CN:

1. **Computer Network**: A computer network is a system that connects multiple computers or devices to share resources and communicate with each other.

2. **Types of Networks**: Common types of networks include LAN (Local Area Network), MAN (Metropolitan Area Network), WAN (Wide Area Network), PAN (Personal Area Network), and VPN (Virtual Private Network).

3. **LAN, MAN, and WAN**:

- **LAN**: Covers a small geographical area, like a single building or campus.

- **MAN**: Spans a city or large campus, linking multiple LANs.

- **WAN**: Covers large geographical areas, connecting multiple LANs and MANs over long distances.

4. **Protocol**: A protocol is a set of rules and conventions that devices follow to communicate and transfer data over a network.

5. **Network Topology**: Network topology refers to the layout or arrangement of devices and cables in a network.

6. **Types of Topologies**: Common network topologies include star, bus, ring, mesh, and tree.

7. **Star Network Topology**:

- **Advantages**: Easy to set up and manage; if one device fails, others remain unaffected.

- **Disadvantages**: If the central hub fails, the whole network goes down.

8. **Bus Network Topology**:

- **Advantages**: Easy to install and requires less cable.

- **Disadvantages**: Limited cable length and number of devices; difficult to troubleshoot if the main cable fails.

9. **Ring Network Topology**: In a ring topology, devices are connected in a circular fashion. Data travels in one direction, and each device has two neighbors, forming a continuous loop.

10. **Number of Cable Links for Six Devices**:

- **Mesh**: Requires 15 cable links (each device connected to every other device).

- **Ring**: Needs 6 cable links (each device connected to two neighbors in a loop).

- **Bus**: Requires 1 main cable with 6 connections for each device.

- **Star**: Needs 6 cable links (each device connected directly to a central hub).

11. **Types of Transmission Medium**: Transmission media can be classified into two main types: guided (wired) and unguided (wireless) media.

12. **Examples of Guided Transmission Media**: Examples include twisted pair cables, coaxial cables, and fiber optic cables.

13. **Examples of Unguided Transmission Media**: Examples include radio waves, microwaves, and infrared waves.

14. **Client-Server, Peer-to-Peer, and Hybrid Architecture**:

- **Client-Server**: Centralized model where clients request resources from a dedicated server.

- **Peer-to-Peer (P2P)**: Decentralized model where each device can act as both client and server.

- **Hybrid**: Combines features of both client-server and peer-to-peer architectures.

15. **Different Network Devices**: Common network devices include routers, switches, bridges, hubs, access points, modems, and gateways.

16. **Router, Switch, Bridge, and Access Point**:

- **Router**: Directs data packets between networks, connecting different IP networks.

- **Switch**: Connects devices within a LAN, using MAC addresses to forward data to the correct device.

- **Bridge**: Connects two network segments, improving network efficiency.

- **Access Point**: Provides wireless connectivity to devices within a network.

17. **Layers in OSI and TCP/IP Model**:

- **OSI Model**: Has 7 layers—Physical, Data Link, Network, Transport, Session, Presentation, and Application.

- **TCP/IP Model**: Has 4 layers—Network Interface, Internet, Transport, and Application.

18. **Function of Each Layer**:

- **Physical Layer**: Transmits raw bit streams over the physical medium.

- **Data Link Layer**: Provides node-to-node data transfer, error detection, and correction.

- **Network Layer**: Handles routing and forwarding of data across networks.

- **Transport Layer**: Manages end-to-end data transfer and error recovery.

- **Session Layer**: Establishes, manages, and terminates communication sessions.

- **Presentation Layer**: Translates data formats and handles encryption/decryption.

- **Application Layer**: Provides network services to end-users and applications.

19. **Unit of Communication at Each Layer**:

- **Physical Layer**: Bits

- **Data Link Layer**: Frames

- **Network Layer**: Packets

- **Transport Layer**: Segments (in TCP) or Datagrams (in UDP)

- **Session, Presentation, and Application Layers**: Data

20. **Function of Data-Link Layer**: The Data Link Layer ensures reliable data transfer between adjacent network nodes, handling framing, MAC addressing, error detection, and flow control.

21. **Definitions**:

- **Flow Control**: A mechanism to regulate data transmission between sender and receiver, ensuring the receiver isn’t overwhelmed with data.

- **Error Control**: A method to detect and correct errors in data transmission, ensuring data integrity.

- **Congestion Control**: A process to prevent network congestion by controlling the flow of data to maintain performance and avoid packet loss.

22. **Design Issues of Data-Link Layer**:

- Framing: Dividing data into manageable frames for transmission.

- Error Control: Detecting and correcting errors in frames.

- Flow Control: Managing data rate to prevent the sender from overwhelming the receiver.

- Link Management: Establishing, maintaining, and terminating links between nodes.

23. **Different Techniques to Framing**:

- Character Count: Uses a field in the header to indicate frame length.

- Byte Stuffing: Uses special characters to mark frame boundaries.

- Bit Stuffing: Inserts extra bits to ensure unique bit patterns for frame boundaries.

- Physical Layer Coding Violations: Uses specific signal patterns to indicate frame boundaries.

24. **CRC (Cyclic Redundancy Check)**: An error-detection technique that adds a calculated checksum to the frame. The receiver recalculates the checksum to check for errors.

25. **Hamming Code**: An error-correcting code that adds parity bits to data, allowing detection and correction of single-bit errors.

26. **Flow Control Protocols**: Flow control protocols manage the rate of data transmission between sender and receiver, ensuring the receiver has time to process data and preventing data overflow. Key flow control protocols include:

- **Simplex Protocol**: A basic protocol where data flows in only one direction, with no acknowledgment or flow control. It's suitable for simple applications where feedback isn't required.

- **Stop-and-Wait Protocol**: In this protocol, the sender transmits a frame and waits for an acknowledgment from the receiver before sending the next frame. It's simple but can be slow, as it waits for acknowledgment after each frame.

- **Sliding-Window Protocol**: This protocol allows multiple frames to be sent before requiring an acknowledgment. Both sender and receiver maintain a window that controls how many frames can be sent and received. This improves efficiency by allowing continuous data flow within the window limit.

- **Go-Back-N Protocol**: A type of sliding-window protocol where the sender can send multiple frames but must retransmit all frames from a lost or damaged frame onward. This ensures data integrity but can lead to retransmission of several frames.

- **Selective-Repeat Protocol**: Another sliding-window protocol where only the frames that were lost or corrupted are retransmitted. It's more efficient than Go-Back-N as it reduces redundant retransmissions and minimizes delays.

27. **Multiple Access Protocols**: These protocols allow multiple devices to share a single communication medium.

- **Pure ALOHA**: A simple protocol where devices transmit data whenever they want. Collisions occur, and any collided frames must be retransmitted, leading to potential inefficiency.

- **Slotted ALOHA**: An improvement over Pure ALOHA where time is divided into slots, and devices only transmit at the beginning of each slot. This reduces collisions and increases efficiency.

- **CSMA (Carrier Sense Multiple Access)**: Devices listen to the medium before transmitting to avoid collisions. If the medium is busy, they wait before attempting to send.

- **WDMA (Wavelength Division Multiple Access)**: Used in optical networks, this protocol assigns different wavelengths (or channels) to different devices for simultaneous data transmission.

- **CSMA/CD (Carrier Sense Multiple Access with Collision Detection)**: Used in wired networks, devices detect collisions while transmitting and stop if a collision is detected, waiting before retransmitting.

- **CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance)**: Used in wireless networks, this protocol tries to avoid collisions by waiting and using mechanisms like acknowledgments to ensure data is successfully received.

28. **Function of the Network Layer**: The Network Layer is responsible for routing, forwarding, and addressing. It determines the best path for data to travel from the source to the destination across multiple networks.

29. **Different Network Layer Protocols**: Common network layer protocols include IP (Internet Protocol), ICMP (Internet Control Message Protocol), ARP (Address Resolution Protocol), RIP (Routing Information Protocol), and OSPF (Open Shortest Path First).

30. **IP Protocol**: The Internet Protocol (IP) is the primary protocol in the Network Layer for sending data packets between devices across networks, using IP addresses for routing.

31. **IP Address**: An IP address is a unique numerical identifier assigned to each device on a network, allowing it to be located and communicate with other devices.

32. **Length of IP Address**: IPv4 addresses are 32 bits long, while IPv6 addresses are 128 bits long.

33. **Address Space of IPv4**: The IPv4 address space is 2^32, providing approximately 4.3 billion unique addresses.

34. **Classes of IP Addresses**: IPv4 addresses are divided into five classes based on network and host portions—Class A, B, C, D, and E.

35. **IP Address Classes**:

- **Class A**: 0.0.0.0 to 127.255.255.255, used for large networks.
- **Class B**: 128.0.0.0 to 191.255.255.255, used for medium-sized networks.
- **Class C**: 192.0.0.0 to 223.255.255.255, used for small networks.
- **Class D**: 224.0.0.0 to 239.255.255.255, reserved for multicast.
- **Class E**: 240.0.0.0 to 255.255.255.255, reserved for experimental purposes.

36. **NAT (Network Address Translation)**: NAT is a technique that allows multiple devices on a local network to share a single public IP address, helping to conserve IP addresses and improve security.

37. **Subnetting and Supernetting**:

- **Subnetting**: The process of dividing a large network into smaller subnetworks to improve management and efficiency.

- **Supernetting**: Combines multiple smaller networks into a larger one, usually to simplify routing and conserve address space.

38. **ARP, RARP, and ICMP**:

- **ARP (Address Resolution Protocol)**: Resolves an IP address to a MAC (physical) address, allowing data to be sent to the correct device on a local network.

- **RARP (Reverse Address Resolution Protocol)**: Converts a device's MAC address to an IP address, mainly used by devices that don't know their IP addresses when they boot up.

- **ICMP (Internet Control Message Protocol)**: Used for error reporting and diagnostics in network communication, such as sending error messages when packets cannot reach their destination.

39. **Routing**: Routing is the process of selecting the best path for data to travel from the source to the destination across a network.

40. **Routing Algorithms and Protocols**: Common routing algorithms include distance-vector, link-state, and path-vector. Protocols using these algorithms include RIP (Routing Information Protocol), OSPF (Open Shortest Path First), and BGP (Border Gateway Protocol).

41. **Distance-Vector Routing**: A routing method where each router shares its routing table with its neighbors, and routes are chosen based on the distance (hop count) to each destination.

42. **Link-State Routing**: Each router independently maps the entire network by sharing information about directly connected links with other routers, creating a global view for optimal routing paths.

43. **Path-Vector Routing**: A routing method that includes the path (sequence of routers) in the route advertisements, allowing the protocol to prevent loops and control path selection.

44. **Routing Protocols**:

- **RIP (Routing Information Protocol)**: A distance-vector protocol that uses hop count as the routing metric, with a maximum hop limit of 15.

- **OSPF (Open Shortest Path First)**: A link-state protocol that calculates the shortest path based on cost, commonly used in large enterprise networks.

- **BGP (Border Gateway Protocol)**: A path-vector protocol used for routing between autonomous systems on the internet, providing control over the path selection.

45. **Function of the Transport Layer**: The Transport Layer provides reliable process-to-process communication, data segmentation, error detection, and flow control, ensuring data is sent accurately and efficiently between applications.

46. **Process-to-Process Communication**: This is a direct communication between two applications (or processes) running on different devices, using unique identifiers like port numbers to facilitate the exchange.

47. **Port Number**: A port number is a unique identifier assigned to each process or service on a device, helping the Transport Layer distinguish between multiple applications.

48. **Categories of Port Numbers**:

   - **Well-Known Ports**: Ranging from 0 to 1023, reserved for standard services like HTTP (80) and FTP (21).

   - **Registered Ports**: Ranging from 1024 to 49151, used by applications registered with the IANA.

   - **Dynamic/Private Ports**: Ranging from 49152 to 65535, available for temporary use by client applications.

49. **Well-Known Port Numbers**: Port numbers from 0 to 1023, reserved for widely used services and protocols such as HTTP (80), HTTPS (443), FTP (21), and SMTP (25).

50. **Private/Ephemeral/Dynamic Port Numbers**: Port numbers from 49152 to 65535, assigned temporarily to client applications for short-term connections, often used by operating systems for automatic assignments.

51. **Registered Ports**: Port numbers from 1024 to 49151, used by specific applications and assigned by IANA for less common but recognized services and software.

52. **Different Transport Layer Protocols**: Common transport layer protocols include TCP (Transmission Control Protocol) and UDP (User Datagram Protocol).

53. **UDP (User Datagram Protocol)**: UDP is a connectionless transport layer protocol that sends data without establishing a connection or guaranteeing delivery, suitable for applications needing fast, efficient data transfer.

54. **Services Provided by UDP**: UDP provides basic data transfer without error checking, flow control, or congestion control. It's used for real-time applications like video streaming, online gaming, and VoIP, where speed is more critical than reliability.

55. **TCP (Transmission Control Protocol)**: TCP is a connection-oriented protocol that ensures reliable data transfer, error checking, and in-sequence data delivery between sender and receiver.

56. **Services Provided by TCP**: TCP offers reliable communication through error detection, flow control, congestion control, ordered data transfer, and retransmission of lost packets.

57. **Flow Control, Error Control, and Congestion Control**:

- **Flow Control**: Manages data transmission rate between sender and receiver, preventing the receiver from being overwhelmed.

- **Error Control**: Detects and corrects errors in data transmission, ensuring accurate delivery.

- **Congestion Control**: Prevents network congestion by managing the data flow rate, adapting to current network conditions.

58. **Connection in TCP**: TCP establishes a reliable connection between sender and receiver before data transfer, using a handshake process to set up the connection.

59. **Three-Way Handshaking Connection Establishment**: The three-way handshake is used to establish a TCP connection, consisting of three steps:

- **SYN**: The client sends a SYN (synchronize) packet to initiate a connection.

- **SYN-ACK**: The server responds with a SYN-ACK (synchronize-acknowledgment) packet to acknowledge the request.

- **ACK**: The client sends an ACK (acknowledgment) packet to confirm the connection, completing the handshake.

60. **Three-Way Handshaking Connection Termination**: In TCP, the connection termination process also involves a handshake, with four steps:

- **FIN**: The client sends a FIN (finish) packet to indicate it wants to terminate the connection.

- **ACK**: The server acknowledges the FIN packet with an ACK.

- **FIN**: The server then sends its own FIN packet to the client.

- **ACK**: The client sends a final ACK to confirm, after which the connection is fully closed.

61. **SCTP (Stream Control Transmission Protocol)**: SCTP is a transport layer protocol similar to TCP and UDP, providing reliable, ordered data delivery but supporting multiple streams within a single connection. It's often used in telecommunication applications due to its ability to prevent message blocking in multi-stream scenarios.

62. **Function of the Application Layer**: The application layer provides end-user services and enables applications to communicate over a network. It interfaces directly with software applications, facilitating functions like file transfers, email, remote login, and web browsing.

63. **Different Application Layer Protocols**: Common application layer protocols include HTTP (Hypertext Transfer Protocol), FTP (File Transfer Protocol), SMTP (Simple Mail Transfer Protocol), DNS (Domain Name System), and DHCP (Dynamic Host Configuration Protocol).

64. **Role of Protocol DHCP (Dynamic Host Configuration Protocol)**: DHCP assigns IP addresses and other network configuration parameters automatically to devices on a network, allowing them to communicate without manual configuration. It streamlines IP management and avoids address conflicts.

65. **Role of Protocol DNS (Domain Name System)**: DNS translates human-readable domain names (like www.example.com) into IP addresses. This allows users to access websites by name rather than remembering numerical IP addresses, serving as the "phone book" of the internet.

66. **Role of Protocol TELNET**: TELNET is a protocol used to provide remote access to a device over a network. It allows users to log in to a remote system and execute commands, but it transmits data (including passwords) in plaintext, making it insecure for modern use.

67. **Role of Protocol FTP**: FTP (File Transfer Protocol) is used for transferring files between a client and a server over a network. It supports both uploading and downloading files, and can be used for managing files on remote systems. However, it is not encrypted by default, making it insecure.

68. **Role of Protocol HTTP**: HTTP (Hypertext Transfer Protocol) is used for transmitting web pages on the internet. It is a request-response protocol where clients (browsers) request web pages from servers and receive responses in the form of HTML content.

69. **Role of Protocol SMTP, POP3, IMAP4, and MIME**:

- **SMTP (Simple Mail Transfer Protocol)**: Used for sending email messages between mail servers.

- **POP3 (Post Office Protocol version 3)**: Used to retrieve emails from a mail server and download them to a client. It deletes messages from the server after download.

- **IMAP4 (Internet Message Access Protocol version 4)**: Similar to POP3, but allows messages to be stored on the server and accessed from multiple devices without being deleted.

- **MIME (Multipurpose Internet Mail Extensions)**: Extends email protocols (like SMTP) to support multimedia content (images, audio, video) and attachments.

70. **Role of Protocol SNMP (Simple Network Management Protocol)**: SNMP is used for managing and monitoring network devices like routers, switches, and servers. It allows administrators to collect performance data, configure devices, and get alerts on network issues.

71. **Security Goals**: The primary security goals in a network are:

- **Confidentiality**: Ensuring that information is only accessible to authorized users.

- **Integrity**: Ensuring that information is not altered or tampered with during transmission.

- **Availability**: Ensuring that information and resources are accessible when needed.

72. **Confidentiality, Integrity, and Availability**:

- **Confidentiality**: Protecting data from unauthorized access (e.g., encryption).

- **Integrity**: Ensuring that data is not tampered with during transmission (e.g., hashing).

- **Availability**: Ensuring that systems and data are accessible when needed, typically through redundancy and failover mechanisms.

73. **Different Security Attacks**: Common security attacks include:

- **Denial of Service (DoS)**: Overloading a system to make it unavailable.

- **Man-in-the-Middle (MitM)**: Intercepting and altering communication between two parties.

- **Phishing**: Tricking individuals into providing sensitive information.

- **SQL Injection**: Inserting malicious SQL code into a web application's database.

- **Malware**: Malicious software that damages or disrupts systems.

74. **Symmetric-Key Cryptography**: Symmetric-key cryptography uses the same secret key for both encryption and decryption. It is fast but requires secure key distribution, as both parties must have the same key.

75. **Asymmetric-Key Cryptography**: Asymmetric-key cryptography uses a pair of keys—one public key for encryption and a corresponding private key for decryption. This eliminates the need for key sharing, but it is slower than symmetric-key cryptography.

76. **Difference Between Symmetric-Key and Asymmetric-Key Cryptography**:

- **Symmetric-Key Cryptography**: Uses a single key for both encryption and decryption; faster but requires secure key distribution.

- **Asymmetric-Key Cryptography**: Uses a pair of keys (public and private); slower but more secure for key exchange and does not require sharing the private key.

77. **How IPSec Provides Security at the Network Layer**: IPSec (Internet Protocol Security) secures IP communications by encrypting and authenticating each IP packet in a communication session. It provides confidentiality, integrity, and authentication, operating at the network layer to protect data traveling between devices on a network.

78. **Purpose of a Firewall and How It Protects a Network**: A firewall is a network security system that monitors and controls incoming and outgoing network traffic based on predefined security rules. It protects a network by blocking unauthorized access while allowing legitimate communication.

79. **SSL and HTTPS**:

- **SSL (Secure Sockets Layer)**: A protocol that provides encryption and authentication for secure communication over a network.

- **HTTPS (Hypertext Transfer Protocol Secure)**: An extension of HTTP that uses SSL/TLS encryption to secure web traffic, ensuring privacy and data integrity between web servers and browsers.

80. **Security in Network, Transport, and Application**:

- **Network Security**: Focuses on protecting the integrity, confidentiality, and availability of data and resources as they are transmitted across or accessed from networks. Includes techniques like encryption, firewalls, and VPNs.

- **Transport Security**: Ensures secure communication between end systems on a network, such as using SSL/TLS for encrypting data over TCP connections (HTTPS).

- **Application Security**: Involves securing the software and services running on systems to prevent attacks like SQL injection, cross-site scripting (XSS), and ensuring that sensitive data is protected within the application itself.