EX NO: 6A	Write a code simulating ARP protocols
Date:	

AIM:

To write a java program for simulating ARP and RARP protocols using TCP.

ALGORITHM:

Client:

1. Start the program
2. Create socket and establish connection with the server.
3. Get the IP address to be converted into MAC address from the user.
4. Send this IP address to server.
5. Receive the MAC address for the IP address from the server.
6. Display the received MAC address
7. Terminate the connection

Server:

1. Start the program
2. Create the socket, bind the socket created with IP address and port number andmake it a listening socket.
3. Accept the connection request when it is requested by the client.
4. Server maintains the table in which IP and corresponding MAC addresses arestored.
5. Receive the IP address sent by the client.
6. Retrieve the corresponding MAC address for the IP address and send it to theclient.
7. Close the connection with the client and now the server becomes a listening serverwaiting for the connection request from other clients
8. Stop

PROGRAM:

Client:

```
import java.io.*;
import java.net.*;
import java.util.*;
  class Clientarp {
      public static void main(String args[]) {
            try {
                  BufferedReader in = new BufferedReader(new InputStreamReader(System.in));
                  Socket clsct = new Socket("127.0.0.1", 139);
                  DataInputStream din = new DataInputStream(clsct.getInputStream());
                  DataOutputStream dout = new DataOutputStream(clsct.getOutputStream());
                  System.out.println("Enter the Logical address(IP):");
                  String str1 = in.readLine();
                  dout.writeBytes(str1 + '\n');
                  String str = din.readLine();

                  System.out.println("The Physical Address is: " + str);
                  clsct.close();
            } catch (Exception e) {
                  System.out.println(e);
            }
      }
}
```

Server:

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

class Serverarp {
      public static void main(String args[]) {
            try {
                  ServerSocket obj = new ServerSocket(139);
                  Socket obj1 = obj.accept();
                  while (true) {
                        DataInputStream din = new DataInputStream(obj1.getInputStream());
```

```
                DataOutputStream dout = new
                DataOutputStream(obj1.getOutputStream());String str = din.readLine();
                String ip[] = { "165.165.80.80", "165.165.79.1" };
                String mac[] = { "6A:08:AA:C2", "8A:BC:E3:FA" };
                for (int i = 0; i < ip.length; i++) {
                    if (str.equals(ip[i])) {
                        dout.writeBytes(mac[i] + '\n');
                        break;
                    }
                }
                obj.close();
            }
        } catch (Exception e) {

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

RESULT :

Thus the program for implementing to display simulating ARP protocols wasexecuted successfully and output is verified.

EX NO:6B	Write a code simulating RARP protocols
Date:	

AIM:

To write a Program for Reverse Address Resolution Protocol (RARP) using UDP

ALGORITHM:

Client:

1. Start the program
2. Create datagram socket
3. Get the MAC address to be converted into IP address from the user.
4. Send this MAC address to server using UDP datagram.
5. Receive the datagram from the server and display the corresponding IP address.
6. Stop the program

Server:

1. Start the program.
2. Server maintains the table in which IP and corresponding MAC addresses arestored.
3. Create the datagram socket
4. Receive the datagram sent by the client and read the MAC address sent.
5. Retrieve the IP address for the received MAC address from the table.
6. Display the corresponding IP address.
7. Stop the program.

PROGRAM:

Client:

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

```
class Clientrarp12 {
public static void main(String args[]) {
    try {
            DatagramSocket client = new DatagramSocket();
            InetAddress addr =
            InetAddress.getByName("127.0.0.1");byte[] sendbyte =
            new byte[1024];
            byte[] receivebyte = new byte[1024];
            BufferedReader in = new BufferedReader(new InputStreamReader(System.in));
            System.out.println("Enter the Physical address (MAC):");
            String str = in.readLine();
            sendbyte = str.getBytes();
            DatagramPacket sender = new DatagramPacket(sendbyte, sendbyte.length, addr, 1309);
            client.send(sender);
            DatagramPacket receiver = new DatagramPacket(receivebyte, receivebyte.length);
            client.receive(receiver);
            String s = new String(receiver.getData());
            System.out.println("The Logical Address is(IP): " + s.trim());
            client.close();
        } catch (Exception e) {
            System.out.println(e);
        }
    }
}
```

Server:

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

class Serverrarp12 {
    public static void main(String args[]) {
        try {
            DatagramSocket server = new
            DatagramSocket(1309);while (true) {
                byte[] sendbyte = new byte[1024];
                byte[] receivebyte = new byte[1024];
                DatagramPacket receiver = new DatagramPacket(receivebyte, receivebyte.length);
                server.receive(receiver);
                String str = new String(receiver.getData());
```

```
                String s = str.trim();
                InetAddress addr = receiver.getAddress();
                 Int port=receiver.getPort();
                 String ip[] = { "165.165.80.80", "165.165.79.1" };
                 String mac[] = { "6A:08:AA:C2", "8A:BC:E3:FA" };
                for (int i = 0; i < ip.length; i++) {
                 if  (s.equals(mac[i])) {
                sendbyte = ip[i].getBytes();
                DatagramPacket sender = new DatagramPacket(sendbyte, sendbyte.length, addr,port);
                 server.send(sender);
                break;
                 }
                 }
                 break;
            }
        } catch (Exception e) {
            System.out.println(e);
        }
    }
}
```

RESULT :

Thus the program for implementing to display simulating RARP protocols wasexecuted successfully and output is verified.

Dr. N.G.P. INSTITUTE OF TECHNOLOGY
COIMBATORE
AUTONOMOUS

Dr. N.G.P. INSTITUTE OF TECHNOLOGY
COIMBATORE
AUTONOMOUS