Lab Assignment on Unit II: (Use JAVA/PYTHON)

Write a program to simulate Go back N and Selective Repeat Modes of Sliding Window
Protocol in peer to peer mode

Demonstrate the packets captured traces using Wireshark Packet Analyzer Tool for peer to peer mode

PROGRAMS

```
/*------------Client Program----------------*/

import java.util.*;
import java.net.*;
import java.io.*;


public class Client{ public static void main(String args[])

    throws Exception{

        BufferedReader br=new BufferedReader(new
        InputStreamReader(System.in));
        System.out.print("Enter the value of m :
        "); int m=Integer.parseInt(br.readLine());
        int x=(int)((Math.pow(2,m))-1);

        System.out.print("Enter no. of frames to be sent:");
        int
        count=Integer.parseInt(br.readLine());
        int data[]=new int[count]; int h=0;

        for(int i=0;i<count;i++)
        {
            System.out.print("Enter data for frame no " +h+ " => ");
            data[i]=Integer.parseInt(br.readLine());
            h=(h+1)%x;
        }

        Socket client=new Socket("localhost",6262);
        ObjectInputStream ois=new ObjectInputStream(client.getInputStream());
        ObjectOutputStream oos = new
ObjectOutputStream(client.getOutputStream());
        System.out.println("Connected with server.");
        boolean flag=false;
        GoBackNListener listener=new
        GoBackNListener(ois,x); listener=new
        GoBackNListener(ois,x); listener.t.start(); int
        strt=0; h=0;
```

```java
        oos.writeObject(x);

        do

        { int c=h;
            for(int i=h;i<count;i++)
            {
                System.out.print("|"+c+"|");
                c=(c+1)%x;
            }
            System.out.println();
            System.out.println();
            h=strt;

            for(int i=strt;i<x;i++)
            {
                System.out.println("Sending frame : " + h);
                h=(h+1)%x;
                System.out.println();
                oos.writeObject(i);
                oos.writeObject(data[i]);
                Thread.sleep(100);
            }

            listener.t.join(3500);

            if(listener.reply!=x-1)
            {
                System.out.println("Noreply from server in 3.5 seconds.
                Resending
data from frame no " + (listener.reply+1));
                System.out.println();
                strt=listener.reply+1;
                flag=false;
            }
            else
            {
                System.out.println("All elements sent successfully. Exiting");
                flag=true;
            }
        }while(!flag); oos.writeObject(-
        1);
    }
}


class GoBackNListener implements Runnable{
```

```java
    Thread t;
    ObjectInputStream ois;
    int reply,x;
    GoBackNListener(ObjectInputStream o,int i)
    {
        t=new
        Thread(this);
        ois=o; reply=-2;
        x=i;
    }

    @Override public
    void run() {
    try
    { int temp=0;
        while(reply!=-
        1)
        { reply=(Integer)ois.readObject();
            if(reply!=-1 && reply!=temp+1)
            reply=temp;
            if(reply!=-1)
            { temp=reply;
                System.out.println("Acknowledgement of frame no " + (reply%x) +
"recieved.");
                System.out.println();
            }
        }
        reply=temp;
    }
    catch(Exception e)
    {
        System.out.println("Exception => " + e);
    }
    }
}


/*-----------------OUTPUT--------------------------


C:\Users\dell\Java\CN\GoBackN>javac Client.java

C:\Users\dell\Java\CN\GoBackN>java Client
Enter the value of m : 7
Enter no. of frames to be sent:5
Enter data for frame no 0 => 1
Enter data for frame no 1 => 2
```

```
Enter data for frame no 2 => 3
Enter data for frame no 3 => 5
Enter data for frame no 4 => 5
Connected with server.
|0||1||2||3||4|
Sending frame : 0

Acknowledgement of frame no 0recieved.

Sending frame : 1

Sending frame : 2

Sending frame : 3

Sending frame : 4

Sending frame : 5

Exceptioninthread"main"java.lang.ArrayIndexOutOfBoundsException:Index5out of
bounds for length 5 at Client.main(Client.java:58)

*/
```

/*----------Server Side-------------*/

```java
import java.net.*;
import java.io.*;
import java.util.*;


public class Server
{ public static void main(String args[]) throws Exception {

      ServerSocket server=new ServerSocket(6262);
      System.out.println("Server established.");
      Socket client=server.accept();
      ObjectOutputStream oos=new
ObjectOutputStream(client.getOutputStream());
      ObjectInputStream ois=new ObjectInputStream(client.getInputStream());

      System.out.println("Client is now
      connected."); int x=(Integer)ois.readObject();
      int k=(Integer)ois.readObject(); int j=0;
      int i=(Integer)ois.readObject();
      boolean flag=true;

      Random r=new Random(6);
```

```java
        int mod=r.nextInt(6);
        while(mod==1||mod==0)
        mod=r.nextInt(6);

        while(true)
        { int c=k;
            for(int h=0;h<=x;h++)
            {
                System.out.print("|"+c+"|");
                c=(c+1)%x;
            }
            System.out.println();
            System.out.println();
            if(k==j)
            {
                System.out.println("Frame "+k+" recieved"+"\n"+"Data:"+j);
                j++;
                System.out.println();
            }
            else
                System.out.println("Frames recieved not in correct
                order"+"\n"+"
Expected frame:" + j +"\n"+ " Recieved frame no :"+ k);

            System.out.println();
            if(j%mod==0 && flag)
            {
                System.out.println("Error found. Acknowledgement not sent.
            "); flag=!flag; j--; }
            else if(k==j-1)
            { oos.writeObject(k);
                System.out.println("Acknowledgement sent");
            }
            System.out.println();

            if(j%mod==0) flag=!flag;

            k=(Integer)ois.readObject()

            ;

            if(k==-1)
                break;

            i=(Integer)ois.readObject();
        }
```

```
            System.out.println("Client finished sending data. Exiting");
            oos.writeObject(-1);
        }
}
```

/*--------------OUTPUT----------------

C:\Users\dell\Java\CN\GoBackN>javac Server.java

C:\Users\dell\Java\CN\GoBackN>java Server
Server established.
Client is now connected.
|0||1||2||3||4||5||6||7||8||9||10||11||12||13||14||15||16||17||18||19||20||2
1||22||23||24||25||26||27||28||29||30||31||32||33||34||35||36||37||38||39||4
0||41||42||43||44||45||46||47||48||49||50||51||52||53||54||55||56||57||58||5
9||60||61||62||63||64||65||66||67||68||69||70||71||72||73||74||75||76||77||7
8||79||80||81||82||83||84||85||86||87||88||89||90||91||92||93||94||95||96||9
7||98||99||100||101||102||103||104||105||106||107||108||109||110||111||112||
113||114||115||116||117||118||119||120||121||122||123||124||125||126||0|

Frame 0 recieved
Data:0

Acknowledgement sent

|1||2||3||4||5||6||7||8||9||10||11||12||13||14||15||16||17||18||19||20||21||
22||23||24||25||26||27||28||29||30||31||32||33||34||35||36||37||38||39||40||
41||42||43||44||45||46||47||48||49||50||51||52||53||54||55||56||57||58||59||
60||61||62||63||64||65||66||67||68||69||70||71||72||73||74||75||76||77||78||
79||80||81||82||83||84||85||86||87||88||89||90||91||92||93||94||95||96||97||
98||99||100||101||102||103||104||105||106||107||108||109||110||111||112||113
||114||115||116||117||118||119||120||121||122||123||124||125||126||0||1|

Frame 1 recieved
Data:1

Error found. Acknowledgement not sent.

2		3		4		5		6		7		8		9		10		11		12		13		14		15		16		17		18		19		20		21		22
23		24		25		26		27		28		29		30		31		32		33		34		35		36		37		38		39		40		41
42		43		44		45		46		47		48		49		50		51		52		53		54		55		56		57		58		59		60
61		62		63		64		65		66		67		68		69		70		71		72		73		74		75		76		77		78		79
80		81		82		83		84		85		86		87		88		89		90		91		92		93		94		95		96		97		98
```

```
|99||100||101||102||103||104||105||106||107||108||109||110||111||112||113||1
14||115||116||117||118||119||120||121||122||123||124||125||126||0||1||2|

Frames recieved not in correct order
 Expected frame:1
 Recieved frame no :2


|3||4||5||6||7||8||9||10||11||12||13||14||15||16||17||18||19||20||21||22||23
||24||25||26||27||28||29||30||31||32||33||34||35||36||37||38||39||40||41||42
||43||44||45||46||47||48||49||50||51||52||53||54||55||56||57||58||59||60||61
||62||63||64||65||66||67||68||69||70||71||72||73||74||75||76||77||78||79||80
||81||82||83||84||85||86||87||88||89||90||91||92||93||94||95||96||97||98||99
||100||101||102||103||104||105||106||107||108||109||110||111||112||113||114|
|115||116||117||118||119||120||121||122||123||124||125||126||0||1||2||3|

Frames recieved not in correct order
 Expected frame:1
 Recieved frame no :3


|4||5||6||7||8||9||10||11||12||13||14||15||16||17||18||19||20||21||22||23||2
4||25||26||27||28||29||30||31||32||33||34||35||36||37||38||39||40||41||42||4
3||44||45||46||47||48||49||50||51||52||53||54||55||56||57||58||59||60||61||6
2||63||64||65||66||67||68||69||70||71||72||73||74||75||76||77||78||79||80||8
1||82||83||84||85||86||87||88||89||90||91||92||93||94||95||96||97||98||99||1
00||101||102||103||104||105||106||107||108||109||110||111||112||113||114||11
5||116||117||118||119||120||121||122||123||124||125||126||0||1||2||3||4|

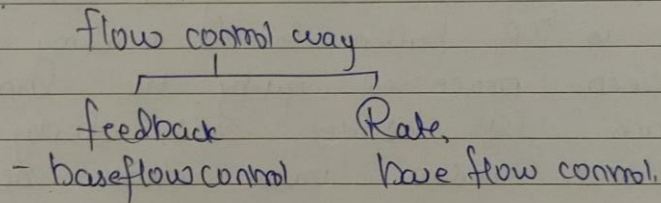Frames recieved not in correct order
 Expected frame:1
 Recieved frame no :4

*/
```

4 Go Back N protocol.
Selective Repeat Protocol.

Q1 Why we need flow control Mechanism?
- flow control mechanism can be classified
by we whether or not the reciving node
sends feedback to the sending node.
flow control is important because it is
possible for a sending computer to
transmit information at a faster rate that
the destination computer can receive and
process it.

Q2 What are the ways to control flow Control at
DLL?

flow control way

feedback   Rate,
- baseflow control   base flow control.

- In these protocols the   - These protocol have built
  sender send frames after   in mechanisms to restrict
  it has received acknouledgeme   the rate of transmission
  from the user. this used   of data without requiring
  in the data link layer   this is used in the.
                                   network layer the transport ly.

Q3 What is Sliding Window. ?
→ The Sliding window is a technique for sending
  multiple frame at a time. It control the data
  packet betn the two device where reliable &
  gradual delivery of data frame is needed. it
  used TCP.

Pushkaraj Chavan TEC 58

Q4. How window size grow and shrink at client side?
- Simple put it is a TCP receive buffer for incoming data that has not been processed yet by the application. The client & server are advertising their window size value as they communicate
- Each TCP header will display the most recent window value which can grow or shrink as the connection progresses

Q5 How window size grow & shrink at server side?
- It is TCP buffer for incoming data has not been processed yet by the application.
Each TCP header will display the most recent window value when can grow or shrink as the connection progresses. In this example the client has TCP receives window of 65,535 bytes and the server has 3,840.

Pushkaraj Chavan TEC 58