

INDEX

S. No	Practical	Signature
1	To study the various Physical layer components of Networking: a) Connectors: BNC, RJ-45 b) Cables: Co-axial, Twisted Pair and UTP	
2	To study about various networking devices used at different layers of OSI: i) Network Adaptor ii) Hubs iii) Switch iv) Router v) Gateways	
3	Explain the installation of LAN card and LAN cabling	
4	To study and implement the OSI reference model	
5	Implementation of Bit Stuffing and Byte/Character stuffing	
6	Implement various error detection technique like Parity check, CRC, and Checksum	
7	Implement Error Correcting technique like Hamming code	
8	Implement the Distance vector Routing Algorithm	
9	Implement the Link State Routing Algorithm	
10	To study about IPV4, IPV6, default subnet mask, net id, host id and classful addressing	
11	To study about the installation and configuration of printers	

12	<p>Explain following network commands with all its suffix and example:</p> <div> <div>a) Ping</div> <div>b) Nslookup</div> <div>c) Ipconfig</div> <div>d) Pathping</div> <div>e) Arp</div> <div>f) Netstat</div> <div>g) Tracert</div> <div>h) Net</div> </div>	
----	---	--

Practical-1

To study the various Physical layer components of Networking.

a) Connectors: BNC, RJ-45 b) Cables: Co-axial, Twisted Pair and UTP

(A) CONNECTORS

A device that eliminates a section of cabling or implements a state of access for network devices, including PCs, hubs, and switches.

Connectors are used to connect the guided (wired) transmission media to devices like the hub, server, workstations etc.

(I) BNC



(Bayonet Neil-Concelman, or sometimes British Naval Connector) connector is used to connect a computer to a coaxial cable in a 10BASE-2 Ethernet network.

(II) RJ-45



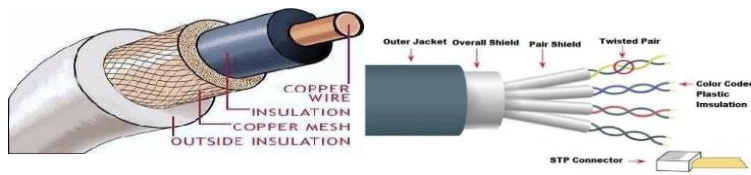
RJ45 is a type of connector commonly used for Ethernet networking. It looks like a telephone jack, but is slightly wider. The "RJ" in RJ45 stands for "registered jack," since it is a standardized networking interface. The "45" simply refers to the number of the interface standard.

(B) CABLE

(i) Co-axial Cable

A coaxial cable is a type of shielded and insulated copper cable that is used in computer networks and to deliver cable TV services to end users. It is used for both baseband and broadband data communication services.

(ii) Twisted Pair Cable



Twisted Pair Cable is a type of guided media.

Twisted pair cables have two conductors that are generally made up of copper and each conductor has insulation. These two conductors are twisted together, thus giving the name twisted pair cables.

(iii) UTP



In an UTP cable, conductors which form a single circuit are twisted around each other in order to cancel out electromagnetic interference (EMI) from external sources.

Unshielded means no additional shielding like meshes or aluminum foil, which add bulk, are used.

Practical-2

To study about various networking devices used at different layers of OSI.

A) Network adapter
Gateways

B) Hubs

C) Switch

D) Router

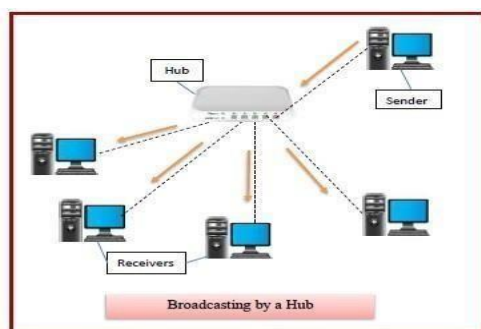
E)

(A) Network adapter

- Network adapters are one of the many pieces that connect us to the internet. They are usually an antenna or card built into your device, but can also be plug-in USB dongles or antennae that allow purely wired devices to receive data wirelessly.
- Network adapters allow computers and other devices to interface with a (LAN) or another type of network in order to access the internet. They can work with wireless connections like Wi-Fi.
- A network adapter is not the same thing as a router. Your router is the gateway that directs wireless traffic to your devices from the internet, while the adapter allows your device to connect to the network and receive that data.
- Examples: Network Interface Card, USB adapters.

(B) Hubs

- A hub operates in the physical layer of the OSI model.
- A hub cannot filter data. It is a non-intelligent network device that sends messages to all ports.
- Transmission mode is half duplex.
- They are passive devices; they don't have any software associated with it.
- They generally have fewer ports of 4/1.



Types of Hubs: -

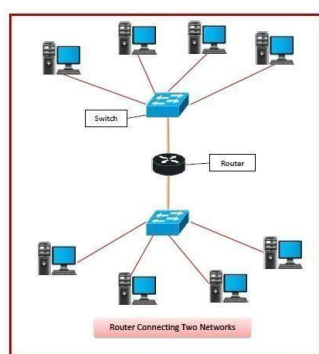
- **Passive Hubs** – Passive hubs connects nodes in a star configuration by collecting wiring from nodes. They broadcast signals onto the network without amplifying or regenerating them. As they cannot extend the distance between nodes, they limit the size of the LAN.
- **Active Hubs** – Active hubs amplify and regenerate the incoming electrical signals before broadcasting them. They have their own power supply and serves both as a repeater as well as connecting centre. Due to their regenerating capabilities, they can extend the maximum distance between nodes, thus increasing the size of LAN.
- **Intelligent Hubs** – Intelligent hubs are active hubs that provide additional network management facilities. They can perform a variety of functions of more intelligent network devices like network management, switching, providing flexible data rates etc.

(C) Switch

- A switch operates in the layer 2, i.e., data link layer of the OSI model.
- It is an intelligent network device that can be conceived as a multiport network bridge.
- It uses MAC addresses (addresses of medium access control sublayer) to send data packets to selected destination ports.
- It uses packet switching technique to receive and forward data packets from the source to the destination device.
- It supports unicast (one-to-one), multicast (one-to-many) and broadcast (one-to-all) communications.
- Transmission mode is full duplex, i.e., communication in the channel occurs in both the directions at the same time. Due to this, collisions do not occur.

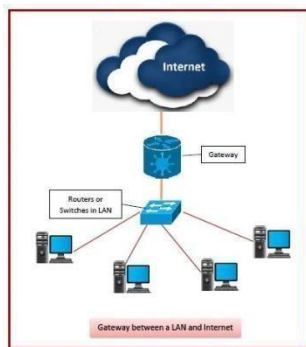
(D) Router

- A router is a layer 3 or network layer device
- It connects different networks together and sends data packets from one network to another.
- A router can be used both in LANs (Local Area Networks) and WANs (Wide Area Networks).



(E) Gateways

- Gateway is located at the boundary of a network and manages all data that inflows or outflows from that network.
- It forms a passage between two different networks operating with different transmission protocols.
- It also stores information about the routing paths of the communicating network.
- It uses packet switching techniques to transmit data across the network.



Practical-3

Explain Installation of LAN card and LAN cabling.

LAN Card

The installation of a LAN (Local Area Network) card, also known as a network interface card (NIC) or Ethernet card, involves adding a hardware component to a computer to enable it to connect to a LAN and communicate with other devices on the network. Here is a general overview of the steps involved in installing a LAN card:

- **Choose the right NIC:** Select a NIC that is compatible with your computer's expansion slot and the type of network you will be connecting to. NICs come in various form factors, such as PCI, PCI Express, or USB, and support different network speeds, such as 10/100 Mbps or Gigabit Ethernet.
- **Power off the computer:** Before installing any hardware component, it's important to power off the computer and unplug it from the electrical outlet to ensure safety.
- **Open the computer case:** Depending on the type of computer you have; you may need to open the computer case to access the expansion slots where the NIC will be installed. Follow the manufacturer's instructions or consult a technician if you are not familiar with the process.
- **Install the NIC:** Insert the NIC into an available expansion slot on the computer's motherboard, aligning it with the slot and firmly pushing it in until it's securely seated. If it's a PCI or PCI Express card, you may need to use a screwdriver to secure it with a screw to the computer case.
- **Connect the network cable:** Once the NIC is installed, connect the network cable from the LAN or Ethernet port on the NIC to the corresponding port on the network switch or router. Make sure it's securely plugged in and properly seated.
- **Close the computer case:** If you had to open the computer case, now is the time to close it and secure it with any screws or latches that were removed.
- **Power on the computer:** Plug the computer back into the electrical outlet and power it on. The operating system should automatically detect the new NIC and install any necessary drivers.

LAN cabling

LAN (Local Area Network) cabling, also known as Ethernet cabling, is the physical infrastructure that connects devices within a local area network. It is commonly used in homes, offices, and other small-scale networking environments.

- **Plan and design:** Before you start the installation, you need to plan and design the LAN cabling layout. This includes determining the locations of the devices that need to be connected, such as computers, switches, and routers, and mapping out the most efficient routes for the cabling.
- **Gather materials and tools:** Once you have a plan in place, you'll need to gather the necessary materials and tools for the installation. This may include Ethernet cables, connectors, patch panels, wall plates, cable management tools, a cable tester, and other accessories.
- **Prepare the installation area:** Next, you'll need to prepare the installation area. This involves clearing any obstructions, such as furniture or debris, from the paths where the cabling will be installed. You may also need to drill holes, install conduit or raceway, and create cable pathways to ensure a neat and organized installation.
- **Install the cabling:** With the area prepared, you can start installing the LAN cabling. This typically involves pulling the Ethernet cables through the designated routes, and securing them in place using clips, brackets, or cable ties. You'll need to properly terminate the cables by attaching connectors, such as RJ-45 connectors, to the ends of the cables using appropriate techniques, such as punch-down or crimping.
- **Test and certify the cabling:** After the cabling is installed, it's important to test and certify the connections to ensure they are functioning correctly. You can use a cable tester to verify that the cables are properly wired and that there are no continuity or performance issues. This helps ensure that your LAN cabling is reliable and capable of supporting the desired network speeds.
- **Label and document:** Finally, it's important to label and document the LAN cabling installation for future reference. You can label the cables and connectors to identify their purpose, such as "PC 1," "Switch Port 2," or "Patch Panel Port 10," and create a documentation record that includes the cabling layout, cable types, and other relevant information. This helps with troubleshooting, maintenance, and future expansions or modifications to the network.

Practical-4

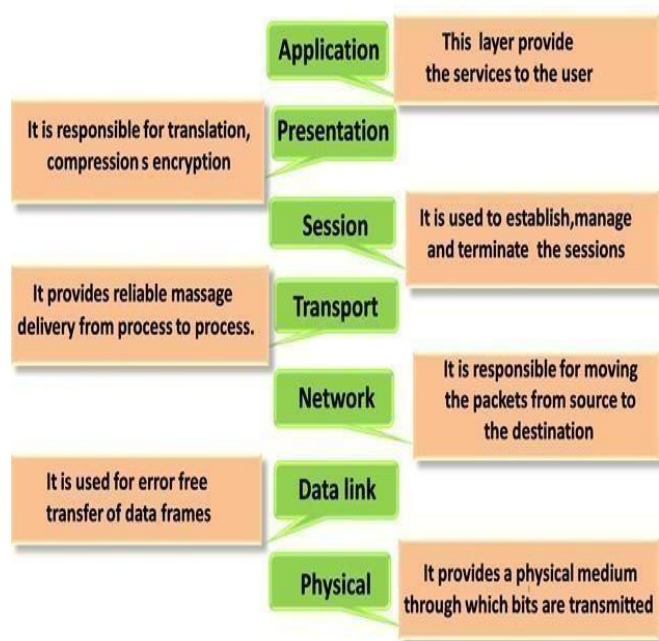
To study and implement the OSI reference model.

OSI Model:

- OSI stands for Open System Interconnection is a reference model that describes how information from a software application in one computer moves through a physical medium to the software application in another computer.
- OSI consists of seven layers, and each layer performs a particular network function.
- OSI model was developed by the International Organization for Standardization (ISO).
- There are the seven OSI layers. Each layer has different functions.

The list of seven layers is given below:

- (1) Physical Layer
- (2) Data-Link Layer
- (3) Network Layer
- (4) Transport Layer
- (5) Session Layer
- (6) Presentation Layer
- (7) Application Layer

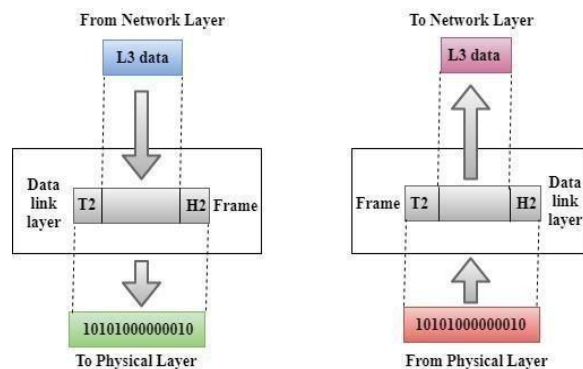


(1) Physical Layer

The physical layer is responsible for the physical cable or wireless connection between network nodes. It defines the connector, the electrical cable or wireless technology connecting the devices, and is responsible for transmission of the raw data is responsible for transmission of the raw data.

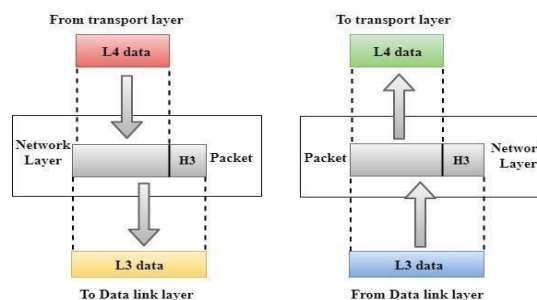
(2) Data Link Layer

The data link layer establishes and terminates a connection between two physically-connected nodes on a network. It breaks up packets into frames and sends them from source to destination. This layer is composed of two parts—Logical Link Control (LLC), which identifies network protocols, performs error checking, and synchronizes frames, and Media Access Control (MAC) which uses MAC addresses to connect devices and define permissions to transmit and receive data.



(3) Network Layer

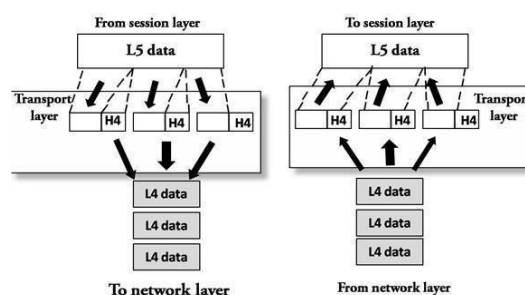
The network layer has two main functions. One is breaking up segments into network packets, and reassembling the packets on the receiving end. The other is routing packets by discovering the best path across a physical network. The network layer uses network addresses (typically IP addresses) to route packets to a destination node



(4) Transport Layer

The transport layer takes data transferred in the session layer and breaks it into “segments” on the transmitting end. It is responsible for reassembling the segments on the receiving end, turning it back into data that can be used by the session layer.

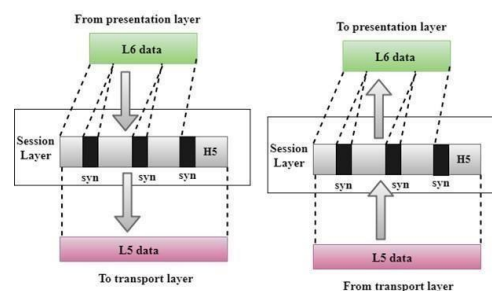
The transport layer carries out flow control, sending data at a rate that matches the connection speed of the receiving device, and error control, checking if data was received incorrectly and if not, requesting it again.



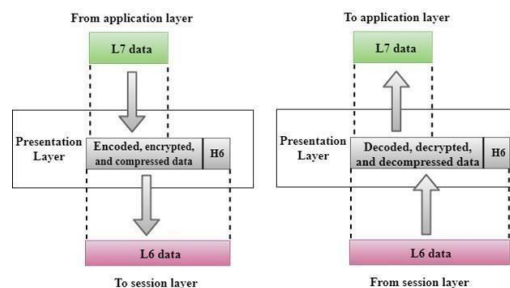
(5) Session Layer

The session layer creates communication channels, called sessions, between devices. It is responsible for opening sessions, ensuring they remain open and functional while data is being transferred, and closing them when communication ends. The

session layer can also set checkpoints during a data transfer—if the session is



interrupted, devices can resume data transfer for the last checkpoint.

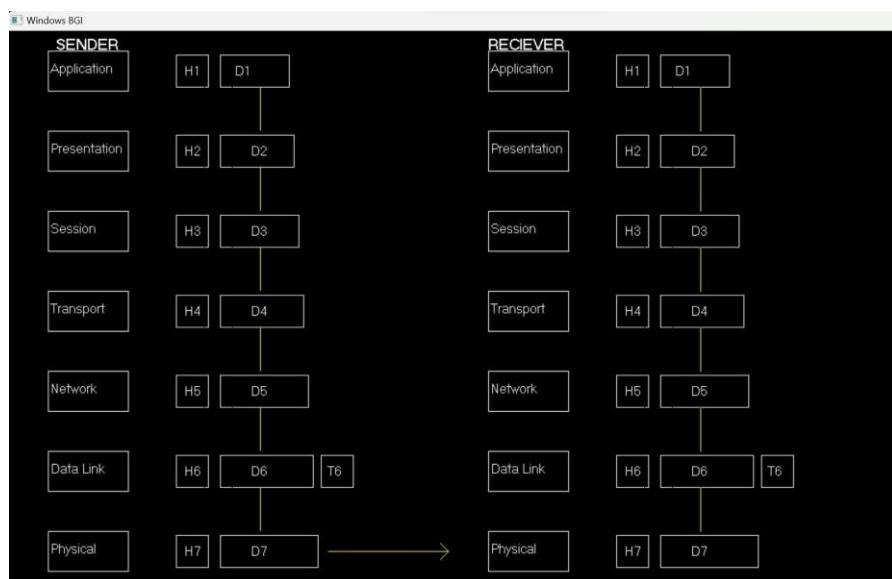
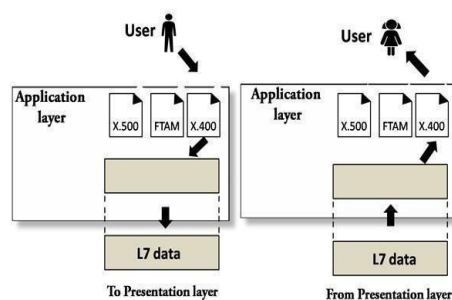


(6) Presentation Layer

The presentation layer prepares data for the application layer. It defines how two devices should encode, encrypt, and compress data so it is received correctly on the other end. The presentation layer takes any data transmitted by the application layer and prepares it for transmission over the session layer.

(7) Application Layer

The application layer is used by end-user software such as web browsers and email clients. It provides protocols that allow software to send and receive information and present meaningful data to users. A few examples of application layer protocols are the Hypertext Transfer Protocol (HTTP), File Transfer Protocol (FTP), Post Office Protocol (POP), Simple Mail Transfer Protocol (SMTP), and Domain Name System (DNS).



Practical-5

Implementation of Bit Stuffing and Byte/Character stuffing

Bit Stuffing

Code

```
#include <bits/stdc++.h>

using namespace std;

vector<int> bitStuffing(vector<int> arr){
    int i=0;
    int count=0;
    vector<int> ans;
    while(i< arr.size()){
        while(i< arr.size() && arr[i]==1 && count<5){
            ans.push_back(1);
            i++;
            count++;
        }
        if(count==5){
            ans.push_back(0);
        }
        if(arr[i]==0){
            ans.push_back(0);
            i++;
        }
        count=0;
    }
    return ans;
}

int main()
{
```

```

int n;
cout << "\nEnter the size of the array: ";
cin >> n;
vector<int> arr(n);
cout << "\nEnter the elements of the array: ";
for (int i = 0; i < n; i++)
{cin >> arr[i];}
cout << "\nElement Before Bit Stuffing: ";
for (int i = 0; i < n; i++)
{cout << arr[i] << " " ;}
cout << endl;
arr =bitStuffing(arr);
cout << "Elements After Bit Stuffing: ";
for (int i= 0; i< arr.size(); i++)
{cout << arr[i] << " " ;}
cout << endl;
return 0;
}

```

```

Enter the size of the array: 14

Enter the elements of the array: 1 0 0 1 1 1 1 1 0 1 1 0 0 1

Element Before Bit Stuffing: 1 0 0 1 1 1 1 1 0 1 1 0 0 0
Elements After Bit Stuffing: 1 0 0 1 1 1 1 1 0 1 0 1 1 0 0

```

Byte Stuffing

Code

```

#include <bits/stdc++.h>
using namespace std;

```

```

int main()
{
    cout << "Flag byte Sequence: f\n";
    cout << "Escape byte Sequence: e\n";
    char flag = 'f';
    char escape = 'e';
    string oM;
    getline(cin, oM);
    cout << "Message to be sent : ";
    string sM = "f";
    for (int i = 0; i < oM.length(); i++)
    {
        if (oM[i] == flag)
        {
            sM += escape;
            sM += flag;
        }
        else if (oM[i] == escape)
        {
            sM += escape;
            sM += escape;
        }
        else
        {
            sM += oM[i];
        }
    }
    sM += flag;
    cout << "\nencoded message : " << sM << endl
         << endl;
    string rM = "";
    for (int i = 1; i <= (sM.length() - 2); i++)
    {
        if(sM[i] == escape)

```

```

{
    if(sm[i + 1] == flag) continue;
    else
    {
        rM += escape;
        i++;
    }
}
else
{
    rM +=sm[i];
}
}
cout << "decoded message    s: " << rM << endl;
return 0;
}

```

Output

```

Flag byte Sequence: f
Escape byte Sequence: e
Netaji Subhas University Of Technology
Message to be sent :
encoded message  : fNeetaji Subhas Univeersity Oef Teechnologyf
decoded message s: Netaji Subhas University Of Technology

```


Practical-6

Implement various error detection technique like Parity check, CRC, and Checksum.

A. Parity Check

Code

```
#include<bits/stdc++.h>
using namespace std;

int main(){
    cout<<"Enter bit of data"<<endl;
    string data;
    cin>>data;

    int count=0;
    for(auto it: data){
        if(it=='1'){
            count++;
        }
    }
    if((count%2)==0){
        cout<<"Number of 1 are even. Hence the data is errorless"<<endl;
    }
    else{
        cout<<"Number of 1 are odd. Hence an error is detected"<<endl;
    }
    return 0;
}
```

Output

```
Enter bit of data
1010011
Number of 1 are even. Hence the data is errorless
```

B. CRC

Code

```
#include <bits/stdc++.h>
using namespace std;
string xor1(string a, string b)
{
    string result = "";

    int n = b.length();

    for (int i= 1; i< n;i++)
    {
        if (a[i] == b[i])
            result += "0";    else
            result += "1";
    }
    return result;
}

string mod2dv(string dividend, string divisor)
{
    int pick = divisor.length();
    string temp = dividend.substr(0, pick);
    int n = dividend.length();
    while (pick < n)
    {
        if (temp[0] == '1')
        {
            temp = xor1(divisor, temp) + dividend[pick];
        }
        else
        {

```

```

        temp = xor1(string(pick, '0'), temp) + dividend[pick];
    }
    pick += 1;
}
if (temp[0] == '1')
{
    temp = xor1(divisor, temp);
}
else
{
    temp = xor1(std::string(pick, '0'), temp);
}
return temp;
}

void encodeData(string data, string key)
{
    int l_key = key.length();
    string appended_data = (data + string(l_key - 1, '0'));
    string remainder = mod2dv(appended_data, key);
    string codeword = data + remainder;
    cout << "Remainder : " << remainder << "\n";
    cout << "Encoded Data (Data + Remainder) : "
        << codeword << "\n";
}

int main()
{
    string data, key;
    cout << "enter data: ";
    cin >> data;
    cout << "enter key: ";
    cin >> key;

```

```

    encodeData(data, key);
    return 0;
}

```

Output

```

enter data: 101011101010
enter key: 1010
Remainder : 010
Encoded Data (Data + Remainder) :101011101010010

```

C. CheckSum

Code

```

#include <bits/stdc++.h>
using namespace std;

string DecimalToBinary(int num)
{
    string str;
    while (num)
    {
        if (num & 1)
            str += '1';
        else
            str += '0';
        num >>= 1;
    }
    reverse(str.begin(), str.end());
    return str;
}

string complement(string a)
{
    string s = "";
    for (int i= 0; i< a.length(); i++)
    {

```

```

        if(a[i] == '0') s += '1';
        else s += '0';
    }
    return s;
}

string addBinary(string a, string b)
{
    int sum = stoi(a, 0, 2) + stoi(b, 0, 2);
    return DecimalToBinary(sum);
}

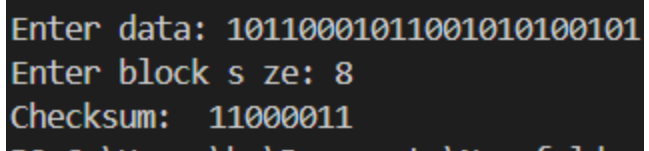
string checksum(string data, int block_size)
{
    string sum = "0";
    int i= 0;
    while(i< data.length())
    {
        string block = data.substr(i, block_size);
        sum = addBinary(sum, block);
        if(sum.length() > block_size)
        {
            sum = sum.erase(0, 1);
            sum = addBinary(sum, "1");
        }
        i+= block_size;
    }
    while(sum.length() < block_size) sum = "0" + sum;
    sum = complement(sum);
    return sum;
}

int main()
{
    string data;
    cout << "Enter data: ";

```

```
getline(cin, data);  
int block_size;  
cout << "Enter block size: ";  
cin >> block_size;  
string checksum_data = checksum(data, block_size);  
cout << "Checksum: " << checksum_data << endl;  
return 0;  
}
```

Output

A screenshot of a terminal window with a black background and white text. It shows the output of a C++ program. The first line is 'Enter data: 10110001011001010100101'. The second line is 'Enter block size: 8'. The third line is 'Checksum: 11000011'.

```
Enter data: 10110001011001010100101  
Enter block size: 8  
Checksum: 11000011
```

Practical-7

Implement Error Correction technique like Hamming code.

Practical-10

To study about IPV4, IPV6, default subnet mask, net id, hostid and classful addressing.

IPv4

IP stands for Internet Protocol and v4 stands for Version Four (IPv4). IPv4 was the primary version brought into action for production within the ARPANET in 1983.

IP version four addresses are 32-bit integers which will be expressed in decimal notation.

Example- 192.0.2.126 could be an IPv4 address.

Parts of ipv4 Networkpart:

- The network part indicates the distinctive variety that's appointed to the network. The network part conjointly identifies the category of the network that's assigned.

Host Part:

- The host part uniquely identifies the machine on your network. This part of the IPv4 address is assigned to every host. For each host on the network, the network part is the same, however, the host half must vary.

Subnet number:

- This is the nonobligatory part of IPv4. Local networks that have massive numbers of hosts are divided into subnets and subnet numbers are appointed to that.

Characteristics of IPv4

- IPv4 could be a 32-Bit IP Address.
- IPv4 could be a numeric address, and its bits are separated by a dot.
- The number of header fields is twelve and the length of the header field is twenty.
- It has Unicast, broadcast, and multicast style of addresses.
- IPv4 supports VLSM (Variable Length Subnet Mask).
- IPv4 uses the Post Address Resolution Protocol to map to the MAC address.
- RIP may be a routing protocol supported by the routed daemon.
- Networks ought to be designed either manually or with DHCP.
- Packet fragmentation permits from routers and causing host.

Advantages of IPv4

- IPv4 security permits encryption to keep up privacy and security.
- IPV4 network allocation is significant and presently has quite 85000 practical routers.
- It becomes easy to attach multiple devices across an outsized network while not NAT.
- This is a model of communication so provides quality service also as economical knowledge transfer.
- IPV4 addresses are redefined and permit flawless encoding.
- Routing is a lot of scalable and economical as a result of addressing is collective more effectively.

Limitations of IPv4

- IP relies on network layer addresses to identify end-points on network, and each network has a unique IP address.
- The world's supply of unique IP addresses is dwindling, and they might eventually run out theoretically.
- If there are multiple host, we need IP addresses of next class.
- Complex host and routing configuration, non-hierarchical addressing, difficult to re-numbering addresses, large routing tables, non-trivial implementations in providing security, QoS (Quality of Service), mobility and multi-homing, multicasting etc. are the big limitation of IPv4 so that's why IPv6 came into the picture.

IPv6

IP v6 was developed by Internet Engineering Task Force (IETF) to deal with the problem of IP v4 exhaustion. IP v6 is a 128-bits address having an address space of 2^{128} which is way bigger than IPv4. In IPv6 we use Colon-Hexa representation. There are 8 groups and each group represents 2 Bytes. The Main reason of IPv6 was the address depletion. Other reasons are related to the slowness of the process due to same unnecessary processing, the need for new options, support for multimedia, and the desperate need for security. IPv6 protocol responds to the above issues using the following main changes in the protocol:

(1) Large address space:

An IPv6 address is 128 bits long compared with the 32 bit address of IPv4, this is a huge (2 raised 96 times) increases in the address space.

(2) Better header format:

IPv6 uses a new header format in which options are separated from the base header and inserted, when needed, between the base header and the upper layer

data. This simplifies and speeds up the routing process because most of the options do not need to be checked by routers.

(3) New options:

IPv6 has new options to allow for additional functionalities.

(1) Allowance for extension:

IPv6 is designed to allow the extension of the protocol if required by new technologies or applications.

(2) Support for more security:

The encryption and authentication options in IPv6 provide confidentiality and integrity of the packet.

In IPv6 representation, we have three addressing methods:

- Unicast
- Multicast
- Anycast

(1) Unicast Address –

Unicast Address identifies a single network interface. A packet sent to a unicast address is delivered to the interface identified by that address.

(2) Multicast Address –

Multicast Address is used by multiple hosts, called as Group, acquires a multicast destination address. These hosts need not be geographically together. If any packet is sent to this multicast address, it will be distributed to all interfaces corresponding to that multicast address.

(3) Anycast Address –

Anycast Address is assigned to a group of interfaces. Any packet sent to an anycast address will be delivered to only one member interface (mostly nearest host possible).

Note: Broadcast is not defined in IPv6.

Subnet mask:

A subnet mask is a 32-bit number created by setting host bits to all 0s and setting network bits to all 1s. In this way, the subnet mask separates the IP address into the network and host addresses.

The “255” address is always assigned to a broadcast address, and the “0” address is always assigned to a network address. Neither can be assigned to hosts, as they are reserved for these special purposes.

The IP address, subnet mask and gateway or router comprise an underlying structure—the Internet Protocol—that most networks use to facilitate inter-device communication.

When organizations need additional subnetworking, subnetting divides the host element of the IP address further into a subnet. The goal of subnet masks are simply to enable the subnetting process. The phrase “mask” is applied because the subnet mask essentially uses its own 32-bit number to mask the IP address.

Class A, B, and C networks have natural masks, or default subnet masks:

- Class A: 255.0.0.0
- Class B: 255.255.0.0
- Class C: 255.255.255.0

You can determine the number and type of IP addresses any given local network requires based on its default subnet mask.

Network ID:

A network ID or NetID is the fragment of IP address that classifies the network for a specified host i.e., it tells us which network the host belongs to, generally comprised of one to up to four octets in dotted-decimal representation.

In dotted-decimal representation, an IP address is divided into four octets, and based on which class the IP address belongs to its octets are further divided into network ID and HOST ID.

For Class A first octet represents network ID as the prefix of the first octet is 0, it uses the remaining 7 bits for network ID, for Class B first and second octets represent network ID the prefix for class B is 10 so it uses the remaining 14 bits for network ID, for Class C first, second and third octet represents network ID the prefix of class C is 110 so it uses the remaining 21 bits for network ID, Class D, and Class E are reserved.

Host ID:

It is the fragment of an IP address that uniquely classifies a host on a specified TCP/IP network. A host ID can be found simply by ANDing the IP address in binary form with its respective default subnet mask (in binary form). The other fragment of an IP address is the network ID, which identifies the network to which the host belongs.

Classful Addressing :

The 32 bit IP address is divided into five sub-classes. These are:

- Class A
- Class B
- Class C
- Class D
- Class E

Each of these classes has a valid range of IP addresses. Classes D and E are reserved for multicast and experimental purposes respectively. The order of bits in the first octet determine the classes of IP address.

IPv4 address is divided into two parts:

- Network ID
- Host ID

The class of IP address is used to determine the bits used for network ID and host ID and the number of total networks and hosts possible in that particular class. Each ISP or network administrator assigns IP address to each device that is connected to its network.

Practical-11

To study about installation and configuration of printers.

The device printer and scanners use the same process for installation and configuration. It is a step-by-step process as given below:

Step 1: Attach the device using a local or network port and connect the power

The device is to be set up on a flat stable surface first after which it is to be connected to host computer with its power off or to the network. Once this is done, connect power to it using power adapter that comes with it or with A/C power cord if the device has built-in power supply and finally the device is to be turned on.

Step 2: Install and Update the Device driver and calibrate the device

Boot up the computer and wait for it to recognize the device and the wizard that appears helps in the configuration process of driver for printer/scanner or CD-ROM that comes with the device can be inserted which guides through the device driver installation procedure.

The device functions once the driver is installed but some devices such as inkjet printers or scanners require to calibrate the device.

Calibration is the process by which a device is brought within functional specifications.

Step 3: Configure options and default settings

Once the driver is installed, default settings and options for printers and scanners can be configured as required.

Step 4: Print/scan a test page

Once all the steps above are completed, can test the output of printer/scanner by printing a test page. Windows has a built-in function for doing this.