Vasad R. Patil
5EIT 8688

Q1] Compression:

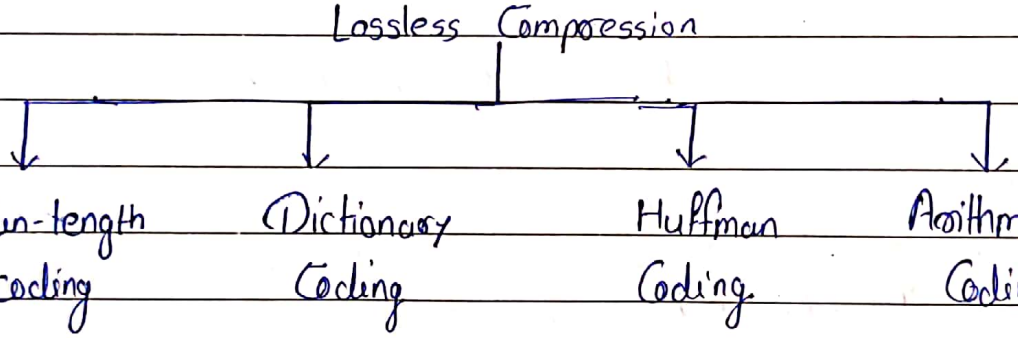
Due to large volume of data exchanged, compression plays an important role in multimedia communication.

In compression, volume of data tobe exchanged is reduced.

1) Lossless Compression:

In lossless compression, the redundant information contained in data is removed.

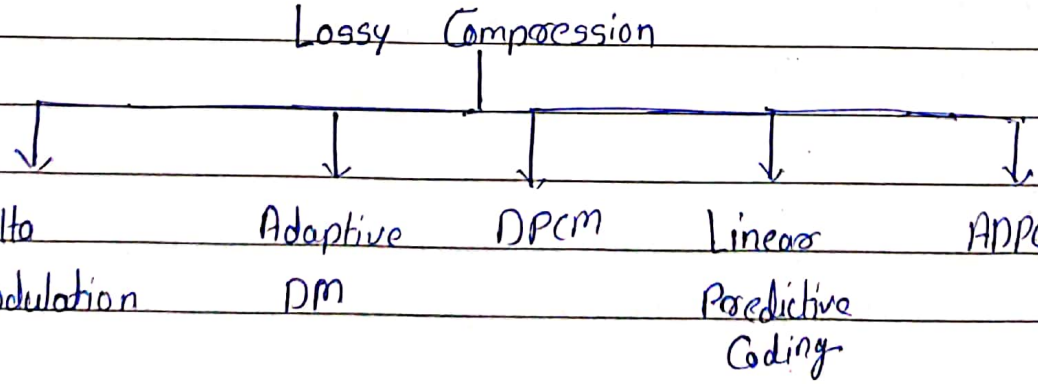
Due to such removal, there is no loss of data which contain information. Hence, it is called lossless compression.



2) Lossy Compression:

There is no limit on the amount of compression in lossy compression

In this method there is loss of information in a controlled manner.



Encoding:

Storing		O/P Code	Addition
W	87	WY	256
Y	89	YS	257
S	83	S*	258
*	42	*W	259
WY	256	WYG	260
G	71	GW	261
WY	256	GWYS	262
S*	258	S*W	263
WYS	262	WYSW	264
WYS	262	WYSG	265
G	71		

Output Codes are 87 89 83 42 256 71 256 258 262 262 71

Q2] Application Layer:

The application layer is present at top of the OSI model. It is layer through which user interacts.

1] Telnet:

Telnet stands for TELecommunications Network. It helps in terminal emulation. It allows telnet client to access resources of telnet server. It is used to managging files on the internet.

2] FTP:

FTP stands for File Transfer Protocol. It is protocol that actually lets us transfer files. It can facilitate this between any 2 machines. But FTP is not just a protocol but it is also a program.

3] TFTP:

The trivial file transfer protocol is stripped down, stock version of FTP, but it's the protocol of choice if you know exactly what you want and where to find.

4] NFS:

It stands for network file system. It allows remote host to mount file systems over a network and interact with those as though they are mounted locally.

5] SMTP:

It stands for Simple Mail Transfer Protocol. It is part of TCP/IP protocol. Using method called store and forward, SMTP moves mail accross network.

6] LDP:

It stands for Line Printer Daemon. It is designed for printer sharing.

7] X window:

It defines protocol for writing GUI based client/server apps. This idea allow a program, called a client, to run on one computer.

8] DNS:

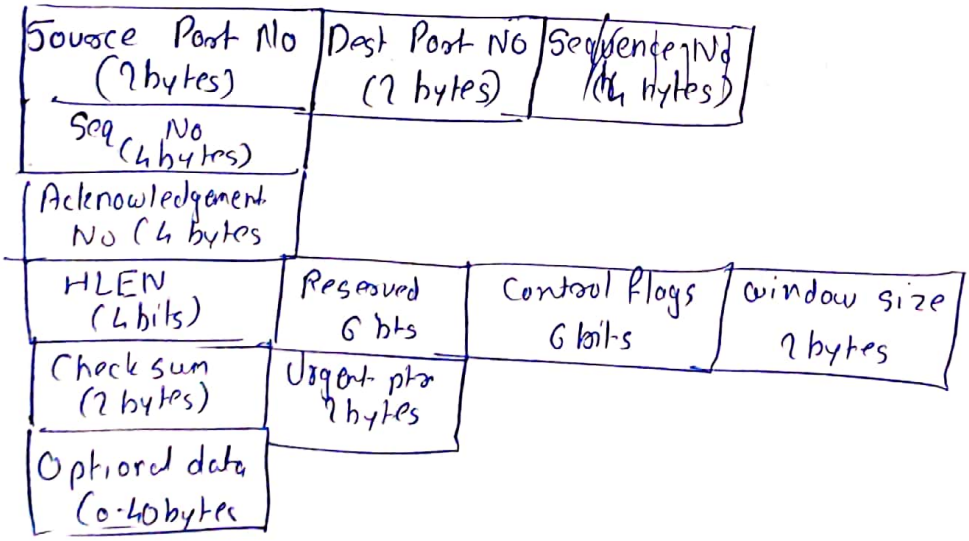
It stands for domain Name Service. Every time on using domain name, a DNS server translets name to IP address.

Q3]

TCP	UDP
1) TCP stands for transmission control protocol.	1) UDP stands for User Datagram Protocol.
2) It is connection oriented.	2) It is connectionless protocol.
3) TCP reads data in form of stream of bytes.	3) UDP contains packets which are transmitted one by one.
4) Header size is 20 bytes	4) Header size is 8 bytes.
5) TCP is slower.	5) UDP is faster as error recovery is not attempted.
6) TCP is heavier as 3 packets are required to setup connection.	6) UDP is lightweight. There is no tracking connections,
7) Does error checking and recovery.	7) Performs error checking but discards erroneous messages.
8) Acknowledgement segments.	8) No acknowledgement segments.

E293 0017 0000 0001 0000 0000 5002 07FF

TCP header contains following fields



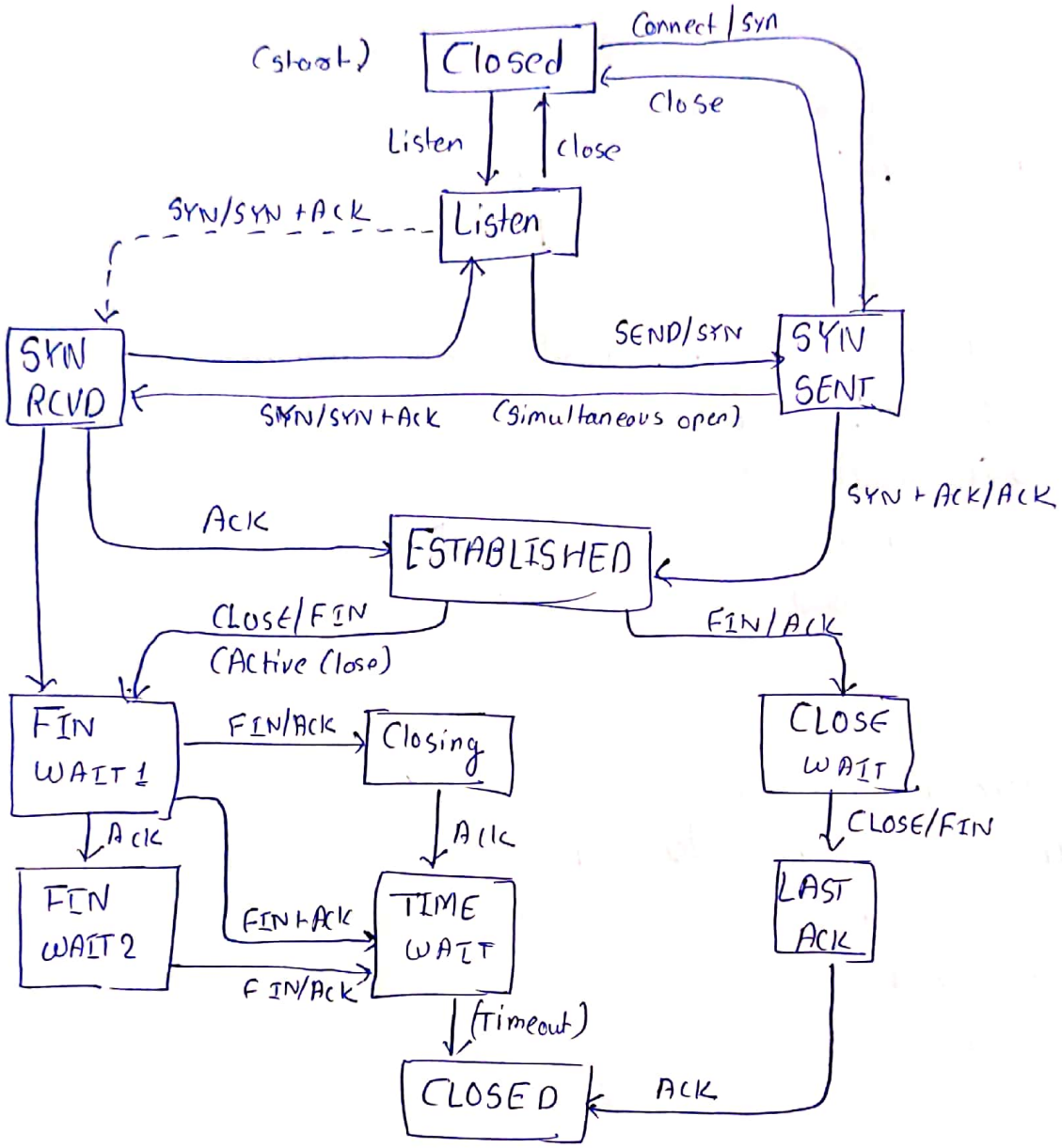
a) Source Port No = $(E793)_{16}$ = 58003
b) Dest Port No = $(0017)_{16}$ = 23
c) Sequence No = $(00000001)_{16}$ = 1
d) Acknowledgement No = $(00000000)_{16}$ = 0
e) Length of header = 5 i.e. header = 5X4 = 20 bytes
f) Type of segment: Combination of reserved field and control field is $(002)_{16}$. The rightmost 6 bits in binary are 000010 which means only SYN bit is set which is used to establish connection.
g) Window size = $(07FF)_{16}$ = 2047 bytes

Q4] TCP State Transition:

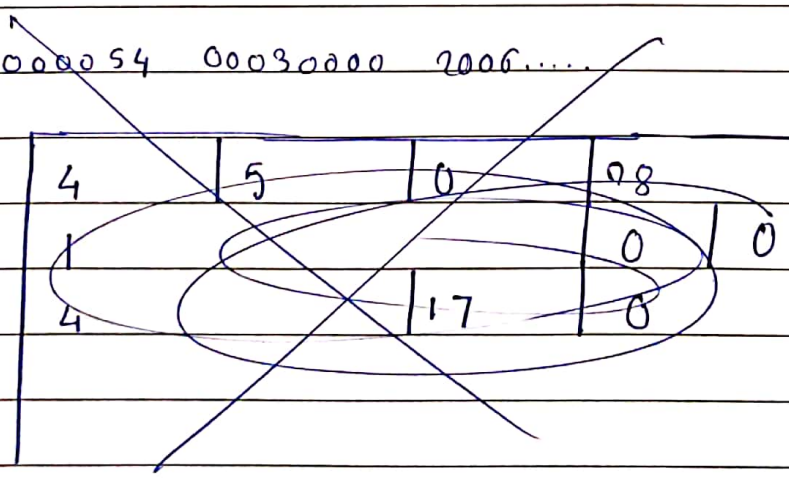
The steps to be followed in TCP connection establishment and release can be represented using finite state machine.

The states in machine are as follows:

CLOSED	No conn active or pending.
LISTEN	Server waiting for incoming call
SYN RCVD	Conn req arrived, wait for ACK
SYN SENT	Application has started an open connection
ESTABLISHED	Normal data transfer state.
FIN WAIT 1	Application said it is finished
FIN WAIT 2	Other side agreed to release
TIMED WAIT	Wait for all packets to die off
CLOSING	Both sides tried closing
CLOSE WAIT	Other side initiated a release
LAST ACK	Wait for ack of FIN of last close



IPV4	IPv6
1) IPv4 is 32-bit IP address	1) IPv6 is 128 bit IP address.
2) Numeric address with bits seperated by dot (.)	2) Alphanumeric address whose binary bits seperated by colon (:)
3) No. of header fields is 12	3) No. of header fields is 8
4) It has checksum fields.	4) Does not has checksum fields.
5) It supports for Virtual Length Subnet Musk (VLSM).	5) Does not support VLSM.
6) IPv4 offers different classes of IP Address. Class A to E	6) IPv6 allows storing unlimited number of IP addresses.
7) Fragmentation done by sending and forwarding routes.	7) Fragmentation done by sender.
8) e.g. 192.168.0.1	8) e.g. 2001:0bd8:0000:0000:0000:ff00:0047:7879



45 00 00 54 00 03 00 00 20 06

a) Header size = 5 X 4 = 20 bytes

b) ∵ length of header is 20 bytes, there are no options

c) Size of data = 84 - 20 = 64 bytes ∵ total length is 84

d) D = 0, M = 0 ~~fragoffset~~ offset = 0 packet is not fragmented

e) ∵ Value of time to live = 32

packet can travel 32 more routers

Q6] Organization granted block 130.56.0.0/16

a) No of valid addresses in each subnet = 62

b) First address in 1st subnet = 130.56.0.1

Last address in 1st subnet = 130.56.0.62

c) First address in 1a^st subnet = 130.56.255.193

Last address in last subnet = 13.56.255.254

Q7]

Dest	Subnet Mask	Interface
178.75.43.0	255.255.255.0	Eth 0
178.75.43.0	255.255.255.128	Eth 1
~~178.75.43~~		
197.17.17.5	255.255.255.224	Eth 3
default		Eth 2

packet 1 : 178.75.43.16

(178.75.43.16) and (178.75.43.0) = (178.75.43.0)

(178.75.43.16) and (255.255.255.128) = (178.75.43.0)

∵ both of the masks are producing same network ID, one with greater numbers of one will selected

i.e. $(10000000)_2 > (0)_2$

128 in binary has more no of 1s than that of 0

∴ Eth 1 will selected.

Packet 2: 192.17.17.10

(192.17.17.10) and (255.255.255.0) = (192.17.17.0)

(192.17.17.10) and (255.255.255.128) = (192.17.17.0)

(192.17.17.10) and (255.255.255.255) = (192.17.17.10)

as it does not match with any network id, it will be forwarded to default.

∴ Eth 2 will selected.

Q8] here,

header + data = 2500 bytes

let, size of IP header = 20 bytes

∴ data = 2480 bytes.

MTU = 500 bytes

here 20 bytes for header of each fragment of data

∴ 480 bytes of data

fragments = 2480/480

≈ 6 fragments

∴ 6×20 = 120 bytes of header will be delivered.

∴ Extra data = (2480+120) − 2500

= 100 bytes

100 bytes of extra data will be received at receiver end.

Q9] HLEN = 5

Total length field = 1000

header size = 5 × 4 = 20 bytes

$(1000)_{10} = (1111101000)_2$

∴ No of bits = 10

~~∴ No of bytes = 2~~

∴ Size of packet allowed = $2^{10} - 1$

= 1023 bytes

Out of 1023 bytes, 20 bytes will be header

∴ Data bytes = 1023 − 20

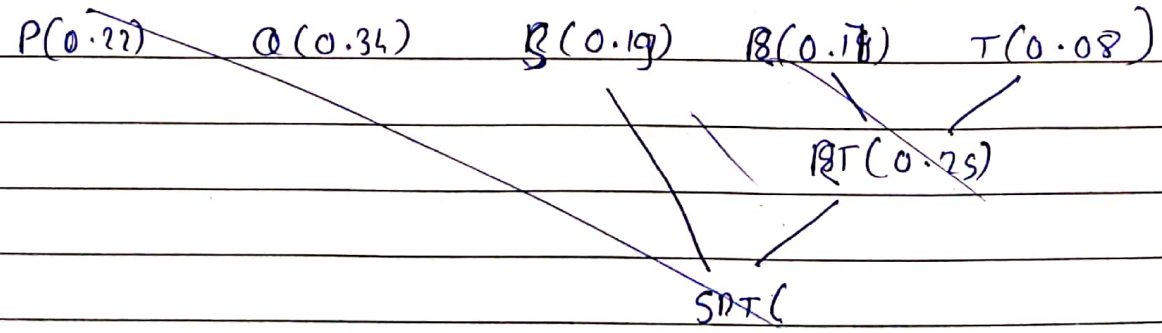
= 1003 bytes

Packet contains 1003 bytes.

Q10]

P	0.22
Q	0.34
R	0.17
S	0.19
T	0.08

R, S, T → RT (0.25)



Q	0.34
RT	0.25
P	0.22
S	0.19

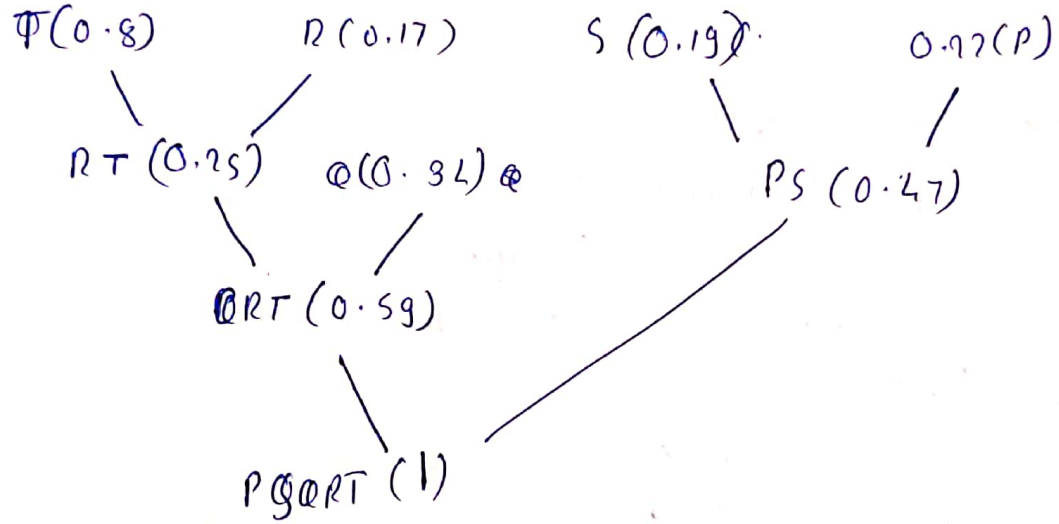
] — PS (0.41)

PS	0.41
Q	0.34
RT	0.25

] — QRT = 0.59

QRT	= 0.59
PS	= 0.41

] — PSQRT = 1.00



Here, no of bits by each P = 2
Q = 2
R = 3
S = 2
T = 3

∴ Expected length of encoded message =

$3 \times 0.08 + 3 \times 0.17 + 2 \times 0.19 + 2 \times 0.22 + 2 \times 0.34$

$= 2.25$

For 100 charecter message = 2.25×100

= 225 bits

Q11]

a) Beakleys Socket:

Beakleys Socket is an Application Programming Interface (API) for internet sockets and Unix domain sockets, used for inter process communication (IPC)

It is commonly used as library of linkable modules.

A socket is an abstract representation for the local endpoint of a network. Beakley Socket API represent it as a file descriptor.

Common functions of library are:

Socket-	Creates a new socket of certain socket type.
bind	Typically used on server side to associate socket with address.
listen	Used on server side and cause TCP socket to enter listening state.
connect	Used on client side to connect server.
close	Used to close socket connection.
send	Used to send message
recv	Used to receive message

b) Piggybanking:

Piggybanking in networking is a technique to utilize available bandwidth more efficiently.

The host does not send acknowledgement immediately but waits for some time and sends it with outgoing packet.

Consider 2 way communication b/w A & B

A sends some data to B

B has to send ack to A

B waits and sends ack with packet in which it contains message for A

This approach is called piggy banking.

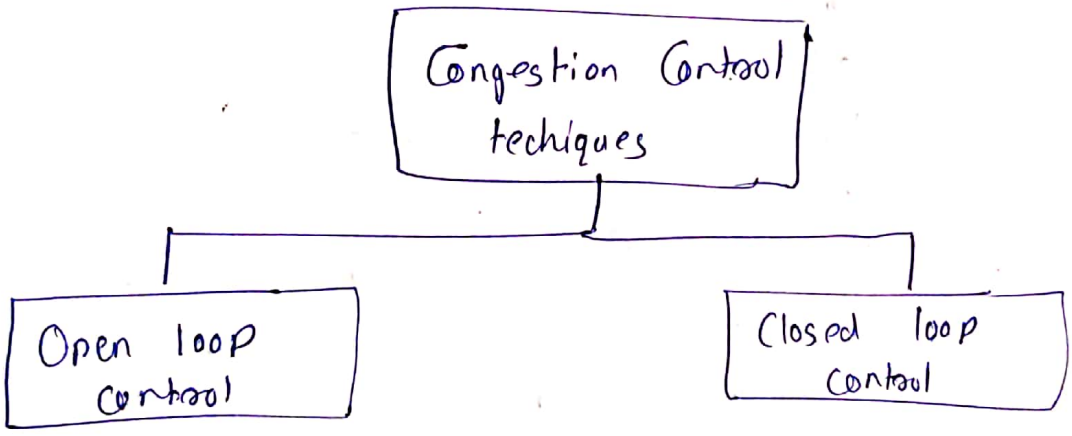
Advantage:

Better Utilization of network bandwidth

Disadvantage:

While waiting to send for ack, transitter may ~~send~~ retransmit ~~new~~ packet

c) Congestion Control Techniques:



Congestion control techniques used to control or prevent congestion.

1) Open loop Control:

In this method Congestion prevented before it happens.

Congestion handeled either by source or destination

Policies Adopted:

1) Retransmission Policy

2) Window Policy

3) Discarding Policy
4) Acknowledgement Policy
5) Admission Policy

2) Closed Loop Control:

This technique is used to treat congestion after it happens.

Techniques used are
1) Backpressure
2) Choke Packet technique
3) Implicit Signaling
4) Explicit Signaling.

Many of these methods handeled by protocols.