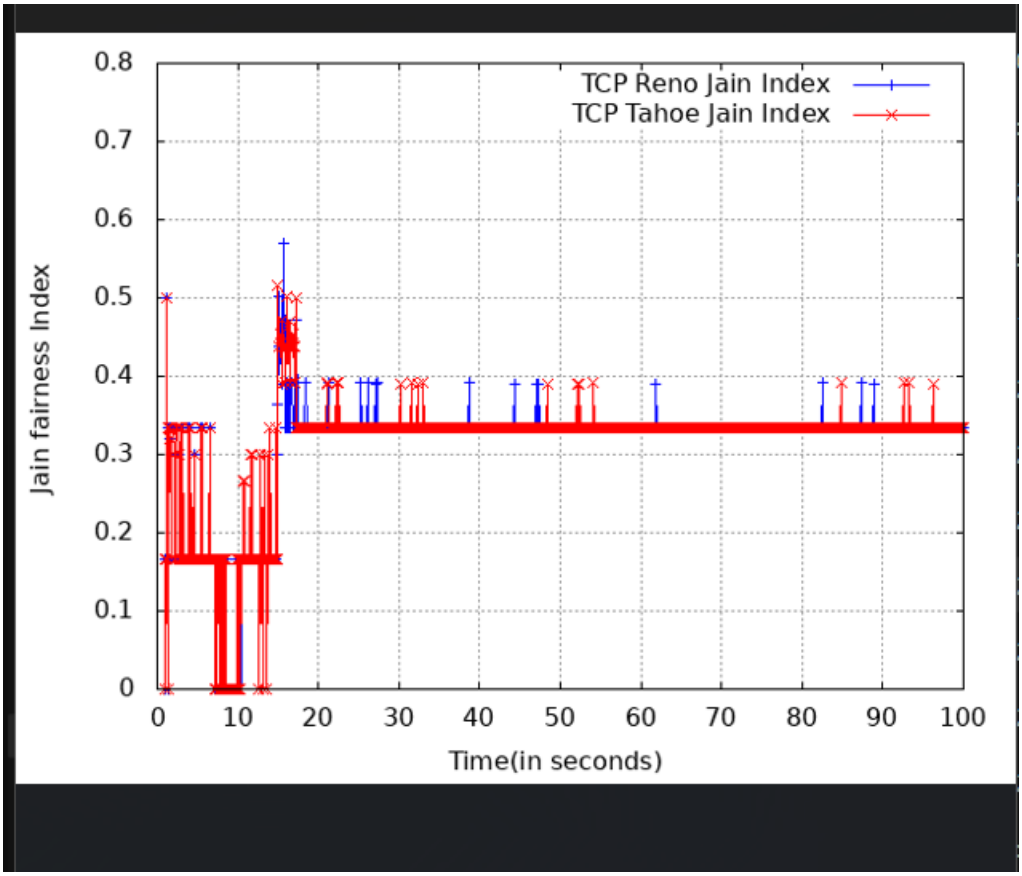
DCCN LAB 7-8

Obtain dropped packets.
Remaining code part will be same.



d) Jain fairness Index (TCP vs Reno)

DCCN LAB 7-8


TCP Reno Jain Index
TCP Tahoe Jain Index
Jain fairness Index
Time(in seconds)
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0
10
20
30
40
50
60
70
80
90
100