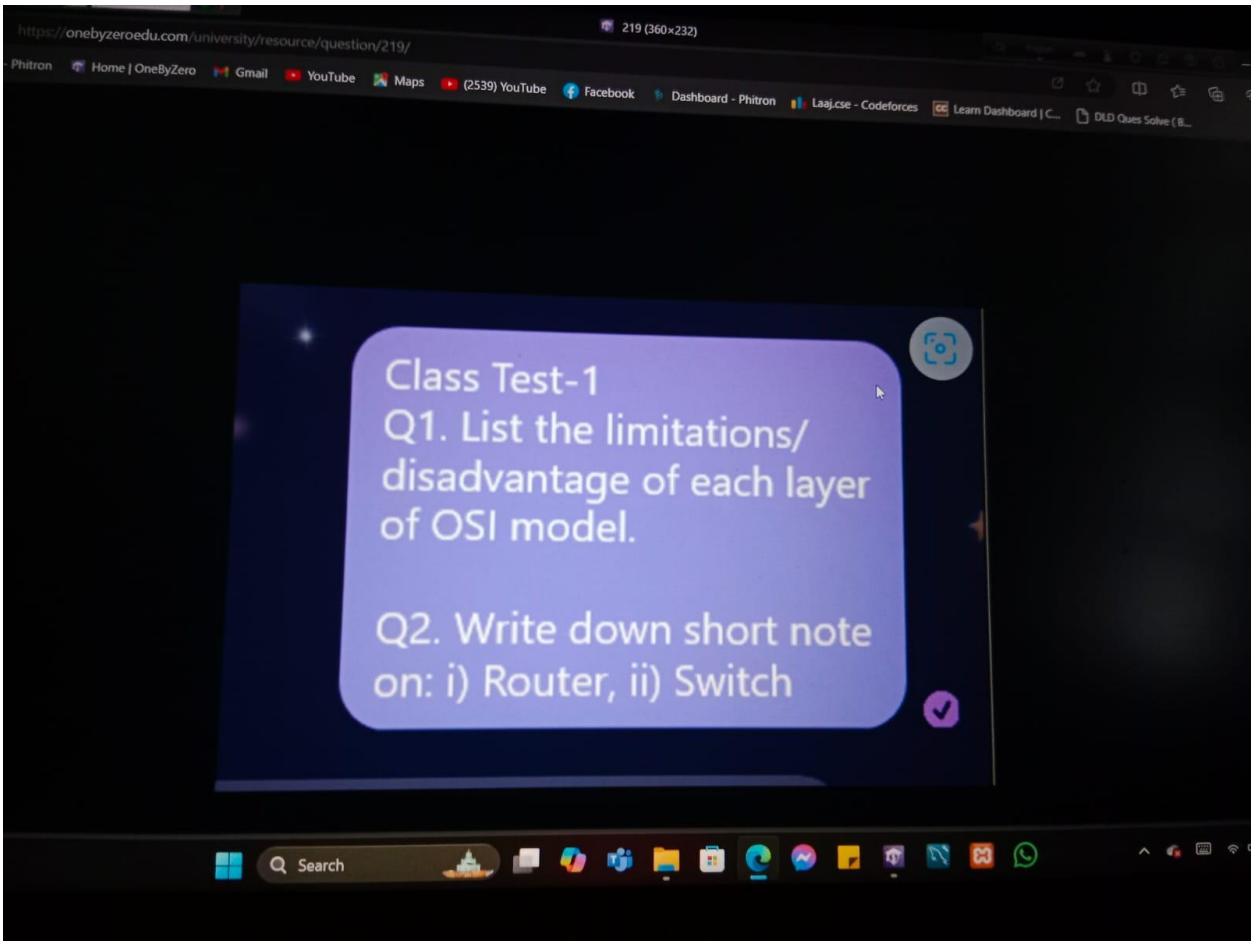



ANSWER

**Q1. Limitations/Disadvantages of Each Layer of the OSI Model**

1. **Physical Layer**:
    - No error correction or detection.
    - Deals only with raw data (bits), lacks data interpretation.
    - Doesn't ensure reliable data delivery.
2. **Data Link Layer**:
    - Limited to devices in the same network segment.
    - Vulnerable to data frame collisions and duplication.
    - Error correction is often inefficient for large networks.
3. **Network Layer**:
    - Does not guarantee data delivery (unreliable).
    - Congestion can occur due to lack of flow control.

- Complex algorithms for routing can increase latency.

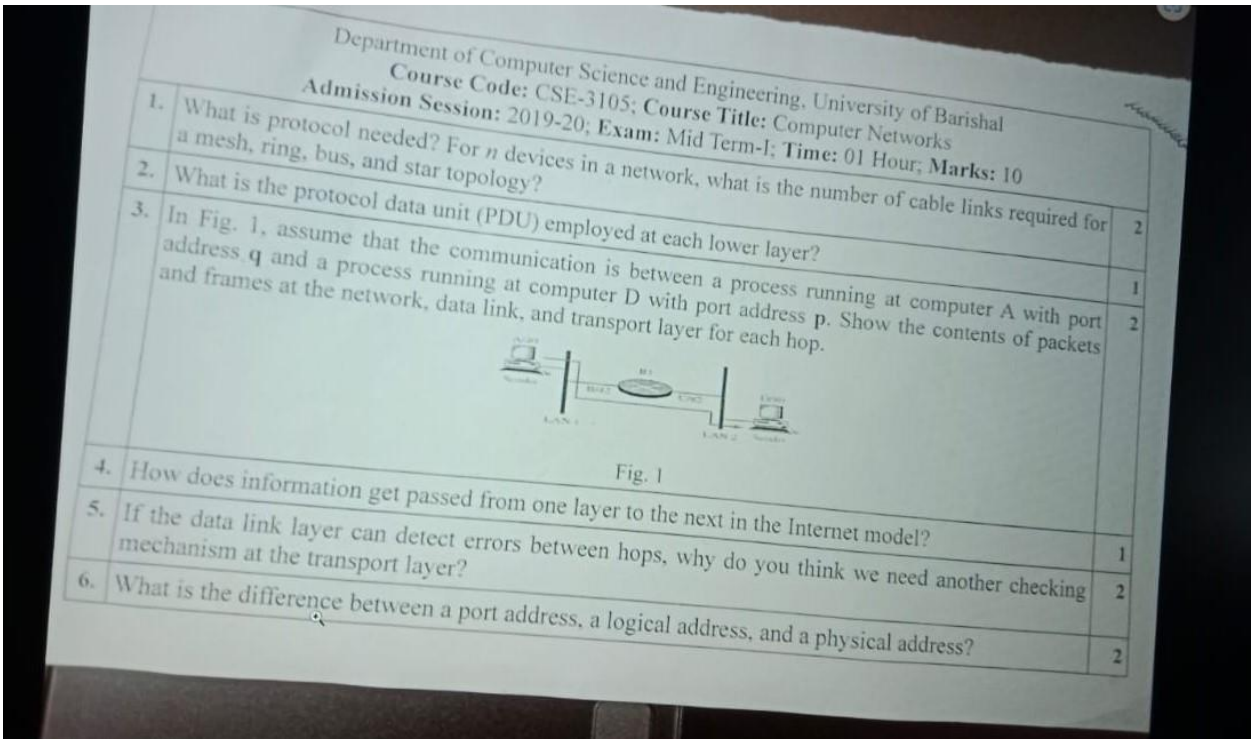
4. **Transport Layer**:
    - Overhead increases with error control and segmentation.
    - Potential delay in retransmissions for lost data.
    - May not adapt well to different network conditions.
5. **Session Layer**:
    - Adds complexity with session management.
    - Rarely used or implemented in modern networking.
    - Can increase overhead due to synchronization mechanisms.
6. **Presentation Layer**:
    - Redundant for many modern applications that handle data formatting.
    - Adds overhead with encryption, compression, or translation.
    - Not always standardized across systems.
7. **Application Layer**:
    - Application-specific protocols can limit interoperability.
    - Vulnerable to security threats due to direct user interaction.
    - Higher risk of inefficiency in poor software design.

---

**Q2. Short Notes**

1. **Router**:
    - A router is a networking device that forwards data packets between computer networks.
    - Operates at the **Network Layer** (Layer 3) of the OSI model.
    - Uses IP addresses to determine the best path for data.
    - Facilitates communication between different networks (e.g., LAN to WAN).
    - Supports advanced functions like traffic management and firewalling.
2. **Switch**:
    - A switch is a device used to connect devices within the same network (e.g., a Local Area Network).

- Operates at the **Data Link Layer** (Layer 2) but can also function at Layer 3 (multilayer switches).
- Uses MAC addresses to forward data to specific devices.
- Reduces network congestion by creating a dedicated communication path.
- Enhances network security and efficiency compared to hubs.



Department of Computer Science and Engineering, University of Barishal
**Course Code:** CSE-3105; **Course Title:** Computer Networks
**Admission Session:** 2019-20; **Exam:** Mid Term-I; **Time:** 01 Hour; **Marks:** 10

| | | |
|---|---|---|
| 1. | What is protocol needed? For *n* devices in a network, what is the number of cable links required for a mesh, ring, bus, and star topology? | 2 |
| 2. | What is the protocol data unit (PDU) employed at each lower layer? | 1 |
| 3. | In Fig. 1, assume that the communication is between a process running at computer A with port address **q** and a process running at computer D with port address **p**. Show the contents of packets and frames at the network, data link, and transport layer for each hop. | 2 |
| 4. | How does information get passed from one layer to the next in the Internet model? | 1 |
| 5. | If the data link layer can detect errors between hops, why do you think we need another checking mechanism at the transport layer? | 2 |
| 6. | What is the difference between a port address, a logical address, and a physical address? | 2 |

Fig. 1

ANSWER

1. What is protocol needed? For n devices in a network, what is the number of cable links required for a mesh, ring, bus, and star topology?

Protocol Needed:
Protocols are essential for ensuring reliable communication between devices in a network. They define rules for data transmission, error checking, flow control, and addressing. Examples include TCP/IP, HTTP, and FTP.
Number of Cable Links:
Mesh Topology:
Total links = n(n−1)2\frac{n(n-1)}{2}2n(n−1)
Example: For n=5n = 5n=5, links = 5(5−1)2=10\frac{5(5-1)}{2} = 1025(5−1)=10.
Ring Topology:
Total links = nnn
Example: For n=5n = 5n=5, links = 5.
Bus Topology:
Only one backbone cable is required, regardless of nnn.

Star Topology:
Total links = nnn (one for each device connected to the central hub).
Example: For n=5n = 5n=5, links = 5.

2. What is the protocol data unit (PDU) employed at each lower layer?
Physical Layer: Bits
Data Link Layer: Frames
Network Layer: Packets
Transport Layer: Segments (TCP) or Datagrams (UDP)

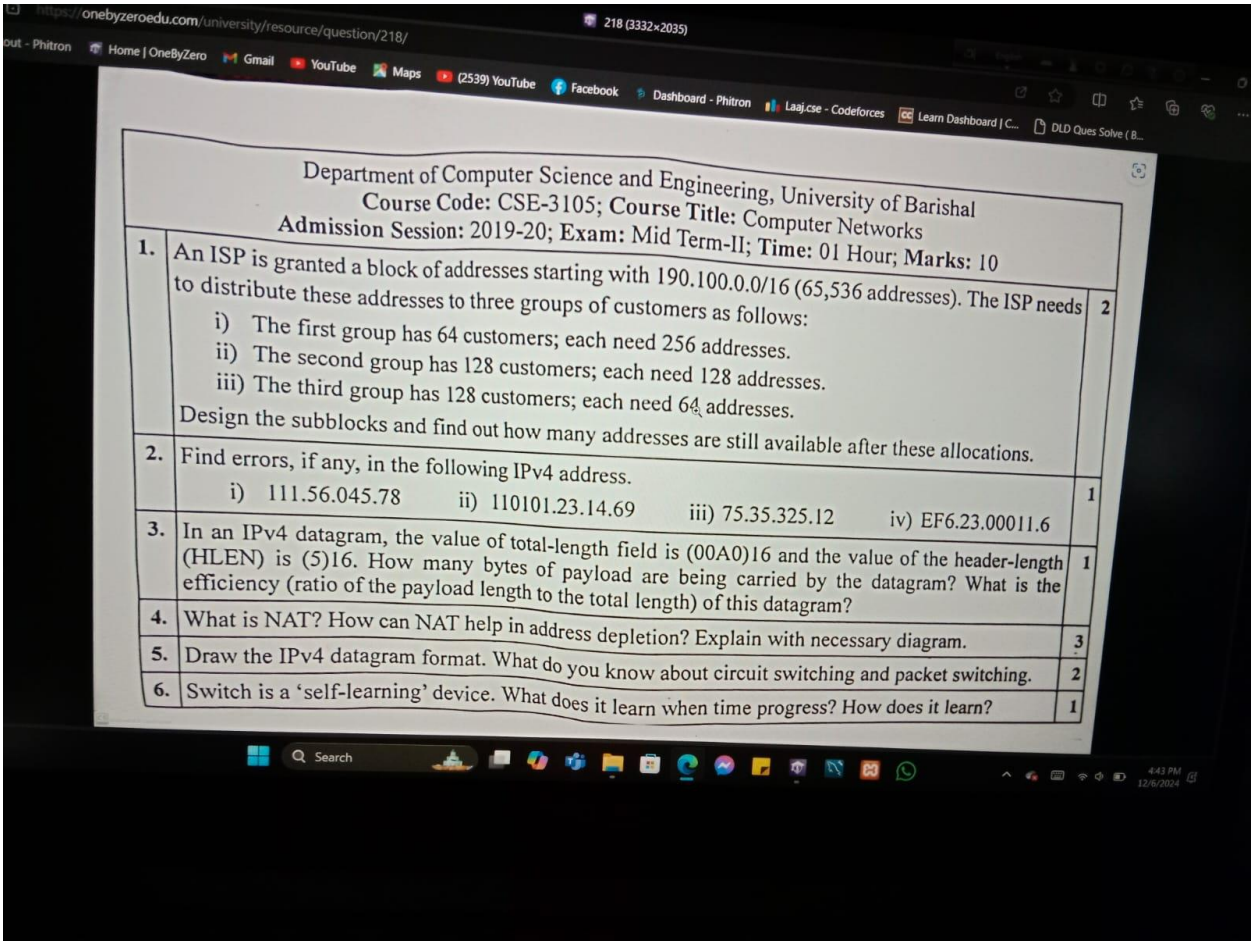
3. Communication between A (port q) and D (port p) through the network. Show contents of packets and frames at the network, data link, and transport layers.
Transport Layer:
Contains the source port (q) and destination port (p). Example:
Header: [Source Port: q, Destination Port: p], Data: [...]
Network Layer:
Contains the source IP (A's IP) and destination IP (D's IP). Example:
Header: [Source IP: A, Destination IP: D], Data: [...]
Data Link Layer:
Contains the source MAC address and destination MAC address. These change at each hop. Example at a specific hop:
Header: [Source MAC: Router1 MAC, Destination MAC: Router2 MAC], Data: [...]

4. How does information get passed from one layer to the next in the Internet model?
Information is passed using a process called encapsulation and decapsulation:
Encapsulation: At the sender side, data starts at the application layer and moves down, where each layer adds its header to the data.
Decapsulation: At the receiver side, data moves up the layers, where each layer removes its corresponding header.

5. If the data link layer can detect errors between hops, why do we need another checking mechanism at the transport layer?
Reason:
Errors can occur during end-to-end communication that the data link layer cannot detect because it operates only at the local network segment.
The transport layer ensures end-to-end reliability by checking for errors such as packet loss, corruption, and sequence issues across the entire path.

6. What is the difference between a port address, a logical address, and a physical address?
Port Address:
A unique number identifying a specific process or application on a device. Example: Port 80 for HTTP.

Logical Address:
The IP address assigned to a device for identification in a network. It changes based on the network. Example: 192.168.1.1.
Physical Address:
The MAC address burned into the hardware of a device's network interface card (NIC). It is unique and permanent. Example: 00:1A:2B:3C:4D:5E.



Department of Computer Science and Engineering, University of Barishal
**Course Code:** CSE-3105; **Course Title:** Computer Networks
**Admission Session:** 2019-20; **Exam:** Mid Term-II; **Time:** 01 Hour; **Marks:** 10

| | | |
|---|---|---|
| 1. | An ISP is granted a block of addresses starting with 190.100.0.0/16 (65,536 addresses). The ISP needs to distribute these addresses to three groups of customers as follows:<br>i) The first group has 64 customers; each need 256 addresses.<br>ii) The second group has 128 customers; each need 128 addresses.<br>iii) The third group has 128 customers; each need 64 addresses.<br>Design the subblocks and find out how many addresses are still available after these allocations. | 2 |
| 2. | Find errors, if any, in the following IPv4 address.<br>i) 111.56.045.78 ii) 110101.23.14.69 iii) 75.35.325.12 iv) EF6.23.00011.6 | 1 |
| 3. | In an IPv4 datagram, the value of total-length field is (00A0)16 and the value of the header-length (HLEN) is (5)16. How many bytes of payload are being carried by the datagram? What is the efficiency (ratio of the payload length to the total length) of this datagram? | 1 |
| 4. | What is NAT? How can NAT help in address depletion? Explain with necessary diagram. | 3 |
| 5. | Draw the IPv4 datagram format. What do you know about circuit switching and packet switching. | 2 |
| 6. | Switch is a 'self-learning' device. What does it learn when time progress? How does it learn? | 1 |

ANSWER

## 1. Subblock Allocation and Remaining Addresses

**Given Block:**

190.100.0.0/16 → Total addresses: $2^{16} = 65,536$

**i) First Group (64 customers, 256 addresses each):**

- Each customer needs $2^8 = 256$ addresses → Prefix length: /24
- Total addresses: $64 \times 256 = 16,384$
- Subblocks: 64 subnets, each of size /24.

**ii) Second Group (128 customers, 128 addresses each):**

- Each customer needs $2^7 = 128$ addresses → Prefix length: /25
- Total addresses: $128 \times 128 = 16,384$
- Subblocks: 128 subnets, each of size /25.

**iii) Third Group (128 customers, 64 addresses each):**

- Each customer needs $2^6 = 64$ addresses → Prefix length: /26
- Total addresses: $128 \times 64 = 8,192$

### iii) Third Group (128 customers, 64 addresses each):

- Each customer needs $2^6 = 64$ addresses → Prefix length: /26
- Total addresses: $128 \times 64 = 8,192$
- Subblocks: 128 subnets, each of size /26.

### Total Allocated Addresses:

$$16,384 + 16,384 + 8,192 = 40,960 \text{ addresses}$$

### Remaining Addresses:

$$65,536 - 40,960 = 24,576 \text{ addresses}$$

## 2. Errors in IPv4 Addresses

**i) 111.56.045.78**

- Invalid: Leading zero in **045** suggests octal, which is not allowed in IPv4. Correct format: **111.56.45.78**

**ii) 110101.23.14.69**

- Invalid: An octet cannot exceed 255. **110101** is out of range.

**iii) 75.35.325.12**

- Invalid: An octet cannot exceed 255. **325** is out of range.

**iv) EF6.23.00011.6**

- Invalid: Hexadecimal notation (EF6) and leading zeros (00011) are not valid in standard IPv4.

## 3. Payload and Efficiency in IPv4 Datagram

**Given:**

- Total Length = $00A0_{16} = 160_{10}$ bytes
- Header Length (HLEN) = $5_{16} = 5 \times 4 = 20$ bytes

**Payload Length:**

$$\text{Payload} = \text{Total Length} - \text{Header Length} = 160 - 20 = 140 \text{ bytes}$$

**Efficiency:**

$$\text{Efficiency} = \frac{\text{Payload Length}}{\text{Total Length}} = \frac{140}{160} = 0.875 \text{ or } 87.5\%$$

4 ANSWER

**What is NAT (Network Address Translation)?**

NAT is a technique used to map private IP addresses within a local network to one or a few public IP addresses. It operates at the router, enabling devices with private IP addresses to communicate with the internet.

**How NAT Helps Address Depletion**

- **Private IPs Reuse**: NAT allows many devices to use private IP addresses (e.g., 192.168.x.x, 10.x.x.x) internally while sharing a single public IP for internet access.
- **Conserving IPv4 Addresses**: By sharing public IP addresses, NAT reduces the need for allocating a unique public IP to every device.

**Diagram of NAT Functionality:**

**Diagram of NAT Functionality:**

plaintext Copy code

```
Private Network                NAT Router                Internet
+-------------+             +-----------------+       +------------------+
| Device A    |   192.168.0.1 --->            |   203.0.113.1 --->       |
| Device B    |   192.168.0.2 --->   NAT      | --->   Internet          |
| Device C    |   192.168.0.3 --->   Router   |                           |
+-------------+             +-----------------+       +------------------+
```

plaintext

1. Devices A, B, and C use private IPs (192.168.0.x).
2. NAT translates these private IPs to the public IP (203.0.113.1) for internet communication.

***6 ANSWER***

**. Switch as a Self-Learning Device**

**What Does a Switch Learn?**

A switch learns the **MAC addresses** of devices connected to its ports.

**How Does a Switch Learn?**

1. **Frame Arrival**: When a frame arrives at a switch, the switch inspects the **source MAC address** in the frame header.
2. **MAC Address Table Update**: It maps the source MAC address to the port on which the frame was received and stores this information in its **MAC address table**.
3. **Forwarding Decision**:
    - If the destination MAC address is already in the table, the switch forwards the frame only to the port associated with that address.
    - If the destination address is not in the table, the switch broadcasts the frame to all ports except the one it arrived on.

**Example of Learning Process:**

- Device A (MAC: AA:BB:CC:DD:EE:01) sends a frame through Port 1.
- The switch learns that MAC **AA:BB:CC:DD:EE:01** is reachable via Port 1.
- This learning process repeats for every device that sends frames through the switch.

By learning and updating the MAC table dynamically, switches minimize unnecessary frame broadcasts and improve network efficiency.

7th batch final

# University of Barishal

## Department of Computer Science and Engineering

**Course Title:** Computer Networks
**Course Code:** CSE-3105
3rd Year 1st Semester Final Examination
**Admission Session:** 2019-2020

**Time:** 03 Hours **Marks:** 60

**N.B.:** Answer any **FIVE** questions out of the followings. All parts of each question must be answered consecutively. Right side of the question shows the maximum marks.

**1.a)** What is computer network? Describe the Network Crieteria. 3

b) What are the advantages of a multi-point connection over a P2P connection? 2

c) Define protocol and Standards in Computer networks. 3

d) What do you mean by ARPANET? Describe the physical topology of computer networks. 4

**2.a)** In the Go-Back-N protocol, the size of the send window can be $2m - 1$, while the size of the receive window is only 1. How can flow control be accomplished when there is a big difference between the size of the send and receive windows? Briefly Explain. 5

**b)** If the data link layer can detect errors between hops, why do you think we need another checking mechanism at the transport layer in OSI Model? What is socket Address? Compare between -TCP & UDP protocol 4

**c)** What do you mean by loopback interface? An organization is assigned the block 2000:1456:2474/48. What is the IPv6 address of an interface in the third subnet if the IEEE physical address of the computer is $(F5\text{-}A9\text{-}23\text{-}14\text{-}7A\text{-}D2)_{16}$? 3

**3.a)** You are given the following network address and subnet mask: 6

**Network address:** 192.168.10.0
**Subnet mask:** 255.255.255.252

| | |
|---|---|
| i) How many subnets? | ii) How many hosts? |
| iii) What are the valid subnets | iv) Fill in the table below |

| Meaning | Subnet 1 | Subnet 2 | Subnet 3 | Subnet 4 | Subnet 5 | Subnet 6 |
|---|---|---|---|---|---|---|
| Subnet address | | | | | | |
| First valid host | | | | | | |
| Last valid host | | | | | | |
| Broadcast address | | | | | | |

**b)** If the data link layer can detect errors between hops, why do you think we need another checking mechanism at the transport layer in OSI model? 3

**c)** How can NAT help in address depletion? Explain with necessary diagram. 3

**4.a)** What is Cryptography? Distinguish between passive and active attacks. 3

**b)** What are the differences between message confidentiality and message integrity? Can you have one without another? Use the additive cipher with $k = 5$ to encrypt the plaintext "BU". Then decrypt the message to get the original plaintext. 4

**c)** *Consider sending 4000-byte IP datagram (including the 20 bytes IP header) into a link that has an MTU of 1200 bytes. Determine the values of the length field and the offset field in each fragment.* 5

ANSWER

**1a) What is a Computer Network? Describe the Network Criteria**

**What is a Computer Network?**

A computer network is a group of interconnected devices (such as computers, servers, routers, and switches) that communicate with each other to share resources, data, and applications. These devices can be connected through wired or wireless means, enabling collaboration and efficient data exchange.

**Network Criteria**

To be effective and reliable, a computer network must meet the following criteria:

1. **Performance:**
    - **Throughput:** Measures how much data is transferred over the network in a specific time.
    - **Latency:** Refers to the delay in transmitting data between sender and receiver.
    - **Jitter:** Refers to variations in packet arrival times.
2. **Reliability:**
    - **Fault Tolerance:** The network's ability to function correctly even in case of a failure.
    - **Recovery Time:** The time it takes for the network to recover after an interruption.
3. **Security:**
    - Protects data and resources from unauthorized access, tampering, and denial-of-service attacks.
    - Includes mechanisms like encryption, authentication, and firewalls.
    -

**1b) Advantages of Multi-Point Connections over Point-to-Point Connections**

**Point-to-Point (P2P) Connection:**

In a P2P connection, a dedicated link connects only two devices.

**Multi-Point Connection:**

In a multi-point connection, multiple devices share a single link.

**Advantages of Multi-Point Connections:**

1. **Cost-Effectiveness:**
    - Fewer cables and interfaces are required compared to multiple P2P links.
2. **Efficient Resource Utilization:**
    - A single communication medium is shared among multiple devices, optimizing bandwidth and reducing resource wastage.
3. **Simplified Network Design:**
    - Adding a new device is straightforward without needing extra dedicated links.
4. **Scalability:**
    - Multi-point connections can support more devices in a network with minimal infrastructure changes.
5. **Flexibility:**
    - Easier to set up for broadcasting and multicasting, as the shared link allows messages to reach multiple devices simultaneously.

Multi-point connections are particularly useful in environments where devices need to communicate frequently without requiring dedicated links.

**1c) Define Protocols and Standards in Computer Networks**

**Protocol**

A protocol is a set of rules and conventions that define how data is transmitted, formatted, and received across a network. It ensures that devices from different manufacturers can communicate effectively.

**Examples of Protocols:**

- **HTTP/HTTPS:** Used for web communication.
- **FTP:** Used for file transfers.
- **TCP/IP:** Manages data transmission over the internet.

**Standards**

Standards are established guidelines or specifications developed by organizations to ensure compatibility and interoperability between different devices and technologies.

**Examples of Standards:**

- **IEEE 802.3:** Specifies Ethernet standards.
- **ISO 7498:** Defines the OSI model.
- **ITU-T:** Sets global telecommunications standards.

**Difference Between Protocols and Standards:**

- **Protocols** are the "rules" for communication.
- **Standards** ensure that the implementation of these rules is consistent across devices and vendors.

**2a) Go-Back-N Protocol: Flow Control with Large Difference Between Send and Receive Windows**

**Go-Back-N Protocol Overview:**

- In the Go-Back-N protocol, the sender can send multiple frames before needing an acknowledgment, but the receiver can only accept one frame at a time (receive window size = 1).
- The **send window** size is 2m−12^m - 12m−1, where mmm is the number of bits used for sequence numbering, allowing multiple frames to be sent without waiting for an acknowledgment.
- The **receive window** size is fixed at 1, meaning the receiver expects frames in a strict sequence and can only accept one frame at a time.

**How Flow Control Works:**

- **Sender Side:** The sender can send multiple frames (up to 2m−12^m - 12m−1) without waiting for an acknowledgment for each frame. However, the sender cannot exceed the available space in the receiver's buffer.
- **Receiver Side:** The receiver only accepts one frame at a time, meaning it will acknowledge the frame once it has been received and processed. If the receiver receives a frame out of order, it will discard it and expect the missing frame to be retransmitted (due to Go-Back-N's nature).

**How Flow Control is Accomplished with Large Send and Small Receive Window:**

- **Acknowledgment System:** The receiver sends an acknowledgment for each correctly received frame, and the sender can only move its window forward after receiving the acknowledgment. This ensures that even with a larger send window, the sender will not overwhelm the receiver since the receiver will only process one frame at a time.
- **Sequence Numbering:** The use of sequence numbers helps the sender and receiver keep track of frames. If a frame is lost or corrupted, the sender will retransmit from the lost frame onward. Since the receiver only accepts one frame at a time, it will process each frame in order, even if there are multiple outstanding frames in the sender's window.

Thus, flow control is maintained by **acknowledging received frames**, and the sender must **wait for acknowledgments** before sliding its window forward, preventing the receiver from being overwhelmed by too many frames at once.

**2b) Why Do We Need Another Checking Mechanism at the Transport Layer if the Data Link Layer Can Detect Errors Between Hops?**

**Error Detection at the Data Link Layer:**

- The **Data Link Layer** is responsible for error detection between two directly connected devices (hops). It typically uses mechanisms such as **Cyclic Redundancy Check (CRC)** to detect errors in transmitted frames.

- If an error is detected, the data link layer can request a retransmission of the frame.

**Need for Error Checking at the Transport Layer:**

Even though the data link layer can detect and handle errors between hops, **additional error checking at the transport layer is necessary for the following reasons:**

1. **End-to-End Error Detection:**
    - The **Data Link Layer** only handles errors that occur during transmission over a single hop (between two directly connected devices). However, data may pass through multiple intermediate devices (routers, switches) before reaching the destination. The **Transport Layer** ensures that errors in the entire end-to-end communication path are detected and handled.
    - The **Transport Layer** provides **end-to-end error checking** using checksums (e.g., in **TCP/UDP**). It ensures the data has not been corrupted in transit over multiple hops, including across networks that are not directly connected.
2. **Error Recovery:**
    - While the data link layer may detect errors and request retransmissions on a hop-by-hop basis, it does not have visibility of the entire communication process. The transport layer (e.g., **TCP**) provides mechanisms to **recover lost or corrupted data** across the entire communication path, ensuring reliable end-to-end data delivery.
3. **Error Detection Over Long Distances:**
    - Data can travel long distances over multiple networks, and errors could be introduced during any part of the journey. The **Transport Layer** provides an additional level of security by verifying data integrity across the entire route, from the sender to the receiver.
4. **Application Layer Consistency:**
    - The **Transport Layer** ensures that data reaches the correct application at the receiving end. If errors go undetected at this layer, the receiving application could be fed corrupted or incomplete data, which may affect its functionality.

**Conclusion:**

The data link layer handles error detection between adjacent nodes, but since data often travels across multiple hops and networks, the **Transport Layer** is needed to ensure that the data is correctly received and processed by the destination application, maintaining data integrity and reliability over the entire communication path.

**2c) Socket Address and Comparison Between TCP and UDP**

**What is a Socket Address?**

A **socket address** is a unique identifier used for communication between two devices over a network. It is a combination of:

- **IP address** (identifying the device on the network)
- **Port number** (identifying the specific application or process on that device)

A socket address allows networked devices to establish a communication link to a specific application or service. In IPv4, it is often represented as:

- **IPv4 address:Port** (e.g., 192.168.1.10:80)

In IPv6, it may look like:

- **IPv6 address:Port** (e.g., [2001:0db8::1]:8080)

**Comparison Between TCP and UDP Protocols:**

| Feature | TCP (Transmission Control Protocol) | UDP (User Datagram Protocol) |
|---|---|---|
| **Connection Type** | Connection-oriented (establishes a connection before transmission) | Connectionless (no need to establish a connection) |
| **Reliability** | Reliable, ensures data delivery through acknowledgments, retransmissions, and error recovery | Unreliable, no guarantees on data delivery |
| **Flow Control** | Yes, uses flow control (windowing) to manage congestion and buffer overflows | No flow control, applications must handle it |
| **Error Detection** | Provides error detection and recovery using checksums, retransmissions, and acknowledgments | Provides error detection using checksums but no error recovery |
| **Speed** | Slower due to connection establishment and error recovery mechanisms | Faster due to lack of connection setup and error recovery |
| **Use Cases** | Used for applications where reliability is crucial (e.g., web browsing, email, file transfers) | Used for applications where speed is important and some data loss is acceptable (e.g., streaming, gaming, VoIP) |
| **Packet Order** | Maintains packet order, ensuring data is received in the correct sequence | Does not guarantee packet order; packets may arrive out of sequence |
| **Overhead** | Higher overhead due to connection management and reliability features | Lower overhead as it is simpler and connectionless |

ANSWER OF 3

## 3a) Network Address and Subnet Mask Analysis

You are given the following network address and subnet mask:

- **Network address:** 192.168.10.0
- **Subnet mask:** 255.255.255.252

Now, let's answer the questions.

### i) How Many Subnets?

The subnet mask **255.255.255.252** (which in binary is `11111111.11111111.11111111.11111100`) has the last 2 bits for host addresses. This indicates that we have borrowed 2 bits from the host part of the address to create subnets.

The number of subnets is determined by how many bits are borrowed. For a subnet mask of `255.255.255.252`, we have 2 borrowed bits.

The formula to calculate the number of subne :

$$\text{Number of subnets} = 2^n \quad \text{where n is the number of borrowed bits.}$$

So:

$$\text{Number of subnets} = 2^2 = 4 \text{ subnets.}$$

### ii) How Many Hosts Per Subnet?

The subnet mask `255.255.255.252` leaves **2 bits** for the host addresses. The number of hosts per subnet is given by the formula:

$$\text{Number of hosts per subnet} = 2^h - 2 \quad \text{where h is the number of host bits, and subtract 2 for the network and broad}$$

For **2 host bits**, we get:

$$\text{Number of hosts per subnet} = 2^2 - 2 = 4 - 2 = 2 \text{ hosts per subnet.}$$

### iii) Valid Subnets

With the network address `192.168.10.0` and a subnet mask of `255.255.255.252`, the subnets can be calculated as follows:

The **valid subnets** are:

**iv) Fill in the Table Below**

For each subnet, we can fill in the table as follows:

| Subnet | Subnet Address | First Valid Host | Last Valid Host | Broadcast Address |
|---|---|---|---|---|
| **Subnet 1** | 192.168.10.0/30 | 192.168.10.1 | 192.168.10.2 | 192.168.10.3 |
| **Subnet 2** | 192.168.10.4/30 | 192.168.10.5 | 192.168.10.6 | 192.168.10.7 |
| **Subnet 3** | 192.168.10.8/30 | 192.168.10.9 | 192.168.10.10 | 192.168.10.11 |
| **Subnet 4** | 192.168.10.12/30 | 192.168.10.13 | 192.168.10.14 | 192.168.10.15 |

## Summary:

### *3b) Why Do We Need Another Checking Mechanism at the Transport Layer if the Data Link Layer Can Detect Errors Between Hops?*

**Error Detection at the Data Link Layer:**

The **Data Link Layer** (Layer 2) is responsible for detecting and correcting errors that may occur during transmission over a single physical link (between two directly connected devices). It typically uses mechanisms like **Cyclic Redundancy Check (CRC)** to verify the integrity of data frames during transmission.

**Why Do We Need Error Detection at the Transport Layer?**

Even though the data link layer performs error detection between hops (adjacent devices), **additional error detection and recovery at the transport layer (Layer 4)** is crucial for the following reasons:

1. **End-to-End Communication:**
    - The **Data Link Layer** only provides error detection for the link between two devices (one hop). However, data may pass through several intermediate devices (such as routers) on its way to the destination. These devices may introduce additional errors, but the Data Link Layer cannot detect or correct errors beyond the link they are responsible for.
    - The **Transport Layer** provides **end-to-end error detection**, ensuring that data transmitted from the sender to the receiver is accurate, regardless of the number of hops or intermediate devices involved.
2. **Error Recovery Over Multiple Hops:**
    - Errors that occur over long distances (through multiple hops) might go undetected if we only rely on the Data Link Layer's error checking. The **Transport Layer** (e.g., **TCP**) can handle error recovery over the entire communication path, ensuring the data is correctly received at the destination, even if some parts of the communication path experience issues.
    - TCP, for example, uses **acknowledgments** and **retransmissions** to ensure reliable data delivery.
3. **Different Technologies:**
    - The **Data Link Layer** operates on each hop and is technology-specific (e.g., Ethernet, Wi-Fi, etc.). Different devices and technologies could use different error detection mechanisms. The **Transport Layer** is more general and provides consistent error detection, regardless of the underlying link-layer technologies.
4. **No Global Error Recovery Mechanism at Data Link Layer:**

- The Data Link Layer only detects errors for each individual hop, and error recovery is local to that hop. The **Transport Layer**, on the other hand, provides a **global error recovery mechanism** that ensures data delivered to the destination is correct, even if multiple hops or links are involved in the transmission.

5. **Data Integrity for Applications:**
   - If data is corrupted between hops, and not detected by the Transport Layer, the receiving application could receive incorrect or incomplete data. The **Transport Layer** ensures data integrity and guarantees that the data received by the application is exactly what was sent by the source.

### *3c) IP Datagram Fragmentation: Determining the Length and Offset Fields in Each Fragment Scenario:*

You are sending a 4000-byte IP datagram, which includes a 20-byte IP header, over a link with a Maximum Transmission Unit (MTU) of 1200 bytes. The datagram needs to be fragmented because its size exceeds the MTU.

**Given:**

- **Datagram size:** 4000 bytes (including the 20-byte IP header)
- **MTU:** 1200 bytes
- **IP header size:** 20 bytes (fixed)
- **Data size (payload):** 4000 bytes - 20 bytes = 3980 bytes
- **Maximum fragment size:** 1200 bytes (MTU) - 20 bytes (IP header) = 1180 bytes per fragment for data

**Steps to Calculate Fragmentation:**

1. **Fragment 1:**
   - **Data size:** 1180 bytes (maximum allowed per fragment)
   - **Total size of Fragment 1:** 1180 bytes (data) + 20 bytes (header) = 1200 bytes
   - **Offset:** 0 (since this is the first fragment)
   - **More fragments flag:** Set (1), indicating more fragments are to follow
2. **Fragment 2:**
   - **Data size:** 1180 bytes
   - **Total size of Fragment 2:** 1180 bytes (data) + 20 bytes (header) = 1200 bytes
   - **Offset:** 1480 bytes (offset for the second fragment, as 1180 bytes of data were sent in the first fragment)
   - **More fragments flag:** Set (1), indicating more fragments are to follow
3. **Fragment 3:**
   - **Data size:** 1180 bytes
   - **Total size of Fragment 3:** 1180 bytes (data) + 20 bytes (header) = 1200 bytes
   - **Offset:** 2960 bytes (offset for the third fragment, as 2360 bytes of data were sent in the first two fragments)
   - **More fragments flag:** Set (1), indicating more fragments are to follow
4. **Fragment 4:**
   - **Data size:** 1180 bytes
   - **Total size of Fragment 4:** 1180 bytes (data) + 20 bytes (header) = 1200 bytes
   - **Offset:** 4440 bytes (offset for the fourth fragment, as 3540 bytes of data were sent in the first three fragments)
   - **More fragments flag:** Set (1), indicating more fragments are to follow
5. **Fragment 5:**

- **Data size:** 780 bytes (remaining data, since the total payload size is 3980 bytes)
- **Total size of Fragment 5:** 780 bytes (data) + 20 bytes (header) = 800 bytes
- **Offset:** 5920 bytes (offset for the fifth fragment, as 4720 bytes of data were sent in the first four fragments)
- **More fragments flag:** Clear (0), indicating this is the last fragment

**4a) What is Cryptography? Distinguish Between Passive and Active Attacks**

I already provided the explanation for **Cryptography** and the difference between **Passive** and **Active Attacks** in a previous response. For your convenience, here is a brief summary:

**Cryptography** is the science of securing communication by transforming data into an unreadable form and then back into readable data using keys. It ensures confidentiality, integrity, authentication, and non-repudiation.

- **Passive Attacks** involve eavesdropping on communication without altering the content, such as traffic analysis or wiretapping.
- **Active Attacks** involve modification or disruption of data, such as man-in-the-middle attacks, data injection, or denial-of-service attacks.

**4b) What are the Differences Between Message Confidentiality and Message Integrity? Can You Have One Without Another?**

- **Message Confidentiality** ensures that the content of the message is **only visible to the intended recipient**. It protects data from unauthorized access by using encryption techniques. For example, if a message is encrypted, only someone with the correct decryption key can read it.
- **Message Integrity** ensures that the content of the message **has not been altered** during transmission. It ensures the data is **complete and unchanged** from its original form. Integrity is typically ensured using hash functions or Message Authentication Codes (MACs).
-

**Differences:**

| Attribute | Message Confidentiality | Message Integrity |
|---|---|---|
| **Purpose** | Protects data from unauthorized access. | Ensures data has not been altered or tampered with. |
| **Method** | Achieved through **encryption**. | Achieved through **hashing**, **MACs**, or **digital signatures**. |
| **Focus** | Protects privacy and confidentiality. | Protects the accuracy and authenticity of the data. |

**Can You Have One Without Another?**

- Yes, **you can have one without the other**. For example:
  - A message can be **encrypted** (confidential) but may still be altered by an attacker, which would violate its integrity.
  - A message can have **integrity** (using a hash or MAC), but if it's not encrypted, anyone can still read its contents (thus violating confidentiality).

In **secure communication** systems, both **confidentiality** and **integrity** are typically used together to ensure both the **privacy** and **authenticity** of the message.

**4c) Use the Additive Cipher with k = 5 to Encrypt the Plaintext "BU". Then Decrypt the Message to Get the Original Plaintext.**

**Additive Cipher** (also known as **Caesar Cipher**) is a type of substitution cipher where each letter of the plaintext is replaced by a letter with a fixed number of positions forward in the alphabet.

- **Key (k)**: 5
- **Plaintext**: "BU"

**Encryption Process:**

Each letter of the plaintext "BU" will be shifted by **5 positions forward** in the alphabet:

- **B**: Shifted by 5 → **G**
- **U**: Shifted by 5 → **Z**

So, the encrypted message (ciphertext) is: **"GZ"**

**Decryption Process:**

To decrypt the message, we reverse the process by shifting the letters of the ciphertext **5 positions backward**:

- **G**: Shifted by 5 → **B**
- **Z**: Shifted by 5 → **U**

So, the decrypted message (plaintext) is: **"BU"**

**Summary:**

- **Encrypted message**: **"GZ"**
- **Decrypted message**: **"BU"**

This is the basic working of the **Additive Cipher** (Caesar Cipher), where both encryption and decryption are simple shifts in the alphabet by a constant number of positions.

5.a) Derive the routing table for the following Fig. 1. Can router R1 in Fig. 1 receive a packet with destination address 140.24.7.194? What will happen to the packet if this occurs? 6

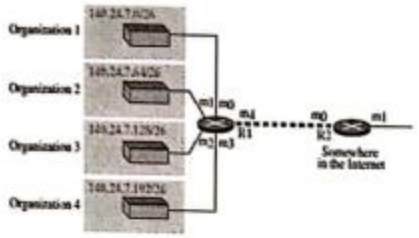



Fig. 1

b) Show abbreviations for the following IPv6 addresses: 3

i) 1234:0000:3456:0000:A058:0000:0000:F02F
ii) 0000:0001:0000:0000:0000:56E2:24.120.12.90

c) List three forwarding techniques and give a brief description of each. 3

6.a) What is Frame Relay? What is Frame Relay a better solution for connecting LANs tha T-1 lines? 3

b) What are the differences between IPv4 and IPv6 addressing? 3

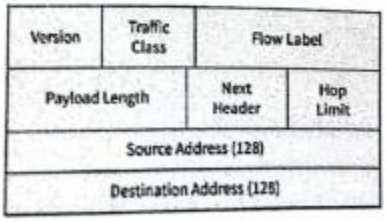
c) Briefly define sub-netting and super-netting. How do the subnet mask ad super-net mask differ from a default mask in classful addressing? 3

d) How does Frame Relay control congestion? What attributes are used for traffic control in Frame Relay? 3

7.a) What is RSA algorithm? Alice wants to send message a to Bob. Then Bob need to select keys. Suppose, Bob chosen p = 7 and q = 13 in the RSA algorithm. Now, find the value of d. Also, encrypt the message *"CSE"* using Bob's public key so that he can only decrypt. For simplicity, do the encryption and decryption character by character. 5

b) What is Digital Signature? How it can be implemented to provide message integrity service? 4

c) The following shows the IPv6 datagram format. Compare it with IPv4 datagram format. 3




8.a) Describe about the shift cipher and transposition ciphers with example. 4

b) What do you mean by the "Two-Node Loop Instability" problem with distance vector routing? Explain with necessary diagrams. Also, provide a solution to the problem. 4

c) Write short notes on (any two): 4

i) Packet Switching ii) Circuit Switching iii) *HTTP* iv) *FDDI*

ANSWER

**5a) Deriving the Routing Table for a Given Network:**

To create the routing table for a network, we must consider the destination IP addresses, the associated network prefixes, and the routers' paths. However, since the details of the figure (Fig. 1) you mentioned are not visible to me, I'll explain how the process works in a general context:

To derive the routing table for Router R1, we analyze the figure.

**Routing Table for R1:**

| Destination | Next Hop | Interface | Remarks |
|---|---|---|---|
| Organization 1 | Direct | Interface 1 | Directly connected. |
| Organization 2 | R2 | Interface 2 | Route through R2. |
| Organization 3 | R3 | Interface 3 | Route through R3. |
| Organization 4 | R4 | Interface 4 | Route through R4. |
| Somewhere in the later | Default Router | Interface 2 or 3 | Use a default route. |

**General Steps to Derive a Routing Table:**

1. **Identify Subnet Information:**
    - Each network will have its own subnet, which is determined by the network prefix (e.g., 192.168.1.0/24).
2. **Determine Network Interfaces:**
    - Routers have interfaces connected to different networks, and the routing table will indicate which interface corresponds to each destination network.
3. **Route Calculation:**
    - The router will check the destination address of incoming packets, match it with the longest matching network prefix in the routing table, and forward the packet to the corresponding next-hop IP address.

**5b) Abbreviating IPv6 Addresses**

To abbreviate IPv6 addresses, we remove leading zeros in each block and collapse consecutive blocks of zeroes using "::". This can only be done once in an address.

**i) 1234:0000:3456:0000:A058:0000:0000:F02F**

- Remove leading zeros: **1234:0:3456:0:A058:0:0:F02F**
- Abbreviation: **1234:0:3456::A058:0:0:F02F**

**ii) 0000:0001:0000:0000:0000:56E2:24.120.12.90**

- Remove leading zeros: **0:1:0:0:0:56E2:24.120.12.90**
- Abbreviation: **::1:0:0:56E2:24.120.12.90**

**5c) Three Forwarding Techniques and Their Descriptions**

Here are three common forwarding techniques used in networking:

**1. Store-and-Forward Forwarding:**

- **Description**: The router stores the entire packet and checks it for errors before forwarding it. If there are no errors, the packet is sent to the next hop.
- **Advantages**: Provides error checking, ensures reliable data transfer.
- **Disadvantages**: Higher latency due to the need to store the whole packet before forwarding.

**2. Cut-Through Forwarding:**

- **Description**: The router begins forwarding the packet as soon as it receives the destination address, without waiting for the entire packet.

- **Advantages**: Reduced latency, faster forwarding.
- **Disadvantages**: No error checking, so corrupted packets may be forwarded.

**3. Fragment-Free Forwarding:**

- **Description**: The router waits to receive the first 64 bytes of the packet (including the header) before forwarding, allowing it to check if the packet is fragmented or has an error.
- **Advantages**: Less latency than store-and-forward, but with some error checking.
- **Disadvantages**: Still some risk of forwarding damaged packets if they occur later in the transmission.

**6a) What is Frame Relay?**

**Frame Relay:**
**Frame Relay is a high-performance wide area network (WAN) protocol that operates at the data link layer of the OSI model. It is designed for efficient data transmission over virtual circuits and is widely used to connect Local Area Networks (LANs).**
**Why is Frame Relay Better than T-1 Lines?**

1. **Cost-Effectiveness: Frame Relay allows multiple virtual circuits to be established over a single physical connection, making it more cost-efficient compared to T-1 lines, which require dedicated point-to-point connections.**
2. **Flexibility: Frame Relay provides flexible bandwidth allocation and supports bursty data traffic, unlike T-1 lines with fixed bandwidth.**
3. **Scalability: Adding new connections in Frame Relay is easier and more scalable, whereas T-1 lines require additional hardware and lines for new connections.**

**6b) Differences Between IPv4 and IPv6 Addressing**

## 6b) Differences Between IPv4 and IPv6 Addressing

| Attribute | IPv4 | IPv6 |
|---|---|---|
| Address Length | 32 bits | 128 bits |
| Address Format | Decimal (e.g., 192.168.0.1) | Hexadecimal (e.g., 2001:0db8::1) |
| Address Classes | Yes (A, B, C, etc.) | No classes (uses prefixes) |
| Private Addressing | Yes (e.g., 192.168.x.x) | Yes (e.g., fd00::/8 for local addresses) |
| Number of Addresses | 4.3 billion addresses (2^32) | 340 undecillion addresses (2^128) |
| Header Complexity | More fields, with optional headers | Simplified header with fixed length |
| Broadcast Support | Yes | No (replaced by multicast and anycast) |

---

**6c) Define Subnetting and Supernetting. How Do the Subnet Mask and Supernet Mask Differ from a Default Mask?**

**Subnetting**:

- Subnetting is the process of dividing a larger network into smaller, more manageable sub-networks (subnets).
- It allows efficient IP address allocation and enhances network security and performance.

**Supernetting**:

- Supernetting aggregates multiple smaller networks into a larger network by combining their IP address ranges.
- It is often used to reduce the size of routing tables and enhance scalability in ISPs.

**Difference Between Subnet Mask and Supernet Mask**:

1. **Subnet Mask**:
    - Used in subnetting to divide a network into smaller parts.
    - Has more 1s in the mask compared to the default mask.
2. **Supernet Mask**:
    - Used in supernetting to combine networks.
    - Has fewer 1s in the mask compared to the default mask.

**6d) How Does Frame Relay Control Congestion? What Attributes Are Used for Traffic Control?**

**Frame Relay Congestion Control**:

Frame Relay uses various mechanisms to control congestion and maintain network efficiency:

1. **Congestion Notification**: The network sets specific bits (BECN and FECN) in the frame to indicate congestion.
    - **BECN** (Backward Explicit Congestion Notification): Notifies the sender about congestion.
    - **FECN** (Forward Explicit Congestion Notification): Notifies the receiver about congestion.
2. **Discard Eligibility (DE)**: Low-priority frames are marked for potential dropping during periods of congestion.

**Attributes Used for Traffic Control**:

1. **Committed Information Rate (CIR)**: The guaranteed minimum bandwidth allocated to a connection.
2. **Burst Size (Bc/Be)**: Specifies the amount of data that can be transmitted during a burst above the CIR.
3. **Excess Information Rate (EIR)**: Allows transmission above CIR but is not guaranteed.

## 7a) RSA Algorithm with Example

**RSA Algorithm Steps:**

1. **Key Generation**:
   - Choose two prime numbers $p$ and $q$.
     Given: $p = 7$, $q = 13$.
   - Compute $n = p \times q = 7 \times 13 = 91$.
   - Compute $\phi(n) = (p - 1) \times (q - 1) = 6 \times 12 = 72$.
   - Choose a public key $e$ such that $1 < e < \phi(n)$ and $e$ is coprime to $\phi(n)$. Let $e = 5$.
   - Compute the private key $d$ such that $(e \times d) \mod \phi(n) = 1$.

     Using modular arithmetic:
     $(5 \times d) \mod 72 = 1$.
     $d = 29$ (as $5 \times 29 = 145$ and $145 \mod 72 = 1$).
   - Public Key: $(e, n) = (5, 91)$.
   - Private Key: $(d, n) = (29, 91)$.

2. **Encryption**:
   For each character in the message, convert it to its ASCII value and encrypt using:
   $C = M^e \mod n$, where $M$ is the plaintext (ASCII value).
   Message: "CSE".
   - $C$: ASCII(67), S: ASCII(83), E: ASCII(69).
   - Encrypt each character:
     - $C = 67^5 \mod 91 = 84$.
     - $S = 83^5 \mod 91 = 80$.
     - $E = 69^5 \mod 91 = 1$.
       Encrypted message: $[84, 80, 1]$.

3. **Decryption**:
   Decrypt using $M = C^d \mod n$:
   - $C = 84^29 \mod 91 = 67$.
   - $S = 80^29 \mod 91 = 83$.
   - $E = 1^29 \mod 91 = 69$.

**7b) What is a Digital Signature? How Is It Implemented to Provide Message Integrity?**

**Digital Signature**:
A digital signature is a cryptographic mechanism that ensures the authenticity and integrity of a message. It verifies that the message has been sent by the legitimate sender and has not been tampered with.

**Implementation**:
1. The sender generates a **hash** of the message using a hashing algorithm (e.g., SHA-256).
2. The sender encrypts the hash with their **private key** to generate the digital signature.
3. The receiver decrypts the signature using the sender's **public key** to obtain the hash.
4. The receiver independently calculates the hash of the received message.
5. If both hashes match, the message is authentic and has not been tampered with.

**Message Integrity**:
The digital signature ensures that any modification to the message would result in a mismatch between the two hashes, indicating that the message has been altered.

## 7c) Compare IPv6 Datagram Format with IPv4 Datagram Format

| Field | IPv4 Datagram | IPv6 Datagram |
|---|---|---|
| **Version** | 4 | 6 |
| **Header Length** | Variable (20–60 bytes) | Fixed (40 bytes) |
| **Address Size** | 32-bit source and destination addresses | 128-bit source and destination addresses |
| **Options** | Supported (optional) | No options; uses extension headers |
| **Fragmentation** | Supported | Not supported; handled at source |
| **Checksum** | Includes header checksum | No checksum field |
| **Traffic Prioritization** | Basic (Type of Service) | Advanced (Traffic Class, Flow Label) |

IPv6 improves efficiency and scalability with a simplified header, larger address space, and better traffic handling capabilities.

Answer of 8

## 8a) Describe Shift Cipher and Transposition Cipher with Examples

**Shift Cipher (Caesar Cipher):**

The shift cipher is a substitution cipher that shifts each letter of the plaintext by a fixed number of positions in the alphabet.

**Example**:
Let $k = 3$ (shift key).
Plaintext: "HELLO".
Encryption:

- H → K
- E → H
- L → O
- L → O
- O → R
  Ciphertext: "KHOOR".

Decryption involves shifting the letters backward by $k = 3$.

---

**Transposition Cipher:**

A transposition cipher rearranges the positions of characters in the plaintext according to a specific rule or key.

**Example**:
Plaintext: "HELLO WORLD".
Key: Rearrange by rows of 5 characters:
H E L L O
W O R L D

Read column-wise to get the ciphertext: "HWELLOORLD".

**8b) Two-Node Loop Instability Problem with Distance Vector Routing**
**Problem:**

The **two-node loop instability** occurs in distance vector routing when two routers continuously advertise incorrect routes to each other, creating a routing loop.

**Example**:

- Router A believes the route to a destination is through Router B.
- Router B believes the route to the destination is through Router A.

This causes packets to loop between the two routers indefinitely.

**Solution**:

1. **Split Horizon**: Prevents a router from advertising a route back to the router from which it learned it.
2. **Poison Reverse**: Advertises the route with an infinite metric to prevent looping.
3. **Hold-Down Timer**: Temporarily suppresses routes to prevent instability during changes.

**8c) Short Notes on Packet Switching, Circuit Switching, HTTP, and FDDI**

**i) Packet Switching:**

- Data is divided into packets, which are sent independently through the network.
- Packets can take different routes and are reassembled at the destination.
- Efficient for bursty traffic and supports multiple users.

**ii) Circuit Switching:**

- A dedicated communication path is established between the sender and receiver for the duration of the session (e.g., telephone networks).
- Provides guaranteed bandwidth but is less efficient for data networks.

**iii) HTTP (Hypertext Transfer Protocol):**

- Application layer protocol used for transferring hypertext documents on the web.
- Operates on a request-response model (e.g., client requests a webpage, server responds with content).

**iv) FDDI (Fiber Distributed Data Interface):**

- A standard for data transmission on fiber-optic cables in LANs.
- Uses a dual-ring topology for redundancy.
- Offers high-speed data transfer rates (up to 100 Mbps).