**SIMULATION OF ARP AND RARP**

EX.NO. : 312217205003

DATE : P.G.ABINAYA

PROGRAM:

**gr1.java:**

import java.util.\*;

import java.io.\*;

import java.lang.\*;

import java.net.\*;

class gr1{

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

{

try{

int[] x = new int[4];

int[] flag= new int[100];

int[] flag1= new int[100];

List<String> arr1 = new ArrayList<String>();

List<String> arr = new ArrayList<String>();

String l="",str="",strr="",wq="",ii="";

Random t = new Random();

int i,j,f,count=0,g=0,ch,k;

//System.out.println("Logical Address");

for (i=0;i<10;i++) {

String ip1 = Integer.toString (t.nextInt(255)) ;

String ip2 = Integer.toString(t.nextInt(255) );

String ip3 = Integer.toString(t.nextInt(255)) ;

String ip4 = Integer.toString(t.nextInt(255));

String ip=ip1+"."+ip2+"."+ip3+"."+ip4;

arr.add(ip);

//System.out.println(arr.get(i));

String p1 = Integer.toHexString(t.nextInt(255)) ;

String p2 = Integer.toHexString(t.nextInt(255) );

String p3 = Integer.toHexString(t.nextInt(255)) ;

String p4 = Integer.toHexString(t.nextInt(255));

String p=p1+":"+p2+":"+p3+":"+p4;

arr1.add(p);

//System.out.println(arr1.get(i)+"\n");

}

FileWriter fw = new FileWriter("file.txt");

FileWriter fw1 = new FileWriter("file1.txt");

flag[0]=1;

flag1[0]=1;

for(i=1;i<arr.size();i++)

{

flag[i]=0;

flag1[i]=0;

}

for (i = 0; i < arr.size(); i++)

{

fw.write(arr.get(i)+"\n");

fw1.write(arr1.get(i)+"\n");

}

fw.close();

fw1.close();

//System.out.println("Physical Address");

System.out.println("Enter choice \n 1)Socket ARP \n 2)Socket RARP \n 3)Exit");

Scanner o=new Scanner(System.in);

ch=o.nextInt();

switch(ch)

{

case 1:{

File zz= new File("file2.txt");

FileWriter fw2 = new FileWriter(zz);

File yy =new File("file3.txt");

FileWriter fw3 = new FileWriter(yy);

File xx= new File("C:/Users/intel/Desktop/overview/networks/file.txt");

FileReader fr=new FileReader(xx);

BufferedReader br=new BufferedReader(fr);

File vv= new File("C:/Users/intel/Desktop/overview/networks/file1.txt");

FileReader fr1=new FileReader(vv);

BufferedReader br1=new BufferedReader(fr1);

ServerSocket s = new ServerSocket(1565);

Socket obj = s.accept();

DataInputStream din = new DataInputStream(obj.getInputStream());

DataOutputStream dout = new DataOutputStream(obj.getOutputStream());

f=0;

String ab=br.readLine();

String cd=br1.readLine();

String bc="00:00:00:00";

dout.writeBytes(ch+"\n");

dout.writeBytes(ab+"\n");

dout.writeBytes(cd+"\n");

for(i=0;i<5;i++)

{

br.readLine();

br1.readLine();

}

str=br.readLine();

dout.writeBytes(str+"\n");

strr=br1.readLine();

dout.writeBytes(strr+"\n");

fw2.write(ab+"\n");

fw2.write(str+"\n");

fw3.write(cd+"\n");

fw3.write(strr+"\n");

System.out.println("IP:"+ab+"\nMAC:"+cd);

System.out.println("\n\nServer broadcast the ARP Request Packet");

System.out.println("-------------------------------------------------------------------------");

System.out.println(" Ethernet(Hardware Type) | IPv4(Protocol Type)");

System.out.println("-------------------------------------------------------------------------");

System.out.println("6(Hardware Length) | 4(Protocol Length) | 1 Request (Operation)");

System.out.println("-------------------------------------------------------------------------");

System.out.println("Sender Hardware(MAC) address "+cd);

System.out.println("-------------------------------------------------------------------------");

System.out.println("Sender Protocol(IP) address "+ab);

System.out.println("-------------------------------------------------------------------------");

System.out.println("Target Hardware(MAC) address "+bc);

System.out.println("-------------------------------------------------------------------------");

System.out.println("Target Protocol(IP) address "+str);

System.out.println("-------------------------------------------------------------------------");

System.out.println("\n\n\n\n\nServer received the ARP Reply Packet from the Client");

System.out.println("-------------------------------------------------------------------------");

System.out.println(" Ethernet(Hardware Type) | IPv4(Protocol Type)");

System.out.println("-------------------------------------------------------------------------");

System.out.println("6(Hardware Length) | 4(Protocol Length) | 2 Reply (Operation)");

System.out.println("-------------------------------------------------------------------------");

System.out.println("Sender Hardware(MAC) address "+strr);

System.out.println("-------------------------------------------------------------------------");

System.out.println("Sender Protocol(IP) address "+str);

System.out.println("-------------------------------------------------------------------------");

System.out.println("Target Hardware(MAC) address "+cd);

System.out.println("-------------------------------------------------------------------------");

System.out.println("Target Protocol(IP) address "+ab);

System.out.println("-------------------------------------------------------------------------");

System.out.println("I got the Client's MAC address!!!!!!!!!!!!");

obj.close();

fw2.close();

fw3.close();

fr.close();

fr1.close();

File uu= new File("file4.txt");

FileWriter fw4 = new FileWriter(uu);

File ww= new File("file5.txt");

FileWriter fw5 = new FileWriter(ww);

FileReader frr=new FileReader("file.txt");

BufferedReader brr=new BufferedReader(frr);

FileReader frr1=new FileReader("file1.txt");

BufferedReader brr1=new BufferedReader(frr1);

brr.readLine();

brr1.readLine();

for(i=0;i<8;i++)

{

wq=brr.readLine();

ii=brr1.readLine();

if(wq.equals(str))

{

brr.readLine();

brr1.readLine();

}

else

{

fw4.write(wq+"\n");

fw5.write(ii+"\n");

}

}

frr.close();

frr1.close();

fw4.close();

fw5.close();

xx.delete();

File ff=new File("file.txt");

uu.renameTo(ff);

File ff1=new File("file1.txt");

vv.delete();

ww.renameTo(ff1);

break;

}

case 2:

{

File zz= new File("file2.txt");

FileWriter fw2 = new FileWriter(zz);

File yy =new File("file3.txt");

FileWriter fw3 = new FileWriter(yy);

File xx= new File("C:/Users/intel/Desktop/overview/networks/file.txt");

FileReader fr=new FileReader(xx);

BufferedReader br=new BufferedReader(fr);

File vv= new File("C:/Users/intel/Desktop/overview/networks/file1.txt");

FileReader fr1=new FileReader(vv);

BufferedReader br1=new BufferedReader(fr1);

ServerSocket s = new ServerSocket(1565);

Socket obj = s.accept();

DataInputStream din = new DataInputStream(obj.getInputStream());

DataOutputStream dout = new DataOutputStream(obj.getOutputStream());

f=0;

String ab=br.readLine();

String cd=br1.readLine();

String bc="00.00.00.00";

dout.writeBytes(ch+"\n");

dout.writeBytes(ab+"\n");

dout.writeBytes(cd+"\n");

for(i=0;i<5;i++)

{

br.readLine();

br1.readLine();

}

str=br.readLine();

dout.writeBytes(str+"\n");

strr=br1.readLine();

dout.writeBytes(strr+"\n");

fw2.write(ab+"\n");

fw2.write(str+"\n");

fw3.write(cd+"\n");

fw3.write(strr+"\n");

System.out.println("IP:"+ab+"\nMAC:"+cd);

System.out.println("\n\nServer received the RARP Request Packet from Client");

System.out.println("-------------------------------------------------------------------------");

System.out.println(" Ethernet(Hardware Type) | IPv4(Protocol Type)");

System.out.println("-------------------------------------------------------------------------");

System.out.println("6(Hardware Length) | 4(Protocol Length) | 1 Request (Operation)");

System.out.println("-------------------------------------------------------------------------");

System.out.println("Sender Hardware(MAC) address "+strr);

System.out.println("-------------------------------------------------------------------------");

System.out.println("Sender Protocol(IP) address "+bc);

System.out.println("-------------------------------------------------------------------------");

System.out.println("Target Hardware(MAC) address "+cd);

System.out.println("-------------------------------------------------------------------------");

System.out.println("Target Protocol(IP) address "+ab);

System.out.println("-------------------------------------------------------------------------");

System.out.println("\n\n\n\n\nServer sends the RARP Reply Packet to the Client");

System.out.println("-------------------------------------------------------------------------");

System.out.println(" Ethernet(Hardware Type) | IPv4(Protocol Type)");

System.out.println("-------------------------------------------------------------------------");

System.out.println("6(Hardware Length) | 4(Protocol Length) | 2 Reply (Operation)");

System.out.println("-------------------------------------------------------------------------");

System.out.println("Sender Hardware(MAC) address "+cd);

System.out.println("-------------------------------------------------------------------------");

System.out.println("Sender Protocol(IP) address "+ab);

System.out.println("-------------------------------------------------------------------------");

System.out.println("Target Hardware(MAC) address "+strr);

System.out.println("-------------------------------------------------------------------------");

System.out.println("Target Protocol(IP) address "+str);

System.out.println("-------------------------------------------------------------------------");

obj.close();

System.out.println("Your Request is Responded!!!!!!!!!!!!!");

fw2.close();

fw3.close();

fr.close();

fr1.close();

File uu= new File("file4.txt");

FileWriter fw4 = new FileWriter(uu);

File ww= new File("file5.txt");

FileWriter fw5 = new FileWriter(ww);

FileReader frr=new FileReader("file.txt");

BufferedReader brr=new BufferedReader(frr);

FileReader frr1=new FileReader("file1.txt");

BufferedReader brr1=new BufferedReader(frr1);

brr.readLine();

brr1.readLine();

for(i=0;i<8;i++)

{

wq=brr.readLine();

ii=brr1.readLine();

if(wq.equals(str))

{

brr.readLine();

brr1.readLine();

}

else

{

fw4.write(wq+"\n");

fw5.write(ii+"\n");

}

} frr.close();

frr1.close();

fw4.close();

fw5.close();

xx.delete();

File ff=new File("file.txt");

uu.renameTo(ff);

File ff1=new File("file1.txt");

vv.delete();

ww.renameTo(ff1);

}

}

}

catch(Exception e)

{

System.out.println(e);

}

}

}

**Clientarp1.java**

import java.io.\*;

import java.util.\*;

import java.net.\*;

public class clientarp1{

public static void main(String args[]){

try{

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

Socket sc = new Socket("127.0.0.1",1565);

DataInputStream din = new DataInputStream(sc.getInputStream());

DataOutputStream dout = new DataOutputStream(sc.getOutputStream());

String gg=din.readLine();

String xy=din.readLine();

String bc=din.readLine();

String str1=din.readLine();

String str=din.readLine();

String xx="00:00:00:00";

String yy="00.00.00.00";

if(gg.equals("1"))

{

System.out.println("IP:"+str1+"\nMAC:"+xx);

System.out.println("\n\nClient receives ARP Request Packet from Server");

System.out.println("-------------------------------------------------------------------------");

System.out.println(" Ethernet(Hardware Type) | IPv4(Protocol Type)");

System.out.println("-------------------------------------------------------------------------");

System.out.println("6(Hardware Length) | 4(Protocol Length) | 1 Request (Operation)");

System.out.println("-------------------------------------------------------------------------");

System.out.println("Sender Hardware(MAC) address "+bc);

System.out.println("-------------------------------------------------------------------------");

System.out.println("Sender Protocol(IP) address "+xy);

System.out.println("-------------------------------------------------------------------------");

System.out.println("Target Hardware(MAC) address "+xx);

System.out.println("-------------------------------------------------------------------------");

System.out.println("Target Protocol(IP) address "+str1);

System.out.println("-------------------------------------------------------------------------");

System.out.println("\n\n\n\n\nClient Sends ARP Reply Packet to Server");

System.out.println("-------------------------------------------------------------------------");

System.out.println(" Ethernet(Hardware Type) | IPv4(Protocol Type)");

System.out.println("-------------------------------------------------------------------------");

System.out.println("6(Hardware Length) | 4(Protocol Length) | 2 Reply (Operation)");

System.out.println("-------------------------------------------------------------------------");

System.out.println("Sender Hardware(MAC) address "+str);

System.out.println("-------------------------------------------------------------------------");

System.out.println("Sender Protocol(IP) address "+str1);

System.out.println("-------------------------------------------------------------------------");

System.out.println("Target Hardware(MAC) address "+bc);

System.out.println("-------------------------------------------------------------------------");

System.out.println("Target Protocol(IP) address "+xy);

System.out.println("-------------------------------------------------------------------------");

System.out.println("Your Request is Responded!!!!!!!!!!!!!");

}

else

{

System.out.println("IP:"+yy+"\nMAC:"+str);

System.out.println("\n\nClient Sends RARP Request Packet to Server");

System.out.println("-------------------------------------------------------------------------");

System.out.println(" Ethernet(Hardware Type) | IPv4(Protocol Type)");

System.out.println("-------------------------------------------------------------------------");

System.out.println("6(Hardware Length) | 4(Protocol Length) | 1 Request (Operation)");

System.out.println("-------------------------------------------------------------------------");

System.out.println("Sender Hardware(MAC) address "+str);

System.out.println("-------------------------------------------------------------------------");

System.out.println("Sender Protocol(IP) address "+yy);

System.out.println("-------------------------------------------------------------------------");

System.out.println("Target Hardware(MAC) address "+bc);

System.out.println("-------------------------------------------------------------------------");

System.out.println("Target Protocol(IP) address "+xy);

System.out.println("-------------------------------------------------------------------------");

System.out.println("\n\n\n\n\nClient receives RARP Reply Packet from Server");

System.out.println("-------------------------------------------------------------------------");

System.out.println(" Ethernet(Hardware Type) | IPv4(Protocol Type)");

System.out.println("-------------------------------------------------------------------------");

System.out.println("6(Hardware Length) | 4(Protocol Length) | 2 Reply (Operation)");

System.out.println("-------------------------------------------------------------------------");

System.out.println("Sender Hardware(MAC) address "+bc);

System.out.println("-------------------------------------------------------------------------");

System.out.println("Sender Protocol(IP) address "+xy);

System.out.println("-------------------------------------------------------------------------");

System.out.println("Target Hardware(MAC) address "+str);

System.out.println("-------------------------------------------------------------------------");

System.out.println("Target Protocol(IP) address "+str1);

System.out.println("-------------------------------------------------------------------------");

System.out.println("I got my IP address!!!!!!!!!!!!");

}

sc.close();

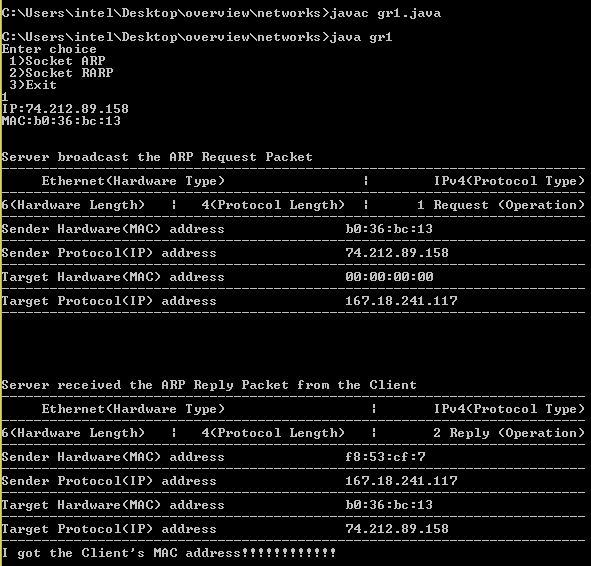
}catch(Exception e){

System.out.println(e);}}}

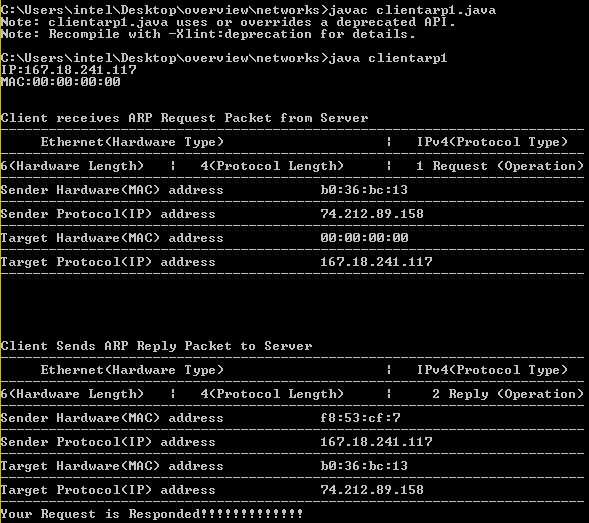
**SAMPLE INPUT/OUTPUT:**

1.)ARP

gr1.java

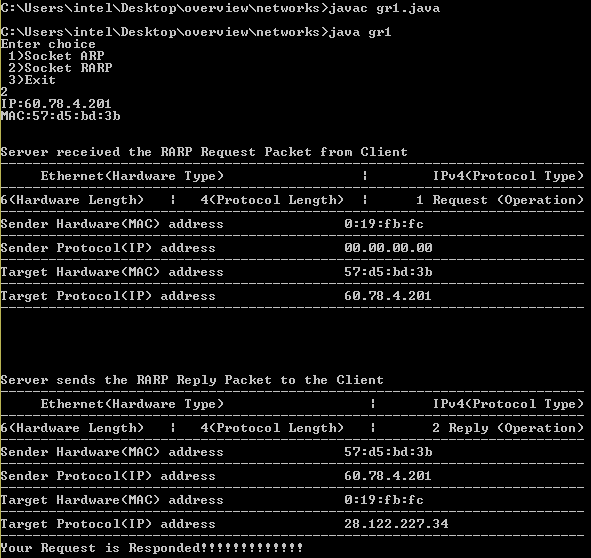


**Clientarp1.java**

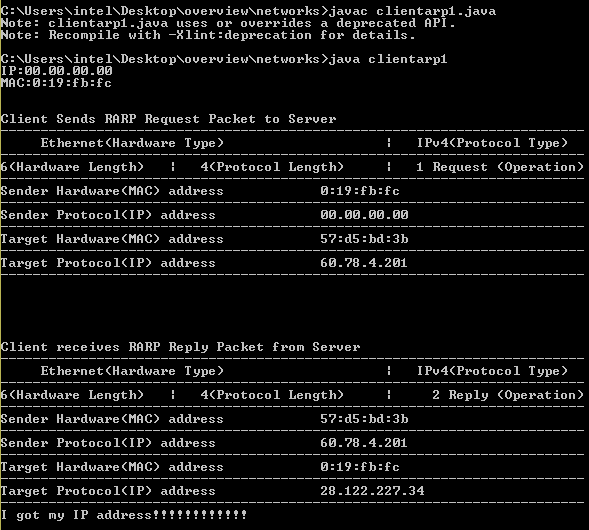


**2.)RARP**

**gr1.java**



**Clientarp1.java**



**SIMULATION OF DNS**

EX.NO. : 312217205003

DATE : P.G.ABINAYA

PROGRAM:

**serverdns.java**

import java.io.\*;

import java.io.IOException;

import java.net.DatagramPacket;

import java.net.DatagramSocket;

import java.net.InetAddress;

import java.net.SocketException;

import java.awt.Desktop;

import java.net.URI;

import java.util.Scanner;

public class serverdns{

public static void main(String[] args) throws IOException

{

try{

int i,flag=0;

String[][] a=new String[5][5];

a[0][0]="www.google.com";

a[0][1]="172.217.11.174";

a[1][0]="www.w3schools.com";

a[1][1]="66.29.212.110";

a[2][0]="www.Dell.com";

a[2][1]="143.166.83.38";

a[3][0]="www.quora.com";

a[3][1]="3.212.29.26";

a[4][0]="www.comlaude.com";

a[4][1]="213.212.81.71";

DatagramSocket ds = new DatagramSocket(1234);

byte[] receive = new byte[65535];

DatagramPacket DpReceive = null;

System.out.println("Server starts Working");

DpReceive = new DatagramPacket(receive, receive.length);

ds.receive(DpReceive);

String b=data(receive).toString();

System.out.println("Client "+b);

for(i=0;i<4;i++)

{

if(b.equals(a[i][0]))

{

System.out.println("Server " + a[i][1]);

Desktop d=Desktop.getDesktop();

d.browse(new URI(b));

flag=1;

}

else if(b.equals(a[i][1]))

{

System.out.println("Server " + a[i][0]);

Desktop d=Desktop.getDesktop();

d.browse(new URI(a[i][0]));

flag=1;

}

}

if(flag==0)

{

System.out.println("Entered ip address or url is wrong");

}

receive = new byte[65535];

}

catch(Exception E){

System.err.println("Exp : "+E.getMessage());

}

}

public static StringBuilder data(byte[] a)

{

if (a == null)

return null;

StringBuilder ret = new StringBuilder();

int i = 0;

while (a[i] != 0)

{

ret.append((char) a[i]);

i++;

}

return ret;

}

}

**clientdns.java**

import java.io.\*;

import java.io.IOException;

import java.net.DatagramPacket;

import java.net.DatagramSocket;

import java.net.InetAddress;

import java.util.Scanner;

public class clientdns

{

public static void main(String args[]) throws IOException

{

Scanner sc = new Scanner(System.in);

DatagramSocket ds = new DatagramSocket();

InetAddress ip = InetAddress.getLocalHost();

System.out.println("Client starts Working");

byte buf[] = null;

byte[] receive = new byte[65535];

String inp = sc.nextLine();

buf = inp.getBytes();

DatagramPacket DpSend = new DatagramPacket(buf, buf.length, ip, 1234);

ds.send(DpSend);

}

public static StringBuilder data(byte[] a)

{

if (a == null)

return null;

StringBuilder ret = new StringBuilder();

int i = 0;

while (a[i] != 0)

{

ret.append((char) a[i]);

i++;

}

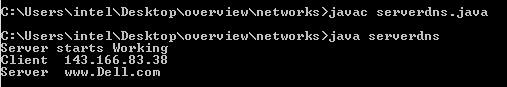
return ret;

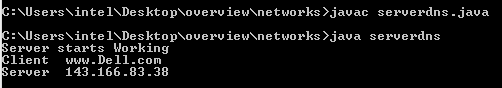
}

}

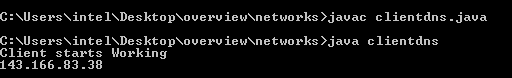
**SAMPLE INPUT\OUTPUT:**

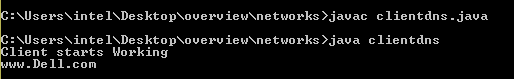
**serverdns.java**

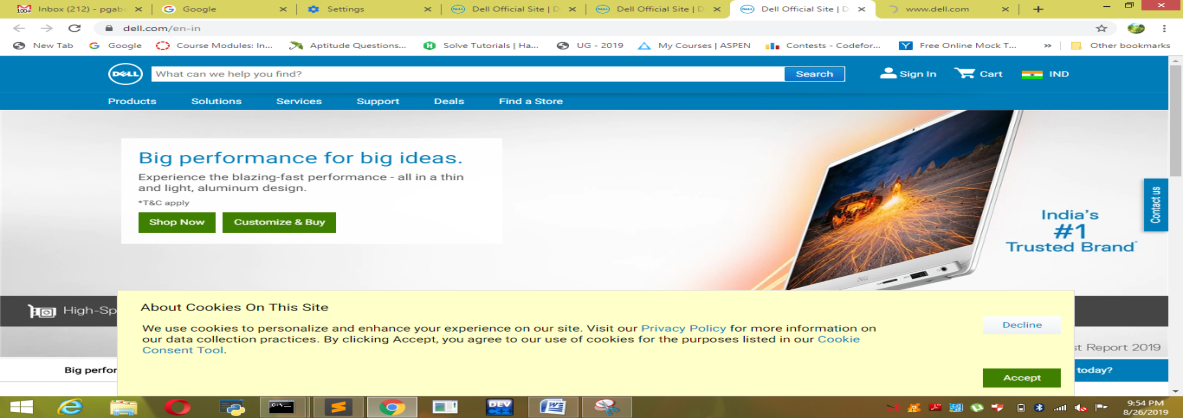




**clientdns.java**







**SIMULATION OF CRC(CYCLIC REDUNDANCY CHECK)**

EX.NO. : 312217205003

DATE : P.G.ABINAYA

**PROGRAM:**

crc1.java

import java.util.\*;

class crc1 {

public static void main(String args[]) {

Scanner scan = new Scanner(System.in);

int n;

System.out.println("Enter the size of the data");

n = scan.nextInt();

int data[] = new int[n];

System.out.println("Enter the data, bit by bit from msb");

for(int i=0 ; i < n ; i++) {

data[i] = scan.nextInt();

}

System.out.println("Enter the size of the divisor");

n = scan.nextInt();

int divisor[] = new int[n];

System.out.println("Enter the divisor, bit by bit from msb");

for(int i=0 ; i < n ; i++) {

divisor[i] = scan.nextInt();

}

int remainder[] = divide(data, divisor,2);

System.out.println("The Remainder is:");

for(int i=0 ; i < remainder.length-1 ; i++) {

System.out.print(remainder[i]);

}

int sent\_data[] = new int[data.length + remainder.length - 1];

System.out.println("\nThe CRC code generated is:");

for(int i=0 ; i < data.length ; i++) {

sent\_data[i]=data[i];

System.out.print(data[i]);

}

for(int i=0 ; i < remainder.length-1 ; i++) {

sent\_data[i+data.length]=remainder[i];

System.out.print(remainder[i]);

}

System.out.println();

int rec\_data[] = new int[data.length + remainder.length - 1];

System.out.println("The Data send by the sender");

for(int i=0;i<sent\_data.length;i++)

{

System.out.print(sent\_data[i]);

}

System.out.println();

System.out.println("Enter the data received bit by bit");

for(int i=0 ; i < sent\_data.length ; i++) {

rec\_data[i] = scan.nextInt();

}

System.out.println("The Data received by the receiver");

for(int i=0;i<sent\_data.length;i++)

{

System.out.print(rec\_data[i]);

}

receive(rec\_data, divisor,1);

}

static int[] divide(int old\_data[], int divisor[],int a)

{

int remainder[] , i;

if(a==2)

{

int data[] = new int[old\_data.length+divisor.length];

System.arraycopy(old\_data, 0, data, 0, old\_data.length);

remainder = new int[divisor.length];

System.arraycopy(data, 0, remainder, 0, divisor.length);

for(i=0 ; i < old\_data.length; i++) {

if(remainder[0] == 1) {

for(int j=1 ; j < divisor.length ; j++) {

remainder[j-1] = exor(remainder[j], divisor[j]);

}

}

else {

for(int j=1 ; j < divisor.length ; j++) {

remainder[j-1] = exor(remainder[j], 0);

}

}

remainder[divisor.length-1] = data[i+divisor.length];

}

}

else

{

int data[] = new int[old\_data.length+divisor.length];

System.arraycopy(old\_data, 0, data, 0, old\_data.length);

remainder = new int[divisor.length];

System.arraycopy(data, 0, remainder, 0, divisor.length);

for(i=0 ; i < old\_data.length-divisor.length+1; i++) {

if(remainder[0] == 1) {

for(int j=1 ; j < divisor.length ; j++) {

remainder[j-1] = exor(remainder[j], divisor[j]);

}

}

else {

for(int j=1 ; j < divisor.length ; j++) {

remainder[j-1] = exor(remainder[j], 0);

}

}

remainder[divisor.length-1] = data[i+divisor.length];

}

}

return remainder;

}

static int exor(int a, int b) {

if(a == b) {

return 0;

}

return 1;

}

static void receive(int data[], int divisor[],int a) {

int remainder[] = divide(data, divisor,a);

for(int i=0;i<remainder.length;i++){

if(remainder[i] != 0) {

System.out.println("\nThere is an error in received data.\n");

return;

}

}

System.out.println("\nData received without any error.\nTransmission Successful!!!!!!!!!!!!!!!!!!!!!!!");

}

}

**SAMPLE INPUT\OUTPUT:**

