18DCE115 CN ASSIGNMENT

Q1) List & explain design issues of data link layers.

↳ 1) Services provided to network layer - Data link layer works as a interface to the network layer. The principal service is transfering data from network layer of sender to network layer of receiver.

2) Frame synchronization - The sorting & ending each frame should be identified.

3) Flow control - Sender must not send data frames at faster rate than receiver can handle.

4) Error Control - Prevent duplication of frames & error should be detected & corrected at destination.

Q2) Explain sliding window mechanism.

↳ Sliding window protocols are data link layer protocols for relable & sequential delivery of data which is used in TCP too.

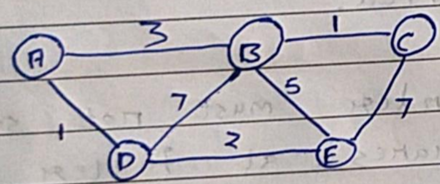
In this sender has buffer called sending window & reciver has buffer called receiving window.

18DCE115

Size of sending window determines sequence number. If sequence number is on bit long, $0 - 2^{n-1}$ seq no. can be assigned. eg. size = 4, seq. no. will be 0, 1, 2, 3, 0, 1, 2, 3...

Size of recieving window is max number of frames that receiver can accept at a time.

Q3) Find the shortest path using Disikstra Algorithm.

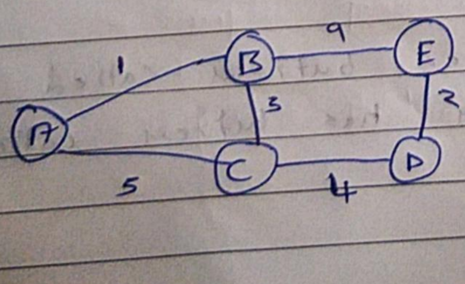


Current Node	Min. distance A	B	C	D	E
A	0	3	∞	1	∞
D	0	3	∞	1	3
B	0	3	4	1	3
E	0	3	4	1	3
C	0	3	4	1	3

suppose A is initial node

Shortest path from A to B is 3
C is 4
D is 1
E is 3

Q4) Find the ~~initial~~ node shortbest past using distance vector algorithm.



↳ Initial tables :-

	A	B	C	D	E
A	0	1	5	∞	∞
B	1	0	3	8	9
C	5	3	0	4	∞
D	∞	∞	4	0	2
E	∞	9	∞	2	0

Using B's table for A, we get A-E = 10
Using D's table for C, we get C-E = 6

⇒ New tables :-

	A	B	C	D	E
A	0	1	5	∞	10
B	1	0	3	∞	9
C	5	3	0	4	6
D	∞	∞	4	0	2
E	10	9	6	2	0

Using C & E's table for B-D, we get min {9+2, 3+4} ∴ BD = 7.
Using B, C, E's table for A-D, we get min {1+7, 5+4, 10+2} ∴ AD = 9.

⇒ Final table :-

	A	B	C	D	E
A	0	1	5	9	10
B	1	0	3	7	9
C	5	3	0	4	6
D	9	7	4	0	2
E	10	9	6	2	0

18DCE115

Q5) List & explain types of delay.

↳ There are 4 types of delay - (i) propogation
(ii) Transmission
(iii) Queuing
(iv) processing

(i) Propagation delay :- Time taken by last bit of packet to reach the destination is called propagation delay. T_p = distance / velocity.

(ii) Transmission delay :- Time taken to transmit a packet from the host to the transmission medium.
T_t = size of data / bandwidth.

(iii) Queuing delay :- The amount of time a packet waits in queue before being processed.

(iv) Processing delay :- Time taken for a packet to be processed.

Q6) List & explain error detection techniques using an example.

↳ The common error detection techniques are
- Simple parity check
- Checksum
- cyclic Redundancy check.

(i) Simple parity check :- In simple parity check, 1 is added to the block if it contains odd number of 1s. otherwise 0 is added.

eg. 100011 → No. of 1 is = 3 = odd => Adding 1.
1000111.

(ii) Checksum :- In this, data is divided into k segments of m bits.

The sender ends the segments are added using 1's compliment to get the sum & sum is complemented to get checksum.

The receivers end again sums & compliments it. if result is 0 then data is accepted, otherwise discarded.

eg. 10011001 11100010 00100100 10000100
k = 4, m = 8

Sender

```
  10011001
  11100010
 ----------
 1 01111011
 ↳        1
 ----------
  01111100
  00100100
 ----------
  10100000
  10000100
 ----------
 100100100
 ↳        1
 ----------
sum = 00100101
Checksum = 11011010
```

Receiver

SUM = 00100101

11011010

00000000

Accepted

(iii) CRC :- It is based on binary division.

A redundant bits called CRC bits are appended to the end of data unit, so resulting data unit becomes exactly divisible by second, predetermined binary number.

eg. 1010000 - Message   CRC generator $= x^3 + 1$
$= 1001$

```
              1011011
1001 ) 1010000 000
       1001
        11000000
        1001
         1010000
         1001
           11000
           1001
            1010
            1001
              11
```

Sender will send 1010000011.

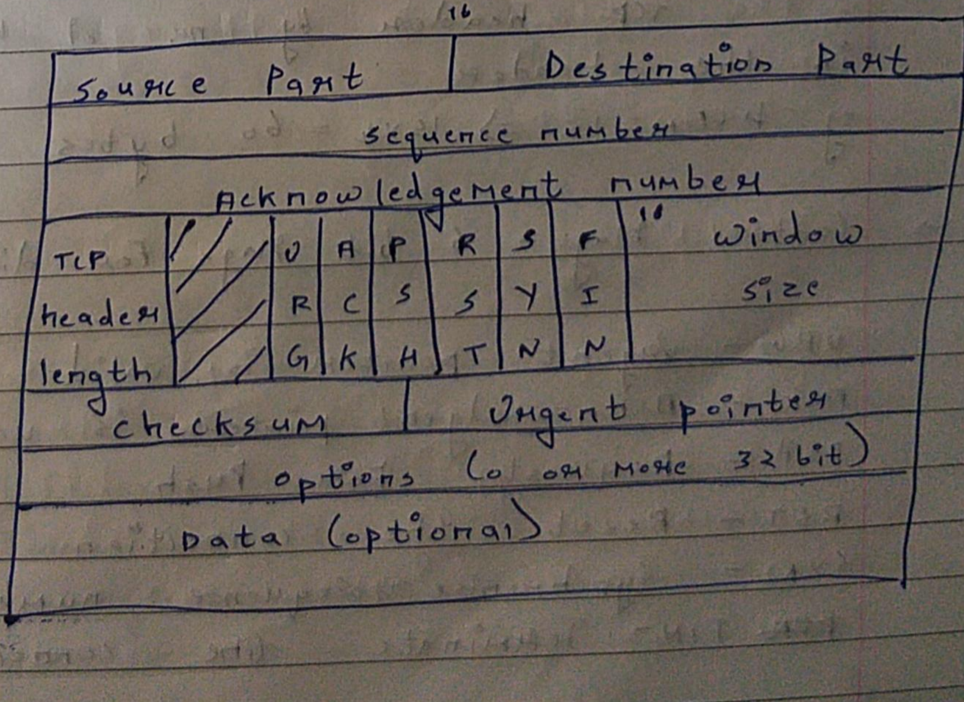
At receiver

```
           1011011
     1001 |1010000011
           1001
            11000011
            1001
             1010011
             1001
              11011
              1001
               1001
               1001
                  0
```

Accepted

Q7) Draw & explain TCP packet header in detail.

↳



18DCE115

→ It can range from 20-60 bytes. 40 bytes are optional

→ Source port & destination port - 16 bit fields that holds the port addr of sender & receiver respec.

→ Sequence no. - 32 bit field that holds sequence number i.e. the byte number of first byte that is sent in that particular segment.

→ Acknowledgement number - 32 bit field that holds acknowledgement number i.e. the byte number that the receiver excepts to receive next.

→ HLEN - 4 bit field that indiates length of TCP header by no. of 4 byte words in the header.
eg. HLEN = 15 ⇒ 15×4 = 60 bytes.

→ Control flags - 6 bit flags for different functions

URG - Urgent pointer is valid
ACK - Acknowledgent pointer is valid
PSH - Request for Push
RST - Reset the connection
SYN - Synchronise sequence numbers
~~PSH~~ FIN - Terminate the connection

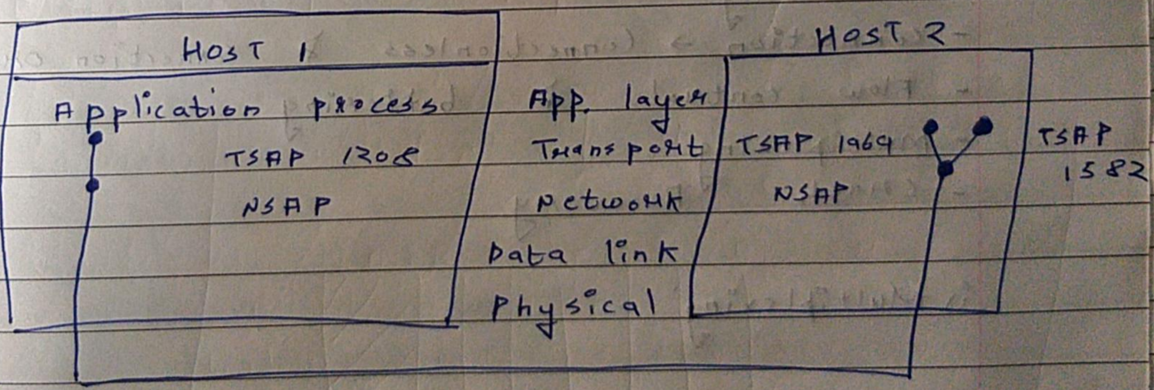
→ Window size :- Tells window size of sending TCP in byte.

→ Checksum :- Holds checksum for error control

→ Urgent pointer :- Used to point to data that is urgently required that needs to reach the receiving process at the earliest.

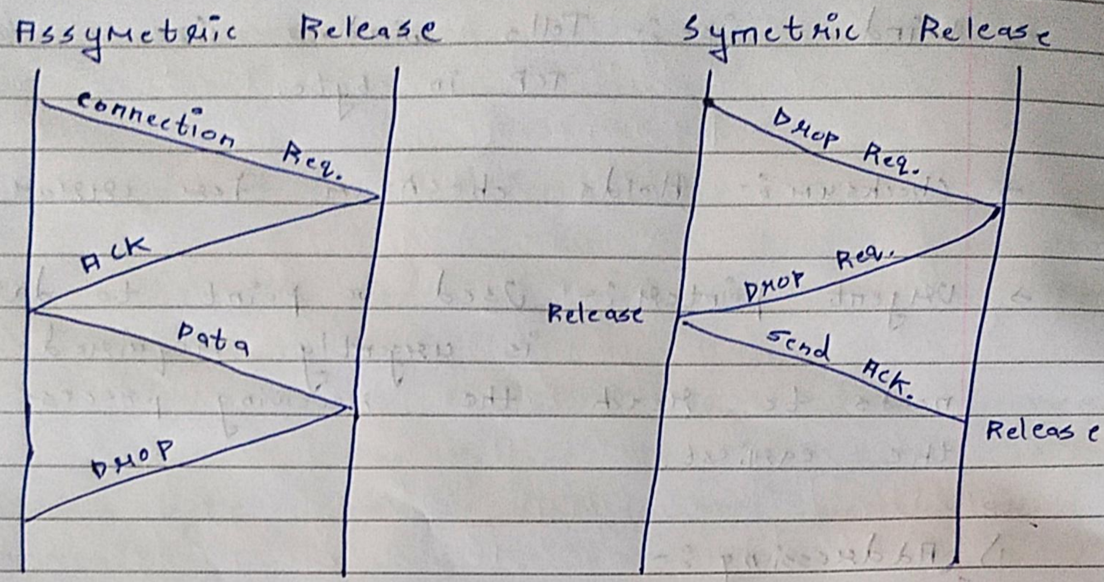
1) Addressing :-

- Application process is connected to the TSAP
- Entity connects to the NSAP
- There are multiple process running within the host.



2) Connection Release :-

- Disconnection b/w two users.
- 2 types :- Assymetric Release & symmetric Relase.



Q8) List out elements of transport protocol & explain any 2 in detail.

↳ Elements of transport layer protocol :-

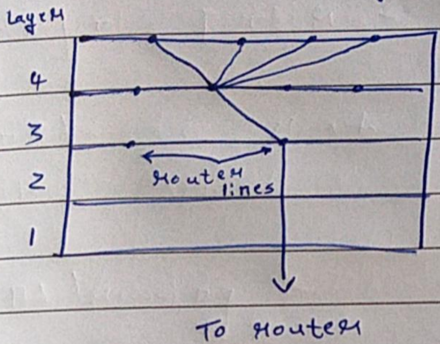
- Addressing
- Connection → Connectionless & Connection Oriented
- Flow control & buffering
- Multiplexing
- Crash Recovery.

i) Multiplexing

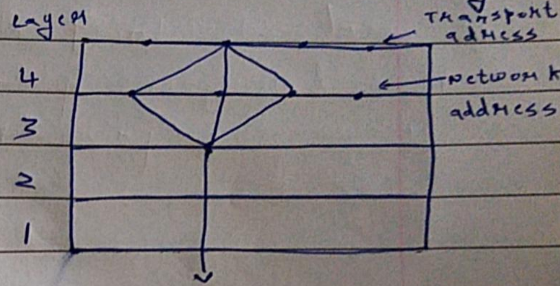
- It is a process of accepting the data from different applications & frowarding to the different applications on different computer.
- At the receiving end, the data is forwarded to correct application.
- This process is called as demultiplexing.

- TCP transmits the packet to the correct application by using the logical channel known as ports.

Upward Multiplexing



Downward Multiplexing



(ii) Crash Recovery

Strategy used by sending host	First ACK then write			First write then ACK		
	AC(W)	AWC	C(AW)	C[AW]	WAC	WC(A)
Always Retransmit	OK	DUP	OK	OK	DUP	DUP
Never Retransmit	LOST	OK	LOST	LOST	OK	OK
Retransmit in S0	OK	DUP	LOST	LOST	DUP	OK
Retransmit in S1	LOST	OK	OK	OK	OK	DUP

OK = Protocol functions correctly
DUP = Protocol generates a duplicate message
LOST = Protocol loses a message

A = Acknowledge
W = Write
C = Crash