Institute of Technology & Management

GIDA Gorakhpur

COMPUTER NETWORKS LAB

Subject Code- KCS 653

Prepared by: **Submitted to:**

PANKAJ   **NAME**

18XXXXXXXXX   **Assistant Professor**

IT 3ʳᵈ YEAR

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING AND INFORMATION TECHNOLOGY

Computer Networks Lab (KCS 653)

INDEX

S. No	Objects	Dated	Grade	Sign.
1.	Write a C program to implement data link layer framing method (Bit Stuffing).			
2.	Write a C program to implement data link layer framing method (Character Stuffing).			
3.	Write a program to implement data link layer framing method (character count).			
4.	Program using TCP Sockets Date and Time Server.			
5.	Implementation of Client-Server Communication using TCP.			
6.	Implementation of TCP/IP ECHO			
7.	Program using UDP socket UDP Chat Server/Client.			
8.	To learn handling and configuration of networking hardware like RJ-45 connector, CAT -6 cable, crimping tool, etc.			

Object 1

Write a C program to implement data link layer framing method (Bit Stuffing).

```c
#include<stdio.h>
#include<conio.h>
#include<string.h>
void main()
{
    int a[20],b[30],i,j,k,count,n;
//  clrscr();
    printf("Enter frame size that is total no. of bits");
    scanf("%d",&n);
    printf("Enter each of the frames that is bits in the form of 0 and 1: ");
    for(i=0; i<n; i++)
        scanf("%d",&a[i]);
    i=0;
    count=1;
    j=0;
    while(i<n)
    {
        if(a[i]==1)
        {
            b[j]=a[i];
            for(k=i+1; a[k]==1 && k<n && count<5; k++)
            {
                j++;
                b[j]=a[k];
                count++;
                if(count==5)
                {
                    j++;
                    b[j]=0;
                }
                i=k;
            }
        }
        else
        {
            b[j]=a[i];
        }
        i++;
        j++;
    }
    printf("After Bit Stuffing :");
    for(i=0; i<j; i++)
        printf("%d",b[i]);
    getch();
}
```

Output:

PROBLEMS   OUTPUT   TERMINAL   DEBUG CONSOLE

PS C:\Users\Pankaj\Desktop\New folder\lab> & 'c:\Users\Pankaj\.vscode\extensions\ms-vscode.cpp
tools-1.5.1\debugAdapters\bin\WindowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In-1cfp0jlk
.yye' '--stdout=Microsoft-MIEngine-Out-fmtx1h5l.3un' '--stderr=Microsoft-MIEngine-Error-iuszafu
p.vxe' '--pid=Microsoft-MIEngine-Pid-tzpnwyxw.czc' '--dbgExe=C:\Program Files\mingw-w64\x86_64-
8.1.0-posix-seh-rt_v6-rev0\mingw64\bin\gdb.exe' '--interpreter=mi'
Enter frame size that is total no. of bits: 6
Enter each of the frames that is bits in the form of 0 and 1):
1
0
0
1
1
0
After Bit Stuffing: 100110
PS C:\Users\Pankaj\Desktop\New folder\lab> █

Object 2

Write a C program to implement data link layer framing method (Character Stuffing).

Source Code:

```c
#include<stdio.h>
#include<conio.h>
#include<string.h>
void main()
{
        char msgIn[999], msgOut[999];
        int i, j=0, len;
//      clrscr();
        printf("Enter the Message: ");
        gets(msgIn);
        len=strlen(msgIn);
        for(i=0;i<len;i++)
        {
                if(msgIn[i]=='s' || msgIn[i]=='d' || msgIn[i]=='e')
                        msgOut[j++]='d';
                        msgOut[j++]=msgIn[i];
        }
        printf("After Stuffing \n");
        printf("%s",msgOut);
        getch();
}
```

Output:

```
PROBLEMS    OUTPUT    TERMINAL    DEBUG CONSOLE

PS C:\Users\Pankaj\Desktop\New folder\lab>  & 'c:\Users\Pankaj\.vscode\extensions\ms-vscode.cpp
tools-1.5.1\debugAdapters\bin\WindowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In-casqy0i5
.ilu' '--stdout=Microsoft-MIEngine-Out-llyvbmcq.idj' '--stderr=Microsoft-MIEngine-Error-iy3iaf3
g.4qw' '--pid=Microsoft-MIEngine-Pid-e3tbhloz.v15' '--dbgExe=C:\Program Files\mingw-w64\x86_64-
8.1.0-posix-seh-rt_v6-rev0\mingw64\bin\gdb.exe' '--interpreter=mi'
Enter the Message: Hello! World
After Stuffing
Hdello! Worldd
PS C:\Users\Pankaj\Desktop\New folder\lab>
```

Object 3

Write a program to implement data link layer framing method (character count).

Source Code:

```
#include<stdio.h>
#include<conio.h>
#include<string.h>
char data[30][30];
int n;
void main()
{
        int i, ch, j;
        char tmp[30][30];
//      clrscr();
        printf("Enter the no of frames:\n");
        scanf("%d",&n);
        for(i=0;i<=n;i++)
        {
                if(i!=0)
                {
                        printf("frame %d: ",i);
                        fflush(stdin);
                        gets(data[i]);
                }
        }
        //saving frame with count and data
        for(i=0;i<=n;i++)
        {
                tmp[i][0]=49+strlen(data[i]);
                tmp[i][1]='\0';
                strcat(tmp[i],data[i]);
        }
        printf("\n\t\t At the Sender: \n");
        printf("Data as frames: \n");
        for(i=1;i<=n;i++)
        {
                printf("Frame %d: ",i);
                puts(tmp[i]);
        }
        printf("Data transmitted");
        for(i=1;i<=n;i++)
        {
                printf("%s",tmp[i]);
        }
        printf("\n\t\t At the Receiver: \n");
        printf("Data Received: \n");
        for(i=1;i<=n;i++)
```

```
        {
                ch=(int)(tmp[i][0]-49);
                for(j=1;j<=ch;j++)
                {
                        data[i][j-1]=tmp[i][j];
                        data[i][j-1]='\0';
                }
                printf("The data after removing count character :\n");
                for(i=1;i<=n;i++)
                        printf("%s",data[i]);
                        printf("\n the data in the frame form: \n");
                        for(i=1;i<=n;i++)
                        {
                                printf("Frame%d:",i);
                                puts(data[i]);
                        }
                getch();
        }
}
```

Output:

PROBLEMS   OUTPUT   TERMINAL   DEBUG CONSOLE

```
Enter the no of frames:
3
frame 1: 0
frame 2: 1
frame 3: 2

                At the Sender:
Data as frames:
Frame 1: 20
Frame 2: 21
Frame 3: 22
Data transmitted202122
                At the Receiver:
Data Received:
The data after removing count character :
12
 the data in the frame form:
Frame1:
Frame2:1
Frame3:2
```

Object 4

Program using TCP Sockets Date and Time Server.

<u>Source Code</u>: **tcpdateserver.java**

```java
import java.net.*;
import java.io.*;
import java.util.*;
class tcpdateserver
{
        public static void main(String args[])
        {
                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);
                        }
        }

}
```

```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:

```
PS C:\Users\Pankaj\Desktop\New folder\lab>  c:; cd 'c:\Users\Pankaj\Desktop\New folder\lab'; &
'c:\Users\Pankaj\.vscode\extensions\vscjava.vscode-java-debug-0.34.0\scripts\launcher.bat' 'C:\
Program Files\AdoptOpenJDK\jdk-11.0.9.101-hotspot\bin\java.exe' '-Dfile.encoding=UTF-8' '-cp' '
C:\Users\Pankaj\AppData\Roaming\Code\User\workspaceStorage\1191e63ca440f28ad0703bff17df7e4c\red
hat.java\jdt_ws\lab_28e706b8\bin' 'tcpdateserver'
The exception is:java.net.BindException: Address already in use: NET_Bind
PS C:\Users\Pankaj\Desktop\New folder\lab>  c:; cd 'c:\Users\Pankaj\Desktop\New folder\lab'; &
'c:\Users\Pankaj\.vscode\extensions\vscjava.vscode-java-debug-0.34.0\scripts\launcher.bat' 'C:\
Program Files\AdoptOpenJDK\jdk-11.0.9.101-hotspot\bin\java.exe' '-Dfile.encoding=UTF-8' '-cp' '
C:\Users\Pankaj\AppData\Roaming\Code\User\workspaceStorage\1191e63ca440f28ad0703bff17df7e4c\red
hat.java\jdt_ws\lab_28e706b8\bin' 'tcpdateclient'
The date/time on server is:Fri Jul 16 21:11:58 IST 2021
PS C:\Users\Pankaj\Desktop\New folder\lab> []
```

Object 5

Implementation of Client-Server Communication using TCP.

<u>**Source Code**</u>: **server.java**

```
import java.io.*;
import java.net.*;
class server{
        public static void main(String args[]){
                String data ="Network Lab";
                try{
                        ServerSocket srvr=new ServerSocket(1234);
                        Socket skt= srvr.accept();
                        System.out.println("Server has connected!!!\n");
                        PrintWriter out=new PrintWriter(skt.getOutputStream(),
true);
                        System.out.print("Sending String: "+data+"\n");
                        out.print(data);
                        out.close();
                        skt.close();
                        srvr.close();
                }
                catch(Exception e){
                        System.out.println("It didn't work");
                }
        }
}
```

<u>**Source Code**</u>: **client.java**

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

                        Socket skt= new Socket("localhost",1234);
                        BufferedReader in=new BufferedReader(new
InputStreamReader(skt.getInputStream()));
                        System.out.println("Received string: ");
                        while(!in.ready()){}
                        System.out.println(in.readLine());
```

```
                    System.out.println("\n");
                    in.close();
            }
            catch(Exception e){
                    System.out.print("Error");
            }
        }
}
```

Output:

Object 6

Implementation of TCP/IP ECHO

Source Code: **tcpEchoServer.java**

```java
import java.net.*;
import java.io.*;
public class tcpEchoServer{
        public static void main(String args[]) throws IOException
        {
                ServerSocket sock=null;
                BufferedReader fromClient=null;
                OutputStreamWriter toClient=null;
                Socket client=null;
                try{
                        sock=new ServerSocket(4000);
                        System.out.println("Server is ready");
                        client=sock.accept();
                        System.out.println("Client Connected");

                        fromClient=new BufferedReader(new
InputStreamReader(client.getInputStream()));

                        toClient=new
OutputStreamWriter(client.getOutputStream());

                        String line;
                        while(true)
                        {
                                line= fromClient.readLine();
                                if((line==null)|| line.equals("bye"))
                                        break;
                                System.out.println("Client["+line+"]");
                                toClient.write("Server["+line+"]\n");
                                toClient.flush();
                        }
                        fromClient.close();
                        toClient.close();
                        client.close();
                        sock.close();
                        System.out.println("Client DisConnected");
                }
```

```java
        catch(IOException ioe){
            System.err.println(ioe);
        }
    }
}
```

Output:

Object 7

Program using UDP socket UDP Chat Server/Client.

<u>Source Code</u>: **udpChatServer.java**

```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("from client<<"+Text);
                        System.out.println("Msg to client:");
                        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("\n Client Quits\n");
                SrvSoc.close();
        }
}
```

Source Code: udpChatClient.java

```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.println("/n Enter 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:

```
Program Files\AdoptOpenJDK\jdk-11.0.9.101-hotspot\bin\java.exe' '-agentlib:jdwp=transport=dt_so
cket,server=n,suspend=y,address=localhost:60474' '-Dfile.encoding=UTF-8' '-cp' 'C:\Users\Pankaj
\AppData\Roaming\Code\User\workspaceStorage\acb0993cfcc852760249ae7a1fe0b903\redhat.java\jdt_ws
\lab_28e706b8\bin' 'udpChatServer'
Server Ready
from client<<Hey, there!




Msg to client:
how are you?
from client<<fine, U?
```

```
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\Pankaj\Desktop\New folder\lab>  & 'c:\Users\Pankaj\.vscode\extensions\vscjava.vscod
e-java-debug-0.34.0\scripts\launcher.bat' 'C:\Program Files\AdoptOpenJDK\jdk-11.0.9.101-hotspot
\bin\java.exe' '-agentlib:jdwp=transport=dt_socket,server=n,suspend=y,address=localhost:60477'
'-Dfile.encoding=UTF-8' '-cp' 'C:\Users\Pankaj\AppData\Roaming\Code\User\workspaceStorage\acb09
93cfcc852760249ae7a1fe0b903\redhat.java\jdt_ws\lab_28e706b8\bin' 'udpChatClient'
Press Enter without text to quit
/n Enter text for server:
Hey, there!
From Server<<how are you?
/n Enter text for server:
fine, U?
From Server<<good
/n Enter text for server:
```

Object 8

AIM :- To learn handling and configuration of network hardware like RJ45, Crimping tool,   Cat 5 Cables etc.

APPARATUS:- Rj-45, Camping tool, Cat 5 Cables etc.

RJ45 Connector:-

A registered jack (RJ) is a standardized physical network interface for connecting telecommunications or data equipment. The physical connectors that registered jacks use are mainly of the modular connector and 50-pin miniature ribbon connector types. The most common twisted-pair connector is an 8-position,8-contact (8P8C) modular plug and jack commonly referred to as an RJ45connector.

CRIMPING TOOL :

A crimping tool is a device used to conjoin two pieces of metal by deforming oneor both of them in a way that causes them to hold each other. The result of thetool's work is called a crimp. A good example of crimping is the process ofaffixing a connector to the end of a cable. For instance, network cables and phone cables are created using a crimping tool (shown below) to join the RJ-45and RJ-11 connectors to both ends of either phone or Cat 5 cable.

CAT 6 CABLES:-

A Category 6 cable (Cat 6 cable) is a type of twisted pair cable standard used specifically in gigabit (Gb) Ethernet-based computer networks. In 2002, it was jointly defined and specified by the Electronics Industries Association and Telecommunication Industries Association (EIA/TIA).
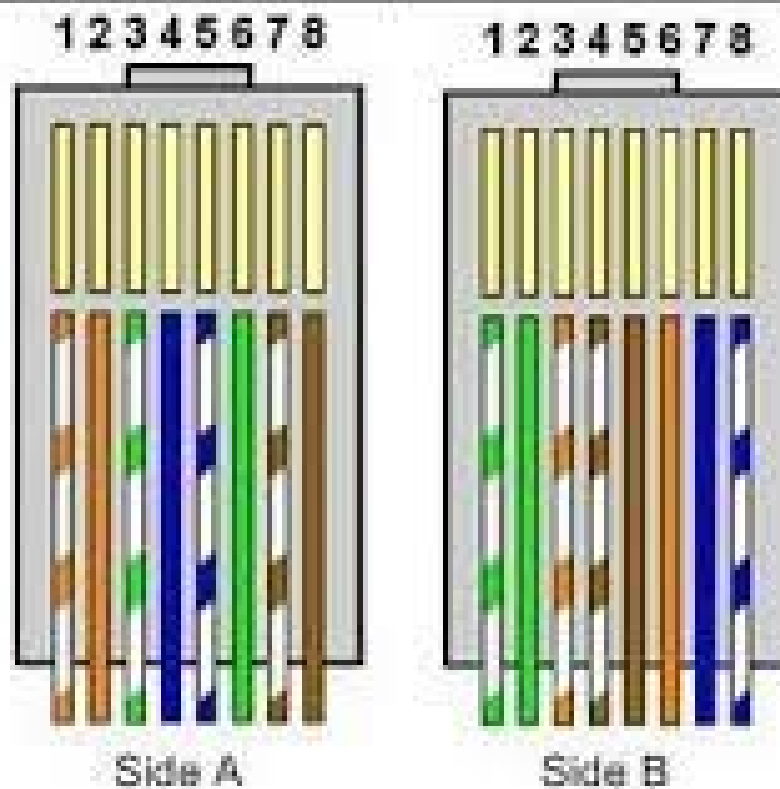
STRAIGHT CONNECTION:-
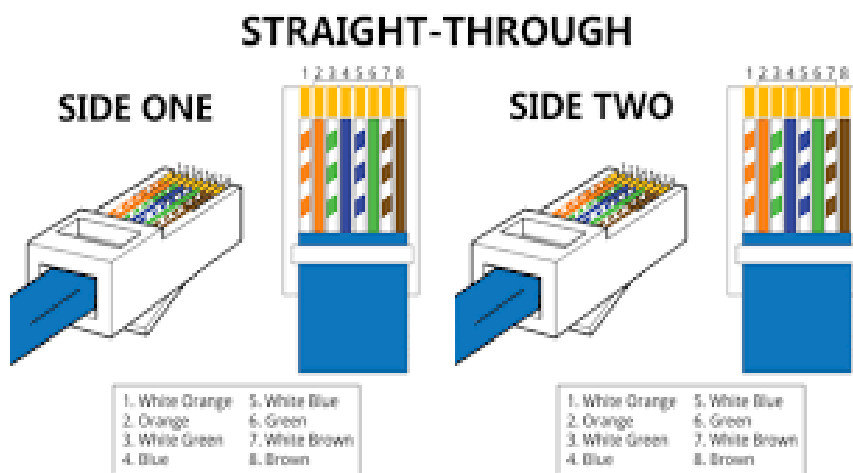
The most common wiring for RJ45 cables is the straight through cable. In this cable layout, all pins are wired one-to-one to the other side.

The pins on the RJ45 connector are assigned in pairs, and every pair carries one differential signal. Each line pair has to be twisted. If UTP or FTP cable is used, the pairs have orange, brown, blue and green colors. The wiring of these cables to RJ45 connectors to make a straight through cable is defined by EIA/TIA 568B. The RJ45 connectors on both ends are wired in the same way. The color scheme is shown below.

Pin ID	side A	side B
1	Orange-white	green-white
2	Orange	green
3	green-white	orange-white
4	blue	brown-white
5	blue-white	Brown
6	green	orange
7	brown-white	Blue
8	brown	blue-white

Side A   Side B

You usually use straight cable to connect different type of devices. This type of cable will be used most of the time and can be used.