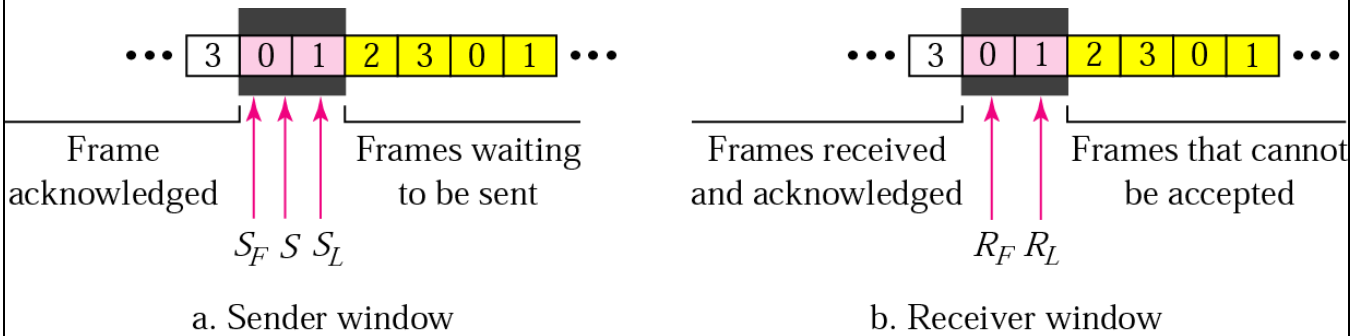


Sub:	Computer Networks				Sub Code:18EC71
Date:	1/02/2021	Duration:	90 Minutes	Max Marks:	50
					Sem / Sec:A,B,C,D

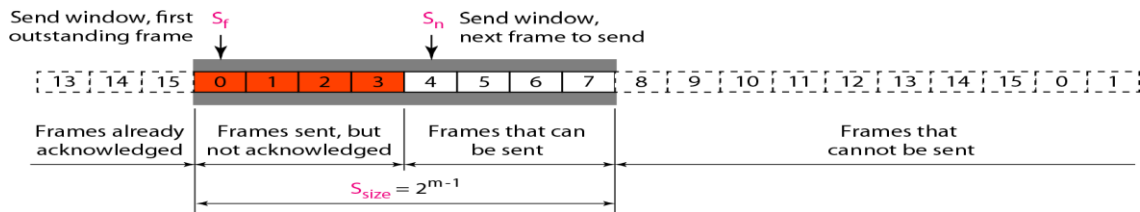
Answer any FIVE FULL Questions

1 Explain Selective Repeat protocol with send window and receive window [10]

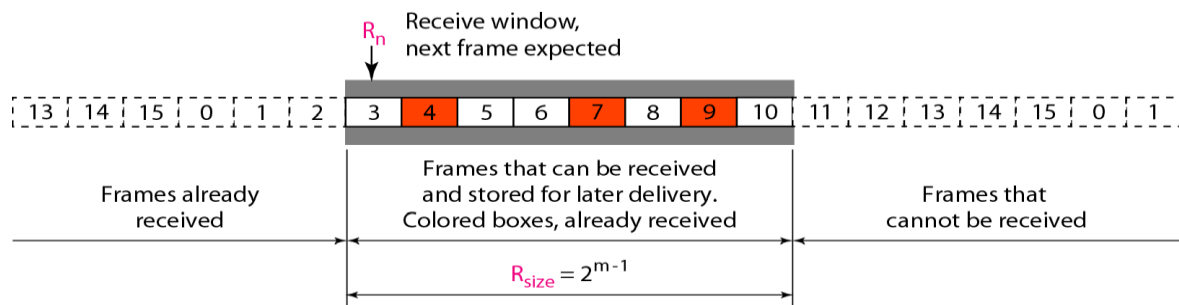
- *The window size is reduced to one half of 2^m*
- *Sender window size = receiver window size = $2^m / 2$*
- *Window size = sequence number/2*
- *If $m = 2$, Window size = $4/2=2$*
- *Sequence number = 0, 1, 2, 3*



Send window for Selective Repeat ARQ



Receive window for Selective Repeat ARQ

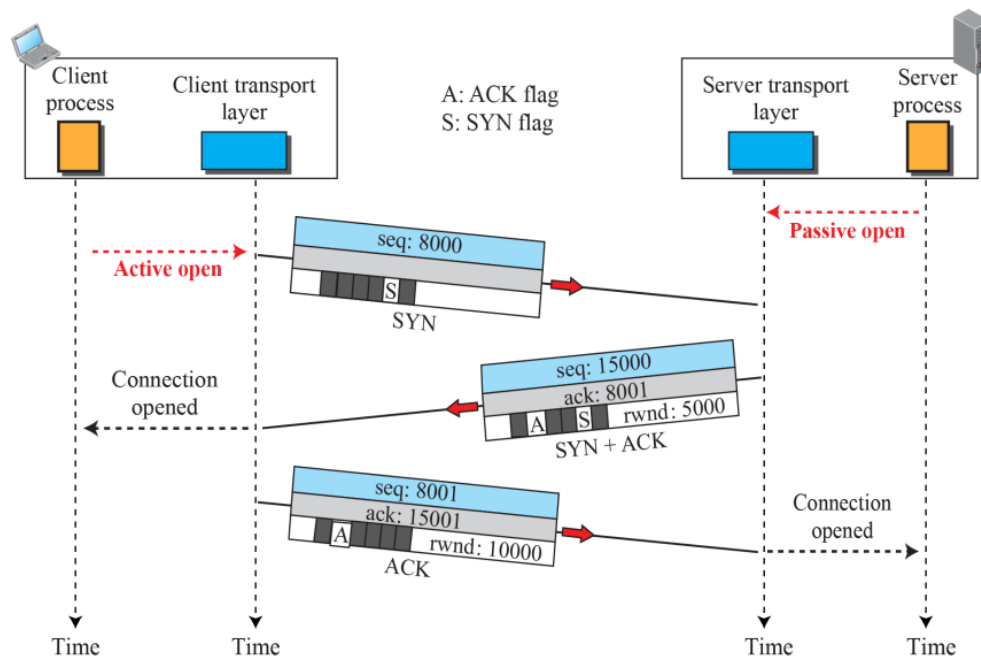


2 Explain connection less and connection oriented service provided by transport layer showing the movement of packets using time line. [10]

A TCP Connection

- It occurs in three phases
- a) Connection Establishment
- b) Data transfer
- c) Connection reset

Connection establishment using three-way handshaking



Each segment has values for all its header fields and perhaps for some of its option fields too.

SYN segment:

- 1) The first TCP segment is shared between client and server with "SYN FLG" set to '1'. The segment is called as "SYN" segment.
- 2) Used for synchronization of sequence numbers.
- 3) It is a control signal and does not contain any data, but consumes one sequence number.

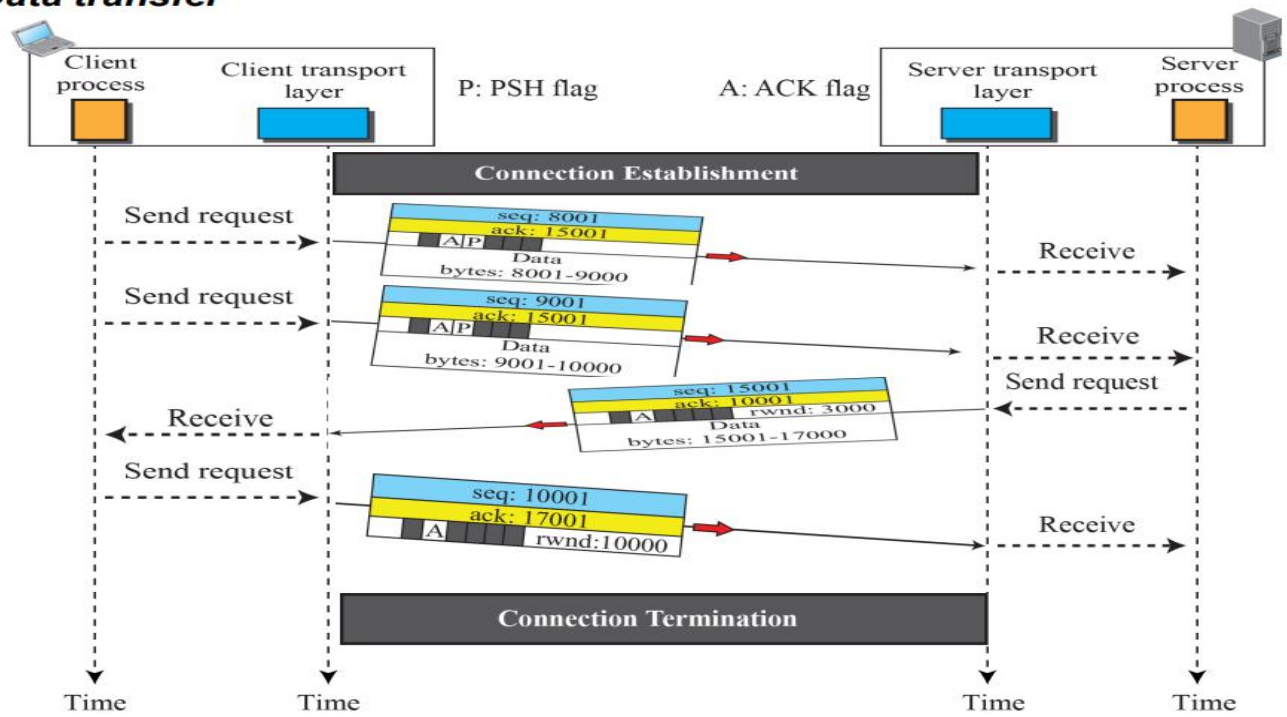
SYN + ACK segment:

The server replies back by sharing the "SYN+ACK" to client by setting an ACK flag and displaying the next sequence number it is expecting to receive from client

ACK segment:

- 1) The client shares the third segment by sharing the "ACK" to server.
- 2) It acknowledges the receipt of the second segment with the ACK flag and acknowledgement number field.

Data transfer



Data transfer:

- After the connection is established the client sends 2000 bytes in one segment
- Then the client shares one more segment with sequence numbers assigned to them.
- As the data is pushed from client to sever hence “ACK” and “PSH” flags are set.
- As soon as the data segments are received by the client it shares the an segment back in which only “ACK” flag is set.
- After receiving the “ACK” segment the client shares the segment to confirm for the segment received.

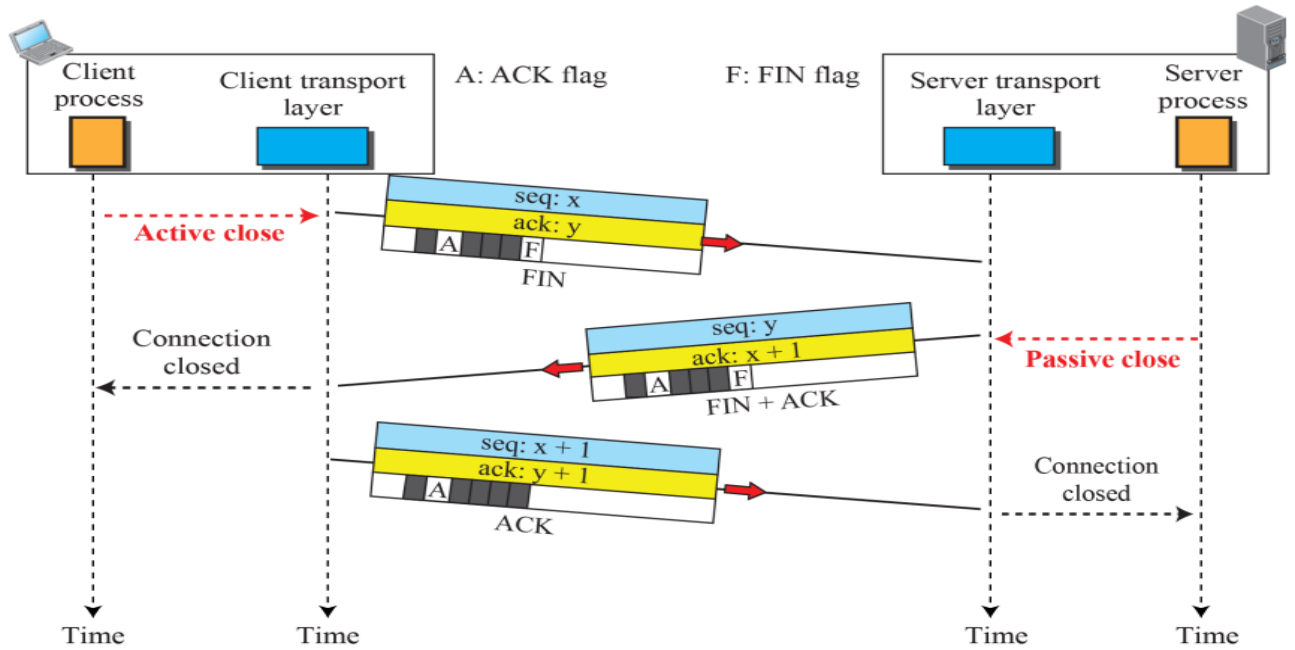
Connection Termination

- For connection termination:

---Three-way handshaking

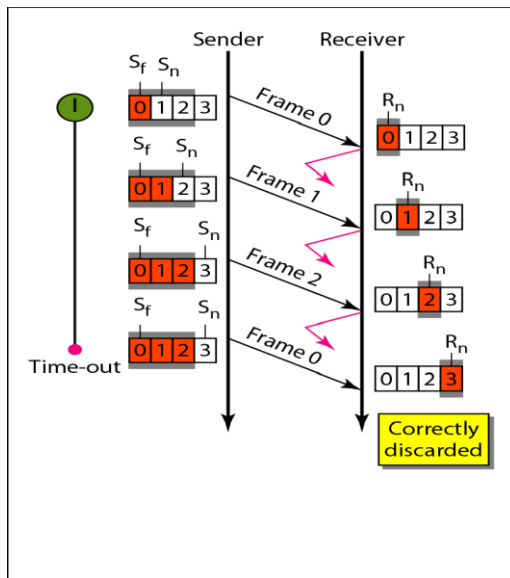
---And four-way handshaking with a half-close option

Connection termination using three-way handshaking

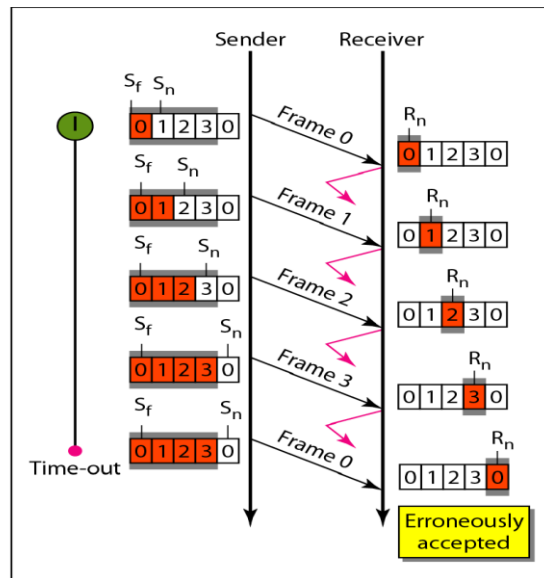


1. In a common situation, the client TCP, after receiving a close command from the client process, sends the first segment, a FIN segment in which the FIN flag is set.
2. Note that a FIN segment can include the last chunk of data sent by the client.
3. It is just a control segment and consumes only one sequence number.

3 Explain why the size of the send window in go back N must be less than 2^m ? [10]
 In Go-Back-N ARQ, the size of the send window must be less than 2^m ;
 the size of the receiver window is always 1



a. Window size $< 2^m$



b. Window size $= 2^m$

4 Discuss general services provided by UDP. List uses/applications of UDP. [10]

UDP Services

- Process-to-Process Communication
- Connectionless services
- Flow Control
- Error Control

- Checksum
- Congestion Control
- Encapsulation and Decapsulation
- Queuing
- Multiplexing and Demultiplexing

Process-to-Process Communication

- UDP provides process-to-process communication using socket addresses, a combination of IP addresses and port numbers.

Connectionless Services

- As mentioned previously, UDP provides a connectionless service. This means that each user datagram sent by UDP is an independent datagram.

Flow Control

- UDP is a very simple protocol. There is no flow control, and hence no window mechanism. The receiver may overflow with incoming messages.

Error Control

- There is no error control mechanism in UDP except for the checksum. This means that the sender does not know if a message has been lost or duplicated.

Checksum

UDP checksum calculation includes three sections: **a pseudoheader, the UDP header, and the data coming from the application layer.**

Congestion Control

- Since UDP is a connectionless protocol, it does not provide congestion control.

Encapsulation and Decapsulation

- To send a message from one process to another, the UDP protocol encapsulates and decapsulates messages.
- Queuing

In UDP, queues are associated with ports.

At the client site, when a process starts, it requests a port number from the operating system

Multiplexing and Demultiplexing

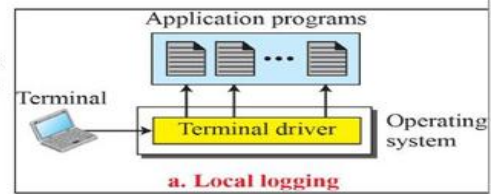
When several processes are used, UDP multiplexing and demultiplexing

5

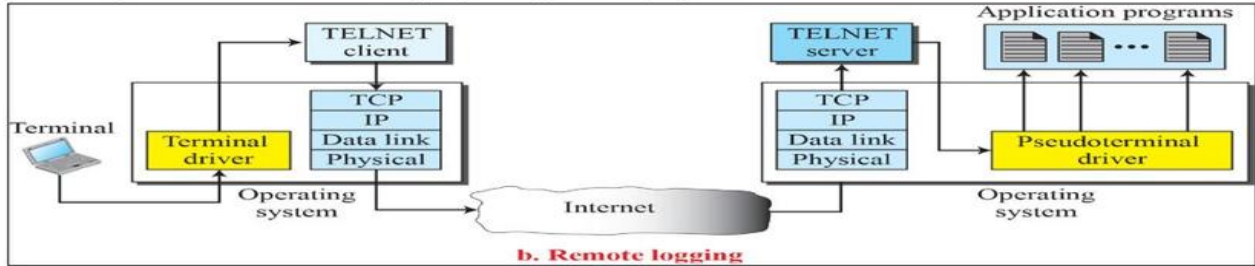
Explain TELNET protocol for remote login process with the help of figure[10]

It is impossible to have a client/server pair for each type of service we need; the number of servers soon becomes intractable. The idea is not scalable. The solution is to have a specific client/server program for a set of common scenarios, but to have some generic client/server programs for the rest.

When a user logs into a local system, it is called *local logging*. As a user types at a terminal or at a workstation running a terminal emulator, the keystrokes are accepted by the terminal driver. The terminal driver passes the characters to the operating system. The operating system interprets the combination of characters and invokes the desired application program or utility.

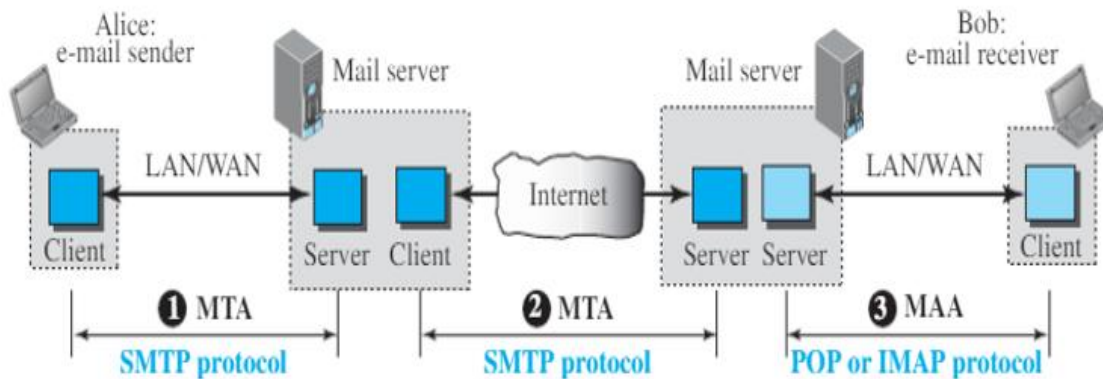


When a user wants to access an application program or utility located on a remote machine, she performs *remote logging*. The user sends the keystrokes to the terminal driver where the local operating system accepts the characters but does not interpret them. The characters are sent to the TELNET client, which transforms the characters into a universal character set called *Network Virtual Terminal (NVT)* characters and delivers them to the local TCP/IP stack. NVT travel through the Internet and arrive at the TCP/IP stack at the remote machine. Here the characters are delivered to the operating system and passed to the TELNET server, which changes the characters to the corresponding characters understandable by the remote computer. However, the characters cannot be passed directly to the operating system because it is not designed to receive characters from a TELNET server; it is designed to receive characters from a terminal driver. The solution is to add a piece of software called a pseudoterminal driver, which pretends that the characters are coming from a terminal. The operating system then passes the characters to the appropriate application program.



6 Explain different protocols used in electronic mail with neat figure.[10]

- 1) SMTP (Simple Mail Transfer Protocol)
- 2) POP (Post Office Protocol 3)



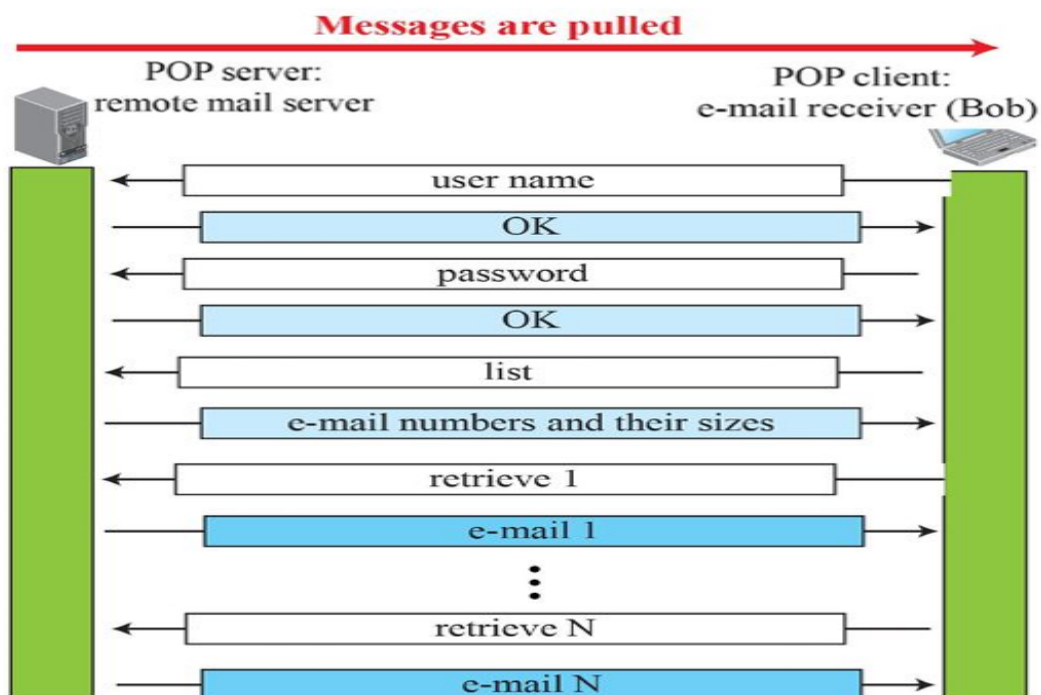
- 1) SMTP uses commands and responses to transfer messages between an MTA client and an MTA server. Each command or reply is terminated by a two-character (carriage return and line feed) end-of-line token.

Commands:

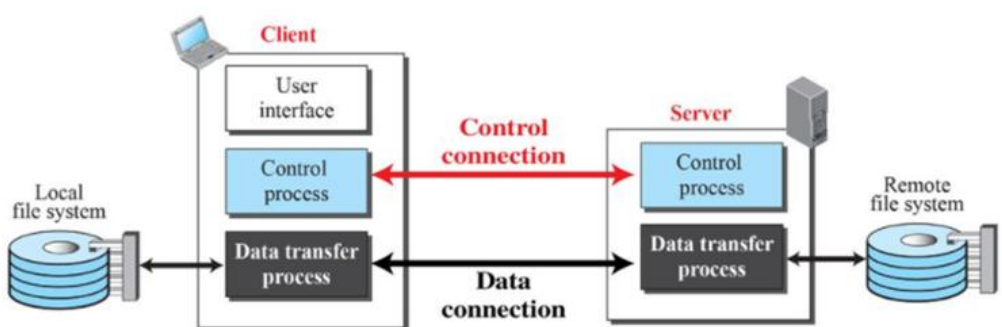
- Commands are sent from the client to the server. The format of a command is shown in next figure.
 - It consists of a keyword followed by zero or more arguments. SMTP defines 14 commands.
 - The first five are mandatory; every implementation must support these five commands.
 - The next three are often used and highly recommended. The last six are seldom used.
- 1) POP is the most commonly used protocol for receiving email over the internet. The "3" refers to the third version of the original POP protocol. A recipient or their email client can download mail periodically from the server using POP3.
 - The client POP3 software is installed on the recipient computer; the server POP3 software is

installed on the mail server.

- Mail access starts with the client when the user needs to download its e-mail from the mailbox on the mail server.
- The client opens a connection to the server on TCP port 110.
- It then sends its user name and password to access the mailbox.
- The user can then list and retrieve the mail messages, one by one.



7 Explain FTP protocol with figure. [10]



The Figure shows a....

- ✓ basic model of FTP. The client has three components: the user interface, the client control process, and the client data transfer process.
- ✓ The server has two components: the server control process and the server data transfer process.
- ✓ The control connection is made between the control processes. The data connection is made between the data transfer processes.

- ***Communication is achieved through commands and responses.***
- ***commands are sent from the client to the server and responses are sent from the server to the client.***
- ***This simple method is adequate for the control connection because we send one command (or response) at a time.***
- ***Each line is terminated with a two-character (carriage return and line feed) end-of-line token.***