~~10/12/21~~
7/04/21

JUNAID. GIRKAR
6000419005?

Q4

a a] link state Routing

- Distance vector routing was used in the ARPANET until 1979, when it was replaced by link state routing
- The idea behind link state routing is fairly simple and can be stated as five parts. Each router must do the following things to make it work:-

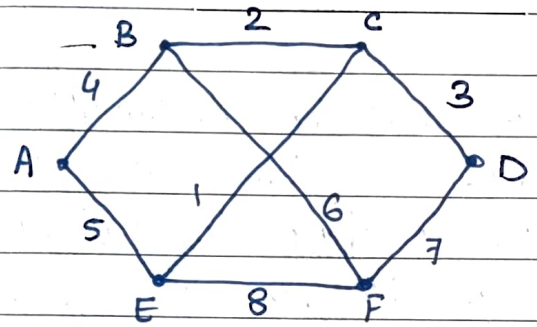
1) Discover its neighbours and learn their network addresses (i.e. learning about the neighbours)
→ when routers are booted, the first task is to learn about its neighbours
→ Does by sending "Hello packets" to end-to-end line
→ Neighbours reply by giving its name.

2) Set the distance or cost metric to each of its neighbours (i.e. setting link costs)
→ cost to reach neighbours can be set automatically or configured by network operators.
→ A common choice is cost $\propto \frac{1}{B.W}$ i.e. more BW = less cost

B.W = Band width
→ Delay of the links can also be a cost
→ Echo packets can be used to find out such delays

JUNAID. GIRKAR
G00041900S7.

3) Construct a packet telling all it has just learnt (i.e. Building link state packets)



A	
B	4
E	5

B	
A	4
C	2
F	6

like wise e

4) send this packets to and receive packets from all other routers (i.e. Distributing the link state routing)
→ For this flooding is used.

So each router has tables of all other routers.

5) Compute the shortest path to every other router (i.e. computing the new routes)

→ For this, algorithms like Dijikstra can be run locally to find out shortest paths to other available neighbours.

JUNAID. GIRKAR
600CU190057

Q5

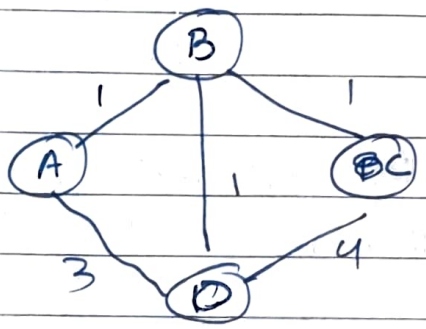
a Pure ALOHA has a vulnerable time of $2 \times T_B$. This is so because there is no rule that defines when the station can send. Slotted ALOHA was invented to improve the efficiency of pure ALOHA.

In slotted ALOHA we divide the time into slots of T_{fr} s and force the station to send only at the beginning of the time slot.

Because a station is allowed to send only at the beginning of the synchronized time slot. If a station misses this moment, it must wait until the beginning of the next time slot. This means that the station which started at the beginning of this slot has already finished sending its frame. Of course there is still the possibility of collision if two stations try to send at the beginning of the same time slot however vulnerable time is now reduced to half i.e T_{fr}.

Slotted ALOHA vulnerable time = T_{fr}.

5 DVR



	A	B	C	D
A	0	1	∞	3
B	1	0	1	1
C	∞	1	0	4
D	∞ 3	1	4	0

	A	B	C	D
A	0	1	∞	∞ 3

→

	A	B	C	D	
A	2	1	3	2	via B
A	6	4	7 ~~8~~	3	via D
A	∞	∞	∞	∞	via C

Minimum =

A	B	C	D
0 ~~3~~	1	2	2

JUNAID. GIRKAR
60004190057

Q6 a]

ANS High-level data link control (HDLC) is a group of communication protocols of the data link layer for transmitting data between network points or nodes. Since it is a data link protocol, data is organized into frames.

HDLC supports two types of transfer modes :-

- Normal response mode (NRM)
- Asynchronous balanced mode (ABM)

HDLC FRAME:

HDLC is a bit-oriented protocol where each frame contains upto six fields. The structure varies according to the type of frame. The fields of a HDLC frame are :-

- Flag
- Address
- Control
- Payload
- FCS

There are 3 types of HDLC frames. The type of frame is determined by the control field of the frame :-

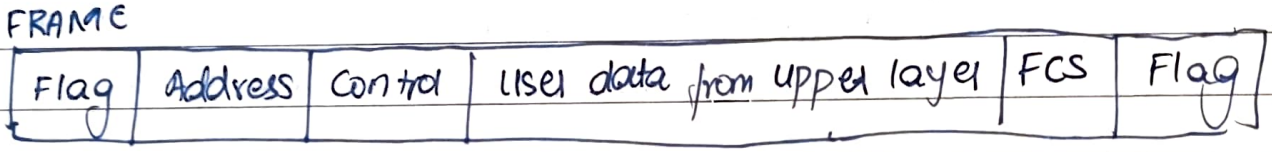
- I-FRAME: I-frames or Information frames carry user data from the network layer. They also include flow and error control information that is piggybacked on user data. The first bit of control field of I-field is 0

JUNAID.GIRKAR
6000419005 7

- S-frame: S frames or supervisory frames do not contain information field. They are used for flow and error control when piggybacking is not required. The first two bits of control field of S-frame is 10

- U-FRAME: U-frames or Un-numbered frames are used for myriad miscellaneous functions, like link management. It may contain an information field if required. The first 2 bits of control field of U-frame is 11

I-FRAME



S-FRAMES

U-Frame.

Junaid.

6b

3 bits reseved for sequence number
Go back and sliding window flow control.

1) size of receiver window = 1

size of sender window = 8 = $2^n = 2^3 = 8$
∴ i.e frames 0 to 7

2) If 4th frame is lost in transit, the receiver won't receive it and hence wont send acknowledgemen

As a result, On timeout, the sender retransmits all the frames in the window