## Practical No 1

**Conversion of decimal number into binary number .**

**Input :**

```
list=[]

decimal_number=input("Enter a decimal number : ")

for i in decimal_number.split("."):

decimal_number=int(i)

decimal_number=bin(decimal_number)

binary_number=decimal_number[2:]

list.append(binary_number)

print("The binary number is : ",end="")

print(*list, sep="." )
```

**Conversion of binary number into decimal number.**

**Input :**

```
list=[]

binary_number=input("Enter a binary number :

") for i in binary_number.split("."):

binary_number=int(i)

binary_number=int(binary_number)

decimal_number=int(i,2)

list.append(decimal_number)

print("The decimal number is : ",end="")

print(*list, sep=".")
```

## Practical No 2 :

**Aim:- Introduction to Wireshark**

Step-1 open wireshark

Step -2 select interfaces(wifi or any other) and Captures packets

Step -3 enter url in browser :- : http://gaia.cs.umass.edu/wireshark-labs/INTRO-wireshark-file1.html

Step_ 4 : open first option in brower where you will see msg " **Congratulations! You've downloaded the first Wireshark lab file!** "

## Practical No 3 :

**Wireshark Lab: Ethernet and ARP**

Step- 1. clear chache( settings , privesy and setting , clear browsing data.

Step- 2. Start up the Wireshark wifi packet

Step- 3. enter URL in browser

https://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file3.html

Step- 4. Apply Filter http and analyze the packets

Step- 5. Apply Filter arp and open first arp request

## Practical No 4 :

**Wireshark Lab: IP**

Step- 1. Download PING PLOTTER - https://www.pingplotter.com/

Step- 2. Start up the Wireshark wifi packet and apply the icmp filter

Step-3. Open new target in pingplotter and enter target gain.cs.umass.edu

Step- 4. download this zip file and extract it –

http://gaia.cs.umass.edu/wireshark-labs/wireshark-traces.zip

Step- 5. In Wireshark go to file-> open_> ip-ethereal-trace-1->open.[with icmp filter]

## Practical No 5 :

**Wireshark Lab: ICMP, study of ping and traceroute command**

Step- 1. Start up the Wireshark wifi packet

Step- 2. Open cmd [Win+R] & Enter :-  ping -n 10 spsu.ac.in & then Enter:- ping -n 10 www.ust.hk

Step- 3. apply the icmp filter

## Practical No 6 :

**Wireshark Lab: UDP**

Step 1. Start wireshark wifi packets & filter udp packets

Step 2. Select one packet.

Step 3. From this packet, go below -> select and User Datagram Protocol ->determine how

many fields there are in the header.- **4 fileds**

## Practical No 7 :

**Wireshark Lab: TCP**

Step 1. Go to website https://gaia.cs.umass.edu/wireshark-labs/alice.txt and download alice.txt.

2. GO to website https://gaia.cs.umass.edu/wireshark-labs/TCP-wireshark-file1.html And upload "alice.txt".

3. Start wireshark wifi packet capture.

4.You will see msg on website "Congratulations!

You've now transferred a copy of alice.txt from your computer to gaia.cs.umass.edu. You should now stop Wireshark packet capture. It's time to start analyzing the captured Wireshark packets!"

5. filter http packets, select any one

6. Then see below transmission control protocols -> Source port :80  Do right click -> Apply as filter -> Selected

7. See Flags & Done

## Practical No 9 :

**Socket programming for UDP and TCP.**

**PROGRAM: //TCP Date Server--tcpdateserver.java**

```
import java.net.*;

import java.io.*;

import java.util.*;

class tcpdateserver {

public static void main (String arg[])

{

ServerSocket ss = null;

Socket cs;
PrintStream ps;
BufferedReader dis;
String inet;
try {
ss = new ServerSocket(4444);
System.out.println("Press Ctrl+C to quit");
while(true)
{
cs = ss.accept();
ps = new PrintStream(cs.getOutputStream()); Date d = new Date();
ps.println(d);
dis = new BufferedReader(new InputStreamReader(cs.getInputStream()));
inet = dis.readLine();
System.out.println("Client System/IP address is :"+ inet);
ps.close(); dis.close(); } }
catch(IOException e)
{ System.out.println("The exception is :" + e); } } }
```

**OUTPUT Server:**

```
$ javac tcpdateserver.java
$ java tcpdateserver
Press Ctrl+C to quit
Client System/IP address is : localhost.localdomain/127.0.0.1
```

**// TCP Date Client--tcpdateclient.java**

```
import java.net.*;
import java.io.*;
class tcpdateclient
{ public static void main (String args[])
{ Socket soc;
BufferedReader dis;
String sdate;
PrintStream ps;
try { InetAddress ia = InetAddress.getLocalHost();
if (args.length == 0) soc = new Socket(InetAddress.getLocalHost(),4444);
else soc = new Socket(InetAddress.getByName(args[0]),4444);
dis = new BufferedReader(new InputStreamReader(soc.getInputStream())); sdate=dis.readLine();
System.out.println("The date/time on server is : " +sdate);
ps = new PrintStream(soc.getOutputStream());
ps.println(ia);
ps.close();
catch(IOException e)
{ System.out.println("THE EXCEPTION is :" + e);
} } }
```

**Output:**

**Client:**

```
$ javac tcpdateclient.java
$ java tcpdateclient
The date/time on server is: Wed Jul 06 07:12:03 GMT 2011
```

Every time when a client connects to the server, server‟s date/time will be returned to the client for synchronization.

## Practical No 10 :

**PROGRAM // UDP Chat Server--udpchatserver.java** import java.io.*;

```
import java.net.*;
class udpchatserver {
public static int clientport = 8040,serverport = 8050;
public static void main(String args[]) throws Exception {
DatagramSocket SrvSoc = new DatagramSocket(clientport);
byte[] SData = new byte[1024];
BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
System.out.println("Server Ready");
while (true) {
byte[] RData = new byte[1024];
DatagramPacket RPack = new DatagramPacket(RData,RData.length);
SrvSoc.receive(RPack);
String Text = new String(RPack.getData());
if (Text.trim().length() == 0) break;
System.out.println("\nFrom Client <<< " + Text );
System.out.print("Msg to Cleint : " );
String srvmsg = br.readLine();
InetAddress IPAddr = RPack.getAddress();
SData = srvmsg.getBytes();
DatagramPacket SPack = new DatagramPacket(SData,SData.length,IPAddr, serverport);
SrvSoc.send(SPack); }
System.out.println("\nClient Quits\n"); SrvSoc.close(); } }
```

**OUTPUT**

**Server**

```
$ javac udpchatserver.java
$ java udpchatserver
Server Ready
From Client <<< are u the SERVER
Msg to Cleint : yes
From Client <<< what do u have to serve
Msg to Cleint : no eatables
Client Quits
```

**// UDP Chat Client--udpchatclient.java**

```
import java.io.*;
import java.net.*;
class udpchatclient {
public static int clientport = 8040,serverport = 8050;
public static void main(String args[]) throws Exception {
BufferedReader br = new BufferedReader(new InputStreamReader (System.in));
DatagramSocket CliSoc = new DatagramSocket(serverport);
InetAddress IPAddr;
String Text;
if (args.length == 0)
IPAddr = InetAddress.getLocalHost(); else IPAddr = InetAddress.getByName(args[0]);
byte[] SData = new byte[1024];
System.out.println("Press Enter without text to quit");
while (true)
{ System.out.print("\nEnter text for server : ");
Text = br.readLine();
SData = Text.getBytes();
DatagramPacket SPack = new DatagramPacket(SData,SData.length, IPAddr, clientport );
CliSoc.send(SPack);
if (Text.trim().length() == 0) break;
byte[] RData = new byte[1024];
DatagramPacket RPack = new DatagramPacket(RData,RData.length);
CliSoc.receive(RPack);
String Echo = new String(RPack.getData()) ;
Echo = Echo.trim();
System.out.println("From Server <<< " + Echo); }
CliSoc.close(); } }
```

**OUTPUT**

**Client**

```
$ javac udpchatclient.java
$ java udpchatclient
Press Enter without text to quit
Enter text for server : are u the SERVER
From Server <<< yes
Enter text for server : what do u have to serve
From Server <<< no eatables
Enter text for server : Ok
```

**- Viren Bagul**