

Sub:	Computer Network And Protocols					Sub Code:	BEC702
Date:	10/10/2025	Duration	90 Minutes	Max Marks:	50	Sem / Sec:	

Answer any FIVE FULL Questions

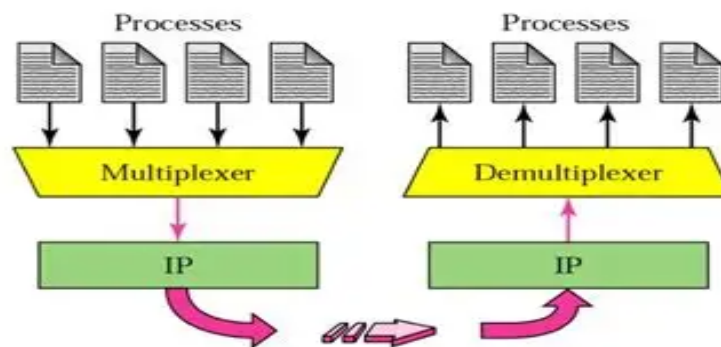
1	With a neat diagram, explain i) encapsulation and decapsulation ii) Multiplexing and Demultiplexing, iii) Addressing. i) Encapsulation and Decapsulation ◆ Encapsulation • When data is sent from sender → receiver, it moves down the OSI/TCP-IP layers. • Each layer adds its own header (and sometimes trailer) around the data. • This process is called Encapsulation. ◆ Decapsulation • At the receiver side, the reverse happens. • Each layer removes its own header and passes the remaining data upward. • This is called Decapsulation. ii) Multiplexing and Demultiplexing ◆ Multiplexing • At the sender side (Transport Layer): Multiple applications (e.g., browser, email, WhatsApp) generate data simultaneously.	[10]
---	---	------

- Transport layer adds **unique identifiers (port numbers)** so they can share the same network path.
- This process is **Multiplexing**.

♦ Demultiplexing

- **At the receiver side:**
The transport layer looks at the **port number** in each segment and delivers data to the correct application.
- This process is **Demultiplexing**.

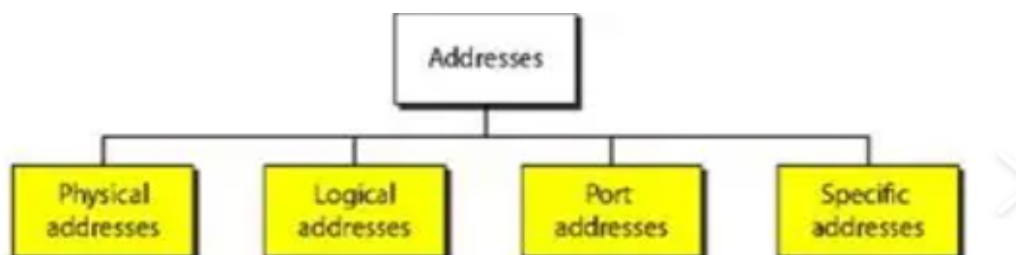
Multiplexing and Demultiplexing

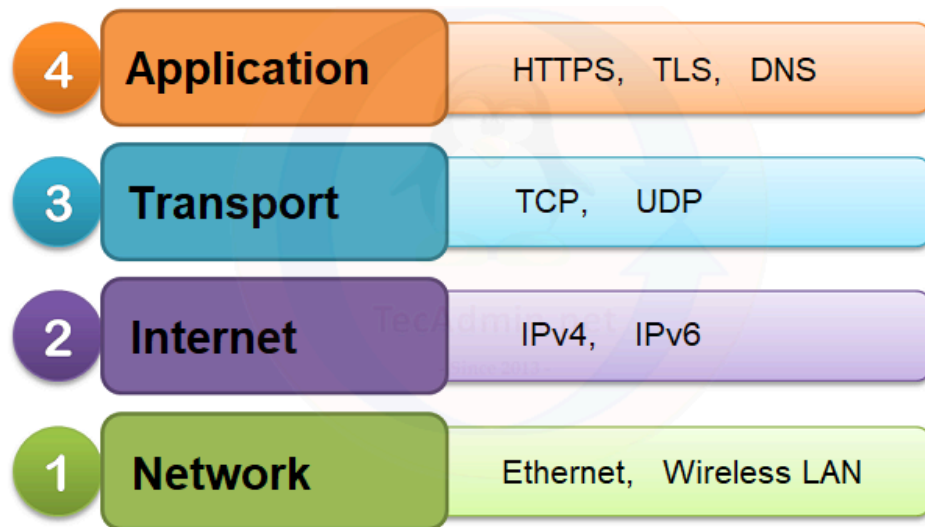


iii) Addressing

For communication, **different layers use different addresses**:

1. **Application Layer** → *Application-specific addresses* (e.g., URL, email ID).
2. **Transport Layer** → *Port numbers* (e.g., HTTP: 80, FTP: 21).
3. **Network Layer** → *Logical address* (e.g., IP address).
4. **Data Link Layer** → *Physical address* (MAC address).





Layers of TCP/IP

1. Application Layer

- Provides services to the user.
- Protocols: **HTTP, FTP, SMTP, DNS, SNMP, Telnet**
- Equivalent to **Application + Presentation + Session** layers of OSI.

2. Transport Layer

- Provides **process-to-process delivery**.
- Ensures reliable or unreliable communication.
- Protocols:
 - **TCP (Transmission Control Protocol)** → Reliable, connection-oriented.
 - **UDP (User Datagram Protocol)** → Unreliable, fast, connectionless.

3. Internet Layer

- Provides **host-to-host delivery** across networks.
- Defines logical addressing (IP addresses).
- Protocols: **IP (IPv4/IPv6), ICMP, IGMP, ARP, RARP.**

4. Network Access Layer (Link Layer)

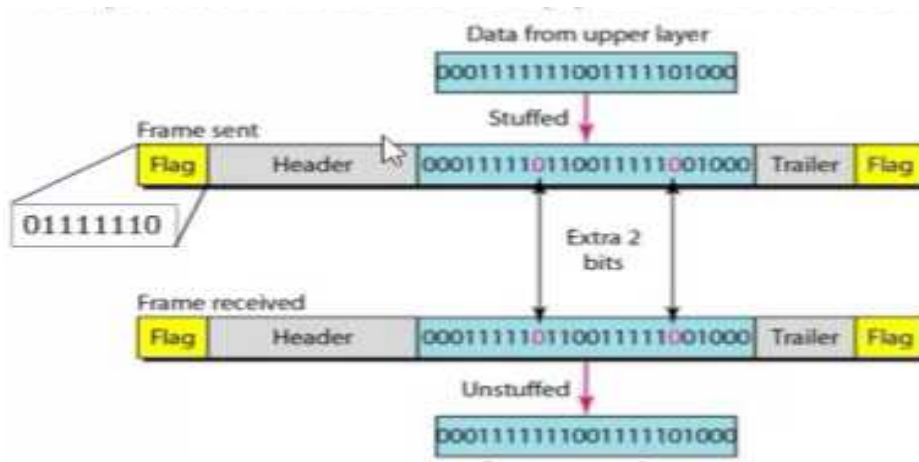
- Provides **node-to-node delivery** over the physical network.
- Deals with **framing, MAC addressing, physical transmission.**
- Protocols: **Ethernet, Wi-Fi, PPP, Frame Relay, ATM.**

3

Describe the concept of (i) Bit stuffing and unstuffing (ii) Byte stuffing and unstuffing with suitable figures and examples.

[10]

Bit Stuffing and Unstuffing



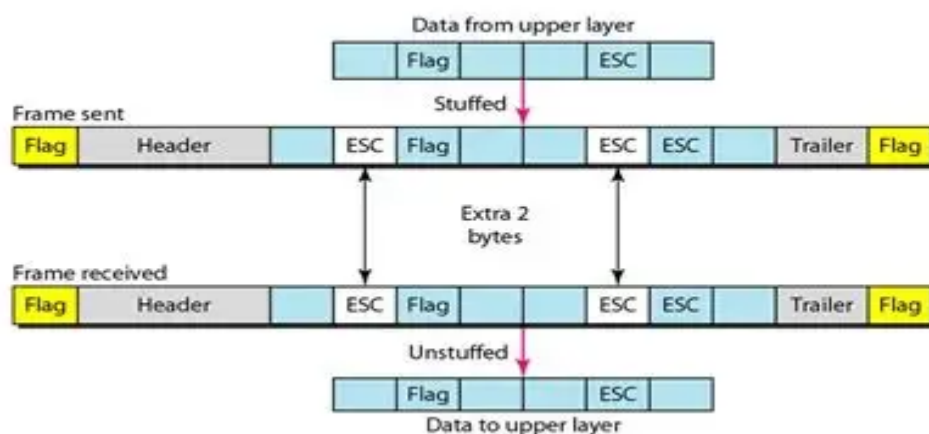
◆ Bit Stuffing

- **Sender side:**
 - Every time 5 consecutive 1's appear in data, the sender **inserts a 0** after them.
- This ensures the flag 01111110 does not occur inside the data field.

◆ Bit Unstuffing

- **Receiver side:**
 - Whenever it finds 5 consecutive 1's followed by a 0, it **removes the 0**.
- This restores the original data.

Byte stuffing and unstuffing



◆ Byte Stuffing

● Sender side:

- Whenever a **special flag byte** (like DLE) appears in data, it **inserts another DLE before it**.

◆ Byte Unstuffing

● Receiver side:

- If it sees **DLE DLE**, it removes one DLE (keeps only original data).

4

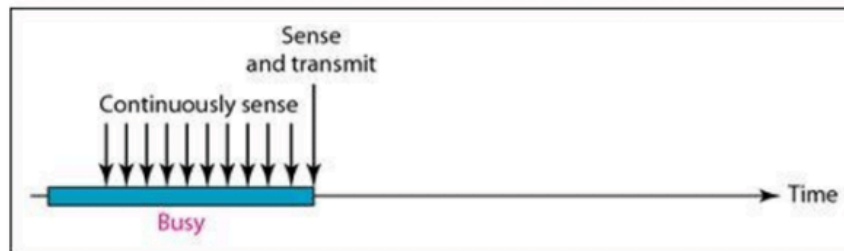
a. Explain the Behaviour of three persistence methods in CSMA with neat figures and flow Diagrams

[10]

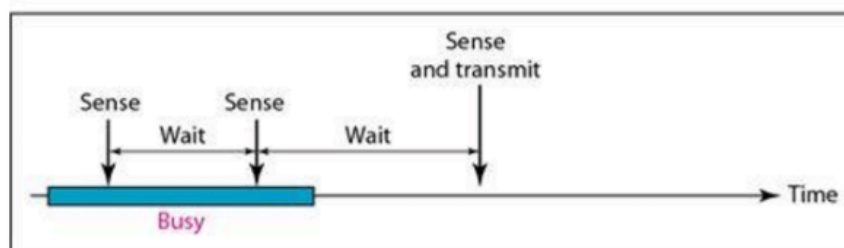
b. Identify if the following 802.3 MAC addresses are unicast, multicast or broadcast :

- 47:20:1B:2E:08:EE
- EE:FF:10:01:11:00
- FF:FF:FF:FF:FF:FF

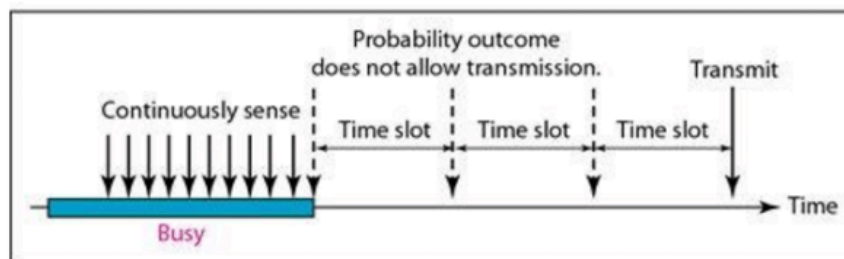
a) three persistence methods in CSMA with neat figures and flow diagrams



a. 1-persistent



b. Nonpersistent



c. p-persistent

- In CSMA, before transmitting, a station **listens (senses)** the channel to check if it is **idle or busy**.
- If idle → station transmits.
- If busy → depends on the **persistence method** used.

There are **three persistence methods**:

1. 1-Persistent CSMA

♦ Behaviour:

- A station that has data to send **continuously senses** the channel.
- If channel is **idle** → **transmit immediately (with probability = 1)**.
- If channel is **busy** → **wait until it becomes idle**, then transmit **immediately**.
- Problem: Causes **collisions**, since multiple stations may start at the same time.

2. Non-Persistent CSMA

♦ Behaviour:

- A station senses the channel.
- If channel is **idle** → **transmit immediately**.
- If channel is **busy** → **wait for a random time**, then sense channel again.
- Advantage: **Reduces collisions** (since stations don't keep waiting aggressively).
- Disadvantage: **Longer delays**.

3. p-Persistent CSMA

♦ Behaviour:

- Used in **slotted channels**.
- If channel is **idle**:
 - Transmit with probability **p**.
 - Wait for next slot with probability **(1 - p)**.
- If channel is **busy** → **wait until next slot** and repeat the process.
- Balances **throughput** and **collision probability**.

1. 47:20:1B:2E:08:EE

- First byte = 47 (hex)
- In binary: 47 = 01000111
- LSB (last bit) = 1 → **Multicast Address** ✓

2. EE:FF:10:01:11:00

- First byte = EE (hex)
- In binary: EE = 11101110
- LSB = 0 → **Unicast Address** ✓

3. FF:FF:FF:FF:FF:FF

- All bits are 1 → This is the **Broadcast Address**

5

a) A pure ALOHA network transmits 200-bit frames on a shared channel of 200 kbps. What is the throughput if the system (all stations together) produces

- a. 1000 frames per second?
- b. 500 frames per second?
- c. 250 frames per second?

b) Find the class of the following IP addresses:

i) 237.14.2.1 ii) 208.25.54.12 iii) 129.14.6.8 iv) 114.34.2.9

Case (a) 1000 frames/sec

- $G = 1000 \times 0.001 = 1$ $G = 1000 \times 0.001 = 1$
- $S = 1 \cdot e^{-2(1)} = e^{-2} = 0.1353$ $S = 1 \cdot e^{-2(1)} = e^{-2} = 0.1353$

So: $S_{fps} = 0.1353 / 0.001 = 135.3$ frames/sec

Case (b) 500 frames/sec

- $G = 500 \times 0.001 = 0.5$ $G = 500 \times 0.001 = 0.5$
- $S = 0.5 \cdot e^{-1} = 0.1839$ $S = 0.5 \cdot e^{-1} = 0.1839$

So:

[10]

$$S_{fps} = 0.1839 / 0.001 = 183.9 \text{ frames/sec}$$

Case (c) 250 frames/sec

- $G = 250 \times 0.001 = 0.25$ $G = 250 \times 0.001 = 0.25$ $G = 250 \times 0.001 = 0.25$
- $S = 0.25 \cdot e^{-0.5} = 0.1516$ $S = 0.25 \cdot e^{-0.5} = 0.1516$ $S = 0.25 \cdot e^{-0.5} = 0.1516$

So: $S_{fps} = 0.1516 / 0.001 = 151.6 \text{ frames/sec}$

(i) 237.14.2.1

- First octet = **237**
- Range 224–239 → **Class D (Multicast)**

(ii) 208.25.54.12

- First octet = **208**
- Range 192–223 → **Class C**

(iii) 129.14.6.8

- First octet = **129**
- Range 128–191 → **Class B**

(iv) 114.34.2.9

- First octet = **114**
- Range 1–126 → **Class A**

6

An organization is granted a block of addresses with the beginning address 14.24.74.0/24. The organization needs to have 3 sub blocks of addresses to use in its three subnets: one sub block of 10 addresses, one sub block of 60 addresses, and one sub block of 120 addresses. Design the sub blocks. Find out the total number of unused addresses.

[10]

ANS:

Subnet A — for 120 hosts (largest)

- For **120 hosts** → need ≥ 120 usable.
Smallest block with usable ≥ 120 is **/25**:
addresses = $2^{32-25} = 2^7 = 128$. usable = $128 - 2 = 126$ (OK).

Hence

- **Network:** 14.24.74.0/25
- **Address range:** 14.24.74.0 — 14.24.74.127 (128 addresses)
- **Usable hosts:** 14.24.74.1 — 14.24.74.126 (126 hosts)
- **Broadcast:** 14.24.74.127

Subnet B — for 60 hosts

- For **60 hosts** → need ≥ 60 usable.
Smallest block is /26:
addresses = $2^{32-26} = 2^6 = 64$. usable = $64 - 2 = 62$ (OK).

Hence

- Next free address after .127 is .128.
- **Network:** 14.24.74.128/26
- **Address range:** 14.24.74.128 — 14.24.74.191 (64 addresses)
- **Usable hosts:** 14.24.74.129 — 14.24.74.190 (62 hosts)
- **Broadcast:** 14.24.74.191

Subnet C — for 10 hosts

- For **10 hosts** → need ≥ 10 usable.
Smallest block is /28:
addresses = $2^{32-28} = 2^4 = 16$. usable = $16 - 2 = 14$ (OK).

Hence

- Next free address after .191 is .192.
- **Network:** 14.24.74.192/28
- **Address range:** 14.24.74.192 — 14.24.74.207 (16 addresses)
- **Usable hosts:** 14.24.74.193 — 14.24.74.206 (14 hosts)
- **Broadcast:** 14.24.74.207

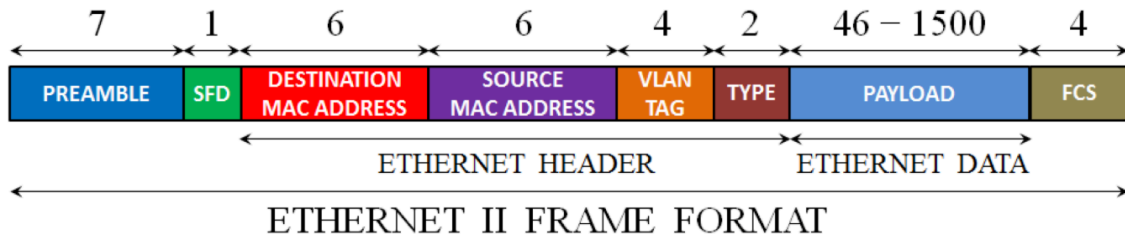
Remaining (unused) addresses

- Addresses left: 14.24.74.208 — 14.24.74.255 → total **48** addresses unused.

Subnet for	Network	Prefix	Address range	Usable hosts (count)	Broadcast
120 hosts	14.24.74.0	/25	.0 – .127	.1 – .126 (126)	.127
60 hosts	14.24.74.128	/26	.128 – .191	.129 – .190 (62)	.191
10 hosts	14.24.74.192	/28	.192 – .207	.193 – .206 (14)	.207
Unused	(leftover)	—	.208 – .255	48 addresses unused	(can be /27 + /28)

7 Explain the IEEE frame format of standard Ethernet with neat diagram. What are the minimum and maximum frame lengths?.

[10]



A. Preamble (7 bytes)

- A pattern of **10101010** repeated 7 times.
- **Purpose:** Allows the receiver to **synchronize** its clock with the sender before actual data arrives.

B. Start Frame Delimiter (SFD) (1 byte)

- Pattern **10101011**.
- Marks the **exact start of the frame**.

C. Destination MAC Address (6 bytes)

- The **hardware address of the receiving device**.
- Tells the network which device should accept the frame.

D. Source MAC Address (6 bytes)

- The **hardware address of the sending device**.

E. Length / Type (2 bytes)

- Specifies either:
 - **Length** of the payload (IEEE 802.3), or

- **Type of protocol** in the payload (Ethernet II, e.g., IP, ARP).

F. Data / Payload (46–1500 bytes)

- The **actual information** being sent.
- If payload < 46 bytes → **padding is added** to meet minimum frame size.

G. Frame Check Sequence (FCS) (4 bytes)

- A **CRC checksum** used by the receiver to detect **errors in transmission**.
- If the FCS check fails → frame is **discarded**.

Minimum and Maximum Frame Length

- **Minimum frame length:** 64 bytes (to ensure CSMA/CD collision detection works).
- **Maximum frame length:** 1518 bytes (excluding preamble) for standard Ethernet.
- **With preamble + SFD**, total bytes transmitted = 1526 bytes.