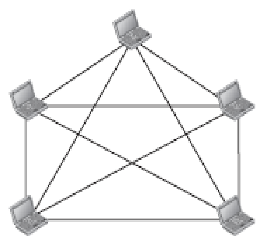
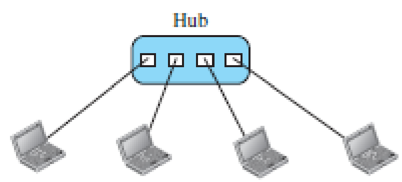
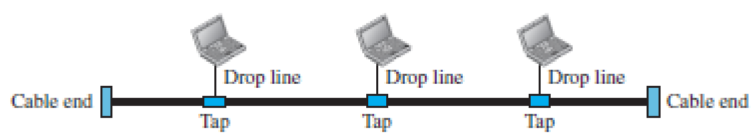
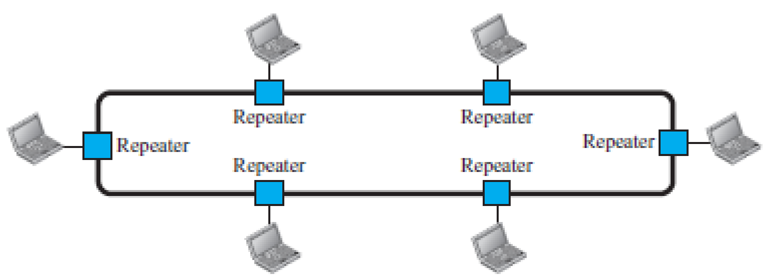
SIES Graduate School of Technology RISE WITH EDUCATION NAAC A+ (Affiliated to University of Mumbai)	***End Semester Examination (R-24) FH 2026*** ***Answer Key with marking scheme***
Branch: CSE (IoT& CSIBCT)	Course: Computer Network
Year/ Semester: SE/IV	Course code: CSEC403
Time: 03 hours	Marks: 80

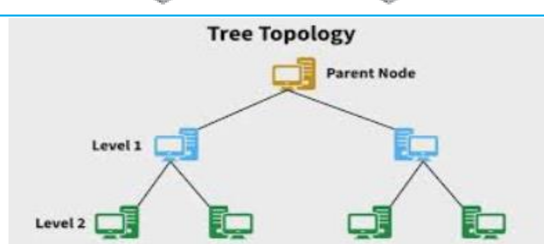
		Marks
Q. 1	Attempt any FOUR. (All questions carry equal marks)	**5**
A.	Any five topologies from the list:Star, Mesh, Ring, Bus and Tree topology, Hybrid Topology with diagram.**1 mark for each topology**	5

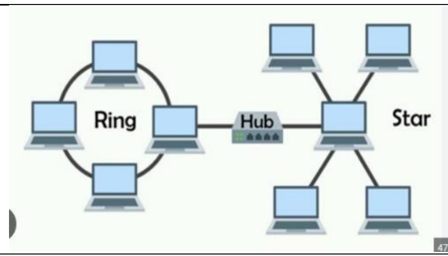
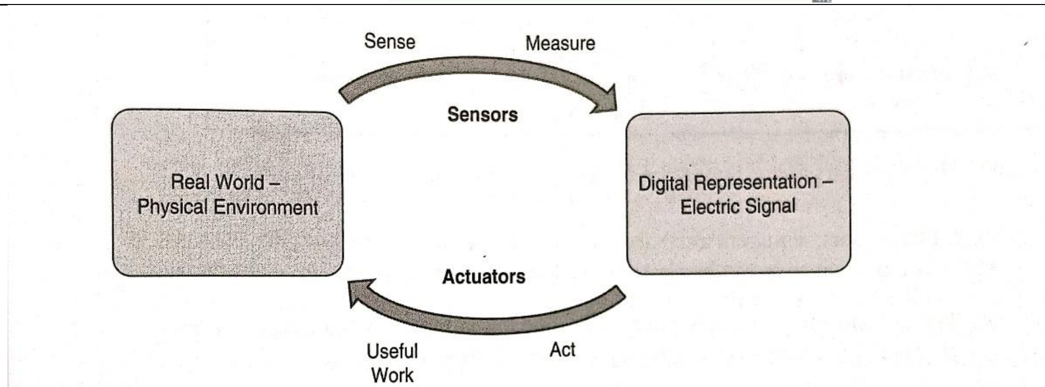










		
B.	 **Figure 3-4** *How Sensors and Actuators Interact with the Physical World* **Diagram-2 Marks** **Explanation:2 Marks**	5
C.	The data-link layer is located between the physical and the network layers. The datalink layer provides services to the network layer and it receives services from the physical layer. It provides node to node communication 1. framing 2. flow control 3. error control 4. MAC Addressing 5. Media Access control Give a brief explanation of all the above points. **1 Marks for each point**	5
D.	An IPv4 address must follow these rules: 1)It contains exactly 4 octets 2)Each octet ranges from 0 to 255 3)Octets are separated by dots (.) i) 221.24.7.8.20-more than 4 octets-**2.5M** ii) 75.45.351.14-one octet is greater than 255-**2.5M**	5

E. 5

PARAMETER	TCP	UDP
Connection	Connection-oriented	Connectionless
Reliability	Reliable as it guarantees data delivery	Unreliable-no guarantee
Speed	Slow	Fast
Ordering of data packets	Ordered and sequential	No order (packets may arrive randomly)
Error Handling	Error detection and correction (retransmission)	Only error detection (no correction)
Flow Control	Yes	No
Congestion Control	Yes	No
Header Size	20-60 bytes(larger)	8bytes(smaller)
Acknowledgement	Yes	No
Use cases	Web browsing, email etc.	VoIP, Video streaming etc.

Each point-0.5 Marks

F. 5

Figure 26.37 *Iterative resolution*

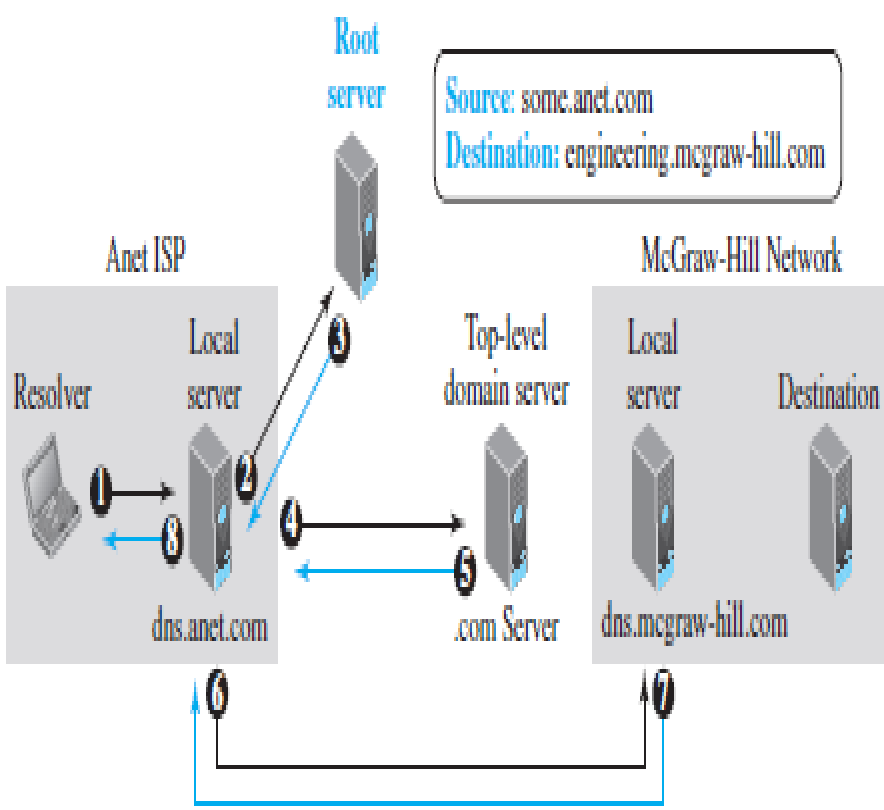
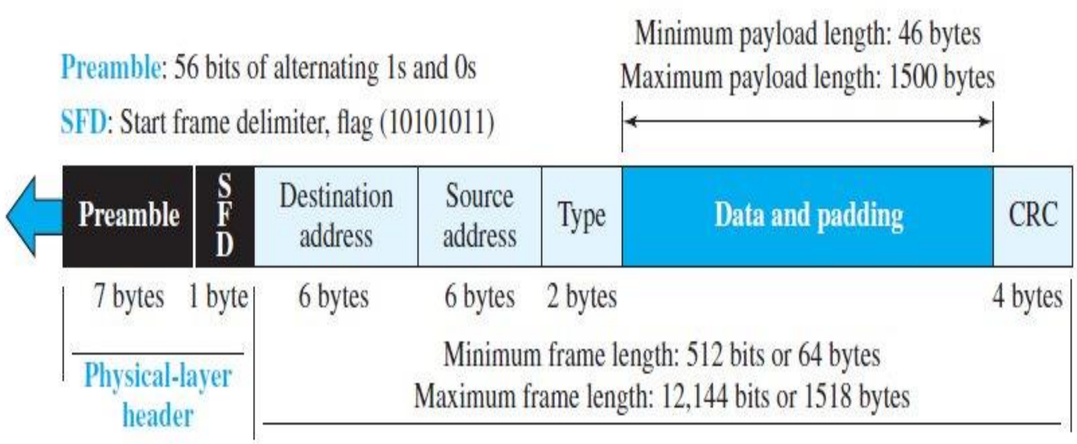


Diagram :2 Marks
Explanation:3 Marks

Q.2	Attempt any FOUR. (All questions carry equal marks)	
A.	Figure 13.3 *Ethernet frame*  **Diagram with proper marking of bytes:4 marks** **Explanation of fields:6 Marks**	10
B.		10

Given

Generator = 10101

① Data bit sequence = 110010101

$G(x) = 1 \cdot x^4 + 0 \cdot x^3 + 1 \cdot x^2 + 0 \cdot x^1 + 1 \cdot x^0$

degree of polynomial = 4.

no. of zeroes to be appended = 4.

∴ appended data = 1100101010000

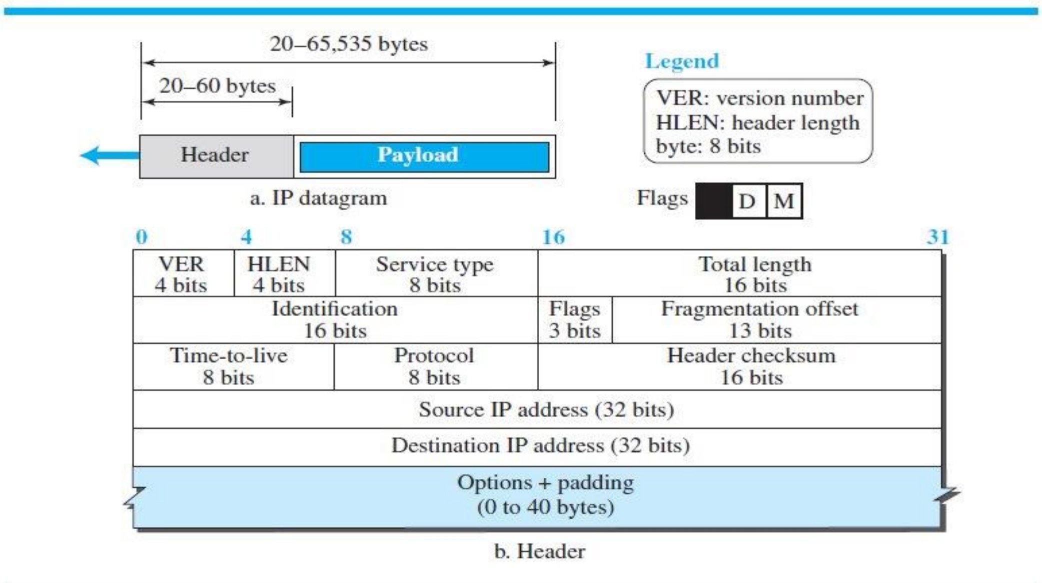
```
           111100111
10101 ) 1100101010000
        10101
        011000
         10101
         011011
          10101
          011100
           10101
           010011
            10101
            001100
             00000
             011000
              00000
              11000
              10101
              011010
               10101
               011110
                10101
                01011
```

Transmitted codeword = 1100101011011

② String of bits received = 110011001100

```
          11111100
10101 ) 110011001100
        10101
        011001
         10101
         011000
          10101
          011010
           10101
           011111
            10101
            01010
             10101
             000000
              000000
              000000
```

As the remainder is zero, no error is ~~detected~~ detected.

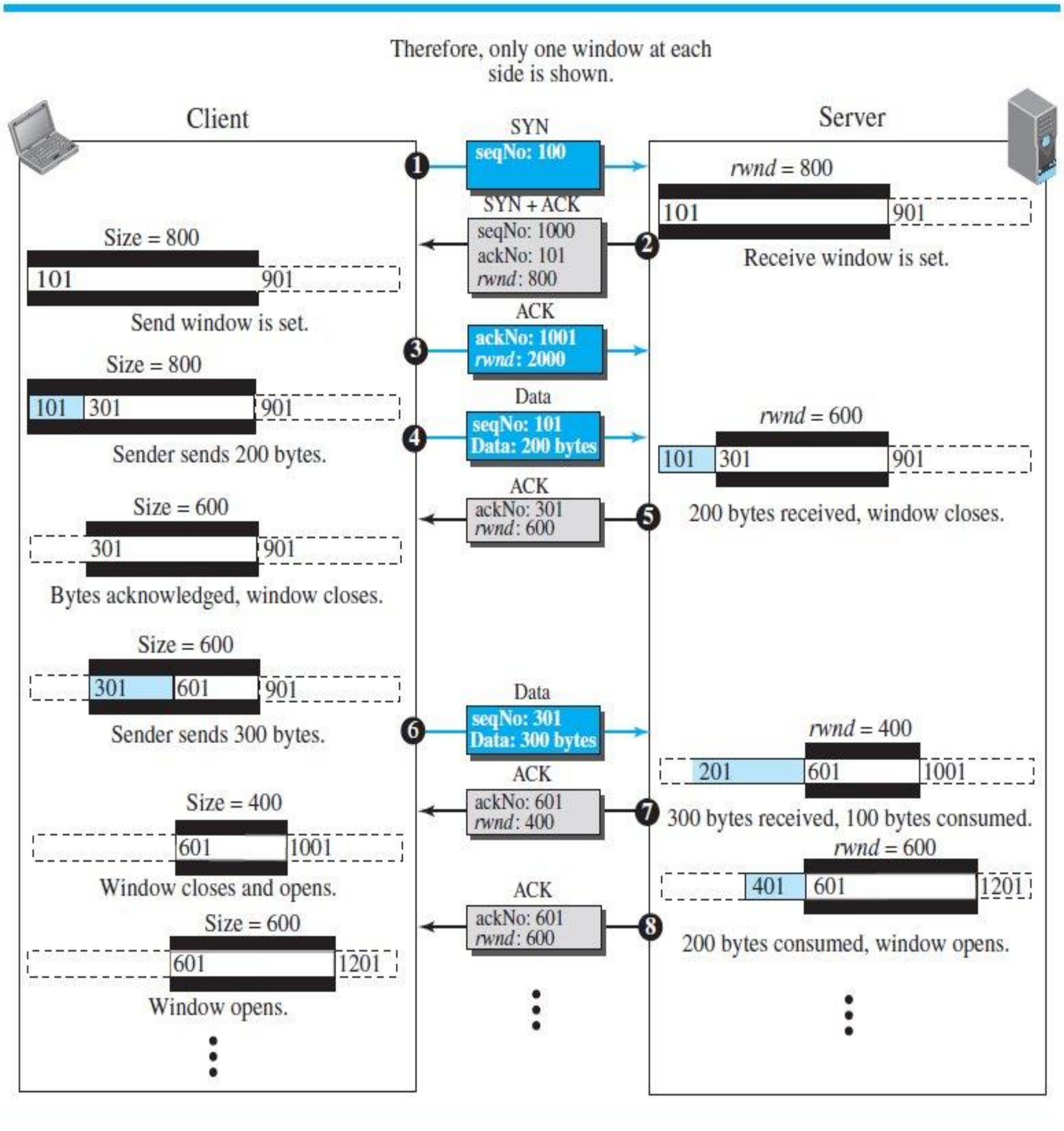
	i)6 Marks **ii)4 Marks**	
C.	**Figure 19.2** *IP datagram*  **Diagram with bytes also marked properly:5 marks** **Explanation:5 Marks**	10
D.	**Distance vector Routing Algorithm with explanation :6 Marks** **Count to infinity Problem:-Two node instability -4Marks**	10

```
Distance_Vector_Routing ( )
{
    // Initialize (create initial vectors for the node)
    D[myself] = 0
```

'ETWORK LAYER

'able 20.1 *Distance-Vector Routing Algorithm for a Node (continued)*

```
    for (y = 1 to N)
    {
        if (y is a neighbor)
            D[y] = c[myself][y]
        else
            D[y] = ∞
    }
    send vector {D[1], D[2], …, D[N]} to all neighbors
    // Update (improve the vector with the vector received from a neighbor)
    repeat (forever)
    {
        wait (for a vector D_w from a neighbor w or any change in the link)
        for (y = 1 to N)
        {
            D[y] = min [D[y], (c[myself][w] + D_w[y ])]     // Bellman-Ford equation
        }
        if (any change in the vector)
            send vector {D[1], D[2], …, D[N]} to all neighbors
    }
} // End of Distance Vector
```

E.	**Figure 24.20** *An example of flow control*  **Diagram :5 Marks** **Explanation 5 Marks**	10

F. (10)

Figure 18.25 *DHCP message format*

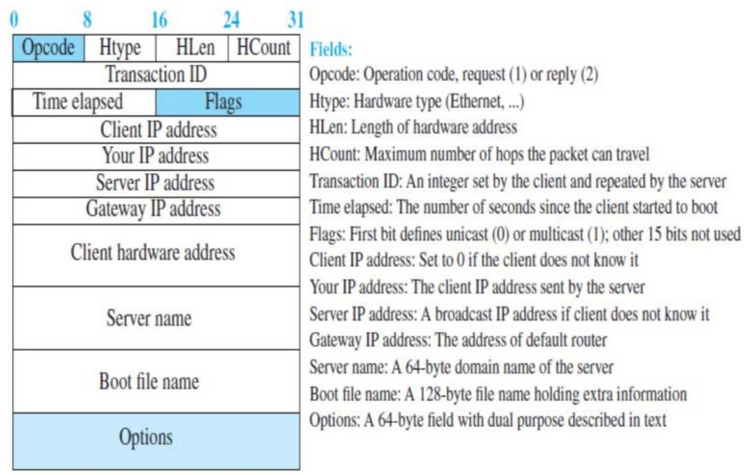


Diagram with proper markings :5Marks
Explanation:5 Marks

Q.3 Attempt any FOUR. (All questions carry equal marks)

A. (5)

Feature	Twisted Pair	Coaxial Cable	Fiber Optic
Bandwidth	Low to Medium	Medium	Very High
Data Rate	Up to 1–10 Gbps	Higher than Twisted Pair	Extremely High (Tbps)
Transmission Distance	Short (up to 100 m)	Moderate (Few km)	Very Long (Tens–Hundreds km)
Cost	Low	Medium	High
Installation	Easy	Moderate	Difficult
Noise Immunity	Low (Susceptible to EMI)	Better Shielding	Very High (Immune to EMI)
Signal Attenuation	High	Moderate	Very Low
Security	Low	Moderate	Very High

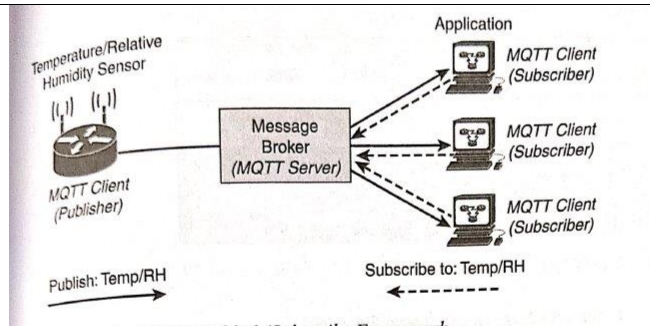
Write any five points.1 Mark for each point.

B. (5)

The channel allocation problem is the problem of sharing a common communication channel among multiple users efficiently. It arises because multiple devices may try to transmit simultaneously, causing collisions and data loss.
Types of channel allocation:
Static allocation: Fixed division of channel (e.g., TDM) – no collisions but inefficient
Dynamic allocation: On-demand access (e.g., ALOHA, CSMA/CD) – better utilization but collisions possible. Ensures efficient, fair, and reliable use of the communication medium.

C. (5)

In this protocol, the sender sends one frame at a time and waits for an acknowledgment before sending the next one. To detect corrupted frames, we need to add a CRC to each data frame. When a frame arrives at the receiver site, it is checked. If its CRC is incorrect, the frame is corrupted and silently discarded. The silence of the receiver is a signal for the sender that a frame was either corrupted or lost. Every time the sender sends a frame, it starts a timer. If an acknowledgment arrives before the timer expires, the timer is stopped and the sender sends the

	next frame (if it has one to send). If the timer expires, the sender resends the previous frame, assuming that the frame was either lost or corrupted. This means that the sender needs to keep a copy of the frame until its acknowledgment arrives. When the corresponding acknowledgment arrives, the sender discards the copy and sends the next frame if it is ready. Only one frame and one acknowledgment can be in the channels at any time.	
D.	ARP Protocol -**3 Marks** RARP PROTOCOL-**2 Marks**	5
E.	1.Retransmission Timer-2 Marks 2.Persistence Timer-1 Mark 3.Keep-Alive Timer-1 Mark 4.Time-Wait Timer-1 Mark	5
F.	 Figure 6-10 *MQTT Publish/Subscribe Framework* **Diagram:2 Marks** **Explanation:3 Marks**	