

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 3. tr w)
$ns trace-all $tf
set nf (open 3. nam w)
$ns namtrace-all $nf

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"

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

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

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
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

set file1 (open file1. tr w)
$tcp0 attach $file1
set file2 (open file2. tr w)
$tcp2 attach $file2
$tcp0 trace cwnd_
$tcp2 trace cwnd_

proc finish { } {
```



```
global ns nf tf
$ns flush-trace
close $tf
close $nf
exec nam 3. nam &
exit 0
}
```

```
$ns at 0.1 "$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
```

AWK File:

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

