|  |  |  |  |
| --- | --- | --- | --- |
| **Name of Student:** | | | **Class:** |
| **Enrollment No:** | | | **Batch:** |
| **Date of Experiment:** | **Date of Submission:** | | **Submitted on:** |
| **Remarks by faculty:** | | | **Grade:** |
| **Signature of student:** | | **Signature of Faculty:** | |

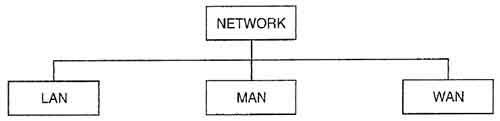
**EXPERIMENT 1:** Study of different types of networks and networking devices

**INTRODUCTION :**

**TYPES OF NETWORK**

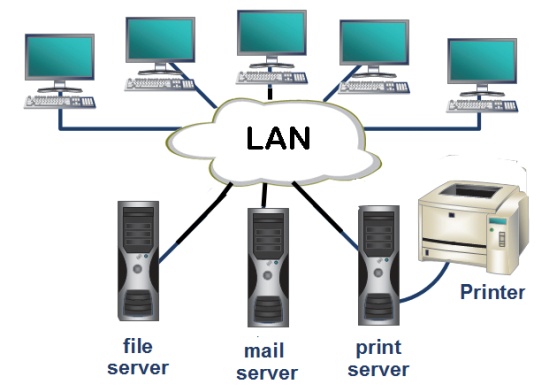
In the today world, Two devices are in network if a process in one device is able to exchange [information](http://ecomputernotes.com/fundamental/information-technology/what-do-you-mean-by-data-and-information) with a process in another device. Networks are known as a medium of connections between nodes (set of devices) or computers. A network is consist of group of [computer](http://ecomputernotes.com/fundamental/introduction-to-computer/what-is-computer) systems, servers, networking devices are linked together to share resources, including a [printer](http://ecomputernotes.com/fundamental/input-output-and-memory/what-is-a-printer-and-what-are-the-different-types-of-printers) or a file server. The connections is established by using either cable media or wireless media. The Network allows computers to connect and communicate with different computers via any medium.

LAN, MAN and WAN are the three major types of the network designed to operate over the area they cover. There are some similarities and dissimilarities between them. One of the major differences is the geographical area they cover, i.e. LAN covers the smallest area; MAN covers an area larger than LAN and WAN comprises the largest of all.



**1**.**Local Area Network (LAN)**

It is also called LAN and designed for small physical areas such as an office, group of buildings or a factory. LANs are used widely as it is easy to design and to troubleshoot. Personal computers and workstations are connected to each other through LANs. We can use different types of topologies through LAN, these are Star, Ring, Bus, Tree etc. LAN can be a simple network like connecting two computers, to share files and network among each other while it can also be as complex as interconnecting an entire building. LAN networks are also widely used to share resources like printers, shared hard-drive etc.

****

**Advantages of LAN**

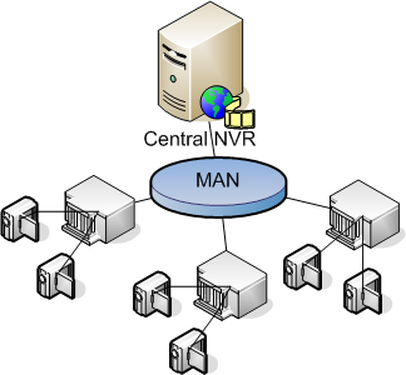
* Resource Sharing: Computer resources like printers, modems, DVD-ROM drives and hard disks can be shared with the help of local area networks. This reduces cost and hardware purchases.
* Software Applications Sharing: It is cheaper to use same software over network instead of purchasing separate licensed software for each client a network.

**Disadvantages of LAN**

* High Setup Cost: Although the LAN will save cost over time due to shared computer resources, but the initial setup costs of installing Local Area Networks is high.
* Privacy Violations: The LAN administrator has the rights to check personal data files of each and every LAN user. Moreover he can check the internet history and computer use history of the LAN user.

**2. Metropolitan Area Network (MAN)**

It was developed in 1980s.It is basically a bigger version of LAN. It is also called MAN and uses the similar technology as LAN. It is designed to extend over the entire city. It can be means to connecting a number of LANs into a larger network or it can be a single cable. It is mainly hold and operated by single private company or a public company.

****

**Advantages of MAN**

* Extremely efficient and provide fast communication via high-speed carriers, such as fibre optic cables.
* It provides a good back bone for large network and provides greater access to WANs.

**Disadvantages of MAN**

* More cable required for a MAN connection from one place to another.
* It is difficult to make the system secure from hackers and industrial espionage(spying) graphical regions.

**3. Wide Area Network (WAN)**

It is also called WAN. WAN can be private or it can be public leased network. It is used for the network that covers large distance such as cover states of a country. It is not easy to design and maintain. Communication medium used by WAN are PSTN or Satellite links. WAN operates on low data rates.

****

**Advantages of WAN**

* Covers a large geographical area so long distance business can connect on the one network.
* Shares software and resources with connecting workstations.

**Disadvantages of WAN**

* Need a good firewall to restrict outsiders from entering and disrupting the network.
* Setting up a network can be an expensive, slow and complicated. The bigger the network the more expensive it is.

**NETWORKING DEVICES**

**1. NIC**

A network interface card (NIC) is a circuit board or card that is installed in a computer so that it can be connected to a network.

A network interface card provides the computer with a dedicated, full-time connection to a network. Personal computers and workstations on a local area network (LAN) typically contain a network interface card specifically designed for the LAN transmission technology

The role of a network interface card is to allow a computer toconnect to the network via an Ethernet cable. The computer can thencommunicate with other computers and devices on the network.

****

**Advantages:**

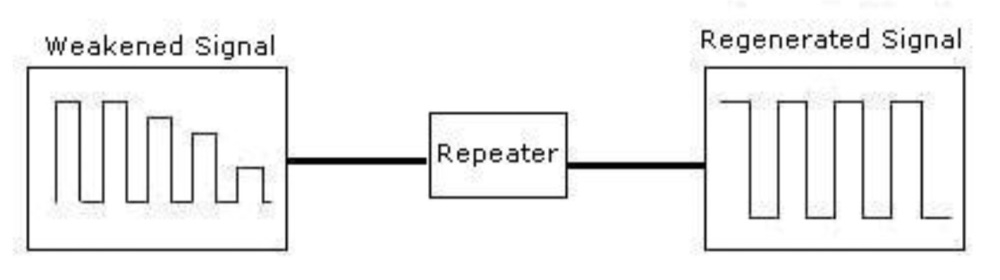
* Network Interface Cards are very cheap to purchase and install.
* Newer Network Interface Cards can reach speeds of up to 2000 Mbps.
* Stable connection between device and cable.

**Disadvantages:**

* Proper configuration is needed in order for the device to work correctly
* Network Interface Cards can be unsafe and data is not secured.
* Replacement need to be replaced in order to handle better speeds.

**2. Repeater**

A Repeater is the simplest facility used for network interconnection.The major function is to receive a network signal from one LAN terminal cable segment and to regenerate and retransmit the signal as it is in its original strength over a one or more other cable segment.



**Advantages:**

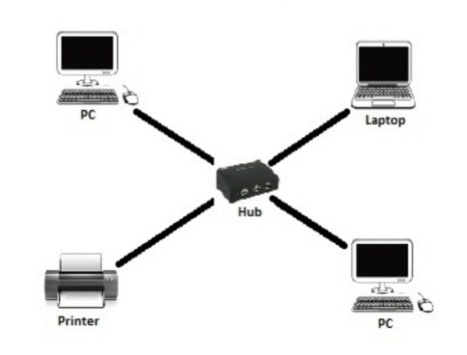
* Repeaters are very simple to connect.
* It is Cost effective
* Repeaters has ability to strengthen signal the weak signal.

**Disadvantages :**

* Repeaters cannot connect different network architectures.
* Repeaters do not reduce network traffic.
* The number of repeaters must be limited.

**3. Hub**

A hub is basically a multiport repeater. A hub connects multiple wires coming from different branches Hubs cannot filter data, so data packets are sent to all connected devices. In other words, collision domain of all hosts connected through Hub remains one. Also, they do not have intelligence to find out best path for data packets which leads to inefficiencies and wastage.

Hub also works on the Data link layer of OSI model.

**Advantages:**

* It can extend total distance of the network.
* It does not affect performance of the network seriously.
* It is cheaper.
* It can connect different media types. .

**Disadvantages :**

* It does not have mechanisms such as collision detection and retransmission of packets.
* It does not operate in full duplex mode.
* It can not connect different network architectures such as token ring and ethernet etc.

**4. Bridge**

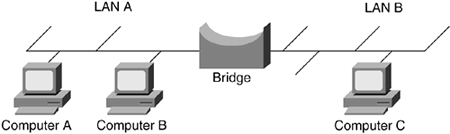
The Bridge is a networking device which is used to divides a LAN into multiple segments. Bridge basically works in a bus topology. A bridge operates at data link layer of OSI model.A bridge is a repeater, with add on functionality of filtering content by reading the MAC addresses of source and destination. It is also used for interconnecting two LANs working on the same protocol. It has a single input and single output port, thus making it a 2 port device

**Advantages:**

* Bridges can extend a network by acting as a repeater
* Bridges can reduce network traffic on a segment by subdividing network communications
* It reduces collision.

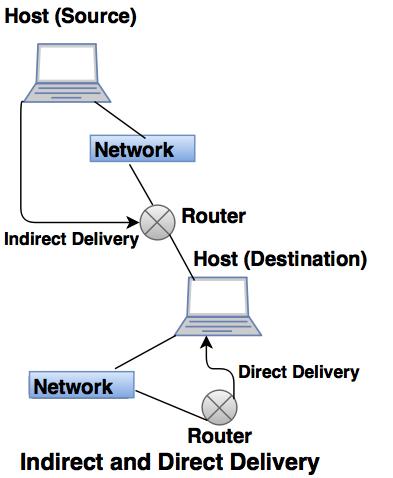
**Disadvantages :**

* It is slower compare to repeaters due to filtering.
* It does not filter broadcasts.
* Bridges are more expensive as compare to repeaters.

****

**5. Router**

A router is a device like a switch that routes data packets based on their IP addresses. Router is mainly a Network Layer device. Routers normally connect LANs and WANs together and have a dynamically updating routing table based on which they make decisions on routing the data packets. Router divide broadcast domains of hosts connected through it.



**Advantages :**

* It provides connection between different network architectures such as ethernet & token ring etc.
* It can choose best path across the internetwork using dynamic routing algorithms.
* It provides sophisticated routing, flow control and traffic isolation.

**Disadvantages :**

* They operate based on routable network protocols.
* They are expensive compare to other network devices.

**6. Gateway**

A gateway, as the name suggests, is a passage to connect two networks together that may work upon different networking models. They basically works as the messenger agents that take data from one system, interpret it, and transfer it to another system. Gateways are also called protocol converters and can operate at any network layer. Gateways are generally more complex than switch or router.

**Advantages :**

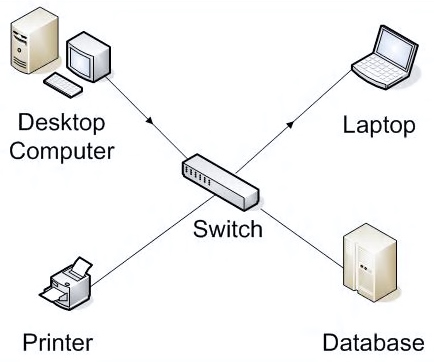
* Direct connections between internal and external hosts are disallowed.
* User-level authentication is supported.
* The application commands are analyzed inside the payload portion of the data packets.

**Disadvantages :**

* Slower than packet filters
* Needs the internal client to know about them.
* Every possible type of connection can not be supported

**7. Switch**

A switch is a multi port bridge with a buffer and a design that can boost its efficiency(large number of  ports imply less traffic) and performance. Switch is data link layer device. Switch can perform error checking before forwarding data, that makes it very efficient as it does not forward packets that have errors and  forward good packets selectively to correct port only.  In other words, switch divides collision domain of hosts, but [broadcast domain](https://en.wikipedia.org/wiki/Broadcast_domain) remains same.



**Advantages :**

* High Efficiency
* Low thermal dissipation
* Can handle a large output current

**Disadvantages :**

* Complicated design
* High parts count
* Cost Factor

**Conclusion –**  
There are many advantages of LAN over MAN and WAN, such as LAN’s provide excellent reliability, high data transmission rate, they can easily be managed, and shares peripheral devices too. Local Area Network cannot cover cities or towns and for that Metropolitan Area Network is needed, which can connect city or a group of cities together. Further, for connecting Country or a group of Countries one requires Wide Area Network.

|  |  |  |  |
| --- | --- | --- | --- |
| **Name of Student:** | | | **Class:** |
| **Enrollment No:** | | | **Batch:** |
| **Date of Experiment:** | **Date of Submission:** | | **Submitted on:** |
| **Remarks by faculty:** | | | **Grade:** |
| **Signature of student:** | | **Signature of Faculty:** | |

**EXPERIMENT 2:** Study of different types of network topologies

**INTRODUCTION :**

**NETWORK TOPOLOGIES:-**

The arrangement of a network which comprises of nodes and connecting lines via sender and receiver is referred as network topology.

**1. Mesh Topology:-**

In mesh topology, every device is connected to another device via particular channel.

Every device is connected with another via dedicated channels. These channels are known as links.If suppose, N number of devices are connected with each other in mesh topology, then total number of ports that is required by each device is ​ N-1. In the Figure 1, there are 5 devices connected to each other, hence total number of ports required is 4.

If suppose, N number of devices are connected with each other in mesh topology, then total number of dedicated links required to connect them is NC2 i.e. N(N-1)/2. In the Figure 1, there are 5 devices connected to each other, hence total number of links required is 5\*4/2 = 10



**Advantages :**

* It is robust.
* Fault is diagnosed easily. Data is reliable because data is transferred among the devices through dedicated channels or links.
* Provides security and privacy.

**Disadvantages :**

* Installation and configuration is difficult.
* Cost of cables are high as bulk wiring is required, hence suitable for less number of devices.
* Cost of maintenance is high.

**2. Star Topology:-**

In star topology, all the devices are connected to a single hub through a cable. This hub is the central node and all others nodes are connected to the central node. The hub can be passive ​in nature i.e. not intelligent hub such as broadcasting devices, at the same time the hub can be intelligent known as active ​hubs. Active hubs have repeaters in them.



**Advantages :**

* If N devices are connected to each other in star topology, then the number of cables required to connect them is N. So, it is easy to set up.
* Each device require only 1 port i.e. to connect to the hub.

**Disadvantages :**

* If the concentrator (hub) on which the whole topology relies fails, the whole system will crash down.
* Cost of installation is high.
* Performance is based on the single concentrator i.e. hub.

**3. Bus Topology:-**

​ Bus topology is a network type in which every computer and network device is connected to single cable. It transmits the data from one end to another in single direction. No bi-directional feature is in bus topology.



**Advantages :**

* If N devices are connected to each other in bus topology, then the number of cables required to connect them is 1 ​which is known as backbone cable and N drop lines are required.
* Cost of the cable is less as compared to other topology, but it is used to built small networks.

**Disadvantages :**

* If the common cable fails, then the whole system will crash down.
* If the network traffic is heavy, it increases collisions in the network. To avoid this, various protocols are used in MAC layer known as Pure Aloha, Slotted Aloha, CSMA/CD etc.

**4. Ring Topology:-**

In this topology, it forms a ring connecting a devices with its exactly two neighbouring devices.  
The following operations takes place in ring topology are :

* One station is known as monitor station which takes all the responsibility to perform the operations.
* To transmit the data, station has to hold the token. After the transmission is done, the token is to be released for other stations to use.
* When no station is transmitting the data, then the token will circulate in the ring.
* There are two types of token release techniques : Early token release releases the token just after the transmitting the data and Delay token release releases the token after the acknowledgement is received from the receiver.



**Advantages :**

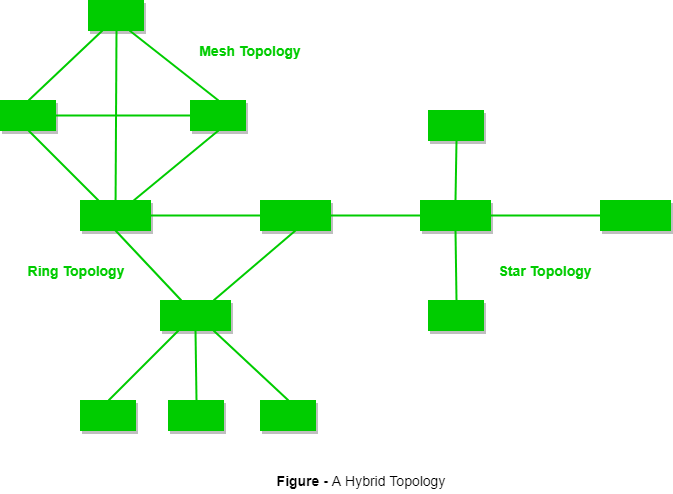
* The possibility of collision is minimum in this type of topology.
* Cheap to install and expand.

**Disadvantages :**

* Troubleshooting is difficult in this topology.
* Addition of stations in between or removal of stations can disturb the whole topology.

**5. Hybrid Topology:-**

This topology is a collection of two or more topologies which are described above. This is a scalable topology which can be expanded easily. It is reliable one but at the same it is a costly topology.

****

**Advantages :**

* Very reliable and easy to detect the Faulty system.
* Troubleshooting is easy.
* It includes both wired and wireless network.
* It is an expandable network.

**Disadvantages :**

* Its design is difficult to understand.
* The [cost](http://www.orosk.com/what-is-cost/) of this network design is high, because of the requirement of a [lot](http://www.orosk.com/what-is-lottocoin/) of [cables](http://www.orosk.com/cables/), cooling systems, etc.

**Conclusion –**  
There are many advantages of the various network topologies and they are widely used today.

|  |  |  |  |
| --- | --- | --- | --- |
| **Name of Student:** | | | **Class:** |
| **Enrollment No:** | | | **Batch:** |
| **Date of Experiment:** | **Date of Submission:** | | **Submitted on:** |
| **Remarks by faculty:** | | | **Grade:** |
| **Signature of student:** | | **Signature of Faculty:** | |

**EXPERIMENT 3:** Study of LAN INSTALLATION AND CONFIGURATION

**INTRODUCTION :**

**RJ 45**

RJ45 is a type of connector commonly used for Ethernet networking. It looks similar to a telephone jack, but is slightly wider. Since Ethernet cables have an RJ45 connector on each end, Ethernet cables are sometimes also called RJ45 cables.

The "RJ" in RJ45 stands for "registered jack," since it is a standardized networking interface.

Each RJ45 connector has eight pins, which means an RJ45 cable contains eight separate wires. If you look closely at the end of an Ethernet cable, you can actually see the eight wires, which are each a different color. Four of them are solid colors, while the other four are striped.

RJ45 cables can be wired in two different ways. One version is called T-568A and the other is T-568B



**Crossover Cable**

A crossover cable is a cable that is used to interconnect two computers by "crossing over" (reversing) their respective pin contacts. Either an [RS-232C](https://searchnetworking.techtarget.com/definition/RS-232C) or a telephone jack connection is possible. A crossover cable is sometimes known as a [null modem](https://searchnetworking.techtarget.com/definition/null-modem).

**Cables Categories**

There are many Ethernet cables that can be bought. Often these cables are supplied free with equipment that uses Ethernet connectivity in some way or another. There are several different varieties of Ethernet cable that can be obtained: speed variations, crossover cables, Cat 5, Cat 5e, Cat6, etc.. The commonly used cables: Cat 5, Cat 5e, Cat 6 all have different levels of performance, and therefore to is necessary to buy or select the right cable for the right application.

Flat Ethernet cable and connector



**Cat-6:**    This cable is defined in TIA/EIA-568-B provides a significant improvement in performance over Cat5 and Cat 5e. During manufacture Cat 6 cables are more tightly wound than either Cat 5 or Cat 5e and they often have an outer foil or braided shielding. The shielding protects the twisted pairs of wires inside the Ethernet cable, helping to prevent crosstalk and noise interference. Cat-6 cables can technically support speeds up to 10 Gbps, but can only do so for up to 55 metres.

| **CATEGORY** | **SHIELDING** | **MAX TRANSMISSION SPEED (AT 100 METERS)** | **MAX BANDWIDTH** |
| --- | --- | --- | --- |
| Cat 3 | Unshielded | 10 Mbps | 16 MHz |
| Cat 5 | Unshielded | 10/100 Mbps | 100 MHz |
| Cat 5e | Unshielded | 1000 Mbps / 1 Gbps | 100 MHz |
| Cat 6 | Shielded or Unshielded | 1000 Mbps / 1 Gbps | >250 MHz |
| Cat 6a | Shielded | 10000 Mbps / 10 Gbps | 500 MHz |
| Cat 7 | Shielded | 10000 Mbps / 10 Gbps | 600 MHz |

**Perform the following steps as directed**

**Step 1:** To make a Direct Cable connection

1. Click **Start**, click **Control Panel**, and then double-click **Network Connections**.

2. Under **Network Tasks**, click **Create a new connection**, and then click **next**.

3. Click **Set up an advanced connection**, and then click **next**.

4. Click **Connect directly to another computer**, and click **next**.

5. Choose the role this machine will play in the communication. If this computer has the information to which you need to gain access, click **Host**. If this computer will access information from the other computer, click **Guest**.

**Step 2:** To Set Up the Host Computer

1. Click the connection device that you want to use for this connection (a parallel or serial port, or an infrared port), and then click **Next**.

2. Grant access to the users who are allowed to connect by selecting the appropriate check boxes, and then click **Next**.

3. Click **Finish** to end the configuration process.

**Step 3:** to Set up the Guest Computer

1. Type a name to identify this connection, and then click **Next**.

2. Click the connection device that you want to use for this connection (a parallel or serial port, or an infrared port), and then click **Next**.

3. Decide whether this connection will be available for all users (click **Anyone's use**), or only for you (click **my use only**), and then click **Next**.

4. Click **Finish** to end the setup process

**Step 4:** To create Windows Workgroup

1. In Windows XP, right click on **My Computer**, select **System Properties**.

2. Select the **Computer Name** tab, click on **Change**.

3. Enter the appropriate **Computer name** and **Workgroup**.

4. Make sure that every computer on your home network references the same workgroup.

**Step 5:**

To Configure TCP/IP

To assign IP address, gateway, subnet mask, DNS

**Step 6:**

To create domain Bring all the PC of Lab under a network using workgroup or domain.

Create client and server

**Conclusions**

Windows workgroup is established and used for sharing and transferring data between physically connected PCs.

|  |  |  |  |
| --- | --- | --- | --- |
| **Name of Student:** | | | **Class:** |
| **Enrollment No:** | | | **Batch:** |
| **Date of Experiment:** | **Date of Submission:** | | **Submitted on:** |
| **Remarks by faculty:** | | | **Grade:** |
| **Signature of student:** | | **Signature of Faculty:** | |

**EXPERIMENT 4:** WRITE A PROGRAM TO IMPLEMENT ERROR CORRECTING TECHNIQUE HAMMING CODE

**SOURCE CODE:**

#include<iostream>

using namespace std;

int main() {

int data[10];

int dataatrec[10],c,c1,c2,c3,i;

cout<<"Enter 4 bits of data one by one\n";

cin>>data[0];

cin>>data[1];

cin>>data[2];

cin>>data[4];

data[6]=data[0]^data[2]^data[4];

data[5]=data[0]^data[1]^data[4];

data[3]=data[0]^data[1]^data[2];

cout<<"\nEncoded data is\n";

for(i=0;i<7;i++)

cout<<data[i];

cout<<"\n\nEnter received data bits one by one\n";

for(i=0;i<7;i++)

cin>>dataatrec[i];

c1=dataatrec[6]^dataatrec[4]^dataatrec[2]^dataatrec[0];

c2=dataatrec[5]^dataatrec[4]^dataatrec[1]^dataatrec[0];

c3=dataatrec[3]^dataatrec[2]^dataatrec[1]^dataatrec[0];

c=c3\*4+c2\*2+c1 ;

if(c==0) {

cout<<"\nNo error while transmission of data\n";

}

else {

cout<<"\nError on position "<<c;

cout<<"\nData sent : ";

for(i=0;i<7;i++)

cout<<data[i];

cout<<"\nData received : ";

for(i=0;i<7;i++)

cout<<dataatrec[i];

cout<<"\nCorrect message is\n";

if(dataatrec[7-c]==0)

dataatrec[7-c]=1;

else

dataatrec[7-c]=0;

for (i=0;i<7;i++) {

cout<<dataatrec[i];

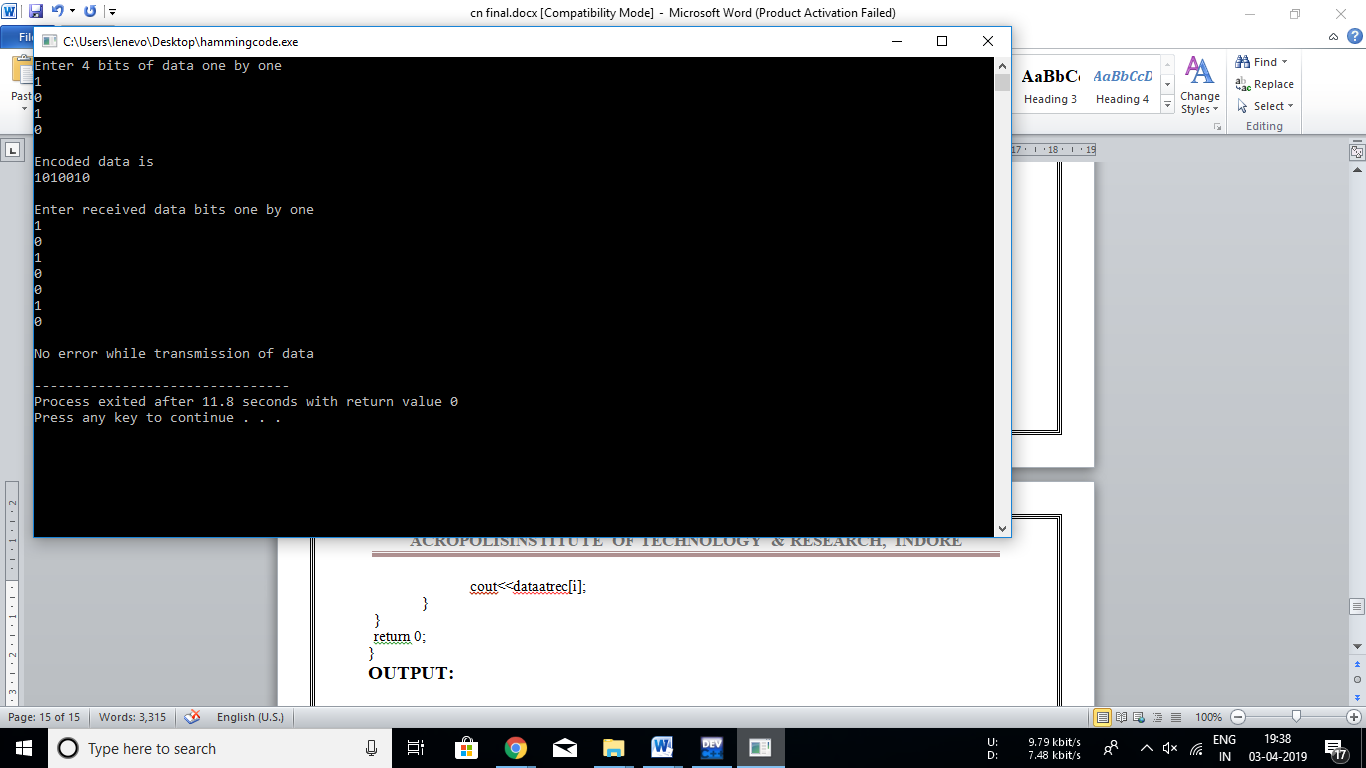
}

}

return 0;

}

**OUTPUT:**



|  |  |  |  |
| --- | --- | --- | --- |
| **Name of Student:** | | | **Class:** |
| **Enrollment No:** | | | **Batch:** |
| **Date of Experiment:** | **Date of Submission:** | | **Submitted on:** |
| **Remarks by faculty:** | | | **Grade:** |
| **Signature of student:** | | **Signature of Faculty:** | |

**EXPERIMENT 5:** WRITE A PROGRAM TO IMPLEMENT TYPES OF FRAMING METHOD

**SOURCE CODE:**

**PROGRAM FOR BIT STUFFING**

#include<iostream>

#include<string.h>

using namespace std;

int main()

{

int a[20],b[30],i,j,k,count,n;

cout<<"Enter frame size : ";

cin>>n;

cout<<"Enter the frame in the form of 0 and 1 :";

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

cin>>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++;

}

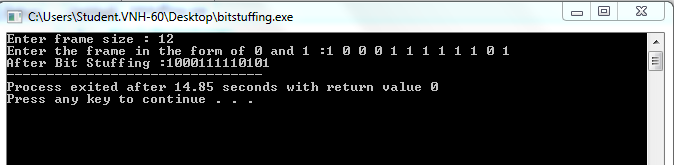
printf("After Bit Stuffing :");

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

cout<<b[i];

}

**OUTPUT:**



**PROGRAM FOR CHARACTER STUFFING**

#include<iostream>

using namespace std;

int main()

{

int a[20],b[30],i,j,k,count,n;

cout<<"Enter frame length:";

cin>>n;

cout<<"Enter input frame :";

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

cin>>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++;

}

cout<<"After stuffing the frame is:";

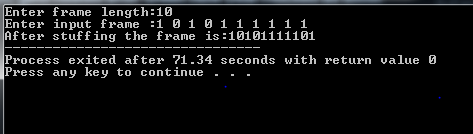
for(i=0;i<j;i++)

cout<<b[i];

return 0;

}

**OUTPUT:**



**PROGRAM FOR CHARACTER COUNT**

#include <iostream>

#include<string.h>

using namespace std;

int main()

{

char str[100];

char a;

cout<<"Enter any text: ";

gets(str);

int count = 0;

for (int i=0; str[i]!='\0';i++)

++ count;

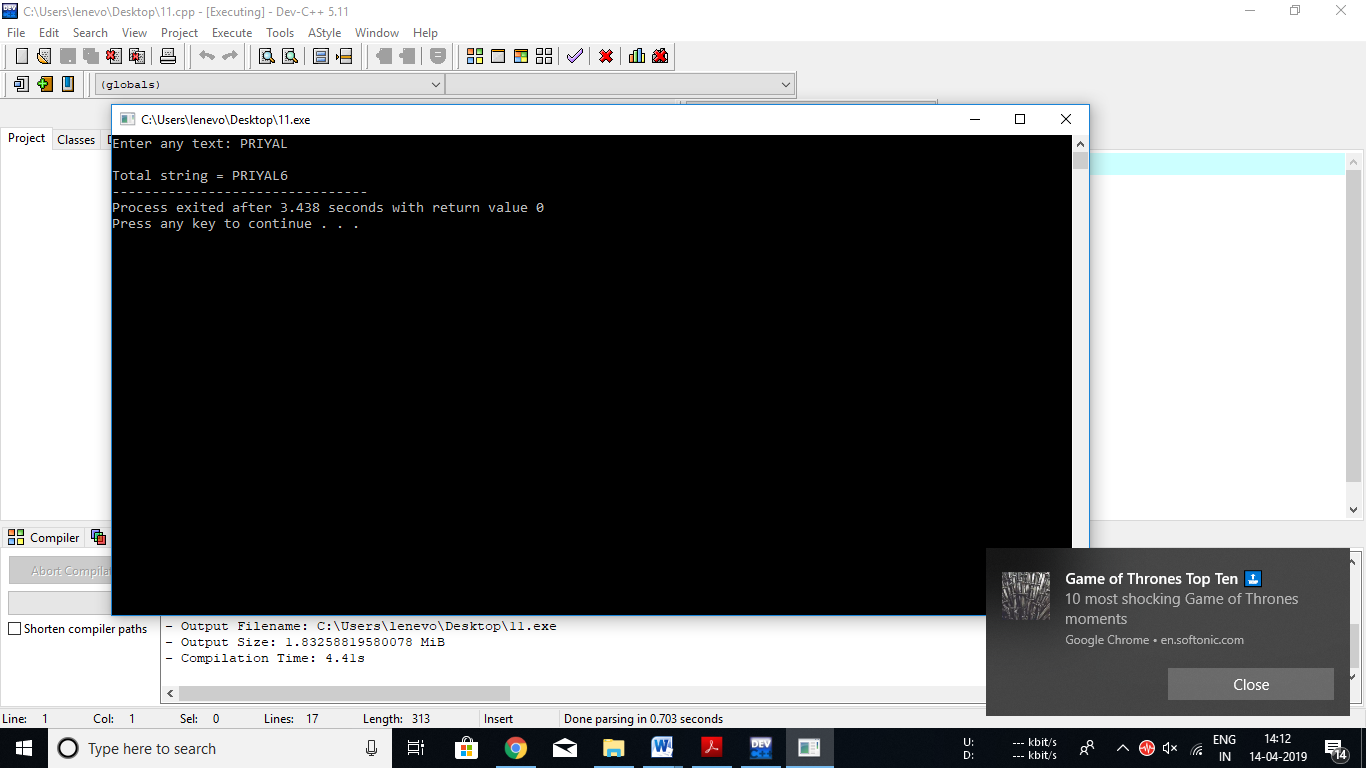
cout<<"\n";

cout<<"Total string = " <<count;

return 0;

}

**OUTPUT:**

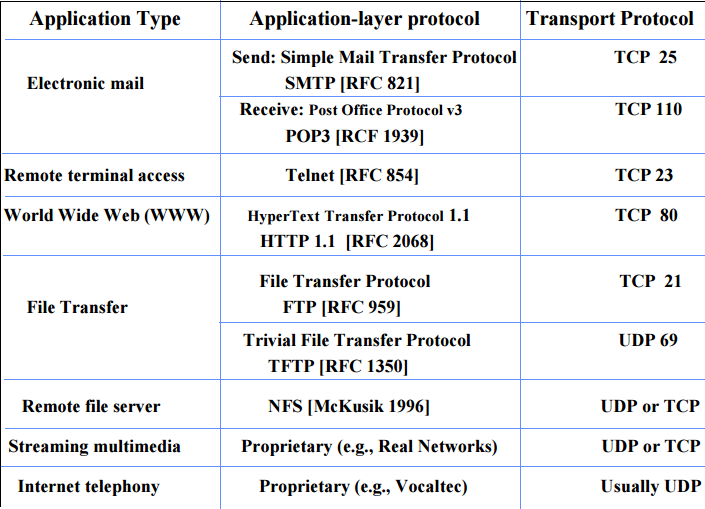


|  |  |  |  |
| --- | --- | --- | --- |
| **Name of Student:** | | | **Class:** |
| **Enrollment No:** | | | **Batch:** |
| **Date of Experiment:** | **Date of Submission:** | | **Submitted on:** |
| **Remarks by faculty:** | | | **Grade:** |
| **Signature of student:** | | **Signature of Faculty:** | |

# EXPERIMENT 6: Study of Application Layer Protocols

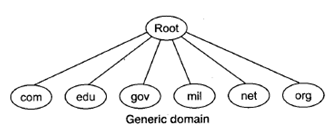
An application layer protocol defines how application processes (clients and servers), running on different end systems, pass messages to each other. In particular, an application layer protocol defines:

* The types of messages, e.g., request messages and response messages.
* The syntax of the various message types, i.e., the fields in the message and how the fields are delineated.
* The semantics of the fields, i.e., the meaning of the information that the field is supposed to contain;
* Rules for determining when and how a process sends messages and responds to messages.



**Domain Name System (DNS):**

* To identify an entity, TCP/IP protocol uses the IP address which uniquely identifies the connection of a host to the Internet.
* DNS is a hierarchical system, based on a distributed database, that uses a hierarchy of Name Servers to resolve Internet host names into the corresponding IP addresses required for packet routing by issuing a DNS query to a name server.
* However, people refer to use names instead of address. Therefore, we need a system that can map a name to an address and conversely an address to name.
* In TCP/IP, this is the domain name system.
* DNS in the Internet: DNS is protocol that can be used in different platforms.
* Domain name space is divided into three categories.
* **Generic Domain:** The generic domain defines registered hosts according, to their generic behaviour. Each node in the tree defines a domain which is an index to the domain name database.



* **Country Domain:** The country domain section follows the same format as the generic domain but uses 2 characters country abbreviations (e.g., US for United States) in place of 3 characters.
* **Inverse Domain:** The inverse domain is used to map an address to a name.

**FTP (File Transfer Protocol):**

* FTP is the standard mechanism provided by TCP/IP for copying a file from one host to another.
* FTP differs form other client-server applications because it establishes 2 connections between hosts.
* Two connections are: Data Connection and Control Connection.
* Data Connection uses PORT 20 for the purpose and control connection uses PORT 21 for the purpose.
* FTP is built on a client-server architecture and uses separate control and data connections between the client and the server.
* One connection is used for data transfer, the other for control information (commands and responses).
* It transfer data reliably and efficiently.

**HTTP (Hypertext Transfer Protocol):**

* This is a protocol used mainly to access data on the World Wide Web (www).
* The Hypertext Transfer Protocol (HTTP) the Web's main application-layer protocol although current browsers can access other types of servers
* A repository of information spread all over the world and linked together.
* The HTIP protocol transfer data in the form of plain text, hyper text, audio, video and so on.
* HTTP utilizes TCP connections to send client requests and server replies.
* It is a synchronous protocol which works by making both persistent and non persistent connections.

**HTTPS (Hypertext Transfer Protocol Secure):**

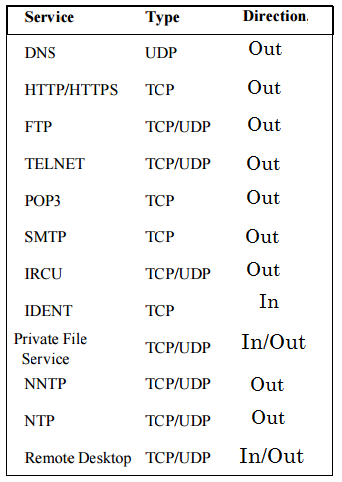
Hypertext Transfer Protocol Secure (HTTPS) is a variant of the standard web transfer protocol (HTTP) that adds a layer of security on the data in transit through a secure socket layer (SSL) or transport layer security (TLS) protocol connection. HTTPS enables encrypted communication and secure connection between a remote user and the primary web server.

**Basic Working of HTTPS**

* Public key and signed certificates are required for the server in HTTPS Protocol.
* Client requests for the https:// page
* When using an https connection, the server responds to the initial connection by offering a list of encryption methods the webserver supports.
* In response, the client selects a connection method, and the client and server exchange certificates to authenticate their identities.
* After this is done, both web server and client exchange the encrypted information after ensuring that both are using the same key, and the connection is closed.
* For hosting https connections, a server must have a public key certificate, which embeds key information with a verification of the key owner's identity.
* Almost all certificates are verified by a third party so that clients are assured that the key is always secure.

**TELNET (Terminal Network):**

* TELNET is client-server application that allows a user to log onto remote machine and lets the user to access any application program on a remote computer.
* TELNET uses the NVT (Network Virtual Terminal) system to encode characters on the local system.
* On the server (remote) machine, NVT decodes the characters to a form acceptable to the remote machine.
* TELNET is a protocol that provides a general, bi-directional, eight-bit byte oriented communication facility.
* Many application protocols are build upon the TELNET protocol.
* Telnet services are used on PORT 23.



|  |  |  |  |
| --- | --- | --- | --- |
| **Name of Student:** | | | **Class:** |
| **Enrollment No:** | | | **Batch:** |
| **Date of Experiment:** | **Date of Submission:** | | **Submitted on:** |
| **Remarks by faculty:** | | | **Grade:** |
| **Signature of student:** | | **Signature of Faculty:** | |

**EXPERIMENT 7: A Case Study on TCL Language**

**1. Introduction**

The Tool Command Language (TCL) was designed by John Ousterhout around the year 1988. There are several motivations behind the development of TCL. First, each tool had its unique languages to invoke commands. As a result, language for one particular tool cannot be used or extended to another tool. However, a general purpose programmable command language can amplify the power of a tool by allowing users to write programs in the command language to extend a tool’s built-in facilities. Second, the number of interactive applications increased significantly (around 1988’s) compared to the number of batch-oriented applications. Each new interactive application requires a new command language to be developed. But command languages developed tend to have insufficient power and clumsy syntax. To overcome these, John introduced the notion of “embeddable command language” and implemented it in TCL language.

**2.1 Design goals of TCL**

According to John Ousterhout, there are three design goals for the TCL language.

1. The language must be very simple and generic so that it can work easily with different applications and does not restrict features that applications can provide.
2. The language must be extensible so that an application can add its own features in TCL language. Moreover, the application-specific features should appear natural, as if they had been designed into the language from the start
3. The first and second goals provide the roadmap for syntactic and semantic notions of TCL languages, while the third goal is more related to machine portability and

Performance.

**2.2 TCL Syntax rules:**

1. **Commands**: A TCL script consists of one or more commands separated by either semi-colons or newlines, except when commands are enclosed with quotation and close brackets. We describe these exceptions later in this section.
2. **Evaluation**: A command is evaluated in two steps. First, a TCL interpreter breaks the command into words and performs substitutions of variables. The first word is considered as a command and the remaining words are passed as arguments of a command (or procedure). Commands might have their own interpreters for arguments.
3. **Words**: Words (or arguments) of a command are separated by white space, except newline and semicolon, which are command separators.
4. **Double quotes**: If the first character of a word is a double-quote (''), the word must be terminated by the next double-quote character. If semi-colons, close brackets, or white space characters (including newlines) appear between the quotes, they are treated as ordinary characters. Command substitution, variable substitution, and backslash substitution are performed on the characters between the quotes. The start and end quotes are not considered as part of a word.
5. **Braces**: If the first character of a word is an open brace (i.e., {) then the word must be terminated by a matching close brace (i.e.,}). For each additional open brace located in a word, there must be an additional close brace. However, an open brace or close brace preceded by a backslash character is not counted as a matching close brace. There is no substitution on the characters between the braces except for backslash-newline substitutions. Moreover, semi-colons, close brackets, or white spaces do not have any special interpretation. The word will consist of exactly the characters between the outer braces, not including the braces themselves.
6. **Command substitution:** If a word resides in an open bracket (i.e., [) then TCL performs command substitution. TCL interpreter is invoked recursively to process characters following the open bracket as a TCL script until terminated by a close bracket (i.e., ]). The result of the script (i.e., the result of its last command) is substituted into the word in place of the brackets and all of the characters between them. There can be any number of command substitutions in a single word.
7. **Variable substitution**: If a word contains a dollar sign ($) followed by one of the forms described below, then TCL performs variable substitution. A variable substitution can be one of the forms: $name (a scalar variable), $name(index) (an element of an array named name at index), and ${name} (a scalar variable).
8. **Backslash substitution**: If backslashes (i.e. \) appear within a word, substitutions occur (e.g., \n for newline or 0xA). Only a specific set of backslash characters are substituted as described in [4].
9. **Comments**: TCL comments begin with a hash character (i.e., #).
10. **Order of substitution**: Each character is processed exactly once by a TCL interpreter as part of words in a command. Substitutions occur from left to right. If command substitution occurs then the nested command is processed entirely by recursive call to the TCL interpreter. For example, command sequences set y [set x 0][incr x][incr x] will set the variable y to the value 012.
11. **Substitution and word boundaries**: Substitutions do not affect word boundaries of a command even if the value of a variable contains spaces.

**2.3 Features of TCL built-in commands**

TCL is a scripting language where most of the language features are implemented by set of built-in commands. These are often sufficient to do various programming tasks. While we describe these built-in commands, we also highlight language design goals that are obtained by commands.

**Data types, variables, and assignment of variables**

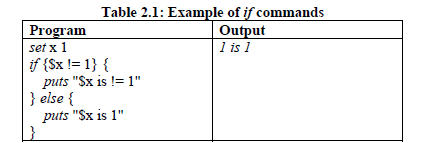
There are two three basic data types namely numeric, string, and list. A variable needs not be declared explicitly like other programming languages such as C. However, naming convention of a variable is similar to ANSI C. Moreover, there are no static typing of variables (i.e., a variable can take integer, float, and string type values.

**Expression, operators, and operands**

A TCL expression consists of a combination of operands, operators, and parentheses. If white space characters are used to separate between operands, operators, and parentheses, then they are ignored by the expression command processor. The expr command evaluates a list of arguments located in square brackets (e.g., expr [2 + 2]), and returns the value.

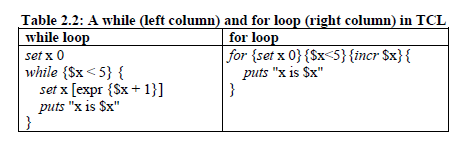
**Conditional statement**

TCL has an if command that supports single and nested if else like statement.

****

**Loop commands**

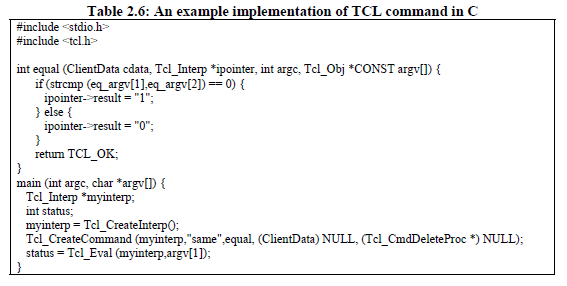
TCL has two loop commands: while and for. The command evaluates test as an expression. The code in body is executed, if test is true. After the code is executed, test is evaluated again as long as test is evaluated as true.

****

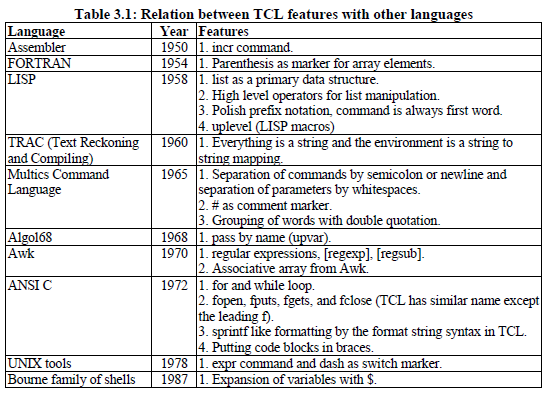
**Integrating TCL with other languages**

TCL can be extended or embedded. The extension means that new commands implemented in other languages (e.g., C) can be run with TCL interpreter just like it’s a built-in command in TCL. Embedding implies that TCL script can be invoked run from other languages through suitable API library functions. We describe both of the approaches in brief below for C programming language.

**TCL commands implemented in C:** The C code that implements a TCL command is called a command procedure. The interface to a command procedure takes an array of 9 values as inputs in a main method that corresponds to the arguments in the TCL script command. The result of the command procedure becomes the result of the TCL command. There are two kinds of command procedures: string-based and object-based. We discuss the string-based interface. Strings are generalized into the Tcl\_Obj type, which can be a string or another native representation like an integer, floating number. Conversions between strings and other types are done in a lazy fashion, and the saved conversions help your scripts run more efficiently. The string-based interface to a C command procedure is much like the interface to the main program.



**3. Comparing TCL features with other languages**

****

**4. TCL features:**

**4.1 Uniformity**

Lack of uniformity implies inconsistencies among syntaxes of a language. Weinberg defined it as “the same things should be done in the same way whenever they occur”. We observe several features of TCL violate uniformity, which are mainly based on its syntax and command features.

**3.2 Compactness**

Compactness criterion is judged by different features of a language which allow expressing statements more concisely. We observe that TCL has several interesting features that go in favor of compactness. These can be found through the rich command sets.

**4.3 Locality**

The locality feature can be justified by features of the language which allows a programmer to find all parts of a code in the same place. Otherwise, one needs to go back and forth for finding any variable or function declaration. TCL has several features which allows better locality, e.g. there is no explicit variable declaration and typing. So, a variable can be set or reset.

**4.4 Linearity**

The linearity feature justifies how easily a program can be understood by reading it sequentially. It is well understood that branching, goto, etc. causes difficulty in understanding a program. TCL has no goto related command. However, the uplevel command allows arbitrary control structure among procedure calls. This command affects linearity considerably.

**5. Conclusion and future work**

This project studies the original design goals of a popular scripting language named TCL (Tool Command Language). We identify three design principles of TCL that include simple and generic language, ability to extend through other languages, and easy gluing together the extension with the language. These goals have been fulfilled by a small set of syntax rules, rich built-in set of command libraries that include structured programming notions, rich data structures and their manipulation.

|  |  |  |  |
| --- | --- | --- | --- |
| **Name of Student:** | | | **Class:** |
| **Enrolment No:** | | | **Batch:** |
| **Date of Experiment:** | **Date of Submission:** | | **Submitted on:** |
| **Remarks by faculty:** | | | **Grade:** |
| **Signature of student:** | | **Signature of Faculty:** | |

**EXPERIMENT 8:** To demonstrate the working of Stop N Wait ARQ

**SOFTWARE USED: -**

**THEORY:-**

**Stop-and-wait ARQ** also can be referred as **Alternating-Bit-Protocol** is a method used in [telecommunications](http://en.wikipedia.org/wiki/Telecommunications) to send information between two connected devices. It ensures that information is not lost due to dropped packets and that packets are received in the correct order. It is the simplest kind of [automatic repeat-request](http://en.wikipedia.org/wiki/Automatic_repeat-request) (ARQ) method. A stop-and-wait ARQ sender sends one [frame](http://en.wikipedia.org/wiki/Frame_%28telecommunications%29) at a time; it is a special case of the general [sliding window protocol](http://en.wikipedia.org/wiki/Sliding_window_protocol) with both transmit and receive window sizes equal to 1. After sending each frame, the sender doesn't send any further frames until it receives an [acknowledgement](http://en.wikipedia.org/wiki/Acknowledgement_%28data_networks%29) (ACK) signal. After receiving a good frame, the receiver sends an ACK. If the ACK does not reach the sender before a certain time, known as the timeout, the sender sends the same frame again.

The above behaviour is the simplest Stop-and-Wait implementation. However, in a real life implementation there are problems to be addressed.

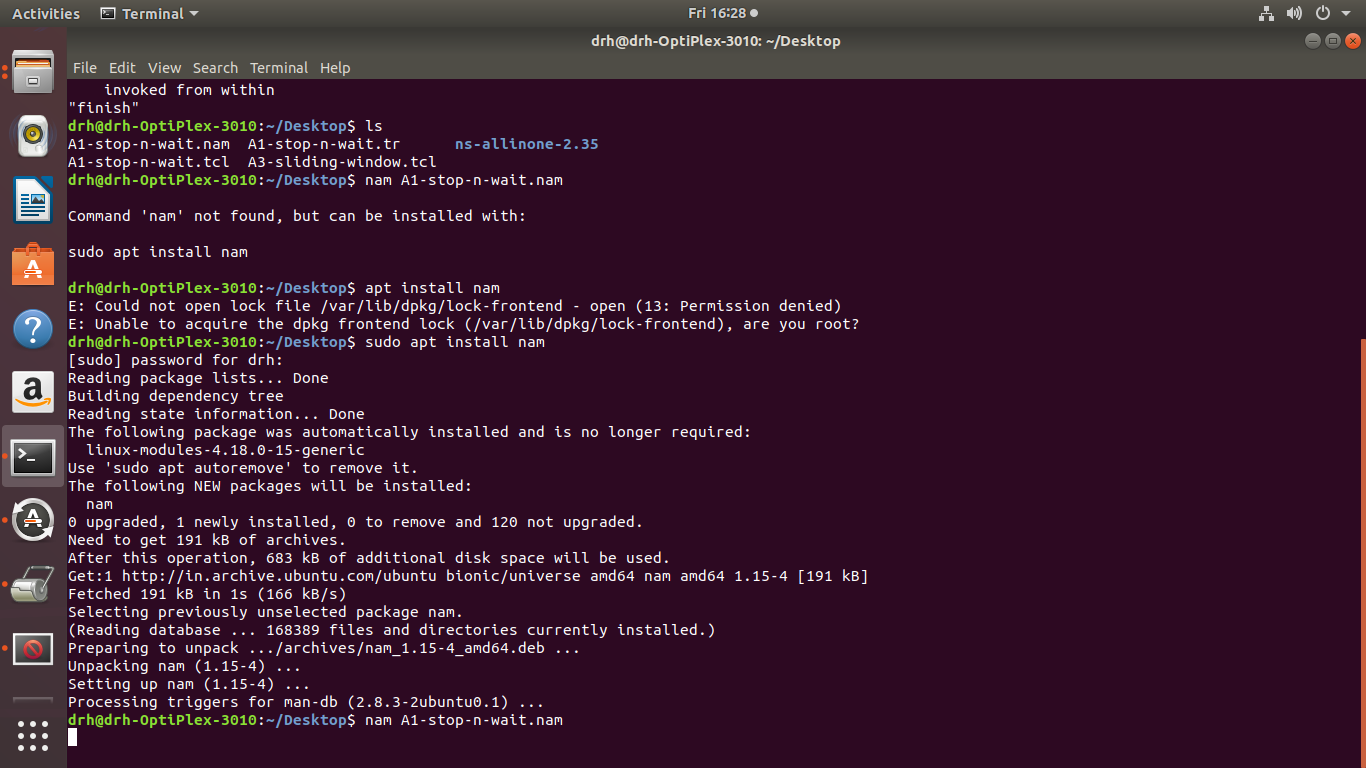
Typically the transmitter adds a [redundancy check](http://en.wikipedia.org/wiki/Redundancy_check) number to the end of each frame. The receiver uses the redundancy check number to check for possible damage. If the receiver sees that the frame is good, it sends an ACK. If the receiver sees that the frame is damaged, the receiver discards it and does not send an ACK—pretending that the frame was completely lost, not merely damaged.

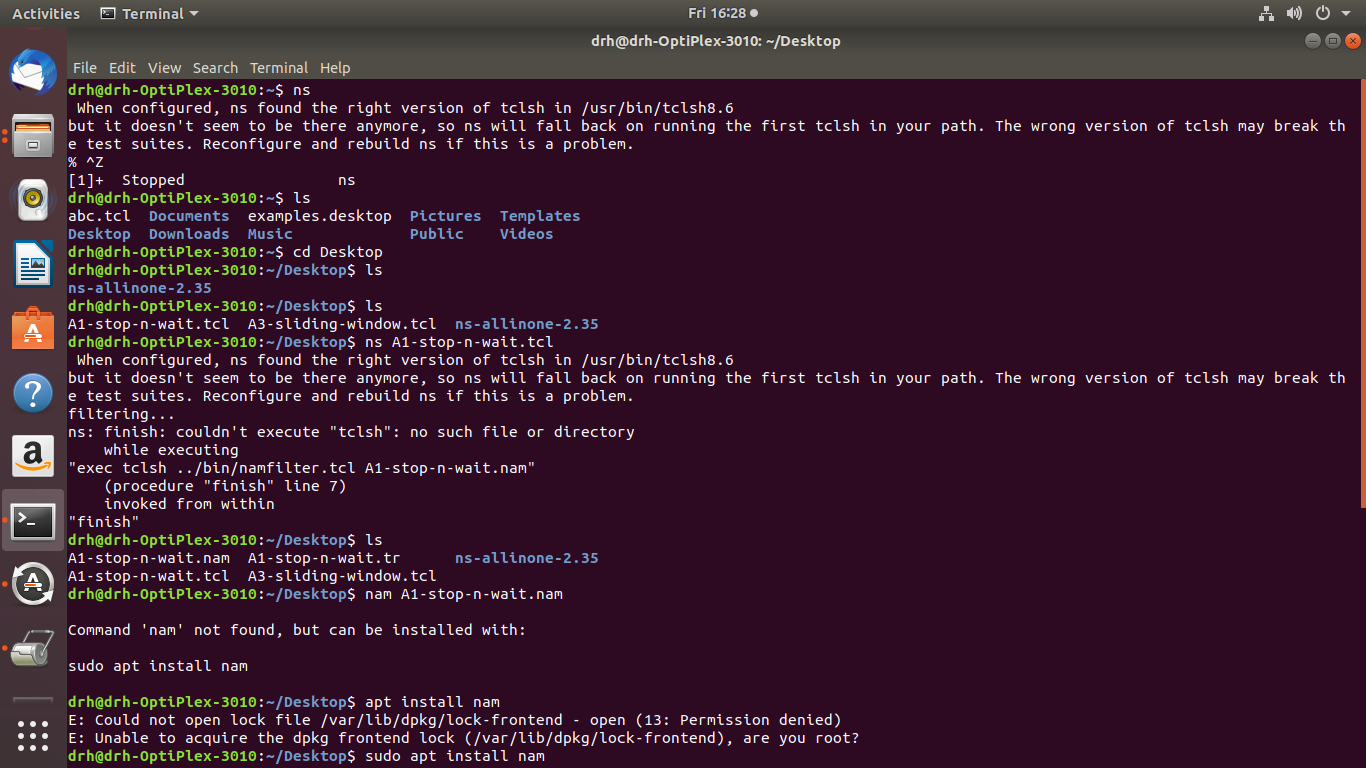
One problem is where the ACK sent by the receiver is damaged or lost. In this case, the sender doesn't receive the ACK, times out, and sends the frame again. Now the receiver has two copies of the same frame, and doesn't know if the second one is a duplicate frame or the next frame of the sequence carrying identical data.

Another problem is when the transmission medium has such a long [latency](http://en.wikipedia.org/wiki/Latency_%28engineering%29) that the sender's timeout runs out before the frame reaches the receiver. In this case the sender resends the same packet. Eventually the receiver gets two copies of the same frame, and sends an ACK for each one. The sender, waiting for a single ACK, receives two ACKs, which may cause problems if it assumes that the second ACK is for the next frame in the sequence.

To avoid these problems, the most common solution is to define a 1 bit *sequence number* in the header of the frame. This sequence number alternates (from 0 to 1) in subsequent frames. When the receiver sends an ACK, it includes the sequence number of the next packet it expects. This way, the receiver can detect duplicated frames by checking if the frame sequence numbers alternate. If two subsequent frames have the same sequence number, they are duplicates, and the second frame is discarded. Similarly, if two subsequent ACKs reference the same sequence number, they are acknowledging the same frame.

Stop-and-wait ARQ is inefficient compared to other ARQs, because the time between packets, if the ACK and the data are received successfully, is twice the transit time (assuming the turnaround time can be zero). The throughput on the channel is a fraction of what it could be. To solve this problem, one can send more than one packet at a time with a larger sequence number and use one ACK for a set. This is what is done in [Go-Back-N ARQ](http://en.wikipedia.org/wiki/Go-Back-N_ARQ) and the [Selective Repeat ARQ](http://en.wikipedia.org/wiki/Selective_Repeat_ARQ).





Output:

