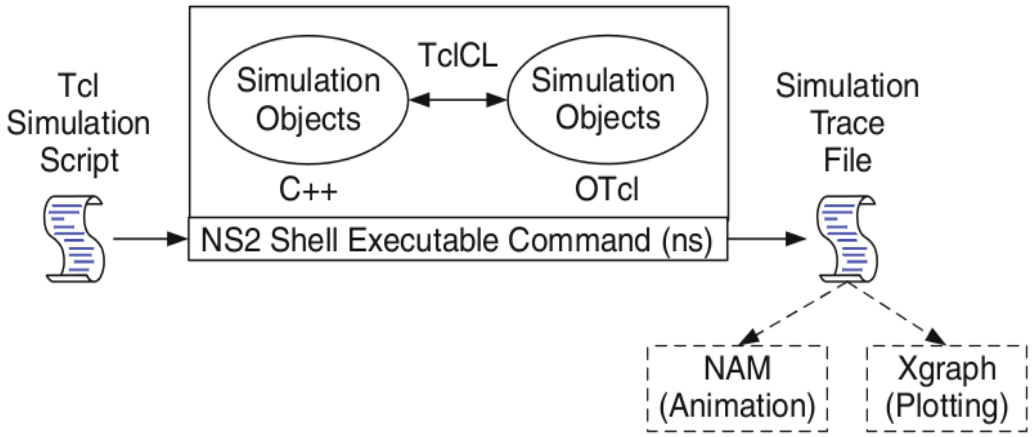
# Introduction to NS-2

**NS2** is an open-source simulation tool that runs on Linux. It is a discreet event simulator targeted at networking research and provides substantial support for simulation of routing, multicast protocols and IP protocols, such as UDP, TCP, RTP and SRM over wired and wireless (local and satellite) networks

- Widely known as NS2, is simply an event driven simulation tool.
- Useful in studying the dynamic nature of communication networks.
- Simulation of wired as well as wireless network functions and protocols (e.g., routing algorithms, TCP, UDP) can be done using NS2.
- In general, NS2 provides users with a way of specifying such network protocols and simulating their corresponding behaviors.

**Basic Architecture of NS2**




## Simple Script Explanation

**set *ns* [new Simulator] :** generates an NS simulator object instance, and assigns it to variable *ns*. What this line does is the following:

- Initialize the packet format (ignore this for now)
- Create a scheduler (default is calendar scheduler)
- Select the default address format (ignore this for now)

The "Simulator" object has member functions that do the following:

- Create compound objects such as nodes and links (described later)
- Connect network component objects created (ex. attach-agent)
- Set network component parameters (mostly for compound objects)
- Create connections between agents (ex. make connection between a "tcp" and "sink")
- Specify NAM display options
- Etc.

***$ns* color *fid color*:** is to set color of the packets for a flow specified by the flow id (fid). This member function of "Simulator" object is for the NAM display, and has no effect on the actual simulation.

***$ns* namtrace-all *file-descriptor*:** This member function tells the simulator to record simulation traces in NAM input format. It also gives the file name that the trace will be written to later by the command ***$ns* flush-trace**. Similarly, the member function **trace-all** is for recording the simulation trace in a general format.

**proc *finish* {}:** is called after this simulation is over by the command ***$ns* at 5.0 "*finish*"**. In this function, post-simulation processes are specified.

**set *n0* [*$ns* node]**: The member function **node** creates a node. A node in NS is compound object made of address and port classifiers.

***$ns* duplex-link *node1 node2 bandwidth delay queue-type*:** creates two simplex links of specified bandwidth and delay, and connects the two specified nodes. In NS, the output queue of a node is implemented as a part of a link; therefore users should specify the queue-type when creating links. The queue can be either of type DropTail or RED.

***$ns* queue-limit *node1 node2 number***: This line sets the queue limit of the two simplex links that connect node1 and node2 to the number specified.

**set tcp [new Agent/TCP]**: This line shows how to create a TCP agent. But in general, users can create any agent or traffic sources in this way. Agents and traffic sources are in fact basic objects (not compound objects), mostly implemented in C++ and linked to OTcl. Therefore, there are no specific Simulator object member functions that create these object instances. To create agents or traffic sources, a user should know the class names these objects (Agent/TCP, Agnet/TCPSink, Application/FTP and so on.

**$ns attach-agent node agent**: The attach-agent member function attaches an agent object created to a node object. Actually, what this function does is call the attach member function of specified node, which attaches the given agent to itself. Therefore, a user can do the same thing by, for example, $n0 attach $tcp. Similarly, each agent object has a member function attach-agent that attaches a traffic source object to itself

***$ns* connect *agent1 agent2***: After two agents that will communicate with each other are created, the next thing is to establish a logical network connection between them. This line establishes a network connection by setting the destination address to each others' network and port address pair.

***$ns* at *time "string"***: This member function of a Simulator object makes the scheduler (scheduler_ is the variable that points the scheduler object created by [new Scheduler] command at the beginning of the script) to schedule the execution of the specified string at given simulation time.

For example, ***$ns* at *0.1 "$cbr start"*** will make the scheduler call a **start** member function of the CBR traffic source object, which starts the CBR to transmit data. In NS, usually a traffic source does not transmit actual data, but it notifies the underlying agent that it has some amount of data to transmit, and the agent, just knowing how much of the data to transfer, creates packets and sends them.

For the application like tcp or udp to run, we need to set two agents and the application which should run in between.When using tcp, we have ftp as the application and tcpsink as the end agent. Connection must be made between tcp and tcpsink, same in udp with cbr and null respectively.

**set tcp [new Agent/TCP]**
**$ns attach-agent $n0 $tcp**

This would make a tcp agent and connect it with the node.

**set ftp [new Application/FTP]**
**$ftp attach-agent $tcp**

Now the ftp is connected with the tcp

**set agent [new Agent/TCPSink]**

**$ns attach-agent $n3 $sink**

Now the tcpsink is set to a node where the tcp packets are received.

The tcp and sink (agents) needs to be connected, such that the network flows.

**$ns connect $tcp $sink**

Same for udp

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

**$ns attach-agent $n2 $udp**

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

**$cbr attach-agent $udp**

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

**$ns attach-agent $n3 $null**

**$ns connect $udp $null**

The file should be saved in .tcl format and should use ns filename.tcl to run

## *Ns2Tracefile format*

| $1 | $2 | $3 | $4 | $5 | $6 | $7 | $8 | $9 | $10 | $11 | $1 2 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Event | Time | From node | To node | Pkt type | Pkt size | Flags | Flowid | Srcaddr | Destaddr | Seqnum | Pktid |

- **Event or type identifier**

```
+: a packet enqueue event
-:a packet dequeue event
r:apacket  receptionevent
d: a packet drop event
c: packet collision at the MAC level
```

- **Time:** at which the packet tracing string is created.
- **Source and destination node:** source and destination ID's of tracing objects
- **Packet type :** type of the packet such as, TCP,UDP, CBR,ACK, FTP etc.

- **Packet size:** size of packets in bytes.
- **Flags:**7digit string

  "-" :disable

  1st: "E"-ECN (Explicit Congestion Notification)echo is enabled. 2nd: "P"-the priority in the IP header is enabled.

  3rd: notinuse

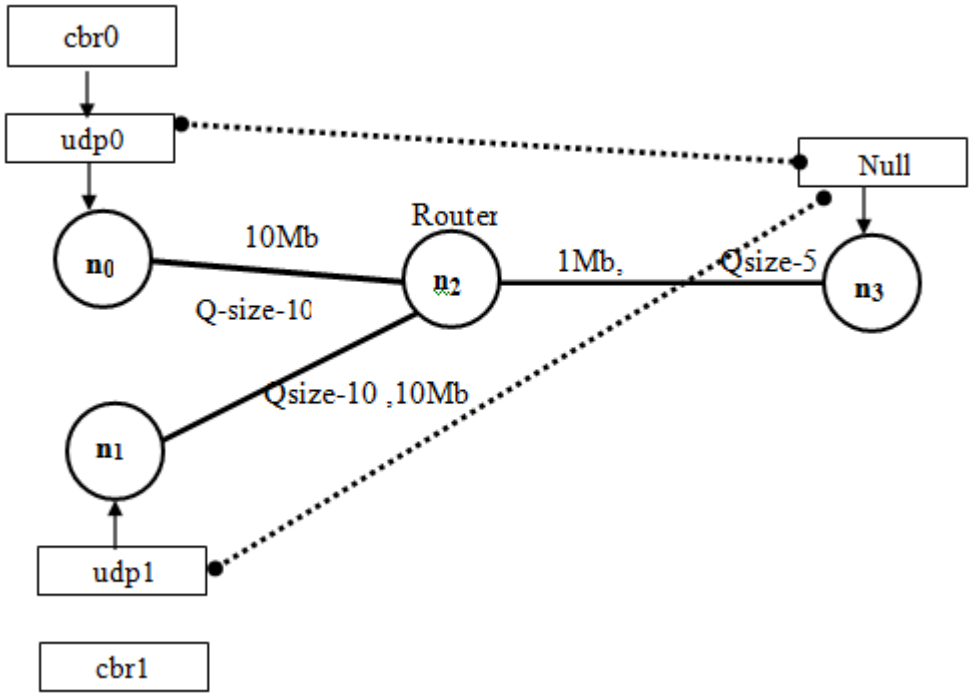
  4th:"A"-Congestion action.

  5th:"E"-Congestion has occurred. 6th:"F"-the TCP fast start is used.

  7th:"N"-Explicit Congestion Notification (ECN)is on.
- **Flow id**
- **Source and destination address:** The format of these two fields is "a.b", where "a" is the address and "b" is the port.
- **Sequence number**
- **Packet unique ID**

**1. Implement three nodes point- to- point network with duplex links between them. Set the queue size and vary the bandwidth and find the number of packets dropped.**

**Topology Design:**




**set ns [ new Simulator ]** `# Creation of simulator object ,S is capital`

set tf [ open lab1.tr w ] `# open a trace file in write mode`

$ns trace-all $tf

set nf [ open lab1.nam w ] `# open a nam file in write mode`

$ns namtrace-all $nf

**# The below code is used to create the 4 nodes.**

set n0 [$ns node]

set n1 [$ns node]

set n2 [$ns node]

set n3 [$ns node]

**#The below used to give color to the packets.**

$ns color 1 "red"

```
$ns color 2 "blue"

# The below used to label the nodes.
$n0 label "Source/udp0"
$n1 label "Source/udp1"
$n2 label "Router"
$n3 label "Destination/Null"

# The below code is used to create the link between the nodes., D & T capital
$ns duplex-link $n0 $n2 10Mb 300ms DropTail
$ns duplex-link $n1 $n2 10Mb 300ms DropTail
$ns duplex-link $n2 $n3 1Mb 300ms DropTail

#The below code is used to set the queue size b/w the nodes
$ns set queue-limit $n0 $n2 10
$ns set queue-limit $n1 $n2 10
$ns set queue-limit $n2 $n3 5

#the below code is used to attach an UDP0 agent to n0, UDP1
#agent to n1 and null agent to n3.

set udp0 [new Agent/UDP]          # attaching transport layer protocols

$ns attach-agent $n0 $udp0
set cbr0 [new Application/Traffic/CBR]    # attach AL protocols
$cbr0 attach-agent $udp0
set null3 [new Agent/Null]        # creating sink(destination) node
$ns attach-agent $n3 $null3

set udp1 [new Agent/UDP]
$ns attach-agent $n1 $udp1
set cbr1 [new Application/Traffic/CBR]
$cbr1 attach-agent $udp1

#The below code sets the udp0 packets to red and udp1 packets to blue color
$udp0 set class_ 1
$udp1 set class_ 2

#The below code is used to connect the agents.
$ns connect $udp0 $null3
```

```
$ns connect $udp1 $null3

#The below code is used to set the packet size to 500., S & M capital

$cbr1 set packetSize_ 500Mb

#The below code is used to set the interval of the packets,
#i.e., Data rate of the packets. if the data rate is high then packets drops are high.

$cbr1 set interval_ 0.005

proc finish { } {
        global ns nf tf
        $ns flush-trace          # clears trace file contents

        exec nam lab1.nam &    # Exe nam visualization

        close $tf
        close $nf
        exit 0                      #Close application
}
$ns at 0.1 "$cbr0 start"          #Scheduler event

$ns at 0.1 "$cbr1 start"

$ns at 10.0 "finish"

$ns run                        # start simulation
```

**AWK Script**

```
BEGIN {
```

```
#include<stdio.h>
count=0;
}
{
        if($1==”d”)             #d: pkt drops
                count++;
} END {
        printf(“Total no. of packets dropped due to congestion is:%d”,count);
}
```

**Output:**

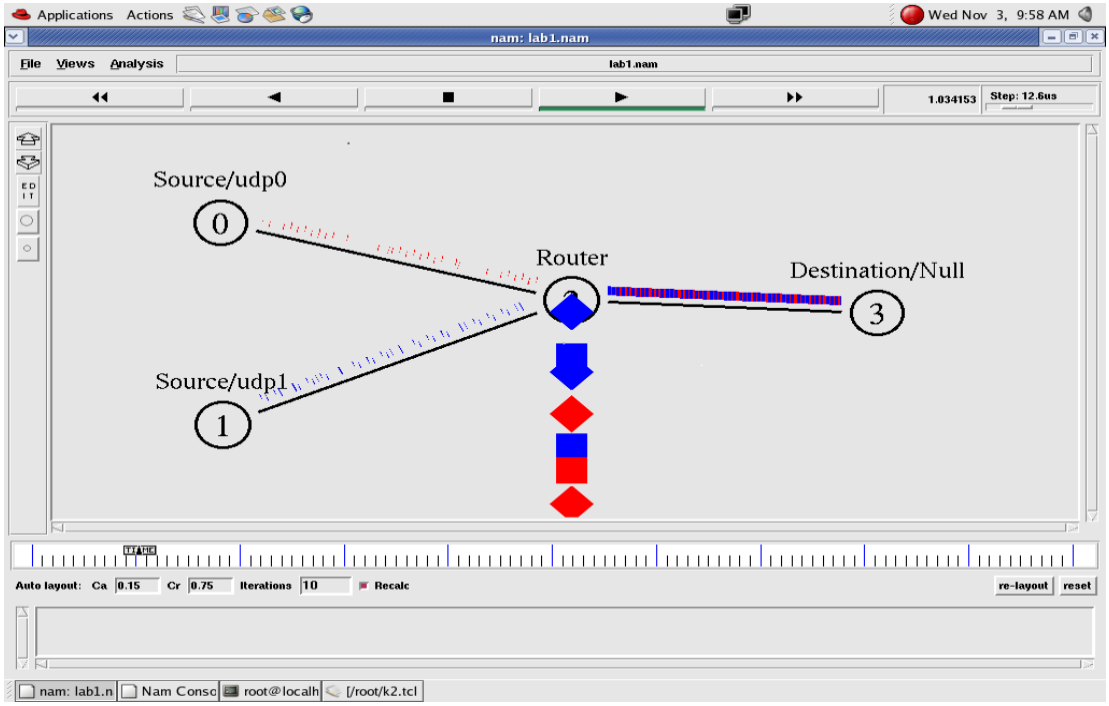
**ns lab1.tcl**

**awk –f lab1.awk lab1.tr**

**The Total no of packets Dropped due to congestion:4560**

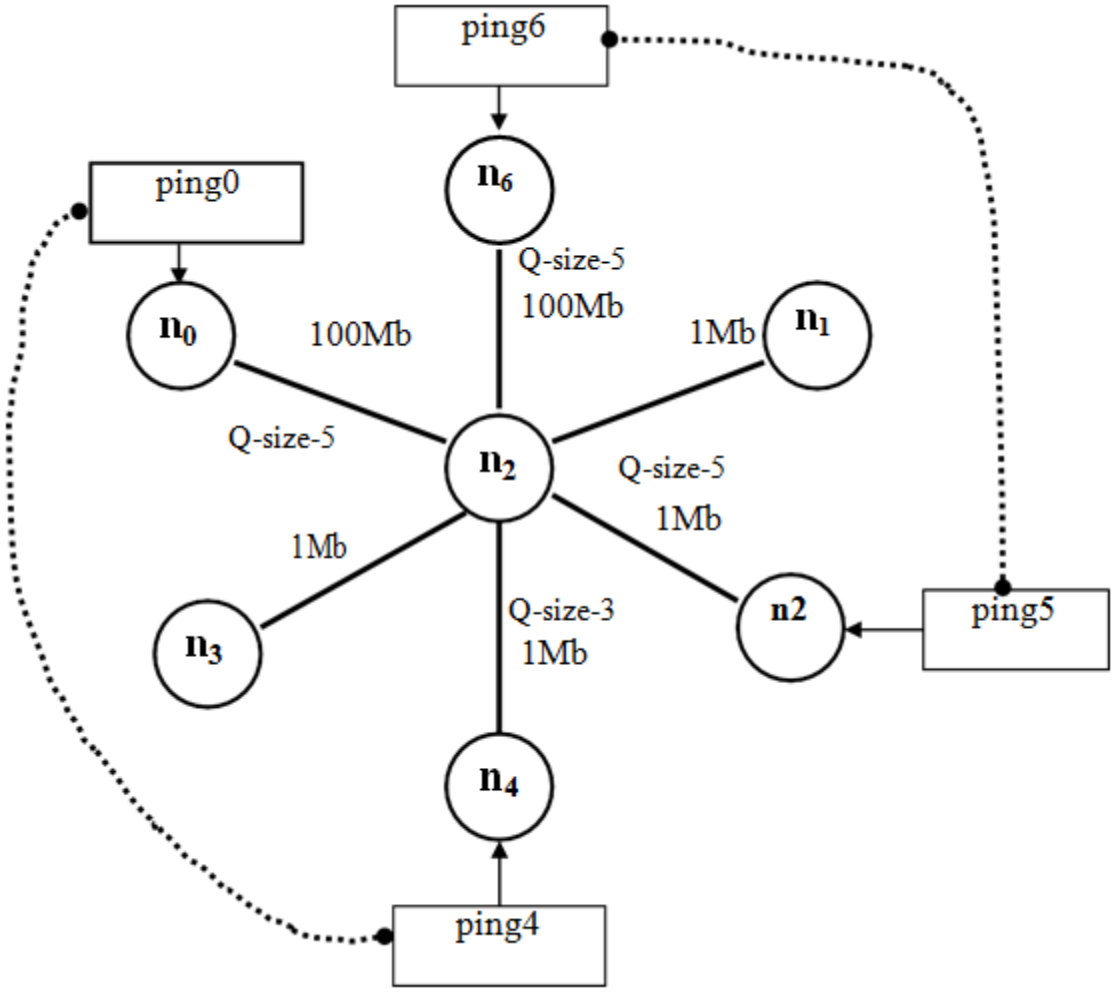
## Snapshot:

**Fig 1: The drop of packets at node 2(Router)**



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

**Topology Design**




## Note:

1. Vary the bandwidth and queue size between the nodes n0-n2 , n2-n4. n6-n2 and n2-n5 and see the number of packets dropped at the nodes.

2. The ping agent can be attached to the nodes by giving
   i. set ping0 [new Agent/Ping]
      $ns attach-agent $n0 $ping0
   ii. The ping agents can be connected using the syntax given below.
      $ns connect $ping0 $ping4

```
set ns [new Simulator]

set tf [open ping.tr w]
$ns trace-all $tf

set nf [open ping.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 $n2 $n6 1Mb 300ms DropTail
$ns duplex-link $n5 $n2 100Mb 300ms DropTail
$ns duplex-link $n2 $n4 1Mb 300ms DropTail
$ns duplex-link $n3 $n2 1Mb 300ms DropTail
$ns duplex-link $n1 $n2 1Mb 300ms DropTail

$ns queue-limit $n0 $n2 5
$ns queue-limit $n2 $n6 2
$ns queue-limit $n2 $n4 3
$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

#set pkt size and interval
$ping0 set packetSize_ 50000
$ping0 set interval_ 0.0001

$ping5 set packetSize_ 60000
$ping5 set interval_ 0.00001

#set pkt color
$ping0 set class_ 1
$ping5 set class_ 2

#connect agents
$ns connect $ping0 $ping4
$ns connect $ping5 $ping6

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

Agent/Ping instproc recv {from rtt} {   #Agent: class, instaproc: member fun, node: mem var
$self instvar node_
puts " The node [$node_ id] received an reply from $from with round trip time of  $rtt"
}

proc finish {} {
global ns nf tf
exec nam ping.nam &        #exe visualization
$ns flush-trace                #simulator method
close $tf
close $nf
exit 0
}

#The below code makes the link down(failure) at 0.9 from n2 to n6 and when the time #becomes 1.5 the link between n2 to n6 is enabled.

$ns rtmodel-at 0.9 down $n2 $n6
$ns rtmodel-at 1.5 up $n2 $n6
```

**#ping messages**

```
$ns at 0.1 "$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"          #exe simulation procedure
$ns run                      # start simulation
```

**AWK Script:**

```
BEGIN{
#include<stdio.h>
count=0;
}
{
        if($1==”d”)
                count++;
} END {
        printf(“Total no. of packets dropped due to congestion is:%d”,count);
```

## Output:

**ns ping.tcl**

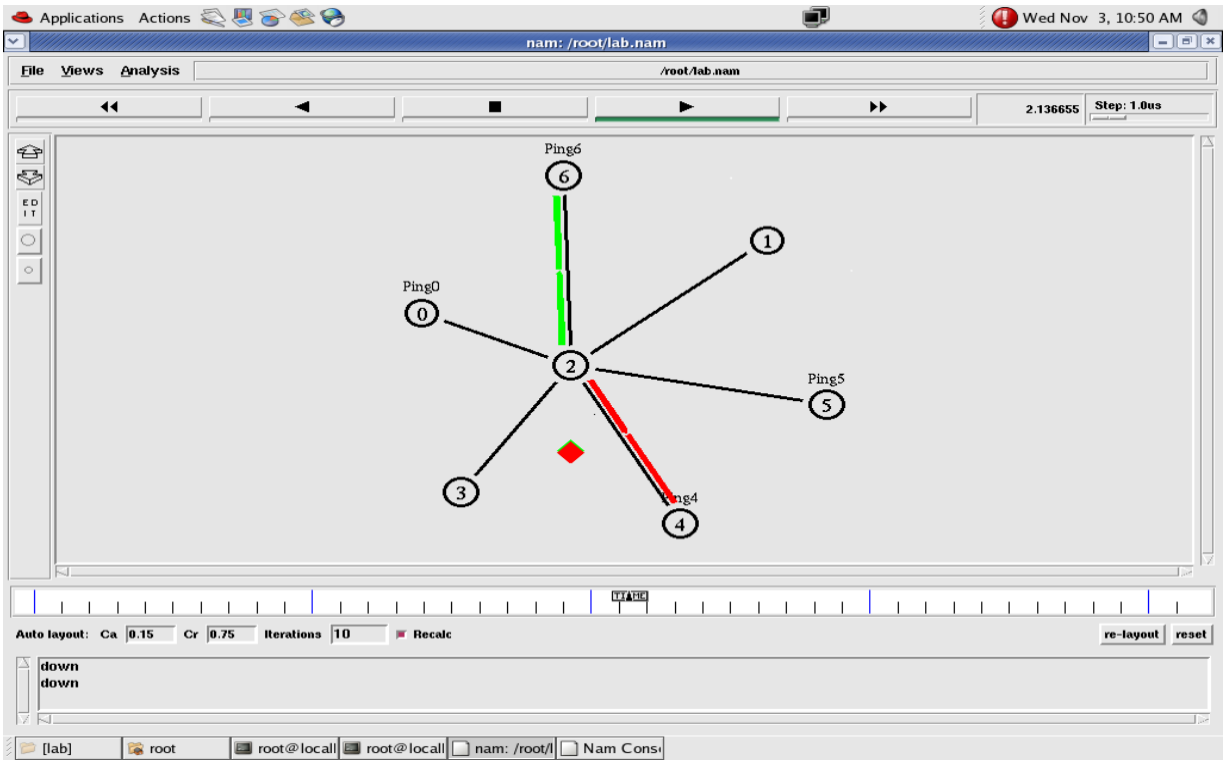
**awk –f ping.awk ping.tr**

**Total no. of packets dropped due to congestion is: 20**

**Snapshot:**

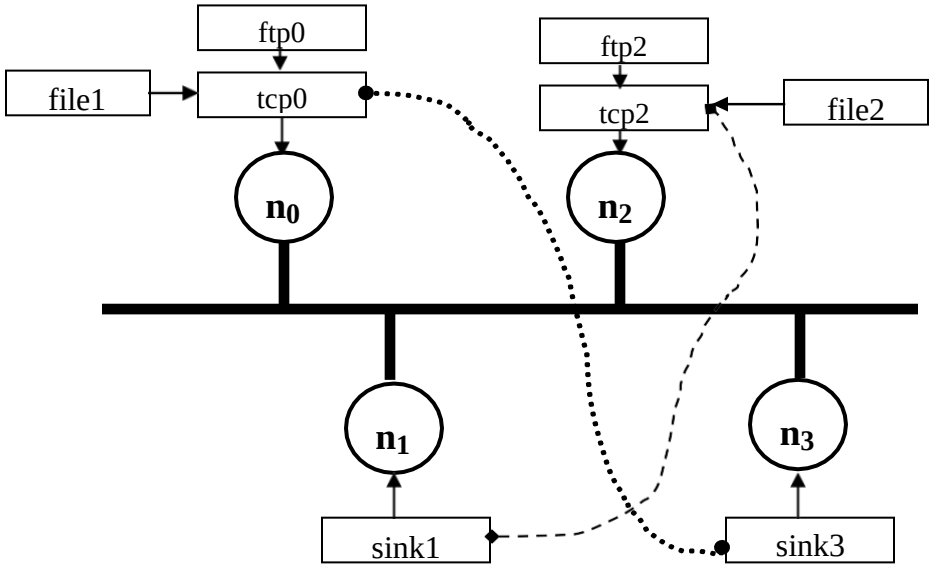
```
dell@ewit: ~
dell@ewit:~$ ns ping.tcl
 The node 0 received an reply from 4 with round trip time of 1604.5
 The node 0 received an reply from 4 with round trip time of 1904.5
 The node 0 received an reply from 4 with round trip time of 2204.5
 The node 5 received an reply from 6 with round trip time of 1685.3
 The node 0 received an reply from 4 with round trip time of 2404.5
 The node 0 received an reply from 4 with round trip time of 2304.5
 The node 5 received an reply from 6 with round trip time of 2065.3
 The node 0 received an reply from 4 with round trip time of 2404.5
 The node 5 received an reply from 6 with round trip time of 2145.3
dell@ewit:~$
```

**Fig 2 : The pinging of the nodes.**



## 3. Implement an Ethernet LAN using n nodes and set multiple traffic nodes and plot congestion window for different source/destination.

**Design:**




**Note:** **The congestion at the nodes can be traced by the following steps:**

**Step 1:** Create a file1.tr and file2.tr

**set file1 [open file1.tr w]**

**set file2 [open file2.tr w]**

**Step 2:** Attach these files (file1.tr & file2.tr) to the agents tcp0 and tcp1 as shown below.

**$tcp0 attach $file1**

**$tcp1 attach $file2**

**Step 3:** To trace the congestion window value we use the trace command to trace the congestion window values.

$tcp0 trace cwnd_

$tcp1 trace cwnd_

Then, the congestion window values are stored in the file1 and file2.

**Step 4:** To plot the graph the steps are given below

**ns lab7.**tcl

**awk -f lab7.awk file1.tr > tcp1**

**awk –f lab7.awk file2.tr > tcp2**

**xgraph –x "Time" -y "cwnd" tcp0 tcp2**

## Program:

```
set ns [new Simulator]

set tf [open lab7.tr w]
$ns trace-all $tf

set nf [open lab7.nam w]
$ns namtrace-all $nf

set n0 [$ns node]
set n1 [$ns node]
set n2 [$ns node]
set n3 [$ns node]
$ns color 1 “red”
$ns color 2 blue”
```

**#Create LAN, all in single line**

```
$ns make-lan "$n0 $n1 $n2 $n3" 10Mb 10ms LL Queue/DropTail Mac/802_3
```

**#Attach Agents to node**

```
set tcp0 [new Agent/TCP]
$ns attach-agent $n0 $tcp0

set ftp0 [new Application/FTP]
$ftp0 attach-agent $tcp0

set sink3 [new Agent/TCPSink]
$ns attach-agent $n3 $sink3

$ns connect $tcp0 $sink3

set tcp2 [new Agent/TCP]
$ns attach-agent $n2 $tcp2

set ftp2 [new Application/FTP]
$ftp2 attach-agent $tcp2

set sink1 [new Agent/TCPSink]
$ns attach-agent $n1 $sink1
```

```
$ns connect $tcp2 $sink1

#set color

$tcp0 set class_ 1
$tcp2 set class_ 2

######To trace the congestion window##########

set file1 [open file1.tr w]      #create file1
$tcp0 attach $file1              #attach file1 to agent tcp0
$tcp0 trace cwnd_
$tcp0 set maxcwnd_ 10

set file2 [open file2.tr w]      #create file2
$tcp2 attach $file2              ]#attach file2 to agent agent tcp2
$tcp2 trace cwnd_

#finish procudeure

proc finish { } {
                global nf tf ns
                $ns flush-trace
                exec nam lab7.nam &
                close $nf
                close $tf
                exit 0           #close application
}

#schedule the traffic agents ftp0 and ftp2

$ns   at   0.1 "$ftp0 start"
$ns   at   1.5 "$ftp0 stop"
$ns   at   2 "$ftp0 start"
$ns   at   3 "$ftp0 stop"
$ns   at   0.2 "$ftp2 start"
```

```
$ns at 2 "$ftp2 stop"

$ns at 2.5 "$ftp2 start"
$ns at 4 "$ftp2 stop"
$ns at 5.0 "finish"
$ns run
```

**AWK Script: (Trace file will have 7 fields)**

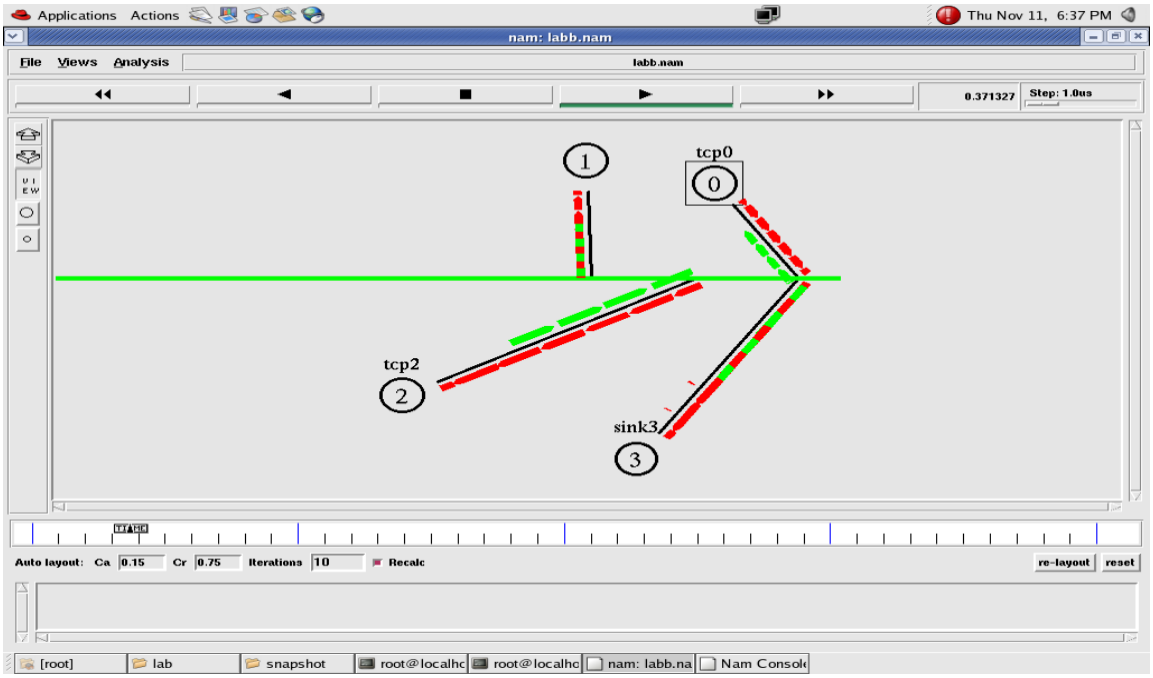
```
BEGIN{
        #include<stdio.h>
}
{
 if($6=="cwnd_")
    printf("%f \t %f \n", $1,$7);        #1: Simulation time, $7:Convlue
}
END {
        puts "DONE"
}
```

**To run :** **ns lab7.tcl**

**awk –f lab7.awk file1.tr>tcp0**

**awk –f lab7.awk file2.tr>tcp2**

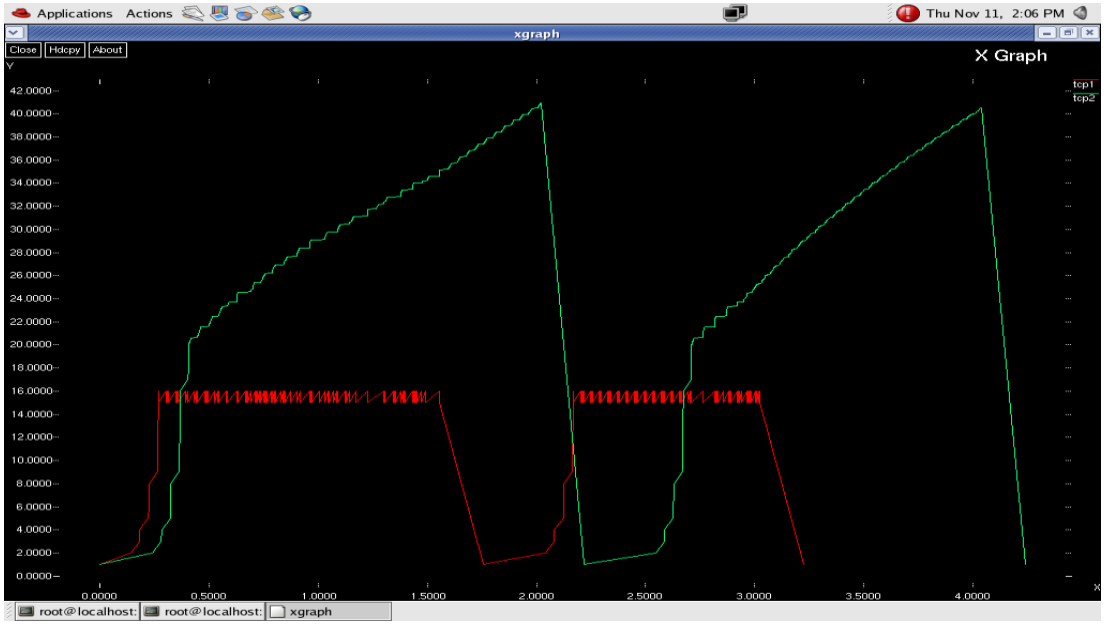
**xgraph –x "time" –y "convalue " tcp0 tcp2**




Note:

To set the foreground and background color of the graph choose the appropriate options by giving **xgraph -**

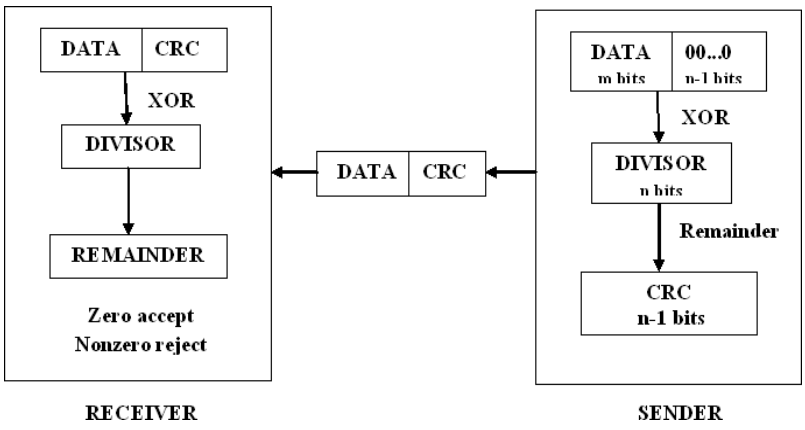
**Fig 3: The congestion window graph(X axis: Time, Y: Conval)**



**4. Develop a program for error detecting code using CRC-CCITT (16- bits).**

The cyclic redundancy check (CRC) is a technique used to detect errors in digital data. CRC is a hash function that detects accidental changes to raw computer data commonly used in digital telecommunications networks and storage devices such as hard disk drives. This technique was invented by W. Wesley Peterson in 1961 and further developed by the CCITT (International Telegraph and Telephone Consultative Committee). Cyclic redundancy checks are quite simple to implement in hardware and can be easily analyzed mathematically. It is one of the better techniques in detecting common transmission errors.

As explained above cyclic redundancy check is also applied to storage devices like hard disks. In this case, check bits are allocated to each block in the hard disk. When a corrupt or incomplete file is read by the computer, the cyclic redundancy error is reported. This could be from another storage device or from CD/DVDs. The common reasons for errors include system crashes, incomplete or corrupt files, or files with lots of bugs.



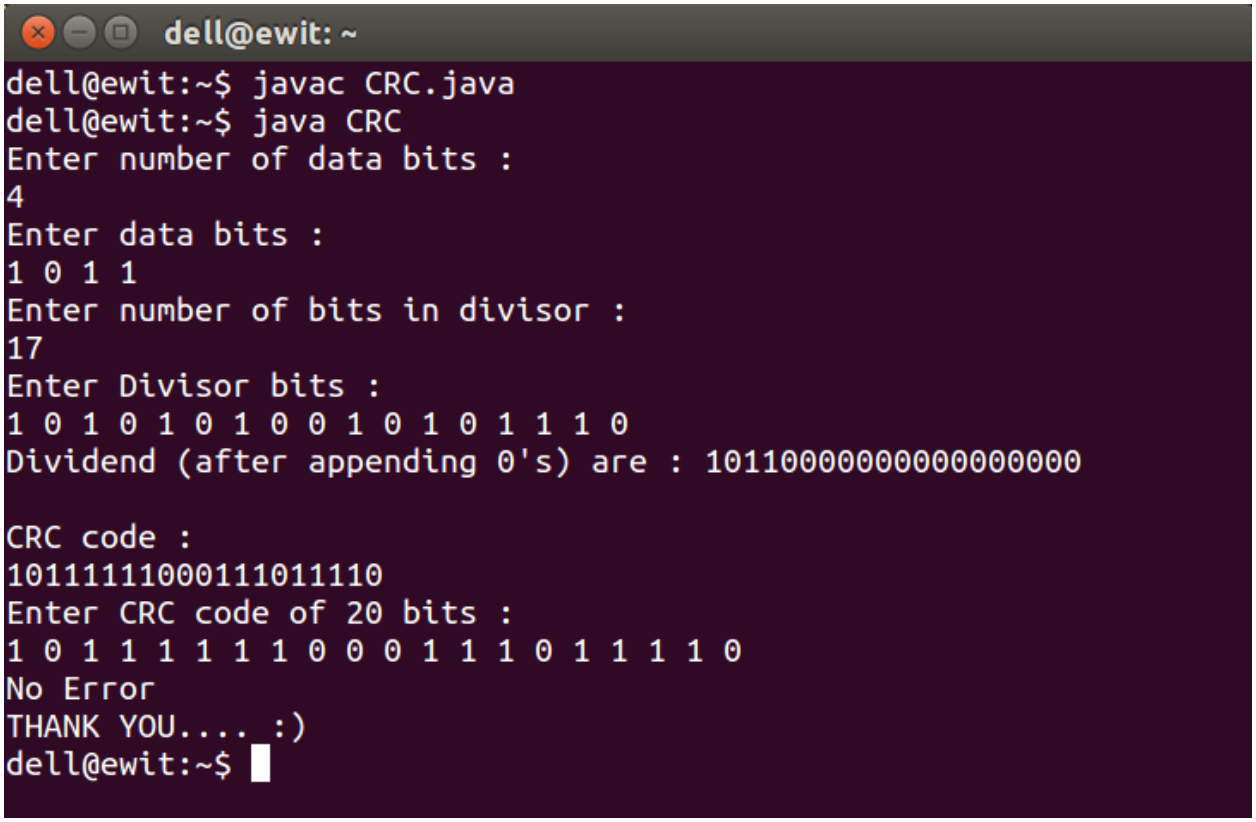

## *Steps:*

- First, a string ofn-1 0sisappendedtothe data unit.
- Thenumberof0sisonelessthanthenumberofbitsinthedivisor which is n bits.
- Then the newly elongated data unit is divided by the divisor using a process called binary division (XOR).
- TheremainderisCRC.TheCRCreplacestheappended0sattheend of the data unit.
- The data unit arrives at the receiver first, followed by the CRC.
- The receiver treats whole string as the data unit and divides it by the same divisor that was used to find the CRC remainder.
- If the remainder is 0 then the data unit is error free. Otherwise it having some error and it must be discarded.

```
import java.io.*;
import java.util.*;
class CRC
{
	public static void main(String a[]) throws IOException
	{
		Scanner sc=new Scanner(System.in);
		int[] message;
		int[] gen;
		int[] app_message;
		int[] rem;
		int[] trans_message;
		int message_bits,gen_bits,total_bits;

		System.out.println("Enter no bits in mwssage:");
		message_bits=sc.nextInt();

		message=new int [message_bits];
		System.out.println("\nEnter message bits:");
		for(int i=0; i<message_bits;i++)
			message[i]= sc.nextInt();

		System.out.println("\nEnter number of bits in gen:");
		gen_bits= sc.nextInt();

		gen=new int[gen_bits];
		System.out.println("\nEnter gen bits:");
		for(int i=0;i<gen_bits;i++)
			gen[i]= sc.nextInt();

		total_bits=message_bits+gen_bits-1;

		app_message=new int[total_bits];
		rem=new int[total_bits];
		trans_message=new int[total_bits];

		for(int i=0;i<message.length;i++)
			app_message[i]=message[i];

		System.out.println("\nMessage bits are:");
		for(int i=0;i<message_bits;i++)
			System.out.print("\t"+message[i]);

		System.out.println("\nGenerators bits are:");
		for(int i=0;i<gen_bits;i++)
			System.out.print("\t"+gen[i]);
```

```
            System.out.println("\nAppended message is:");
            for(int i=0;i<app_message.length;i++)
                  System.out.print("\t"+app_message[i]);

            for(int j=0;j<app_message.length;j++)
                  rem[j]=app_message[j];

            rem=computecrc(app_message,gen,rem);

            for(int i=0;i<app_message.length;i++)
                  trans_message[i]=(app_message[i]^rem[i]);

            System.out.println("\nTransmitted message from the transmitter is:");
            for(int i=0;i<trans_message.length;i++)
                  System.out.print("\t"+trans_message[i]);

            System.out.println("\nEnter received message of"+total_bits+"bits at receiver end:");
            for(int i=0;i<trans_message.length;i++)
                  trans_message[i]= sc.nextInt();

            System.out.println("\nReceived message is:");
            for(int i=0;i<trans_message.length;i++)
                  System.out.print("\t"+trans_message[i]);

            for(int j=0;j<trans_message.length;j++)
                  rem[j]=trans_message[j];

            rem=computecrc(trans_message,gen,rem);
            for(int i=0;i<rem.length;i++)
            {
                  if(rem[i]!=0)
                  {
                        System.out.println("\nThere is error in the received message");
                        break;
                  }
                  if(i==rem.length-1)
                        System.out.println("\nThere is no erron in the received mesage!!");
            }
      }
      static int[] computecrc(int app_message[],int gen[], int rem[])
      {
            int current=0;
            while(true)
            {
                  for(int i=0;i<gen.length;i++)
                        rem[current+i]=(rem[current+i]^gen[i]);
                  while(rem[current]==0 && current!=rem.length-1)
```

```
                    current++;
                if((rem.length-current)<gen.length)
                    break;
            }
            return rem;
        }
}
```



**5. Develop a program to implement a sliding window protocol in the data link layer**

**6. Develop a program to find the shortest path between vertices using bellman-ford and path vector routing algorithm.**

## Steps:

***Input :*** Graph and a source vertex *src* .

***Output:*** Shortest distance to all vertices from *src*. If there is a negative weight cycle, then shortest distances are not calculated, negative weight cycle is reported.

**1.** This step initializes distances from source to all vertices as in finite and distance to source itself as 0. Create an array dist[] of size |V| with all values as infinite except dist[src] where src is source vertex.

**2.** This step calculates shortest distances. Do following |V|-1 times where |V| is the number of vertices in given graph.

**a)** Do following for each edge u-v.

If dist[v]>dist[u]+weigh to f edge uv, then update
list[v]. dist[v] = dist[u] + weight of edge uv

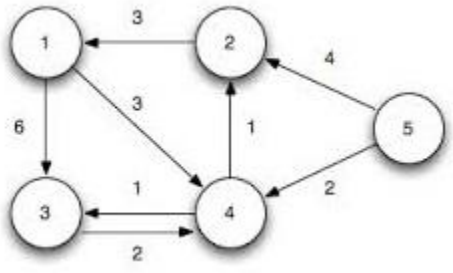
**3.** Thisstepreportsifthereisanegativeweightcycleingraph.Dofollowingfor each edge u-v.

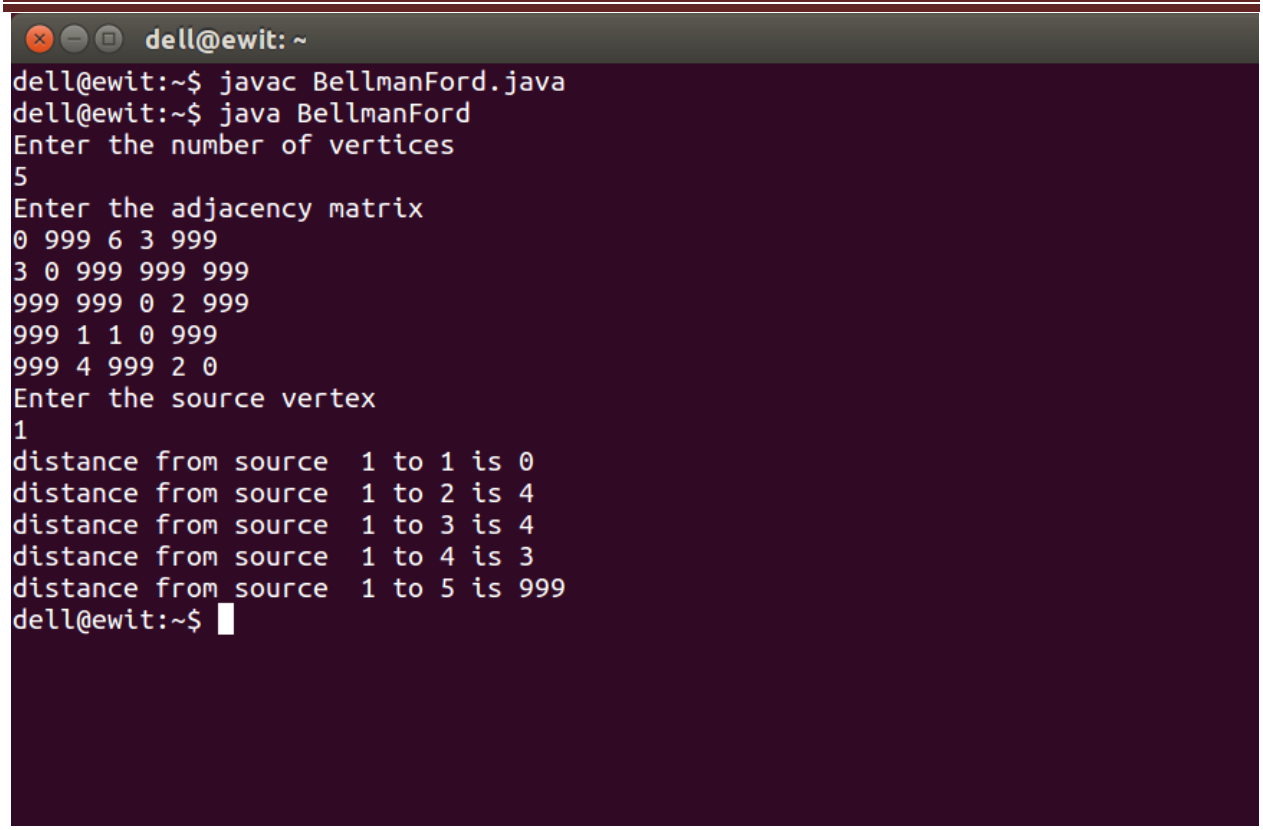
If dist[v] >dist[u] + weight of edge uv, then
"Graph contains negative weight cycle".

The idea of step 3 is, step 2 guarantees shortest distances if graph doesn't contain negative weight cycle. If we iterate through all edges one more time and get a shorter path for any vertex, then there is a negative weight cycle.

```
import java.util.Scanner;
public class BellmanFord
{
	private int D[];
	private int NoV;
	public static final int MAX_VALUE = 999;

	public BellmanFord(int NoV)
	{
		this.NoV = NoV;
		D = new int[NoV + 1];
	}
	public void BellmanFordEvaluation(int source, int A[][])
	{
		for (int node = 1; node <= NoV; node++)
		{
			D[node] = MAX_VALUE;
		}
		D[source] = 0;
		for (int node = 1; node <= NoV - 1; node++)
		{
			for (int i = 1; i <= NoV; i++)
			{
				for (int j = 1; j <= NoV; j++)
				{
					if (A[i][j] != MAX_VALUE)
					{
						if (D[j] > D[i] + A[i][j])
							D[j] = D[i]+ A[i][j];
					}
				}
			}
		}
		for (int i = 1; i <= NoV; i++)
		{
			for (int j = 1; j <= NoV; j++)
			{
				if (A[i][j] != MAX_VALUE)
				{
					if (D[j] > D[i]+ A[i][j])
					{
						System.out.println("The Graph contains negative
egde cycle");
						return;
					}
				}
```

```
                        }
                }
                for (int vertex = 1; vertex <= NoV; vertex++)
                {
                        System.out.println("distance of source  " + source + " to "+ vertex + " is "
+ D[vertex]);
                }
        }

        public static void main(String... arg)
        {
                int NoV = 0;
                int source;
                Scanner scanner = new Scanner(System.in);

                System.out.println("Enter the number of vertices");
                NoV = scanner.nextInt();
                int A[][] = new int[NoV + 1][NoV + 1];

                System.out.println("Enter the adjacency matrix");
                for (int i = 1; i <= NoV; i++)
                {
                        for (int j = 1; j <= NoV; j++)
                        {
                                A[i][j] = scanner.nextInt();
                        }
                }

                System.out.println("Enter the source vertex");
                source = scanner.nextInt();
                BellmanFord bellmanford = new BellmanFord(NoV);
                bellmanford.BellmanFordEvaluation(source, A);
                scanner.close();
        }
}
```




dell@ewit: ~
dell@ewit:~$ javac BellmanFord.java
dell@ewit:~$ java BellmanFord
Enter the number of vertices
5
Enter the adjacency matrix
0 999 6 3 999
3 0 999 999 999
999 999 0 2 999
999 1 1 0 999
999 4 999 2 0
Enter the source vertex
1
distance from source 1 to 1 is 0
distance from source 1 to 2 is 4
distance from source 1 to 3 is 4
distance from source 1 to 4 is 3
distance from source 1 to 5 is 999
dell@ewit:~$

## 7. Using TCP/IP sockets, write a client – server program to make the client send the file name and to make the server send back the contents of the requested file if present.

### *Methods and description*

α. **Public Server Socket(int port)throws IOException:** Attempts to create a server socket bound to the specified port. An exception occurs if the port is already bound by another application.

β. **public Socket accept() throws IOException:** Waits for an incoming client. This method blocks until either a client connects to the server on the specified port or the socket times out, assuming that the time-out value has been set using the set So Timeout() method. Otherwise, this method blocks indefinitely.

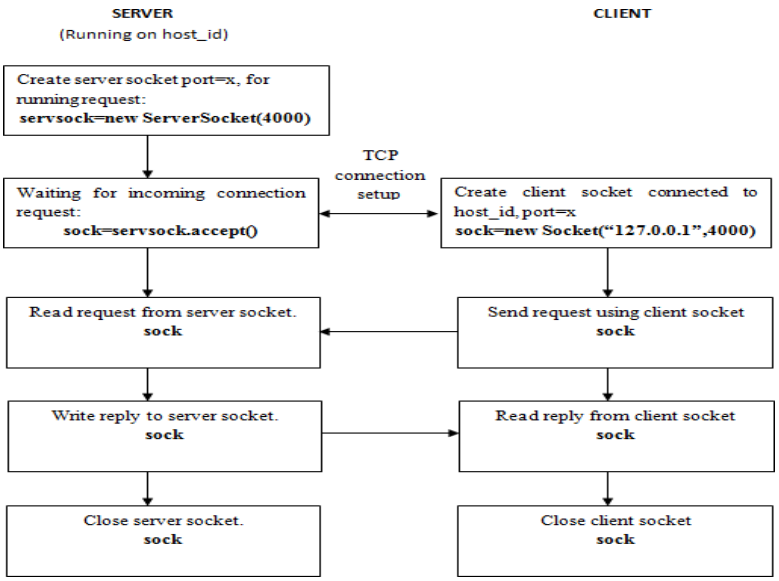
χ. **public Socket(Inet Address host, int port) throws IOException:** This method attempts to connect to the specified server at the specified port. If this constructor does not throw an exception, the connection is successful and the client is connected to the server.

δ. **Public Inet Address get Inet Address()**: This method returns the address of the other computer that this socket is connected to.

ε. **Public int getPort()**:Returns the port the socket is bound to on there mote machine.

ϕ. **Public InputStreamgetInputStream() throws IOException**: Returns the input stream of the socket. The input stream is connected to the output stream of the remote socket.

γ. **Public OutputStreamgetOutputStream() throws IOException**: Returns the output stream of the socket. The output stream is connected to the input stream of the remote socket.




**// TCP Server**

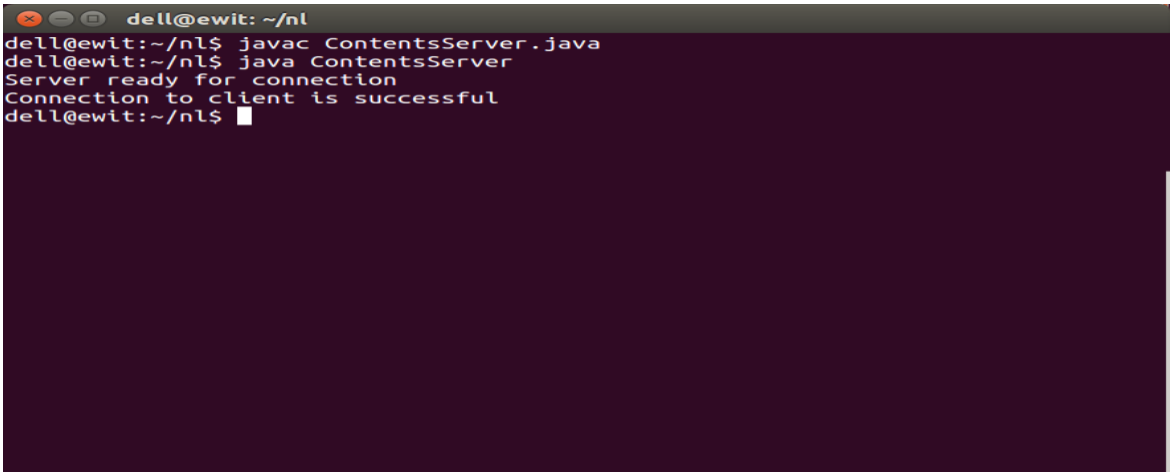
```
import java.net.*;
import java.io.*;
public class TCPS
{
public static void main(String[] args) throws Exception
{
ServerSocket sersock=new ServerSocket(4000);
System.out.println("Server ready for connection");
Socket sock=sersock.accept();
System.out.println("Connection Is successful and waiting for chatting");
InputStream istream=sock.getInputStream();
BufferedReader fileRead=new BufferedReader(new InputStreamReader(istream));
String fname=fileRead.readLine();
BufferedReader ContentRead=new BufferedReader(new FileReader(fname));
OutputStream ostream=sock.getOutputStream();
PrintWriter pwrite=new PrintWriter(ostream,true);
String str;
while((str=ContentRead.readLine())!=null){
pwrite.println(str);
}
sock.close();
sersock.close();
pwrite.close();
fileRead.close();
ContentRead.close();
}
}
```

**//TCP Client**

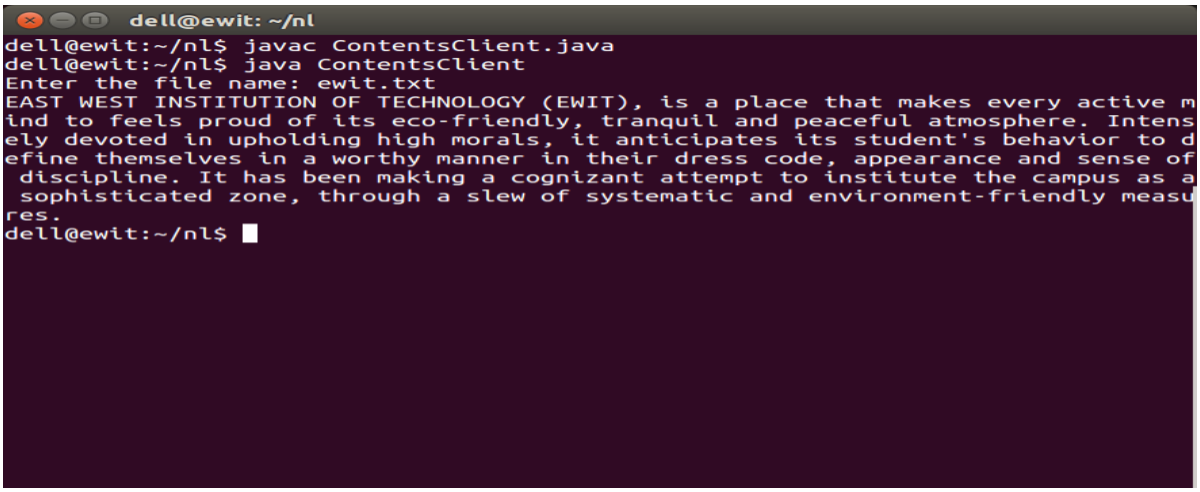
```
import java.net.*;
import java.io.*;
public class TCPC
{
public static void main(String[] args) throws Exception
{
Socket sock=new Socket("127.0.01",4000);
System.out.println("Enter the filename");
BufferedReader keyRead=new BufferedReader(new InputStreamReader(System.in));
String fname=keyRead.readLine();
OutputStream ostream=sock.getOutputStream();
PrintWriter pwrite=new PrintWriter(ostream,true);
```

```
pwrite.println(fname);
InputStream istream=sock.getInputStream();
BufferedReader socketRead=new BufferedReader(new InputStreamReader(istream));
String str;
while((str=socketRead.readLine())!=null)
{
System.out.println(str);
}
pwrite.close();
socketRead.close();
keyRead.close();
}
}
```

**TCPServer:**




**TCPClient:**



## 8. Develop a program on datagram socket for client/server to display the messages on client side, typed at the server side.

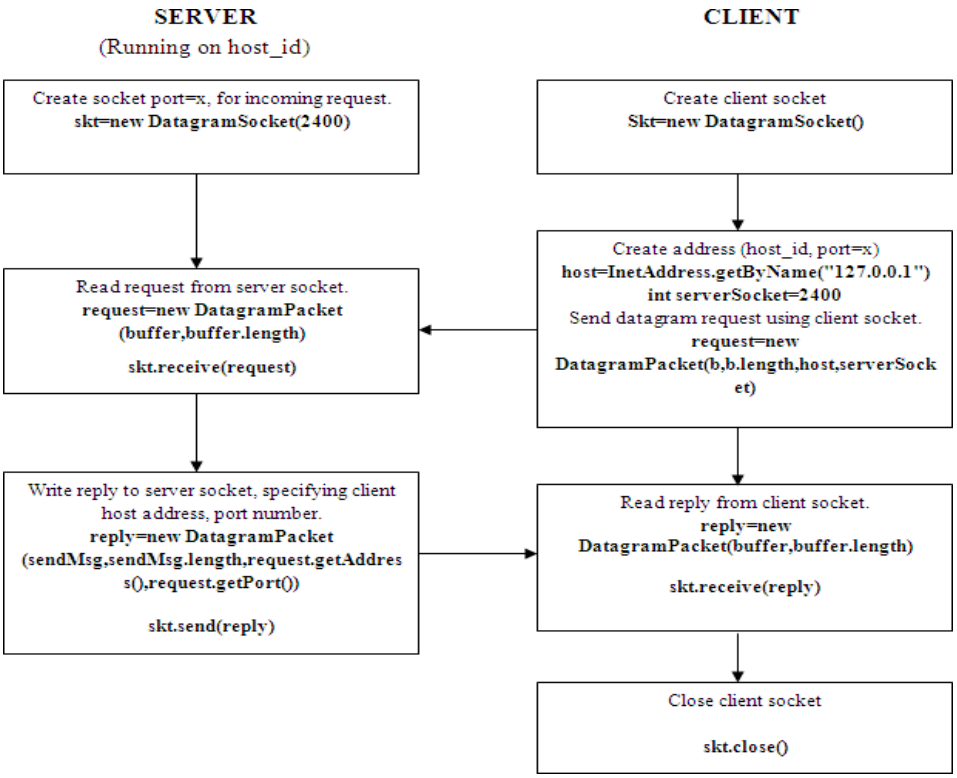



Fig 8.1:UDP client/server communication flow.

### *Methods and description*

α. **DatagramSocket(intport) throws SocketEeption:** it creates a datagram socket and binds it with the given Port Number.

β. **DatagramPacket(byte[]buffer,intlength):** it creates a datagram packet. This constructor is used to receive the packets.

χ. **DatagramPacket(byte[] buffer, int length, InetAddress address, int port):**it creates a datagram packet. This constructor is used to send the packets.

***UDP Server Source code:***

```
import java.io.*;
importjava.net.*;
importjava.util.*;
 public class udps
{
publicstaticvoid main(String[]args)
{
DatagramSocketskt=null;
Scannersc=newScanner(System.in); try
{
skt=newDatagramSocket(2400
); byte[] buffer=new
byte[1000]; while(true)
{
DatagramPacketrequest=newDatagramPacket(buffer,buffer.length);
skt.receive(request);
Stringmessage=sc.nextLine();
byte[]sendMsg=message.getByte
s(); DatagramPacket reply=new
DatagramPacket(sendMsg,sendMsg.length,request.getAddress(),request.getPort());
skt.send(reply);
}
}
catch(Exceptionex)
{
}
}
}
```

***UDP Client Source code:***

```
import java.io.*;
importjava.net.*;
publicclassudpc
{
        publicstaticvoid main(String[]args)
        {
DatagramSocketskt;
try
    {
        skt=new DatagramSocket();
          Stringmsg="textmessage";
        byte[]b=msg.getBytes();
        InetAddresshost=InetAddress.getByName("127.0.0.1");
        intserverSocket=2400;
        DatagramPacketrequest=newDatagramPacket(b,b.length,host,serverSocket);
        skt.send(request);
        byte[]buffer=newbyte[1000];
        DatagramPacketreply=newDatagramPacket(buffer,buffer.length);
skt.receive(reply);
System.out.println("clientrecieved:"+newString(reply.getData()));
skt.close();
    }
catch(Exceptionex){}
}
}
```

Fig 8.2: UDP server

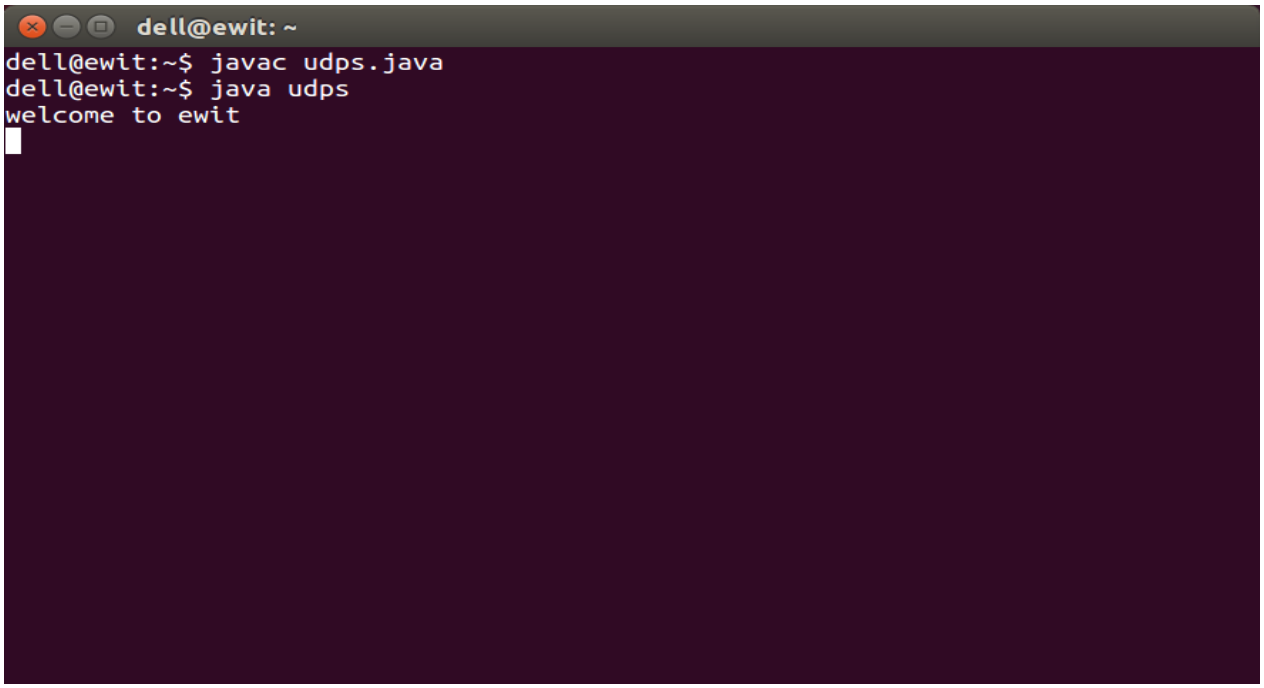
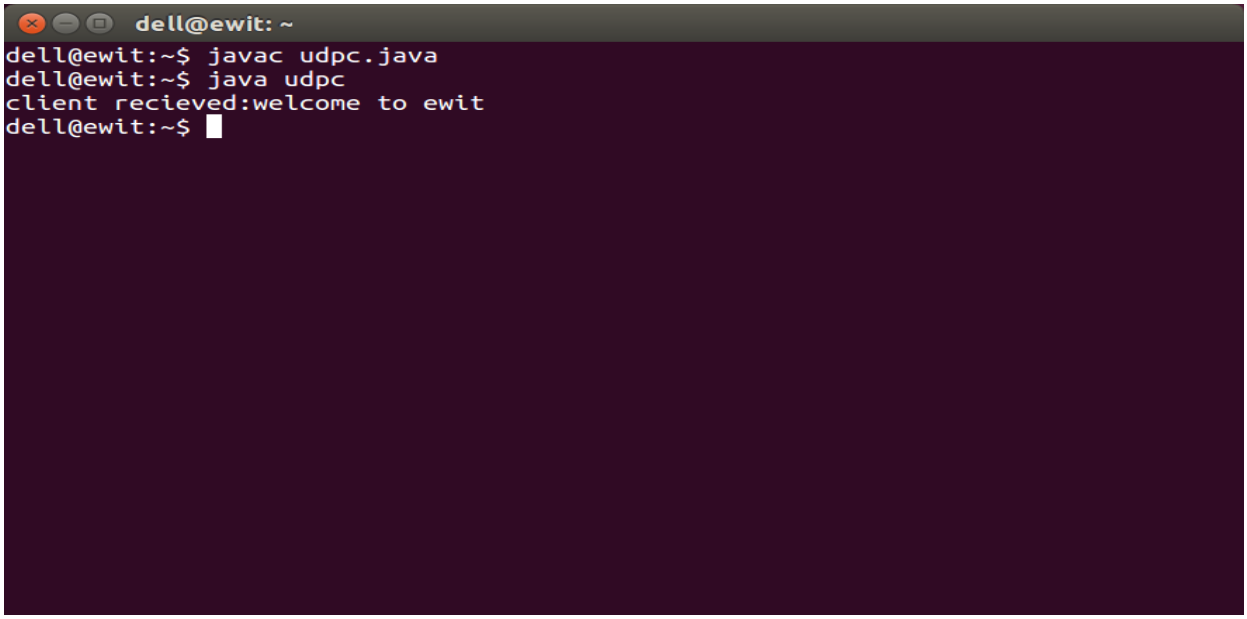



Fig 8.2: UDP Client



**9. Write a program for simple RSA algorithm to encrypt and decrypt the data.**

RSA is an algorithm used by modern computers to encrypt and decrypt messages. It is an asymmetric cryptographic algorithm. Asymmetric means that there are two different keys (one is public key and another is private key). This is also called public key cryptography, because one of the m can be given to everyone. The other key must be kept private. It is based on the fact that finding the factors of an integer is hard (the factoring problem). RSA stands for Ron Rivest, Adi Shamir and Leonard Adleman, who first publicly described it in 1978.

**Following are the steps of RSA algorithm.**

**Key Generation**

1. Generate two large prime numbers **p** and **q**, such that **p!=q**.
2. Let **n=p*q.**
3. Let **t=(p-1)*(q-1)**
4. Choose a small numbere ,co-prime tot, with
   **GCD(t,e)=1**and **1<e<t.**
5. Find d, such that
   **d*emodt=1**. Publish **e** and
   **n** as **public key**. Keep **d**
   and **t** as **secret key**.

**Encryption**

**Cipher=(Message)$^{e}$modn**

**Decryption**

**Message=(Cipher)$^{d}$modn**

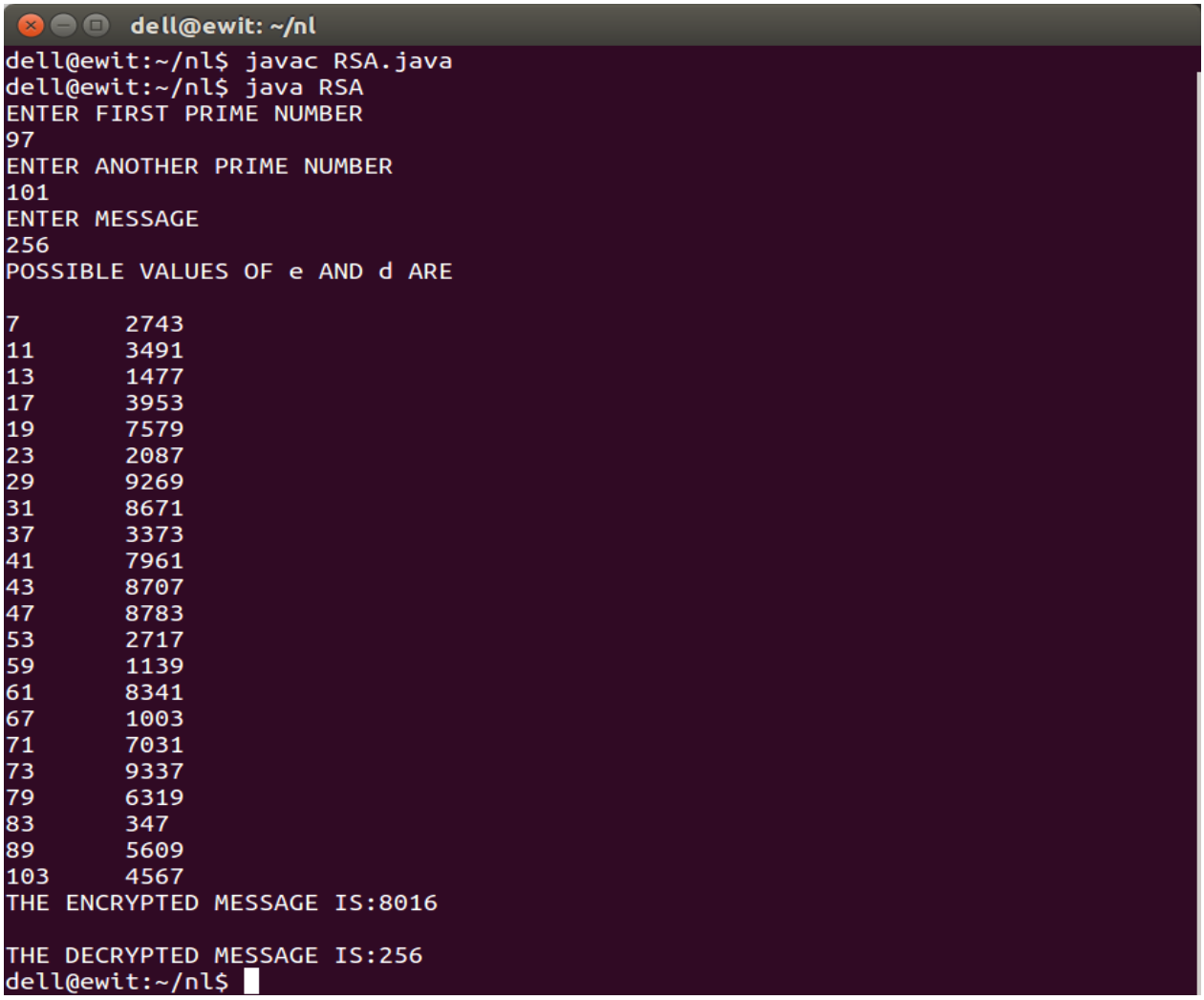
importjava.util.Scanner;

```
public class RSA
{
        publicstaticintp,q,n,t,flag,msg,m,temp;
        public staticint e[]=new int[100];
         public staticint d[]=new int[100];
        public static int prime( intpr)
        {
                inti;

                Doublea=(Math.sqrt(p
                r)); m=a.intValue();

                for(i=2;i<=m;i++)
                {
                        if(pr%i==0)
                                return0;
                }

                return1;

        }
        publicstaticvoidce()
        {
                intk=0;

                for(inti=2;i<t;i++)
                {
                        if(t%i==0)
                                conti

                        nue;

                        flag=prime(i

                        );

                        if(flag==1&&i!=p&&i!=q)
                        {

                                e[k]=i;
                                flag=cd(e[k]);
                                if(flag>0)
                                {
                                        d[k]=fl
                                        ag;
                                        k++;
                                }
```

```
                        if(k==99)
                    break;
                    }
            }
        }

        publicstaticintcd( intx)
        {
            intk=1;

            while(true)
            {
                k=k+t;

                if(k%x==0)
                    return(k/x);
            }

        }

        publicstaticvoid encrypt()
        {

            intpt,ct,key=e[0],k;

            pt=msg;
            k=1;

            for(intj=0;j<key;j++)
            {
                k=k*pt;
                k=k%n;
            }

            ct=k;
            temp
            =ct;
            System.out.println("\nTHEENCRYPTEDMESSAGEIS:"+ct);

        }

        Publicstatic void decrypt()
        {
            intpt,ct,key=d[0],k;
            ct=temp;
k=1;
```

```
            for(intj=0;j<key;j++)
            {
                    k=k
                    *ct;
                    k=k
                    %n;
            }

            pt=k;

           System.out.println("\nTHEDECRYPTEDMESSAGEIS:"+pt);
    }

    publicstaticvoidmain(Stringargs[])
    {
            Scanner sc=new Scanner(System.in);
            System.out.println("ENTERFIRSTPRIMENUMBER"
            ); p=sc.nextInt();
            flag=prime(p);
            if(flag==0)
            {
                    System.out.println("WRONGINPU
                    T"); System.exit(1);
            }
            System.out.println("ENTERANOTHERPRIMENUMBER");
            q=sc.nextInt();
            flag=prime(q);
            if(flag==0||p==q)
            {
                    System.out.println("WRONGINPU
                    T"); System.exit(1);
            }

            System.out.println("ENTERMESSAG
            E"); msg=sc.nextInt();
            n=p*q;
            t=(p-1)*(q-1);

            ce();

            System.out.println("POSSIBLEVALUESOFeANDdARE");
            for (int i=0;i< m-1;i++)
                    System.out.printf("\n%d\t%d",e[i],d[i]);

            encrypt();
            decrypt();
    }
```

}




**10. Develop a program for congestion control using leaky bucket algorithm**.

To understand this concept first we have to know little about traffic shaping.

**Traffic Shaping:** This is a mechanism to control the amount and the rate of the traffic sent to the network.

Two techniques can shape traffic:

1. Leaky Bucket
2. Token Bucket.

Suppose we have a bucket in which we are pouring water in a random order but we have to get water in a fixed rate, for this we will make a hole at the bottom of the bucket. It will ensure that watercomingoutisinsomefixedrate.Andalsoifbucketisfullwewillstoppouringinit. The input rate can vary, but the output rate remains constant. Similarly, in networking, a technique called leaky bucket can smooth out bursty traffic. Bursty chunks are stored in the bucket and sent out at an average rate.

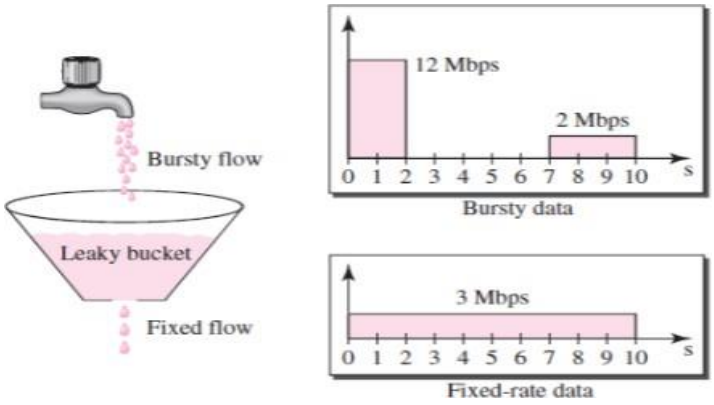



Fig 10.1: Leaky bucket scenario

In the above figure, we assume that the network has committed a bandwidth of 3 Mbps for a host. The use of the leaky bucket shapes the input traffic to make it conform tot his commitment. In Figure the host sends a burst of data at a rate of 12 Mbps for 2 s, for a total of 24 M bits of data. The host is silent for 5 s and then sends data at a rate of 2 Mbps for 3 s, for a total of 6 M bits of data. In all, the host has sent 30 M bits of data in 10 s. The leaky bucket smoothes the traffic by sending out data at a rate of 3Mbps during the same10s.Without the leaky bucket, the beginning burst may have hurt the network by consuming more bandwidth than is set aside for this host. We can also see that the leaky bucket may prevent congestion.

```
import java.util.Random;
import java.util.Scanner;
public class Leaky
{
public static int bsize=0,packet,tgen,j=1;
public static String stop;
public static final int bmax=1024;
public static final int orate=100;
public static final int delay=1500;
public static Random r=new Random();
public static Random t=new Random();
public class generating extends Thread
{
public void run()
{
while(stop==null)
{
tgen=t.nextInt(3000);
packet=r.nextInt(512);
if(bsize+packet<bmax)
{
bsize=bsize+packet;
System.out.printf("%13d%10d%15d%20d\n",j++,packet,bsize,bmax-bsize);
}
else
System.out.println("Bucket OverFlow, "+packet+" size of packet discarded");
try{Thread.sleep(tgen);}catch(Exception e){};
}
}
}
public class leaking extends Thread
{
public void run()
{
while(true)
{
if(bsize>0 && bsize-orate>0) //output packet rate is 100bytes
{
bsize=bsize-orate;
System.out.printf("%38d%20d%15d\n",bsize,(bmax-bsize),orate);
}
```

```
else
{
System.out.printf("%38d%20d%15d\n",0,bmax,bsize);
bsize=0;
if(stop!=null)
return;}
try{Thread.sleep(delay);}catch(Exception e){};
}
}
}
//@SuppressWarnings("resource")
public static void main(String[] args)
{
Leaky le=new Leaky();
Scanner in=new Scanner(System.in);
generating g=le.new generating();
leaking l=le.new leaking();
System.out.println("Started");
System.out.println("Output Rate is:"+orate+"\nAnd it is flowing at interval:"+
((float)delay/1000)+"sec");
System.out.println("Enter any key to stop input");
System.out.printf("Packet number | Input Packet | Bucket filled | Remaining space|Output rate");
System.out.println();
g.start();
try{Thread.sleep(10);}catch(Exception e){};
l.start();
stop=in.next();
}
}
```

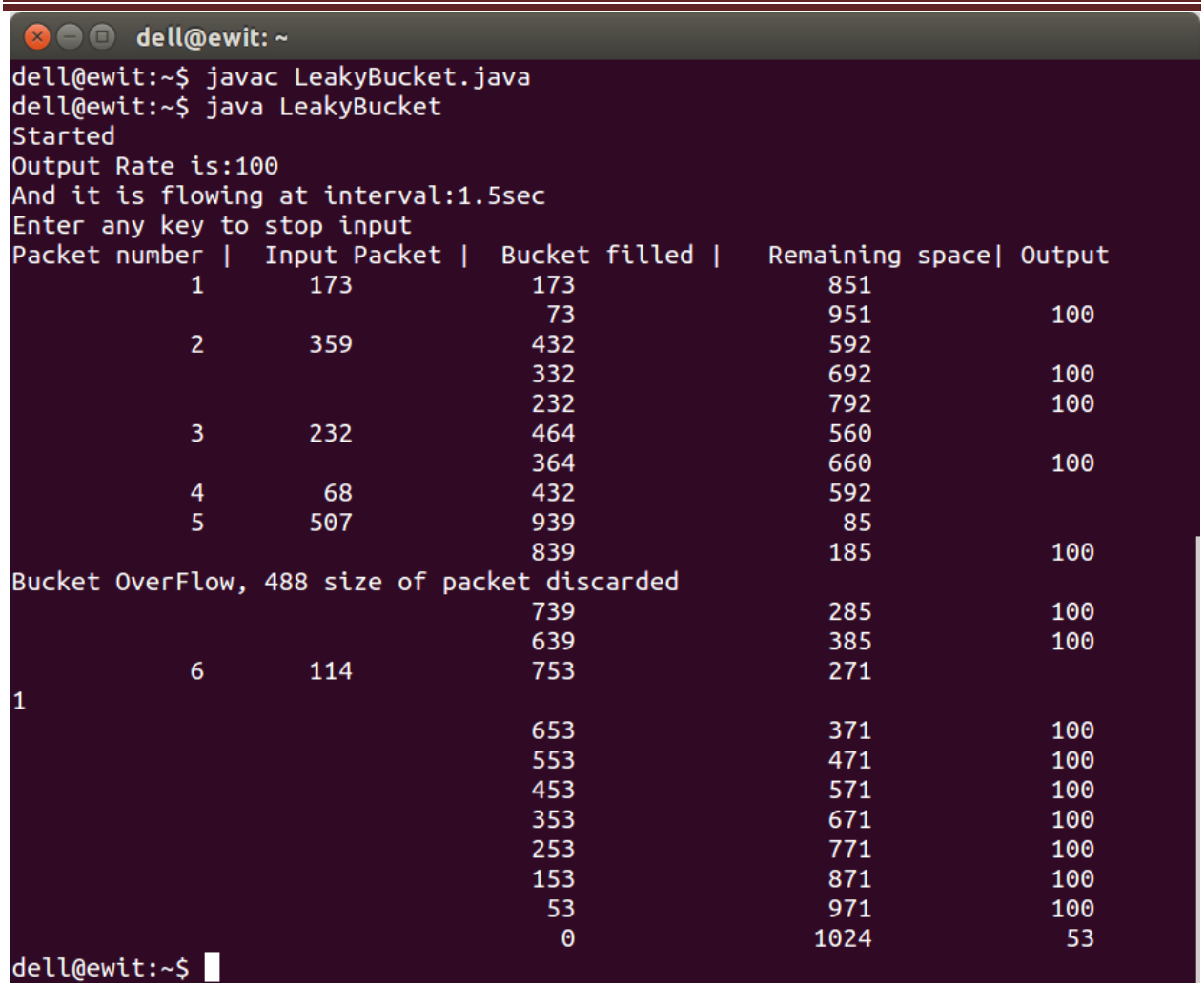



Fig 10.2: Congestion control using Leaky bucket algorithm.

# VIVA QUESTION AND ANSWER

**1) What is a Link?**
A link refers to the connectivity between two devices. It includes the type of cables and protocols used in order for one device to be able to communicate with the other.

**2) What are the layers of the OSI reference model?**
There are 7 OSI layers: Physical Layer, Data Link Layer, Network Layer, Transport Layer, Session Layer, Presentation Layer and Application Layer.

**3) What is backbone network?**
A backbone network is a centralized infrastructure that is designed to distribute different routes and data to various networks. It also handles management of bandwidth and various channels.

**4) What is a LAN?**
LAN is short for Local Area Network. It refers to the connection between computers and other network devices that are located within a small physical location.

**5) What is a node?**
A node refers to a point or joint where a connection takes place. It can be computer or device that is part of a network. Two or more nodes are needed in order to form a network connection.

**6) What are routers?**
Routers can connect two or more network segments. These are intelligent network devices that store information in its routing table such as paths, hops and bottlenecks. With this info, they are able to determine the best path for data transfer. Routers operate at the OSI Network Layer.

**7) What is point to point link?**
It refers to a direct connection between two computers on a network. A point to point connection does not need any other network devices other than connecting a cable to the NIC cards of both computers.

**8) What is anonymous FTP?**
Anonymous FTP is a way of granting user access to files in public servers. Users that are allowed access to data in these servers do not need to identify themselves, but instead log in as an anonymous guest.

**9) What is subnet mask?**
A subnet mask is combined with an IP address in order to identify two parts: the extended network address and the host address. Like an IP address, a subnet mask is made up of 32 bits.

**10) What is the maximum length allowed for a UTP cable?**
A single segment of UTP cable has an allowable length of 90 to 100 meters. This limitation can be overcome by using repeaters and switches.

**11) What is data encapsulation?**
Data encapsulation is the process of breaking down information into smaller manageable chunks before it is transmitted across the network. It is also in this process that the source and destination addresses are attached into the headers, along with parity checks.

**12) Describe Network Topology**
Network Topology refers to the layout of a computer network. It shows how devices and cables are physically laid out, as well as how they connect to one another.

**13) What is VPN?**
VPN means Virtual Private Network, a technology that allows a secure tunnel to be created across a network such as the Internet. For example, VPNs allow you to establish a secure dialup connection to a remote server.

**14) Briefly describe NAT.**

NAT is Network Address Translation. This is a protocol that provides a way for multiple computers on a common network to share single connection to the Internet.

**15) What is the job of the Network Layer under the OSI reference model?**

The Network layer is responsible for data routing, packet switching and control of network congestion. Routers operate under this layer.

**16) How does a network topology affect your decision in setting up a network?**

Network topology dictates what media you must use to interconnect devices. It also serves as basis on what materials, connector and terminations that is applicable for the setup.

**17) What is RIP?**

RIP, short for Routing Information Protocol is used by routers to send data from one network to another. It efficiently manages routing data by broadcasting its routing table to all other routers within the network. It determines the network distance in units of hops.

**18) What are different ways of securing a computer network?**

There are several ways to do this. Install reliable and updated anti-virus program on all computers. Make sure firewalls are setup and configured properly. User authentication will also help a lot. All of these combined would make a highly secured network.

**19) What is NIC?**

NIC is short for Network Interface Card. This is a peripheral card that is attached to a PC in order to connect to a network. Every NIC has its own MAC address that identifies the PC on the network.

**20) What is WAN?**

WAN stands for Wide Area Network. It is an interconnection of computers and devices that are geographically dispersed. It connects networks that are located in different regions and countries.

**21) What is the importance of the OSI Physical Layer?**

The physical layer does the conversion from data bits to electrical signal, and vice versa. This is where network devices and cable types are considered and setup.

**22) How many layers are there under TCP/IP?**

There are four layers: the Network Layer, Internet Layer, Transport Layer and Application Layer.

**23) What are proxy servers and how do they protect computer networks?**

Proxy servers primarily prevent external users who identifying the IP addresses of an internal network. Without knowledge of the correct IP address, even the physical location of the network cannot be identified. Proxy servers can make a network virtually invisible to external users.

**24) What is the function of the OSI Session Layer?**

This layer provides the protocols and means for two devices on the network to communicate with each other by holding a session. This includes setting up the session, managing information exchange during the session, and tear-down process upon termination of the session.

**25) What is the importance of implementing a Fault Tolerance System? Are there limitations?**

A fault tolerance system ensures continuous data availability. This is done by eliminating a single point of failure. However, this type of system would not be able to protect data in some cases, such as in accidental deletions.

**26) What does 10Base-T mean?**

The 10 refers to the data transfer rate, in this case is 10Mbps. The word Base refers to base band, as oppose to broad band. T means twisted pair, which is the cable used for that network.

**27) What is a private IP address?**

Private IP addresses are assigned for use on intranets. These addresses are used for internal networks and are not routable on external public networks. These ensures that no conflicts are

present among internal networks while at the same time the same range of private IP addresses are reusable for multiple intranets since they do not "see" each other.

**28) What is NOS?**

NOS, or Network Operating System, is specialized software whose main task is to provide network connectivity to a computer in order for it to be able to communicate with other computers and connected devices.

**29) What is DoS?**

DoS, or Denial-of-Service attack, is an attempt to prevent users from being able to access the internet or any other network services. Such attacks may come in different forms and are done by a group of perpetuators. One common method of doing this is to overload the system server so it cannot anymore process legitimate traffic and will be forced to reset.

**30) What is OSI and what role does it play in computer networks?**

OSI (Open Systems Interconnect) serves as a reference model for data communication. It is made up of 7 layers, with each layer defining a particular aspect on how network devices connect and communicate with one another. One layer may deal with the physical media used, while another layer dictates how data is actually transmitted across the network.

**31) What is the purpose of cables being shielded and having twisted pairs?**

The main purpose of this is to prevent crosstalk. Crosstalks are electromagnetic interferences or noise that can affect data being transmitted across cables.

**32) What is the advantage of address sharing?**

By using address translation instead of routing, address sharing provides an inherent security benefit. That's because host PCs on the Internet can only see the public IP address of the external interface on the computer that provides address translation and not the private IP addresses on the internal network.

**33) What are MAC addresses?**

MAC, or Media Access Control, uniquely identifies a device on the network. It is also known as physical address or Ethernet address. A MAC address is made up of 6-byte parts.

**34) What is the equivalent layer or layers of the TCP/IP Application layer in terms of OSI reference model?**

The TCP/IP Application layer actually has three counterparts on the OSI model: the Session layer, Presentation Layer and Application Layer.

**35) How can you identify the IP class of a given IP address?**

By looking at the first octet of any given IP address, you can identify whether it's Class A, B or C. If the first octet begins with a 0 bit, that address is Class A. If it begins with bits 10 then that address is a Class B address. If it begins with 110, then it's a Class C network.

**36) What is the main purpose of OSPF?**

OSPF, or Open Shortest Path First, is a link-state routing protocol that uses routing tables to determine the best possible path for data exchange.

**37) What are firewalls?**

Firewalls serve to protect an internal network from external attacks. These external threats can be hackers who want to steal data or computer viruses that can wipe out data in an instant. It also prevents other users from external networks from gaining access to the private network.

**38) Describe star topology**

Star topology consists of a central hub that connects to nodes. This is one of the easiest to setup and maintain.

**39) What are gateways?**

Gateways provide connectivity between two or more network segments. It is usually a computer that runs the gateway software and provides translation services. This translation is a key in allowing different systems to communicate on the network.

**40) What is the disadvantage of a star topology?**

One major disadvantage of star topology is that once the central hub or switch get damaged, the entire network becomes unusable.

**41) What is SLIP?**

SLIP, or Serial Line Interface Protocol, is actually an old protocol developed during the early UNIX days. This is one of the protocols that are used for remote access.

**42) Give some examples of private network addresses.**

10.0.0.0 with a subnet mask of 255.0.0.0
172.16.0.0 with subnet mask of 255.240.0.0
192.168.0.0 with subnet mask of 255.255.0.0

**43) What is tracert?**

Tracert is a Windows utility program that can used to trace the route taken by data from the router to the destination network. It also shows the number of hops taken during the entire transmission route.

**44) What are the functions of a network administrator?**

A network administrator has many responsibilities that can be summarize into 3 key functions: installation of a network, configuration of network settings, and maintenance/troubleshooting of networks.

**45) Describe at one disadvantage of a peer to peer network.**

When you are accessing the resources that are shared by one of the workstations on the network, that workstation takes a performance hit.

**46) What is Hybrid Network?**

A hybrid network is a network setup that makes use of both client-server and peer-to-peer architecture.

**47) What is DHCP?**

DHCP is short for Dynamic Host Configuration Protocol. Its main task is to automatically assign an IP address to devices across the network. It first checks for the next available address not yet taken by any device, then assigns this to a network device.

**48) What is the main job of the ARP?**

The main task of ARP or Address Resolution Protocol is to map a known IP address to a MAC layer address.

**49) What is TCP/IP?**

TCP/IP is short for Transmission Control Protocol / Internet Protocol. This is a set of protocol layers that is designed to make data exchange possible on different types of computer networks, also known as heterogeneous network.

**50) How can you manage a network using a router?**

Routers have built in console that lets you configure different settings, like security and data logging. You can assign restrictions to computers, such as what resources it is allowed access, or what particular time of the day they can browse the internet. You can even put restrictions on what websites are not viewable across the entire network.

**51) What protocol can be applied when you want to transfer files between different platforms, such between UNIX systems and Windows servers?**

Use FTP (File Transfer Protocol) for file transfers between such different servers. This is possible because FTP is platform independent.

**52) What is the use of a default gateway?**Default gateways provide means for the local networks to connect to the external network. The default gateway for connecting to the external network is usually the address of the external router port.

**53) One way of securing a network is through the use of passwords. What can be considered as good passwords?**

Good passwords are made up of not just letters, but by combining letters and numbers. A password that combines uppercase and lowercase letters is favorable than one that uses all upper case or all lower case letters. Passwords must be not words that can easily be guessed by hackers, such as dates, names, favorites, etc. Longer passwords are also better than short ones.

**54) What is the proper termination rate for UTP cables?**

The proper termination for unshielded twisted pair network cable is 100 ohms.

**55) What is netstat?**

Netstat is a command line utility program. It provides useful information about the current TCP/IP settings of a connection.

**56) What is the number of network IDs in a Class C network?**

For a Class C network, the number of usable Network ID bits is 21. The number of possible network IDs is 2 raised to 21 or 2,097,152. The number of host IDs per network ID is 2 raised to 8 minus 2, or 254.

**57) What happens when you use cables longer than the prescribed length?**

Cables that are too long would result in signal loss. This means that data transmission and reception would be affected, because the signal degrades over length.

**58) What common software problems can lead to network defects?**

Software related problems can be any or a combination of the following:

- client server problems
- application conflicts
- error in configuration
- protocol mismatch
- security issues
- user policy and rights issues

**59) What is ICMP?**

ICMP is Internet Control Message Protocol. It provides messaging and communication for protocols within the TCP/IP stack. This is also the protocol that manages error messages that are used by network tools such as PING.

**60) What is Ping?**

Ping is a utility program that allows you to check connectivity between network devices on the network. You can ping a device by using its IP address or device name, such as a computer name.

**61) What is peer to peer?**

Peer to peer are networks that does not reply on a server. All PCs on this network act as individual workstations.

**62) What is DNS?**

DNS is Domain Name System. The main function of this network service is to provide host names to TCP/IP address resolution.

**63) What advantages does fiber optics have over other media?**

One major advantage of fiber optics is that is it less susceptible to electrical interference. It also supports higher bandwidth, meaning more data can be transmitted and received. Signal degrading is also very minimal over long distances.

**64) What is the difference between a hub and a switch?**

A hub acts as a multiport repeater. However, as more and more devices connect to it, it would not be able to efficiently manage the volume of traffic that passes through it. A switch provides a better alternative that can improve the performance especially when high traffic volume is expected across all ports.

**65) What are the different network protocols that are supported by Windows RRAS services?**

There are three main network protocols supported: NetBEUI, TCP/IP, and IPX.

**66) What are the maximum networks and hosts in a class A, B and C network?**

For Class A, there are 126 possible networks and 16,777,214 hosts
For Class B, there are 16,384 possible networks and 65,534 hosts
For Class C, there are 2,097,152 possible networks and 254 hosts

**67) What is the standard color sequence of a straight-through cable?**

orange/white, orange, green/white, blue, blue/white, green, brown/white, brown.

**68) What protocols fall under the Application layer of the TCP/IP stack?**

The following are the protocols under TCP/IP Application layer: FTP, TFTP, Telnet and SMTP.

**69) You need to connect two computers for file sharing. Is it possible to do this without using a hub or router?**

Yes, you can connect two computers together using only one cable. A crossover type cable can be use in this scenario. In this setup, the data transmit pin of one cable is connected to the data receive pin of the other cable, and vice versa.

**70) What is ipconfig?**

Ipconfig is a utility program that is commonly used to identify the addresses information of a computer on a network. It can show the physical address as well as the IP address.

**71) What is the difference between a straight-through and crossover cable?**

A straight-through cable is used to connect computers to a switch, hub or router. A crossover cable is used to connect two similar devices together, such as a PC to PC or Hub to hub.

**72) What is client/server?**

Client/server is a type of network wherein one or more computers act as servers. Servers provide a centralized repository of resources such as printers and files. Clients refers to workstation that access the server.

**73) Describe networking.**

Networking refers to the inter connection between computers and peripherals for data communication.

Networking can be done using wired cabling or through wireless link.

**74) When you move the NIC cards from one PC to another PC, does the MAC address gets transferred as well?**

Yes, that's because MAC addresses are hard-wired into the NIC circuitry, not the PC. This also means that a PC can have a different MAC address when the NIC card was replace by another one.

**75) Explain clustering support**

Clustering support refers to the ability of a network operating system to connect multiple servers in a fault-tolerant group. The main purpose of this is the in the event that one server fails, all processing will continue on with the next server in the cluster.

**76) In a network that contains two servers and twenty workstations, where is the best place to install an Anti-virus program?**

An anti-virus program must be installed on all servers and workstations to ensure protection. That's because individual users can access any workstation and introduce a computer virus when plugging in their removable hard drives or flash drives.

**77) Describe Ethernet**.

Ethernet is one of the popular networking technologies used these days. It was developed during the early 1970s and is based on specifications as stated in the IEEE. Ethernet is used in local area networks.

**78) What are some drawbacks of implementing a ring topology?**

In case one workstation on the network suffers a malfunction, it can bring down the entire network. Another drawback is that when there are adjustments and reconfigurations needed to be performed on a particular part of the network, the entire network has to be temporarily brought down as well.

**79) What is the difference between CSMA/CD and CSMA/CA?**

CSMA/CD, or Collision Detect, retransmits data frames whenever a collision occurred. CSMA/CA, or Collision Avoidance, will first broadcast intent to send prior to data transmission.

**80) What is SMTP?**

SMTP is short for Simple Mail Transfer Protocol. This protocol deals with all Internal mail, and provides the necessary mail delivery services on the TCP/IP protocol stack.

**81) What is multicast routing?**

Multicast routing is a targeted form of broadcasting that sends message to a selected group of user, instead of sending it to all users on a subnet.

**82) What is the importance of Encryption on a network?**

Encryption is the process of translating information into a code that is unreadable by the user. It is then translated back or decrypted back to its normal readable format using a secret key or password. Encryption help ensure that information that is intercepted halfway would remain unreadable because the user has to have the correct password or key for it.

**83) How are IP addresses arranged and displayed?**

IP addresses are displayed as a series of four decimal numbers that are separated by period or dots. Another term for this arrangement is the dotted decimal format. An example is 192.168.101.2

**84) Explain the importance of authentication.**

Authentication is the process of verifying a user's credentials before he can log into the network. It is normally performed using a username and password. This provides a secure means of limiting the access from unwanted intruders on the network.

**85) What do mean by tunnel mode?**

This is a mode of data exchange wherein two communicating computers do not use IPSec themselves. Instead, the gateway that is connecting their LANs to the transit network creates a virtual tunnel that uses the IPSec protocol to secure all communication that passes through it.

**86) What are the different technologies involved in establishing WAN links?**

Analog connections - using conventional telephone lines; Digital connections - using digitalgrade telephone lines; switched connections - using multiple sets of links between sender and receiver to move data.

**87) What is one advantage of mesh topology?**

In the event that one link fails, there will always be another available. Mesh topology is actually one of the most fault-tolerant network topology.

**88) When troubleshooting computer network problems, what common hardware-related problems can occur?**

A large percentage of a network is made up of hardware. Problems in these areas can range from malfunctioning hard drives, broken NICs and even hardware startups. Incorrectly hardware configuration is also one of those culprits to look into.

**89) What can be done to fix signal attenuation problems?**

A common way of dealing with such a problem is to use repeaters and hub, because it will help regenerate the signal and therefore prevent signal loss. Checking if cables are properly terminated is also a must.

**90) How does dynamic host configuration protocol aid in network administration?**

Instead of having to visit each client computer to configure a static IP address, the network administrator can apply dynamic host configuration protocol to create a pool of IP addresses known as scopes that can be dynamically assigned to clients.

**91) Explain profile in terms of networking concept?**

Profiles are the configuration settings made for each user. A profile may be created that puts a user in a group, for example.

**92) What is sneakernet?**

Sneakernet is believed to be the earliest form of networking wherein data is physically transported using removable media, such as disk, tapes.

**93) What is the role of IEEE in computer networking?**

IEEE, or the Institute of Electrical and Electronics Engineers, is an organization composed of engineers that issues and manages standards for electrical and electronic devices. This includes networking devices, network interfaces, cablings and connectors.

**94) What protocols fall under the TCP/IP Internet Layer?**

There are 4 protocols that are being managed by this layer. These are ICMP, IGMP, IP and ARP.

**95) When it comes to networking, what are rights?**

Rights refer to the authorized permission to perform specific actions on the network. Each user on the network can be assigned individual rights, depending on what must be allowed for that user.

**96) What is one basic requirement for establishing VLANs?**

A VLAN requires dedicated equipment on each end of the connection that allows messages entering the Internet to be encrypted, as well as for authenticating users.

**97) What is IPv6?**

IPv6 , or Internet Protocol version 6, was developed to replace IPv4. At present, IPv4 is being used to control internet traffic, butis expected to get saturated in the near future. IPv6 was designed to overcome this limitation.

**98) What is RSA algorithm?**

RSA is short for Rivest-Shamir-Adleman algorithm. It is the most commonly used public key encryption algorithm in use today.

**99) What is mesh topology?**

Mesh topology is a setup wherein each device is connected directly to every other device on the network. Consequently, it requires that each device have at least two network connections.