

2. Program 2: Simulate the transmission of ping messages over a network topology consisting of 6 nodes and find the number of packets dropped due to congestion.

```
#Create Simulator
set ns [new Simulator]

#Use colors to differentiate the traffic
$ns color 1 Blue
$ns color 2 Red

#Open trace and NAM trace file
set ntrace [open prog2.tr w]
$ns trace-all $ntrace
set namfile [open prog2.nam w]
$ns namtrace-all $namfile

#Finish Procedure
proc Finish {} {
    global ns ntrace namfile

#Dump all trace data and close the file
$ns flush-trace
close $ntrace
close $namfile

#Execute the nam animation file
exec nam prog2.nam &

#Find the number of ping packets dropped
puts "The number of ping packets dropped are "
exec grep "^d" prog2.tr | cut -d " " -f 5 | grep -c "ping" &
exit 0
}

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

#Connect the nodes
for {set j 0} {$j < 5} {incr j} {
    $ns duplex-link $n($j) $n([expr ($j+1)]) 0.1Mb 10ms DropTail
}

#Define the recv function for the class 'Agent/Ping'
Agent/Ping instproc recv {from rtt} {
    $self instvar node_
    puts "node [$node_ id] received ping answer from $from with round trip time $rtt
}
```

```
ms"  
}
```

```
#Create two ping agents and attach them to n(0) and n(5)  
set p0 [new Agent/Ping]  
$p0 set class_ 1  
$ns attach-agent $n(0) $p0
```

```
set p1 [new Agent/Ping]  
$p1 set class_ 1  
$ns attach-agent $n(5) $p1  
$ns connect $p0 $p1
```

```
#Set queue size and monitor the queue  
#Queue size is set to 2 to observe the drop in ping packets  
$ns queue-limit $n(2) $n(3) 2  
$ns duplex-link-op $n(2) $n(3) queuePos 0.5
```

```
#Create Congestion  
#Generate a Huge CBR traffic between n(2) and n(4)  
set tcp0 [new Agent/TCP]  
$tcp0 set class_ 2  
$ns attach-agent $n(2) $tcp0  
set sink0 [new Agent/TCPSink]  
$ns attach-agent $n(4) $sink0  
$ns connect $tcp0 $sink0
```

```
#Apply CBR traffic over TCP  
set cbr0 [new Application/Traffic/CBR]  
$cbr0 set packetSize_ 500  
$cbr0 set rate_ 1Mb  
$cbr0 attach-agent $tcp0  
#Schedule events  
$ns at 0.2 "$p0 send"  
$ns at 0.4 "$p1 send"  
$ns at 0.4 "$cbr0 start"  
$ns at 0.8 "$p0 send"  
$ns at 1.0 "$p1 send"  
$ns at 1.2 "$cbr0 stop"  
$ns at 1.4 "$p0 send"  
$ns at 1.6 "$p1 send"  
$ns at 1.8 "Finish"
```

```
#Run the Simulation  
$ns run
```