

- 2.** Implement transmission of ping messages/trace route over a network topology consisting of 6 nodes and find the number of packets dropped due to congestion.

```
set ns [new Simulator]
```

```
set tf [open lab2.tr w]
```

```
$ns trace-all $tf
```

```
set nf [open lab2.nam w]
```

```
$ns namtrace-all $nf
```

```
set n0 [$ns node]
```

```
set n1 [$ns node]
```

```
set n2 [$ns node]
```

```
set n3 [$ns node]
```

```
set n4 [$ns node]
```

```
set n5 [$ns node]
```

```
set n6 [$ns node]
```

```
$n0 label "Ping0"
```

```
$n4 label "Ping4"
```

```
$n5 label "Ping5"
```

```
$n6 label "Ping6"
```

```
$n2 label "Router"
```

```
$ns color 1 "red"
```

```
$ns color 2 "green"
```

```
$ns duplex-link $n0 $n2 100Mb 300ms DropTail
```

```
$ns duplex-link $n1 $n2 1Mb 300ms DropTail
```

```
$ns duplex-link $n3 $n2 1Mb 300ms DropTail
```

```
$ns duplex-link $n5 $n2 100Mb 300ms DropTail
```

```
$ns duplex-link $n2 $n4 1Mb 300ms DropTail
```

```
$ns duplex-link $n2 $n6 1Mb 300ms DropTail
```

```
$ns queue-limit $n0 $n2 5
```

```
$ns queue-limit $n2 $n4 3
```

```
$ns queue-limit $n2 $n6 2
```

```
$ns queue-limit $n5 $n2 5
```

#The below code is used to connect between the ping agents to the node n0, n4 , n5 and n6.

```
set ping0 [new Agent/Ping]
```

```
$ns attach-agent $n0 $ping0
```

```
set ping4 [new Agent/Ping]
```

```
$ns attach-agent $n4 $ping4
```

```
set ping5 [new Agent/Ping]
```

```
$ns attach-agent $n5 $ping5
```

```
set ping6 [new Agent/Ping]
```

```
$ns attach-agent $n6 $ping6
```

```
$ping0 set packetSize_ 50000
```

```
$ping0 set interval_ 0.0001
```

```
$ping5 set packetSize_ 60000
```

```
$ping5 set interval_ 0.00001
```

```
$ping0 set class_ 1
```

```
$ping5 set class_ 2
```

```
$ns connect $ping0 $ping4
```

```
$ns connect $ping5 $ping6
```

```
#Define a 'recv' function for the class 'Agent/Ping'
```

```
#The below function is executed when the ping agent receives a reply from  
the destination
```

```
Agent/Ping instproc recv {from rtt} {
```

```
$self instvar node_
```

```
puts " The node [$node_ id] received an reply from $from with round trip  
time of $rtt"
```

```
}
```

```
proc finish {} {
```

```
global ns nf tf
```

```
exec nam lab2.nam &
```

```
$ns flush-trace
```

```
close $tf
```

```
close $nf
```

```
exit 0
```

```
}
```

```
#Schedule events
```

```
$ns at 0.1 "$ping0 send"
```

```
$ns at 0.2 "$ping0 send"
```

```
$ns at 0.3 "$ping0 send"
```

```
$ns at 0.4 "$ping0 send"
```

\$ns at 0.5 "\$ping0 send"
\$ns at 0.6 "\$ping0 send"
\$ns at 0.7 "\$ping0 send"
\$ns at 0.8 "\$ping0 send"
\$ns at 0.9 "\$ping0 send"

\$ns at 1.0 "\$ping0 send"
\$ns at 1.1 "\$ping0 send"
\$ns at 1.2 "\$ping0 send"
\$ns at 1.3 "\$ping0 send"
\$ns at 1.4 "\$ping0 send"
\$ns at 1.5 "\$ping0 send"
\$ns at 1.6 "\$ping0 send"
\$ns at 1.7 "\$ping0 send"
\$ns at 1.8 "\$ping0 send"

\$ns at 0.1 "\$ping5 send"
\$ns at 0.2 "\$ping5 send"
\$ns at 0.3 "\$ping5 send"
\$ns at 0.4 "\$ping5 send"
\$ns at 0.5 "\$ping5 send"
\$ns at 0.6 "\$ping5 send"
\$ns at 0.7 "\$ping5 send"
\$ns at 0.8 "\$ping5 send"
\$ns at 0.9 "\$ping5 send"

```
$ns at 1.0 "$ping5 send"  
$ns at 1.1 "$ping5 send"  
$ns at 1.2 "$ping5 send"  
$ns at 1.3 "$ping5 send"  
$ns at 1.4 "$ping5 send"  
$ns at 1.5 "$ping5 send"  
$ns at 1.6 "$ping5 send"  
$ns at 1.7 "$ping5 send"  
$ns at 1.8 "$ping5 send"
```

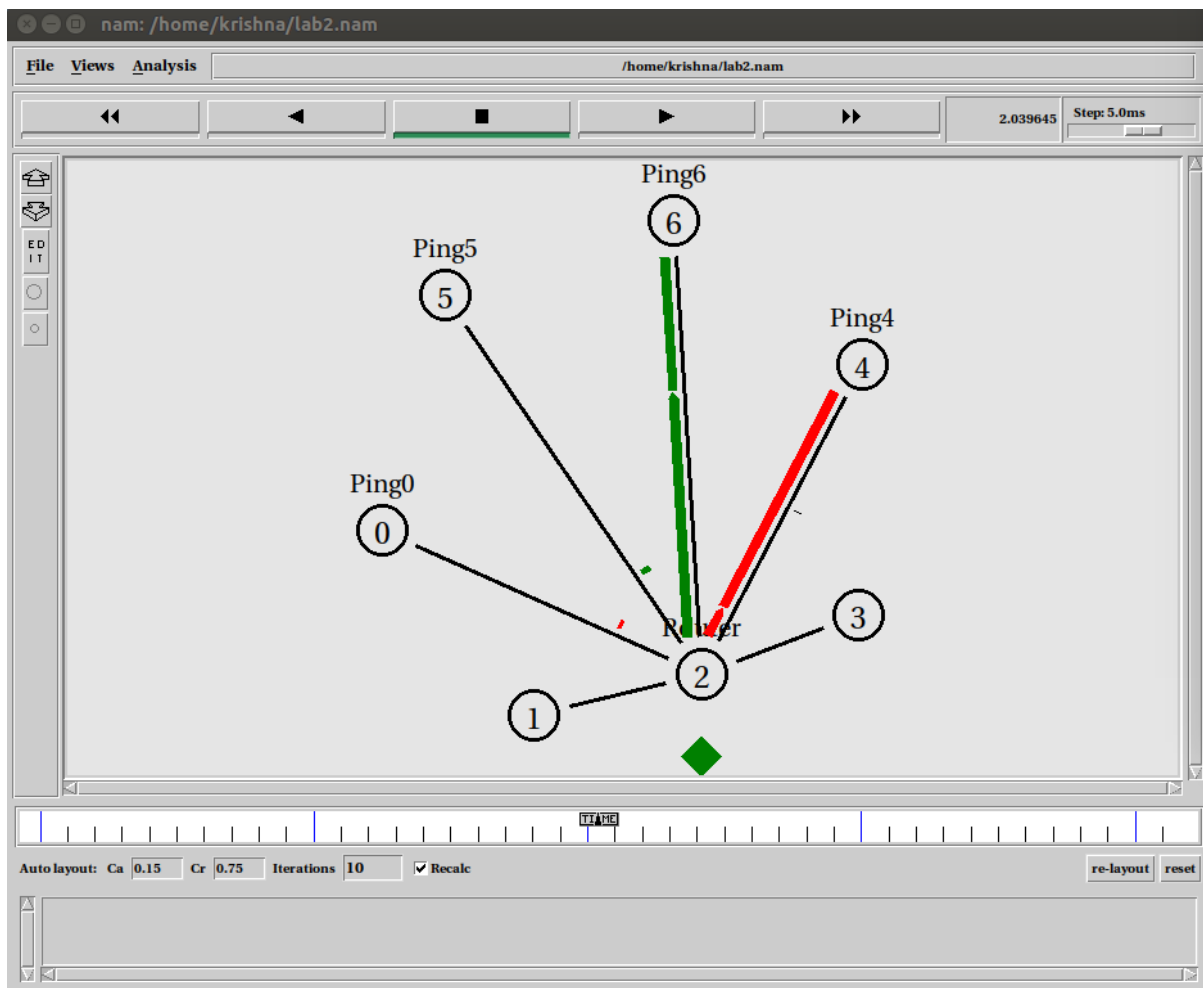
```
$ns at 5.0 "finish"
```

```
$ns run
```

AWK:

```
BEGIN{  
count=0;  
}  
{  
if($1=="d")  
count++;  
}  
END{  
printf("The Total no of Packets Drop is :%d\n\n", count);  
}
```

Topology:



Output:

```
krishna@ubuntu: ~  
krishna@ubuntu:~$ vi lab2.tcl  
krishna@ubuntu:~$ vi lab2.awk  
krishna@ubuntu:~$ awk -f lab2.awk lab2.tr  
The Total no of Packets Drop is :24  
krishna@ubuntu:~$
```