

# Computer networks lab

## Lab 08

### **Question no.01:**

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

```
# Set routing protocol
```

```
$ns rtproto DV
```

```
# Open nam tracefile
```

```
set nf [open prob1.nam w]
```

```
# Open tracefile
```

```
set nt [open trace.tr w]
```

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

```
$ns trace-all $nt
```

```
#Define a 'finish' procedure
```

```
proc finish {} {
```

```
global ns nf nt
```

```
$ns flush-trace
```

```
close $nf
```

```
close $nt
```

```
exec nam -a prob1.nam &
```

```
exit 0
```

```
}
```

```
# create 7 nodes
```

```
puts "create 7 nodes now....."
```

set n1 [\$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]

puts "create connections now....."

# Create connection

\$ns duplex-link \$n1 \$n2 0.512Mb 5ms DropTail

\$ns duplex-link \$n2 \$n3 0.512Mb 5ms DropTail

\$ns duplex-link \$n3 \$n4 0.512Mb 5ms DropTail

```
$ns duplex-link $n4 $n5 0.512Mb 5ms DropTail
```

```
$ns duplex-link $n5 $n6 0.512Mb 5ms DropTail
```

```
$ns duplex-link $n6 $n7 0.512Mb 5ms DropTail
```

```
$ns duplex-link $n7 $n1 0.512Mb 5ms DropTail
```

```
puts "Create agents and attach to appropriate nodes..."
```

```
# Create a CBR traffic source
```

```
set cbr [new Application/Traffic/CBR]
```

```
$cbr set packetSize_ 500 ;# Adjust the packet size as needed
```

```
$cbr set interval_ 0.005 ;# Adjust the interval to control the rate
```

```
# Create a UDP agent and attach it to a node
```

```
set udp [new Agent/UDP]
```

```
$ns attach-agent $n1 $udp ;# Attach the UDP agent to node n1
```

# Create a Null agent (sink) and attach it to a node

```
set null [new Agent/Null]
```

```
$ns attach-agent $n4 $null ;# Attach the Null agent to node n4 for example
```

# Connect the CBR source to the UDP agent

```
$cbr attach-agent $udp
```

# Connect the UDP agent to the Null agent

```
$ns connect $udp $null
```

```
puts "schedule transmitting packets..."
```

```
#schedule transmitting packets
```

```
$ns at 0.02 "$cbr start"
```

```
puts "link down node2 and node3....."
```

```
$ns rtmodel-at 0.40 down $n2 $n3
```

```
$ns rtmodel-at 1.0 up $n2 $n3
```

```
$ns at 1.5 "$cbr stop"
```

```
$ns at 2.00 "finish"
```

```
# Create a queue monitor
```

```
set qm0_1 [new QueueMonitor]
```

```
set qm0_6 [new QueueMonitor]
```

```
# Link the queue monitor to the appropriate queues
```

```
$ns monitor-queue $n1 $n2 $qm0_1
```

```
$ns monitor-queue $n6 $n7 $qm0_6
```

```
# Define a procedure to write queue statistics to a file
```

```
proc record_queue_stats {} {
```

```
    global qm0_1 qm0_6
```

```
set time 0.5 ;# Set the time interval for recording queue stats
```

```
set file0_1 [open qm0_1.tr w]
```

```
set file0_6 [open qm0_6.tr w]
```

```
puts $file0_1 "Time\tPacketsInQueue"
```

```
puts $file0_6 "Time\tPacketsInQueue"
```

```
while {$time <= 2.0} {
```

```
    puts $file0_1 "$time\t[$qm0_1 pkts]"
```

```
    puts $file0_6 "$time\t[$qm0_6 pkts]"
```

```
    update time $time + 0.5
```

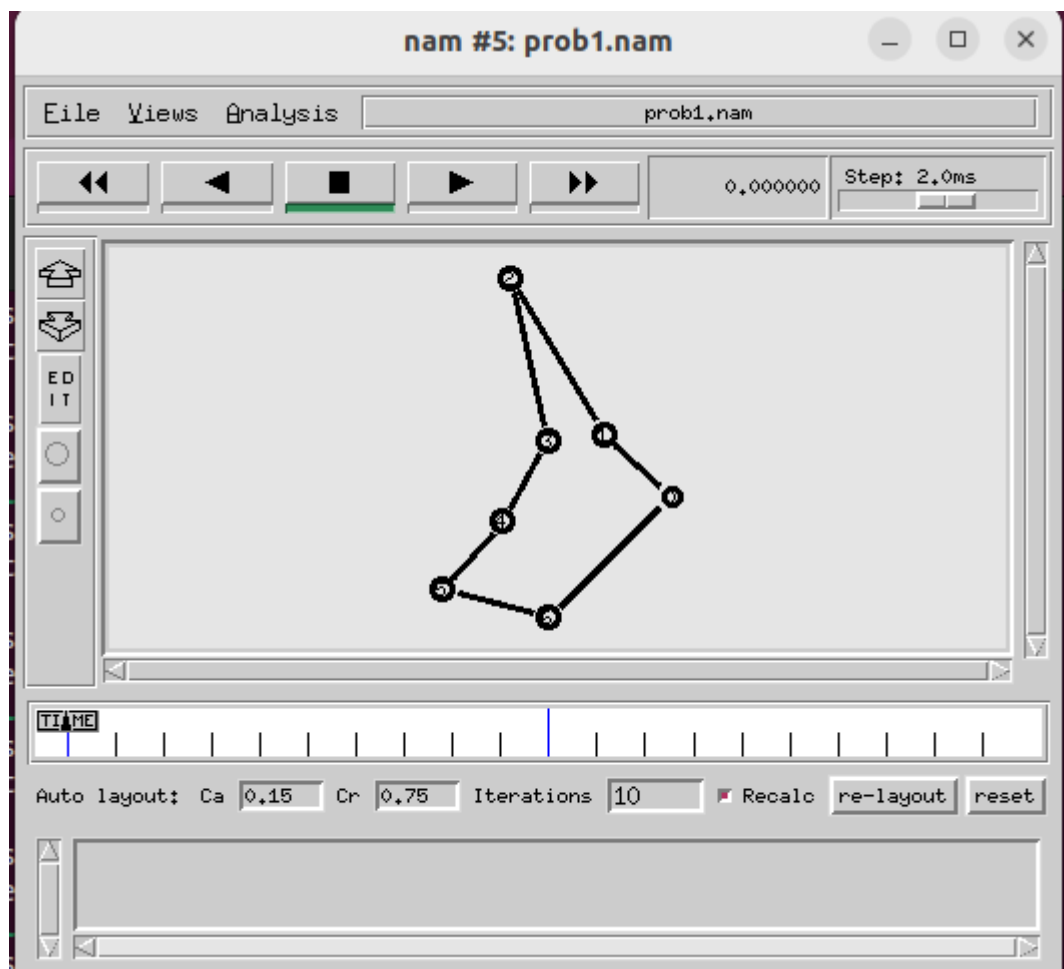
```
}
```

```
close $file0_1
```

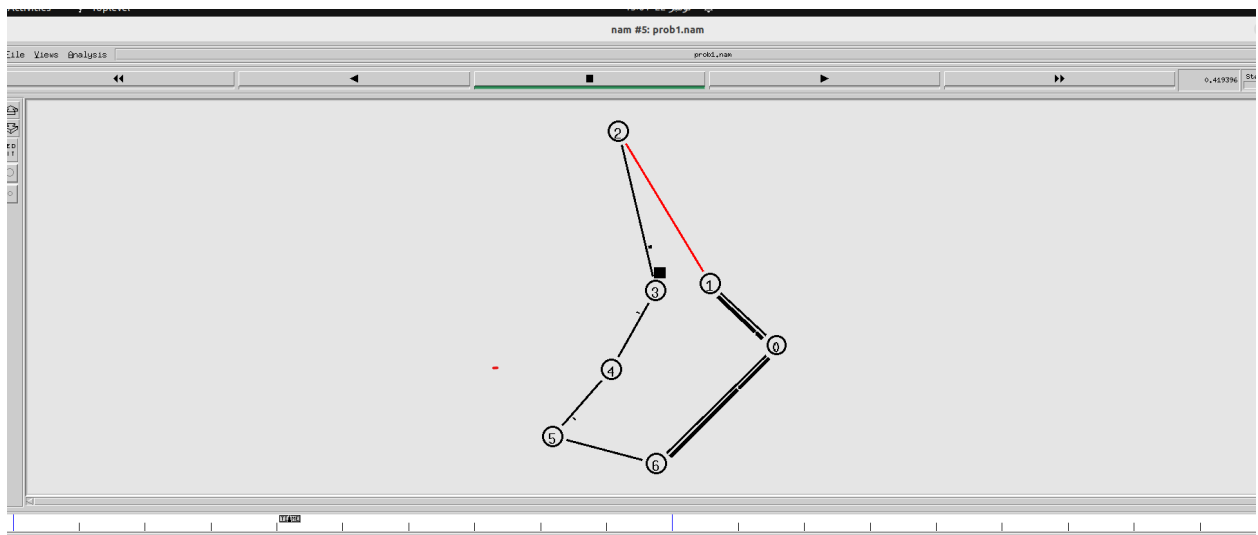
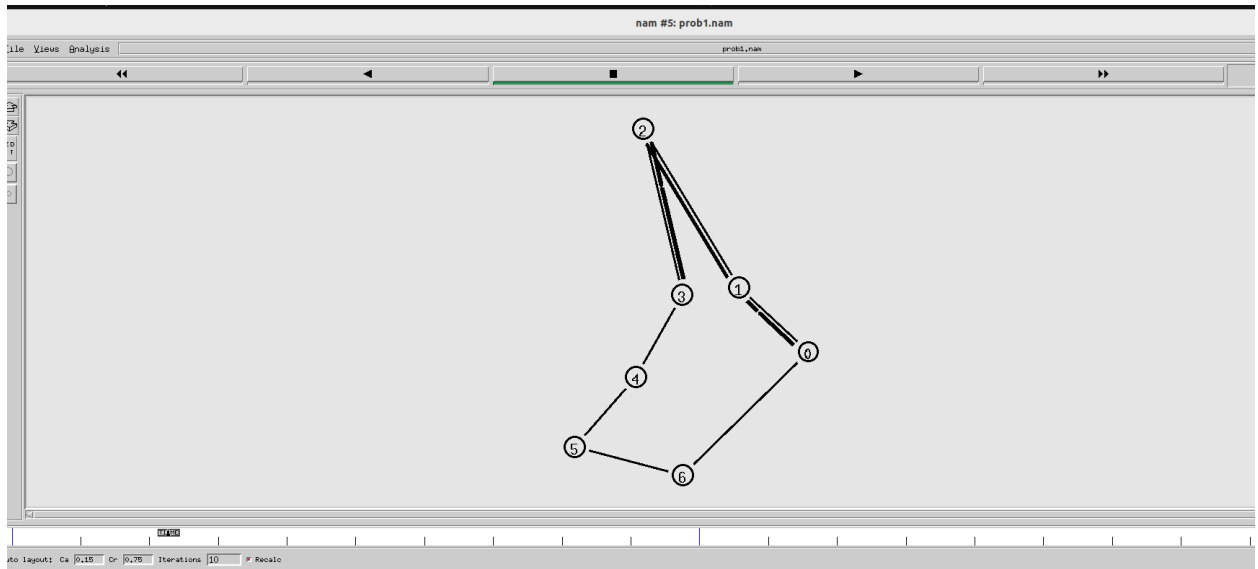
```
close $file0_6
```

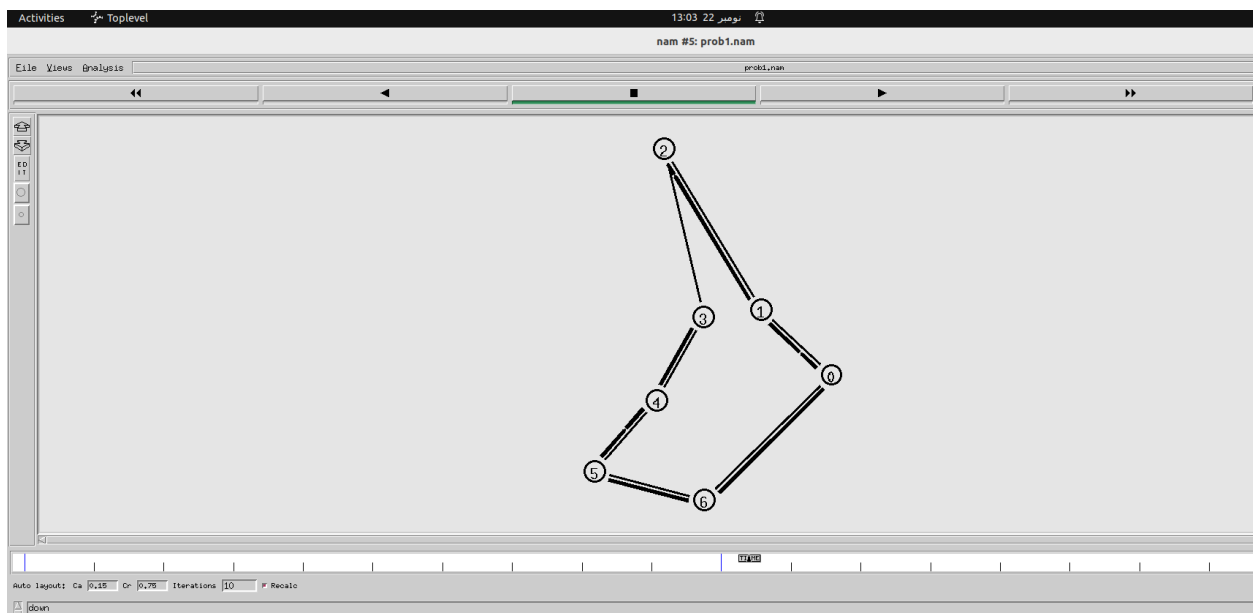
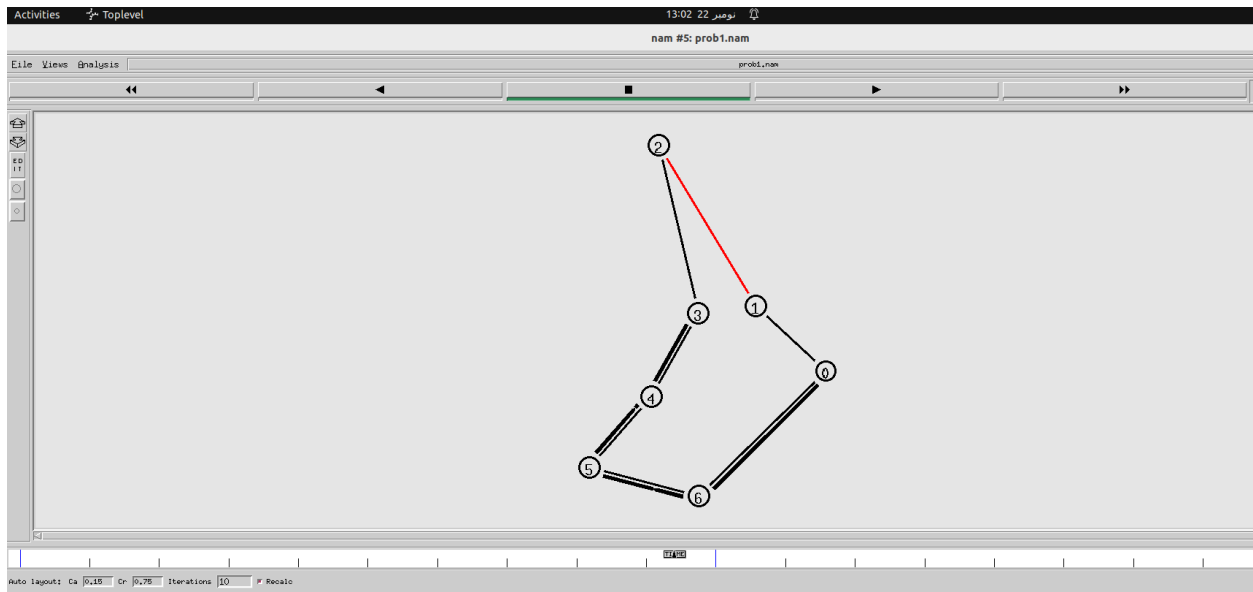
```
}
```

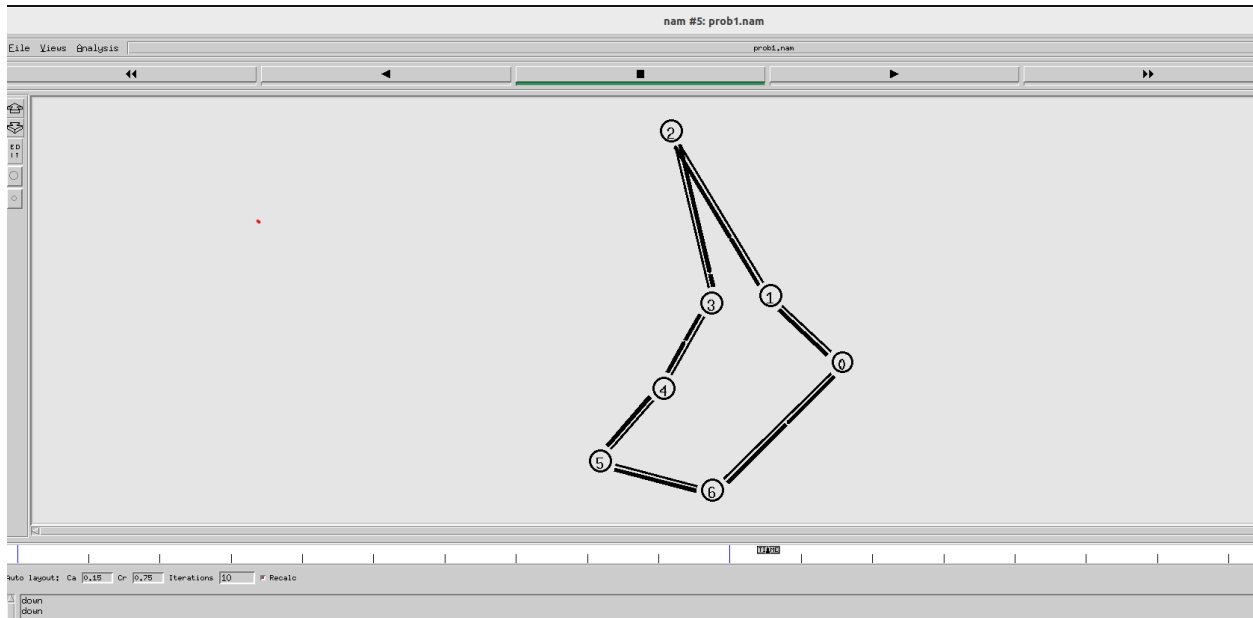
\$ns run











a) Initially, the shortest path was chosen as according to the distance vector algorithm i.e 0-1-2-3. It is a type of algorithm used by packets to discover routes from source to destination. Usually the bellman ford algorithm is used. Packets travel via ports that lead from one router to another. Distance determines the number of hops a packet would take. These distance and direction algorithms help to decide the path of the packet

b) After the links fail, the packets follow an alternative path to the destination as the current shortest path is no longer able to be used. The alternate path is detected during the initialization of DV.(0-6-5-4-3)

c) If we remove the DV routing algorithm, the network would no longer have a protocol to route packets. The consequences would depend upon the underlying NS2 defaults.

question no.02:

set ns [new Simulator]

# Set routing protocol

\$ns rtpeto DV

```
# Open nam tracefile
```

```
set nf [open prob1.nam w]
```

```
# Open tracefile
```

```
set nt [open trace.tr w]
```

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

```
$ns trace-all $nt
```

```
#Define a 'finish' procedure
```

```
proc finish {} {
```

```
global ns nf nt
```

```
$ns flush-trace
```

```
close $nf
```

```
close $nt
```

```
exec nam -a prob1.nam &
```

```
exit 0
```

```
}
```

```
# Creating Nodes
```

```
for {set i 0} {$i < 7} {incr i} {
```

```
    set n($i) [$ns node]
```

```
}
```

```
# Creating Links
```

```
for {set i 1} {$i < 7} {incr i} {
```

```
    $ns duplex-link $n(0) $n($i) 512Kb 10ms SFQ
```

```
}
```

```
# Orienting The nodes
```

\$ns duplex-link-op \$n(0) \$n(1) orient left-up

\$ns duplex-link-op \$n(0) \$n(2) orient right-up

\$ns duplex-link-op \$n(0) \$n(3) orient right

\$ns duplex-link-op \$n(0) \$n(4) orient right-down

\$ns duplex-link-op \$n(0) \$n(5) orient left-down

\$ns duplex-link-op \$n(0) \$n(6) orient left

\$ns color 1 blue

\$ns color 2 red

# TCP\_Config

set tcp0 [new Agent/TCP]

\$tcp0 set class\_1

\$ns attach-agent \$n(1) \$tcp0

```
set sink0 [new Agent/TCPSink]
```

```
$ns attach-agent $n(4) $sink0
```

```
$ns connect $tcp0 $sink0
```

```
# UDP_Config
```

```
set udp0 [new Agent/UDP]
```

```
$udp0 set class_2
```

```
$ns attach-agent $n(2) $udp0
```

```
set null0 [new Agent/Null]
```

```
$ns attach-agent $n(5) $null0
```

```
$ns connect $udp0 $null0
```

```
# CBR Config
```

```
set cbr0 [new Application/Traffic/CBR]
```

\$cbr0 set rate\_ 256Kb

\$cbr0 attach-agent \$udp0

set ftp0 [new Application/FTP]

\$ftp0 attach-agent \$tcp0

\$ns rtmodel-at 0.5 down \$n(0) \$n(5)

\$ns rtmodel-at 0.9 up \$n(0) \$n(5)

\$ns rtmodel-at 0.7 down \$n(0) \$n(4)

\$ns rtmodel-at 1.2 up \$n(0) \$n(4)

\$ns at 0.1 "\$ftp0 start"

\$ns at 1.5 "\$ftp0 stop"

\$ns at 0.2 "\$cbr0 start"



\$ns at 1.3 "\$cbr0 stop"

\$ns at 2.0 "finish"

\$ns run

