

**AIM**

To study and implement the functioning of sliding window protocol.

**Software required:**

- (i) JDK
- (ii) command prompt

**Program****Server programs**

```

import java.lang.* ;
import java.net.*;
import java.io.*;
class bits
{
    public static void main (String ar[])
    {
        try
        {
            BufferedInputStream in;
            ServerSocket ss = new ServerSocket(47);
            System.out.println("Waiting for connection . . .");
            Socket s = ss.accept();
        }
    }
}
  
```

```
System.out.println("Received request to send frames");
in = new BufferedInputStream(s.getInputStream());
DataOutputStream out = new
DataOutputStream(s.getOutputStream());
int p = in.read();
System.out.println("Sending ...");
for (int i = 1; i <= p; i++)
{
    System.out.println("Sending frame number " + i);
    out.write(i);
    out.flush();
    System.out.println("Waiting for acknowledgement ...");
    Thread.sleep(1000);
    int a = in.read();
    System.out.println("Received acknowledgement for
frame number: " + i + " as " + a);
}
out.flush();
in.close();
out.close();
s.close();
ss.close();
System.out.println("Quitting ...");
}
catch (Exception e)
{
}
```

## client program

```
import java.lang.System;  
import java.net.*;  
import java.io.*;  
import java.math.*;  
class bitc bitc  
{  
    public static void main (String args[])  
    {  
        try  
        {
```

```
        InetAddress addr = InetAddress.getByName ("LocalHost");  
        System.out.println (addr);  
        Socket s = new Socket (addr, 47);
```

```
DataOutputStream out = new DataOutputStream (s.getOutputStream ());
```

```
BufferedInputStream in = new BufferedInputStream  
(s.getInputStream ());
```

```
BufferedInputStream inn = new BufferedInputStream  
(s.getInputStream ());
```

```
BufferedReader b = new BufferedReader (new InputStreamReader  
(System.in));
```

Expt. No. \_\_\_\_\_

```

int flag = 0;
System.out.println("connected");
System.out.println("Enter the number of frame to
be requested to server:");

```

```
int c = Integer.parseInt(b.readLine());
```

```
out.write(c);
```

```
out.flush();
```

```
int l, j = 0;
```

```
while (j < c)
```

```
{
```

```
l = in.read();
```

```
System.out.println("Received frame number:" + i);
```

```
System.out.println("Sending acknowledgement for
frame number:" + i);
```

```
out.write(i);
```

```
out.flush();
```

```
j++;
```

```
{
```

```
out.flush();
```

```
in.close();
```

```
in.close();
```

```
out.close();
```

System.out.println ("Q waiting ...");

Thread.sleep(1000);

{

catch (Exception e)

{}

{}

Result: Successfully studied and implemented  
the following of sliding windows protocol.

**AIM**

Implementation of DHCP and Telnet protocol

configuration on Router

Router >

Router > enable

Router # configure terminal

Router # enable Password 123

Router (config) # host name cse

cse (config) # exit

cse # exit

Press RETURN to get started.

cse > enable

password : 123

cse # configure terminal

cse (config) # interface fast ethernet 0/0

cse (config-if) # ip address 192.168.10.100 255.

255.255.0

cse (config) # no shutdown

% LINK -5-CHANGED: Interface fast ethernet 0/0,  
changed state to up

? LINEPROTO -5-UPDOWN: Line protocol on interface  
fast ethernet 0/0, changed state to up

```
cse (config-if) # exit  
cse (config) # ip dhcp pool cse  
cse (dhcp-config) # default-router 192.168.  
10.100  
cse (dhcp-config) # network 192.168.10.0.  
255.255.255.0
```

### Result

Thus the DHCP experiment completed successfully.

AIM :

Implementation of telnet protocol

configuration on Router:

configuration of router is same as previous  
with following addition:

cse (config) # line vty 0 2

cse (config-line) # password CISCO

cse (config-line) # login

Result

Thus the telnet experiment completed successfully.

**AIM****Implementation of VLAN.****Configuration on switch**

Switch &gt; enable

Switch #

Switch # configure terminal

Enter configuration commands, one per line. End with CNTL/Z

Switch (config) # vlan 2

Switch (config - vlan) # name IT

Switch (config - vlan) # exit

Switch (config) # vlan 3

Switch (config - vlan) # name BCE

Switch (config - vlan) # exit

Switch (config) #

Switch (config) # interface fast ethernet 0/1

Switch (config - if) # switchport mode access

Switch (config - if) # switchport access vlan 2

Switch (config - if) # exit

Switch (config) #

Switch (config) # interface fast ethernet 0/2

Switch (config - if) # switchport mode access

Switch (config - if) # switchport access vlan 2

Switch (config - if) # exit

Switch (config) # interface fast ethernet 0/3

switch (config-if) # switch port mode access

switch (config-if) # switch port access vlan 3

switch (config-if) # exit

switch (config) # interface fast ethernet 0/4

switch (config-if) # switch port mode access

switch (config-if) # switch port access vlan 3

switch (config-if) # exit

switch (config) #

switch (config) # interface fast ethernet 0/5

switch (config-if) # switch port mode access

switch (config-if) # switch port access vlan 2

switch (config-if) # exit

switch (config) # interface fast ethernet 0/6

switch (config-if) # switch port mode access

switch (config-if) # switch port access vlan 3

switch (config-if) # exit

## configuration on Router

Router > enable

Router # configure terminal

Enter configuration commands one per line. End  
with CNTL/Z

Router (config) # interface fast ethernet 0/0

Router (config-if) # ip address 192.168.10.5 255.  
255.255.0

Router (config-if) # no shutdown

'.' LINK - 5 - CHANGED : Interface fast ethernet 0/0, changed state to up

Router (config-if) # no shutdown  
'.' LINK

Router '.' LINEPROTO - 5 - UPDOWN : Line protocol on interface fast ethernet 0/0, changed state to up

Router (config-if) # exit

Router (config) # interface fast ethernet 1/0

Router (config-if) # ip address 192.168.10.5 255.  
255.255.0

Router (config-if) # no shutdown

'.' LINK - 5 - CHANGED : Interface fast ethernet 1/0, changed state to up.

'.' LINEPROTO - 5 - UPDOWN : Line protocol on interface fast ethernet 1/0, changed state to up

Router (config-if) # exit

Router (config) # do shutdown showrun  
Translating "show run" --- domain server  
(255.255.255.255)

### Result

Thus the experiment was completed successfully.