3. Implement an Ethernet LAN using n nodes and set multiple traffic nodes and plot

congestion window for different source / destination

set ns [new Simulator]

set tf [open lab3.tr w]

$ns trace-all $tf

set nf [open lab3.nam w]

$ns namtrace-all $nf

# Node creation

set n0 [$ns node]

$n0 color "magenta"

$n0 label "src1"

set n1 [$ns node]

set n2 [$ns node]

$n2 color "magenta"

$n2 label "src2"

set n3 [$ns node]

$n3 color "blue"

$n3 label "dest2"

set n4 [$ns node]

set n5 [$ns node]

$n5 color "blue"

$n5 label "dest1"

# Create LAN (single line)

$ns make-lan "$n0 $n1 $n2 $n3 $n4" 100Mb 100ms LL Queue/DropTail Mac/802\_3

# Duplex link

$ns duplex-link $n4 $n5 1Mb 1ms DropTail

# TCP0 connection

set tcp0 [new Agent/TCP]

$ns attach-agent $n0 $tcp0

set ftp0 [new Application/FTP]

$ftp0 attach-agent $tcp0

$ftp0 set packetSize\_ 500

$ftp0 set interval\_ 0.0001

set sink5 [new Agent/TCPSink]

$ns attach-agent $n5 $sink5

$ns connect $tcp0 $sink5

# TCP2 connection

set tcp2 [new Agent/TCP]

$ns attach-agent $n2 $tcp2

set ftp2 [new Application/FTP]

$ftp2 attach-agent $tcp2

$ftp2 set packetSize\_ 600

$ftp2 set interval\_ 0.001

set sink3 [new Agent/TCPSink]

$ns attach-agent $n3 $sink3

$ns connect $tcp2 $sink3

# Trace files

set file1 [open file1.tr w]

$tcp0 attach $file1

set file2 [open file2.tr w]

$tcp2 attach $file2

# Trace cwnd (must have underscore)

$tcp0 trace cwnd\_

$tcp2 trace cwnd\_

# Finish procedure

proc finish {} {

global ns nf tf

$ns flush-trace

close $tf

close $nf

exec nam lab3.nam &

exit 0

}

# Event scheduling

$ns at 0.1 "$ftp0 start"

$ns at 5 "$ftp0 stop"

$ns at 7 "$ftp0 start"

$ns at 0.2 "$ftp2 start"

$ns at 8 "$ftp2 stop"

$ns at 14 "$ftp0 stop"

$ns at 10 "$ftp2 start"

$ns at 15 "$ftp2 stop"

$ns at 16 "finish"

$ns run

Step2: # AWK script to print time vs congestion window (cwnd)

BEGIN {

}

{

# Check if 6th field is "cwnd\_"

if ($6 == "cwnd\_") {

printf("%f\t%f\n", $1, $7)

}

}

END {

}

-