Program No. 3

Implement an Ethernet LAN using n nodes and set multiple traffic nodes and plot congestion window for different source / destination using NS2 Simulator.

Program:

(TCL file)

```
set ns [new Simulator]
set nf [open lab3.nam w]
$ns namtrace-all $nf
set nd [open lab3.tr w]
$ns trace-all $nd
$ns color 1 Blue
$ns color 2 Red
proc finish { } {
global ns nf nd
$ns flush-trace
close $nf
close $nd
exec nam lab3.nam &
exit 0
}
set n0 [$ns node]
set nl [$ns node]
set n2 [$ns node]
set n3 [$ns node]
set n4 [$ns node]
set n5 [$ns node]
set n6 [$ns node]
set n7 [$ns node]
set n8 [$ns node]

$n7 shape box
$n7 color Blue
$n8 shape hexagon
$n8 color Red
$ns duplex-link $nl $n0 2Mb l0ms DropTail
$ns duplex-link $n2 $n0 2Mb l0ms DropTail
$ns duplex-link $n0 $n3 1Mb 20ms DropTail
$ns make-lan "$n3 $n4 $n5 $n6 $n7 $n8" 512Kb 40ms LL Queue/DropTail
Mac/802_3
$ns duplex-link-op $nl $n0 orient right-down
$ns duplex-link-op $n2 $n0 orient right-up
$ns duplex-link-op $n0 $n3 orient right
$ns queue-limit $n0 $n3 10
set tcp1 [new Agent/TCP]
$ns attach-agent $nl $tcp1
set sink1 [new Agent/TCPSink]
$ns attach-agent $n7 $sink1
$ns connect $tcp1 $sink1
$tcp1 set class_ 1
$tcp1 set packetSize_ 200
set ftp1 [new Application/FTP]
$ftp1 attach-agent $tcp1
```

```
set tfile [open cwnd.tr w]
$tcp1 attach $tfile
$tcp1 trace cwnd_
set tcp2 [new Agent/TCP/Reno]
$ns attach-agent $n2 $tcp2
set sink2 [new Agent/TCPSink]
$ns attach-agent $n8 $sink2
$ns connect $tcp2 $sink2
$tcp2 set class_ 2
$tcp2 set packetSize_ 200
set ftp2 [new Application/FTP]
$ftp2 attach-agent $tcp2
set tfile2 [open cwnd2.tr w]
$tcp2 attach $tfile2
$tcp2 trace cwnd_
$ns at 0.5 "$ftp1 start"
$ns at 1.0 "$ftp2 start"
$ns at 5.0 "$ftp2 stop"
$ns at 5.0 "$ftp1 stop"
$ns at 5.5 "finish"
$ns run
```

.awk file:

Lab3.awk

```
BEGIN{
}
{
     if ($6=="cwnd_")
     printf("%f\t%f\t\n",$1,$7);
}
END{
}
```

Commands list

cd /mnt/to destination folder

```
gedit 3.tcl
gedit lab3.awk
ns 3.tcl
awk -f lab3awk file1.tr>file10
awk -f lab3.awk file2.tr>file11
xgraph file10 file11
```

Program No. 4:

Write a NS2 script to implement the operation of Stop and Wait Protocol.

Program:

```
set ns [new Simulator]
$ns color 1 Blue
# set nam output file
set nf [open out.nam w]
$ns namtrace-all $nf

# destructor
proc finish {} {
global ns nf
$ns flush-trace
close $nf
exec nam out.nam &
exit 0
}

# create two new nodes and create labels for them
set n0 [$ns node]
set n1 [$ns node]
$ns at 0.0 "$n0 label \" Sender\" "
$ns at 0.0 "$n1 label \"Receiver\" "

# set up a new duplex link
$ns duplex-link $n0 $n1 1Mb 200ms DropTail
$ns duplex-link-op $n0 $n1 orient right

# create a new TCP agent
set tcp [new Agent/TCP]
# attach the agent to first node
$ns attach-agent $n0 $tcp
$tcp set fid_ 1
$tcp set window_ 1
$tcp set maxcwnd_ 1
$ns add-agent-trace $tcp tcp
$ns monitor-agent-trace $tcp
set tcpsink [new Agent/TCPSink]
$ns attach-agent $n1 $tcpsink

$ns connect $tcp $tcpsink
set ftp [new Application/FTP]
$ftp attach-agent $tcp

$ns at 0.5 "$ftp start"
$ns at 3.0 "$ns detach-agent $n0 $tcp ; $ns detach-agent $n1 $tcpsink"
$ns at 1.0 "$ns trace-annotate \"send packet 1\""
$ns at 1.4 "$ns trace-annotate \"recieve ack 1\""
$ns at 2.0 "$ns trace-annotate \"send packet 2\""
$ns at 2.5 "$ns trace-annotate \"receive ack 2\""
$ns at 3.2 "$ns trace-annotate \"send packet 3\""
$ns at 3.5 "$ns trace-annotate \"receive ack 3\""
$ns at 3.8 "$ns trace-annotate \"send packet 4\""
$ns at 4.0 "finish"
```

```
$ns run
```

Command list

cd /mnt/ to destination folder

gedit lab4.tcl

ns lab4.tcl

Program 5 or 10: (ESS) NS sim (3)

```
#create Simulator class
set ns [new Simulator]
#open trace file
set nt [open lab2.tr w]
$ns trace-all $nt
#create Topography object
set topo [new Topography]
#define grid size
$topo load_flatgrid 1000 1000
#open namtrace file
set nf [open lab2.nam w]
$ns namtrace-all-wireless $nf 1000 1000
#specify mobile node Parameter configuration
$ns node-config -adhocRouting DSDV \
-llType LL \
-macType Mac/802_11 \
-ifqType Queue/DropTail \
-ifqLen 20 \
-phyType Phy/WirelessPhy \
-channelType Channel/WirelessChannel \
-propType Propagation/TwoRayGround \
-antType Antenna/OmniAntenna \
-topoInstance $topo \
-agentTrace ON \
-routerTrace ON

#create a General Operation Director(god) object that stores the total number of mobile nodes.
create-god 4
#create nodes and label them
set n0 [$ns node]
set n1 [$ns node]
set n2 [$ns node]
set n3 [$ns node]

$n0 label "tcp0"
$n1 label "sink0"
$n2 label "bs1"
$n3 label "bs2"

#give initial x, y, z coordinates to nodes
$n0 set X_ 110
$n0 set Y_ 500
$n0 set Z_ 0

$n1 set X_ 600
$n1 set Y_ 500
$n1 set Z_ 0

$n2 set X_ 300
$n2 set Y_ 500
$n2 set Z_ 0
$n3 set X_ 450
```

```
$n3 set Y_ 500
$n3 set Z_ 0

#attach agent and application to nodes and connect them
set tcp0 [new Agent/TCP]
$ns attach-agent $n0 $tcp0
set ftp0 [new Application/FTP]
$ftp0 attach-agent $tcp0

set sink1 [new Agent/TCPSink]
$ns attach-agent $n1 $sink1

$ns connect $tcp0 $sink1

#schedule the event
$ns at 0.5 "$ftp0 start"

#set up a destination for mobile nodes. They move to <x><y> coordinates at <s>m/s.
$ns at 0.3 "$n0 set dest 110 500 10"
$ns at 0.3 "$n1 set dest 600 500 20"
$ns at 0.3 "$n2 set dest 300 500 30"
$ns at 0.3 "$n3 set dest 450 500 30"

$ns at 10.0 "$n0 set dest 100 550 5"
$ns at 10.0 "$n1 set dest 630 450 5"
$ns at 70.0 "$n0 set dest 170 680 5"
$ns at 70.0 "$n1 set dest 580 380 5"

$ns at 120.0 "$n0 set dest 140 720 5"
$ns at 135.0 "$n0 set dest 110 600 5"
$ns at 140.0 "$n1 set dest 600 550 5"
```

```
$ns at 155.0 "$n0 set dest 89 500 5"

$ns at 190.0 "$n0 set dest 100 440 5"

$ns at 210.0 "$n1 set dest 700 600 5"

$ns at 240.0 "$n1 set dest 650 500 5"


proc finish { } {

global ns nt nf

$ns flush-trace

exec nam lab2.nam &

close $nt

close $nf

exit 0

}


$ns at 400 "finish"

$ns run
```

AWK file (p2.awk)

```
BEGIN{
PktsSent=0;
PktsRcvd=0;
PktsAtRTR=0;
}
{
if(($1=="s")&&($4=="RTR")&&($7=="tcp")) PktsAtRTR++;
if(($1=="s")&&($4=="AGT")&&($7=="tcp")) PktsSent++;
if(($1=="r")&&($4=="AGT")&&($7=="tcp")) PktsRcvd++;
}

END{
print " Number of Packets Sent :" PktsSent
print " Number of Packets Received :" PktsRcvd
print " Pacjet Delivery Ratio :" PktsRcvd/PktsSent*100
print " Routing Load :" PktsAtRTR/PktsRcvd
}
```

Commands list:

```
cd /mnt/ to destination folder
gedit p2.tcl
gedit p2.awk
ns p2.tcl
awk -f p2.awk lab2.tr
```