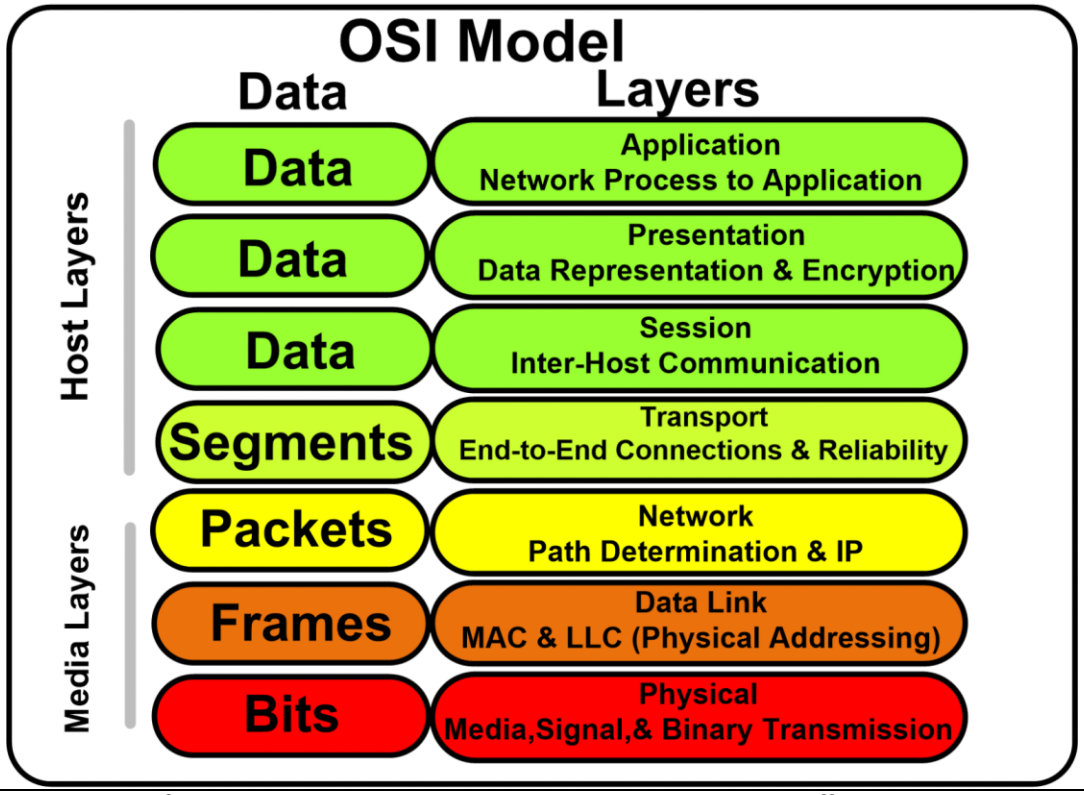
(Affiliated to University of Mumbai)	***End Semester Examination (R-24/R-25) SH 2026 Answer Key with marking scheme***
Branch: CE	Course: Computer Networks
Year/ Semester: III/V	Course code: CEC503
Time: 03 hours	Marks: 80

		Marks
Q. 1	All questions are compulsory.	
A.	Explain the OSI reference model with a neat diagram. Describe the functions of any five layers.? **Sol:** OSI (Open Systems Interconnection) model is a seven-layer reference model that explains how data is transmitted between devices over a network. Functions of the seven layers: 1. Physical: Transmits raw bits through cables, fibre or wireless signals. 2. Data Link: Creates frames, uses MAC addresses and detects transmission errors. 3. Network: Routes packets between networks using logical IP addresses. 4. Transport: Provides process-to-process delivery, reliability, flow control and segmentation. 5. Session: Establishes, manages and terminates communication sessions. 6. Presentation: Handles data translation, encryption/decryption and compression. 7. Application: Provides network services to applications, such as HTTP, FTP and SMTP. Any five correctly explained layer functions satisfy the question.	



B. Explain the framing techniques used at the Data Link Layer. Differentiate between character count, byte stuffing, and bit stuffing?

Sol:

Framing divides a continuous stream of bits into identifiable units called frames at the Data Link Layer.

Technique	Working	Limitation
Character count	A header field specifies the number of bytes in the frame.	If the count is corrupted, frame boundaries may be lost.
Byte stuffing	Special flag bytes mark frame boundaries. An escape byte is inserted when a flag or escape byte occurs in the data.	Adds extra bytes to the data.
Bit stuffing	A 0 is inserted after every sequence of five consecutive 1s in the data. The receiver removes the inserted 0.	Adds extra bits when required.

Example of bit stuffing:

- Data: `01111110`
- Stuffed data: `011111010`

The inserted `0` prevents the data from being mistaken for a flag sequence.

C.

Differentiate between IPv4 and IPv6 based on address size, address representation, header format, and address configuration?

Sol:

IPv6 provides a much larger address space, simplified base-header processing and improved support for modern networks.

Feature	IPv4	IPv6
Address size	32 bits	128 bits
Representation	Dotted decimal, e.g. `192.168.1.10`	Hexadecimal separated by colons, e.g. `2001:db8::10`
Header	Variable length, 20–60 bytes	Fixed base header of 40 bytes
Configuration	Manual or DHCP	SLAAC, DHCPv6 or manual
Address capacity	Approximately 2^{32} addresses	2^{128} addresses
Broadcast	Supports broadcast	No broadcast; uses multicast instead

D.

Given: 192.168.1.0/24, to be divided into 4 equal subnets.

Subnet mask

- 4 subnets need 2 extra bits ($2^2 = 4$).
- New prefix = /24 + 2 = **/26**
- Subnet mask = **255.255.255.192**

Usable hosts per subnet

- Host bits remaining = 32 − 26 = 6
- Usable hosts = $2^6 - 2$ = **62**

First subnet

- Network address = **192.168.1.0**
- Broadcast address = **192.168.1.63**
- (Usable range: 192.168.1.1 to 192.168.1.62)

E.

Differentiate between circuit switching, message switching, and packet switching with respect to connection establishment, data transmission, delay, resource utilization, and applications.

Sol:

Feature	Circuit Switching	Message Switching	Packet Switching
Connection	Dedicated path established first	No dedicated path	Usually no dedicated path
Transmission	Continuous data stream	Entire message is forwarded	Message is divided into packets
Storage	Limited intermediate storage	Store-and-forward of entire message	Store-and-forward of packets
Delay	Setup delay, then predictable transmission	Often high	Generally efficient; delay varies
Resource use	Resources may remain reserved	Shared resources	Shared resources
Example	Traditional telephone network	Historical telegraph/store-and-forward systems	Internet

Circuit switching provides dedicated resources, message switching forwards complete messages, and packet switching divides data into smaller packets for efficient network transmission.

F.	Difference each carries equal marks	
Q.2A	Consider the following network with link costs: A–B = 1, A–C = 5, B–C = 3, B–E = 9, C–D = 4, and D–E = 2. Apply the Distance Vector Routing algorithm with router A as the source. Show the distance-vector updates and determine the shortest path and minimum cost from A to every other router. Explain how routers exchange routing information and update their routing tables **Sol:** Given link costs: A–B = 1, A–C = 5, B–C = 3, B–E = 9, C–D = 4, D–E = 2. The Distance Vector algorithm uses the Bellman–Ford equation: $$D_x(y) = \min_v \{c(x,v) + D_v(y)\}$$ Here, $D_x(y)$ is the estimated cost from router x to destination y, and $c(x,v)$ is the cost of the direct link to neighbour v. Step 1: Initial distances from A	

Destination	Initial cost
A	0
B	1
C	5
D	∞
E	∞

Step 2: Exchange distance vectors and update routes

- To C: direct cost is 5, but via B the cost is $1 + 3 = 4$. Update C to 4.
- To D: via C, cost is $4 + 4 = 8$.
- To E: via B, cost is $1 + 9 = 10$; via D, cost is $8 + 2 = 10$.

After the distance vectors have converged, the final routing table at A is:

Destination	Shortest path	Minimum cost
A	A	0
B	A → B	1
C	A → B → C	4
D	A → B → C → D	8
E	A → B → C → D → E	10

There is also an equal-cost route to E: A → B → E, with cost 10.

Each router periodically sends its distance vector to its neighbouring routers. On receiving a vector, it adds the cost of the link to that neighbour and updates its routing table whenever a lower-cost route is found. Updates continue until the network converges.

OR

Q.2B.

Consider a network with the following link costs:

P–Q = 6, P–R = 3, Q–R = 2, Q–S = 5, R–S = 3, R–T = 7, and S–T = 1.

Apply Dijkstra's shortest-path algorithm with P as the source router.

Sol:

Given:

P–Q = 6, P–R = 3, Q–R = 2, Q–S = 5, R–S = 3, R–T = 7, S–T = 1.

Source vertex: P.

Step 1: Initialize distances

$$d(P) = 0, \quad d(Q) = \infty, \quad d(R) = \infty, \quad d(S) = \infty, \quad d(T) = \infty$$

After processing P, the distances are Q = 6 and R = 3.

Step 2: Update distances iteratively

Selected vertex	Q	R	S	T
P	6	3	∞	∞
R	5	3	6	10
Q	5	3	6	10
S	5	3	6	7
T	5	3	6	7

The values shown are the current minimum distances from P; the selected vertex is finalized at each step. After processing R, Q improves from 6 to 5, S becomes 6, and T initially becomes 10. Processing S then improves T to 7.

Step 3: Shortest paths

Destination	Shortest path	Cost
P	P	0
R	P → R	3
Q	P → R → Q	5
S	P → R → S	6

T	P → R → S → T	7

Dijkstra's algorithm repeatedly selects the unvisited vertex with the smallest tentative distance and relaxes its outgoing edges until all reachable vertices are finalized.

Q.3A

Explain the working of Stop-and-Wait ARQ and Sliding Window protocols used for reliable data transmission at the Data Link Layer. Compare these protocols in terms of their working principles, transmission efficiency, acknowledgement mechanism, and error recovery.

Further, explain the working of the following sliding-window ARQ protocols with suitable, labelled diagrams:

1. Go-Back-N ARQ: Explain how the sender retransmits the lost or damaged frame and all subsequent frames in the current window.
2. Selective Repeat ARQ: Explain how the sender retransmits only the lost or damaged frames and how the receiver handles out-of-order frames.

Discuss the advantages and limitations of Go-Back-N ARQ and Selective Repeat ARQ with respect to bandwidth utilization, retransmissions, and receiver complexity.

Sol:

Stop-and-Wait ARQ: The sender transmits one frame and waits for an acknowledgement (ACK). If the ACK does not arrive before the timeout, the frame is retransmitted. Sequence numbers help detect duplicate frames.

Sliding Window Protocol: The sender may transmit multiple frames before receiving acknowledgements. A window controls how many frames can be outstanding, improving link utilization.

Feature	Stop-and-Wait ARQ	Sliding Window
Frames outstanding	One	Multiple
Efficiency	Lower on long-delay links	Generally higher
Acknowledgement	ACK for each frame	Cumulative or selective ACKs, depending on protocol

Error recovery	Retransmit the unacknowledged frame	Depends on the ARQ variant

Go-Back-N ARQ

- The sender can transmit several frames within its window.
- If a frame is lost or damaged, the receiver discards subsequent out-of-order frames.
- After detecting the loss, the sender retransmits the missing frame and the following outstanding frames.

Example: If frame 2 is lost while frames 3 and 4 are sent, the sender retransmits frames 2, 3 and 4.

Selective Repeat ARQ

- The receiver can accept and buffer out-of-order frames.
- Only the lost or damaged frame is retransmitted.
- Individual acknowledgements or selective acknowledgements identify successfully received frames.

Example: If frame 2 is lost, frames 3 and 4 can be buffered; the sender retransmits only frame 2.

Comparison: Go-Back-N is simpler but may waste bandwidth retransmitting correctly received frames. Selective Repeat uses bandwidth more efficiently but requires more receiver buffering, tracking and protocol complexity.

OR

Q.3B

Explain Pure ALOHA, Slotted ALOHA, CSMA/CD and CSMA/CA. Compare their operation and explain their applications in Ethernet and wireless LANs.

Sol:

Pure ALOHA: A station transmits whenever it has data. If a collision occurs, it waits a random time and retransmits. Its theoretical maximum throughput is approximately 18.4%.

Slotted ALOHA: Time is divided into slots, and transmission begins only at slot boundaries. This reduces collisions. Its theoretical maximum throughput is approximately 36.8%.

CSMA/CD (Carrier Sense Multiple Access with Collision Detection): A station listens before transmitting, detects collisions while transmitting, aborts transmission and retries after a random backoff. It was used in shared, half-duplex Ethernet.

CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance): A wireless station senses the channel and uses random backoff to reduce collisions. Wi-Fi may also use RTS/CTS to help address hidden-terminal problems.

Protocol	Main mechanism	Typical application
Pure ALOHA	Transmit anytime	Historical radio networks
Slotted ALOHA	Transmit at slot boundaries	Slotted random-access systems
CSMA/CD	Detect collisions	Shared, half-duplex Ethernet
CSMA/CA	Avoid collisions	IEEE 802.11 Wi-Fi

Modern switched full-duplex Ethernet generally does not experience collisions, so CSMA/CD is not used in normal operation.

Q.4A

Explain the different phases of TCP congestion control: Slow Start, Congestion Avoidance, Fast Retransmit, and Fast Recovery. Illustrate how the congestion window changes when packet loss occurs.

Sol:

TCP congestion control adjusts the sender's transmission rate according to network conditions. The principal mechanisms are:

1. Slow Start: The congestion window (`cwnd`) starts small and grows approximately exponentially, commonly doubling each round-trip time while below the slow-start threshold.
2. Congestion Avoidance: Once `cwnd` reaches the slow-start threshold (`ssthresh`), growth becomes approximately linear, typically around one MSS per RTT.
3. Fast Retransmit: The sender retransmits a missing segment after receiving three duplicate ACKs, without waiting for a timeout.
4. Fast Recovery: In TCP Reno-style behaviour, the sender reduces its congestion window after loss and avoids restarting Slow Start immediately after triple-duplicate ACKs.

Illustrative example

Suppose `cwnd` grows as 1, 2, 4, 8, 9, 10 MSS.

- During Slow Start, it grows approximately exponentially.
- During Congestion Avoidance, it grows more slowly.
- When loss is detected by three duplicate ACKs, TCP Reno typically sets `ssthresh` to about half the flight size and enters Fast Recovery.
- On a retransmission timeout, TCP typically reduces the window much more sharply and returns to Slow Start.

The exact window changes depend on the TCP variant and whether loss is detected through duplicate ACKs or a timeout.

OR

Q.4B

Explain the services of the Transport Layer, including process-to-process delivery, multiplexing, buffering and connection management. Compare TCP and UDP and discuss their suitability for reliable data transfer and real-time applications.

Sol:

The Transport Layer provides communication between application processes on different hosts.

Main services:

- Process-to-process delivery: Uses port numbers to identify application processes.
- Multiplexing and demultiplexing: Combines data from multiple applications and delivers received data to the correct process.
- Buffering: Temporarily stores data during transmission and reception.
- Connection management: Establishes, maintains and terminates connections when a connection-oriented protocol is used.
- Reliability and flow control: TCP provides acknowledgements, retransmission, and flow control.

Feature	TCP	UDP
Connection	Connection-oriented	Connectionless
Reliability	Reliable, ordered byte stream	No built-in delivery guarantee
Retransmission	Supported	Not provided by UDP itself
Overhead	Higher	Lower

Typical applications	Web connections, email, file transfer	DNS queries, real-time media, many online games

TCP is suitable when reliable; ordered delivery is required. UDP is useful when low latency and low overhead are important, and the application can tolerate loss or manage reliability itself.

Q.5A

Given: block 124.5.73.0/24, which has $2^8 = 256$ addresses.

Step 1: Address requirement and slash notation

Group	Businesses	Addresses each	Host bits	Prefix	Total addresses
Second group	4	32 (2^5)	5	**/27**	128
First group	3	16 (2^4)	4	**/28**	48
Third group	4 households	4 (2^2)	2	**/30**	16

Total allocated = 128 + 48 + 16 = **192 addresses**.

Step 2: Allocation of sub-blocks (largest block first, so there are no gaps or overlaps)

Group	Sub-block	Address range
Second group (32 each)	124.5.73.0/27	.0 to .31
	124.5.73.32/27	.32 to .63
	124.5.73.64/27	.64 to .95
	124.5.73.96/27	.96 to .127
First group (16 each)	124.5.73.128/28	.128 to .143
	124.5.73.144/28	.144 to .159
	124.5.73.160/28	.160 to .175
Third group (4 each)	124.5.73.176/30	.176 to .179
	124.5.73.180/30	.180 to .183
	124.5.73.184/30	.184 to .187
	124.5.73.188/30	.188 to .191

Step 3: Unallocated addresses

Unallocated = 256 − 192 = **64 addresses**, the range 124.5.73.192 to 124.5.73.255 (which can be written as the block 124.5.73.192/26).

OR

Q.5B

(a) Host bits and borrowed bits

- 5 host bits give $2^5 - 2 = 30$ usable hosts, which is not enough.
- 6 host bits give $2^6 - 2 = 62$ usable hosts, which is enough.
- **Host bits required = 6**, so **bits borrowed = 8 − 6 = 2**.

(b) Subnet mask and number of subnets

- Binary: 11111111.11111111.11111111.**11**000000
- Dotted decimal: **255.255.255.192** (/26)
- Number of subnets = 2^2 = **4**

(c) Subnet table

	Subnet Network address Usable host range Broadcast address 1 195.5.20.0 195.5.20.1 to 195.5.20.62 195.5.20.63 2 195.5.20.64 195.5.20.65 to 195.5.20.126 195.5.20.127 3 195.5.20.128 195.5.20.129 to 195.5.20.190 195.5.20.191 4 195.5.20.192 195.5.20.193 to 195.5.20.254 195.5.20.255 **(d) Destination 195.5.20.150** • Last octet: 150 = 10010110, mask octet 192 = 11000000. • ANDing gives 10000000 = 128, so the network address is 195.5.20.128. • The packet belongs to **subnet 3** (broadcast address 195.5.20.191). • Each subnet has 62 usable addresses and 50 are needed, so **12 usable addresses** are unused in each subnet.	

Q.6A	**Content** **Marks** Need for host configuration (1.5) and what DHCP provides (0.5) 2 Ports 67 (server) and 68 (client), and why UDP 2 Neat diagram showing the four messages with ports 3 Working of the four steps (2) and lease renewal (1) 3	
	OR	
Q.6B	**Content** **Marks** Neat diagram 3 UA, MTA and MAA roles 2 Importance of the MTA 2 POP3 and IMAP 3	

********** All the Best**********