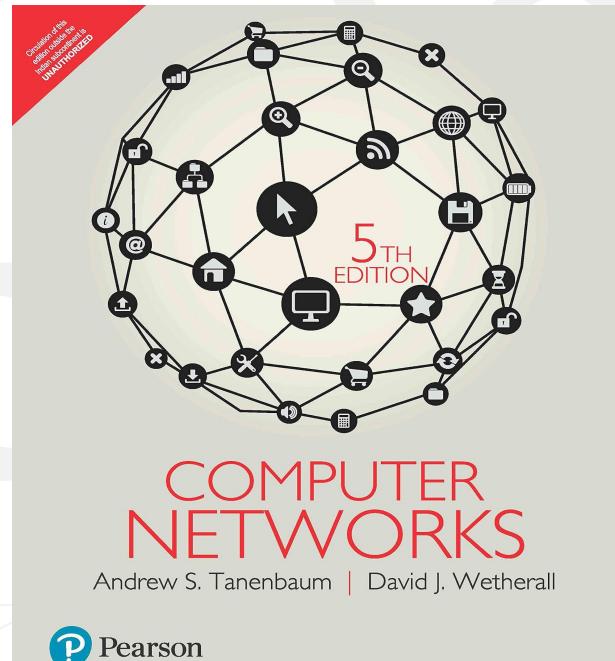
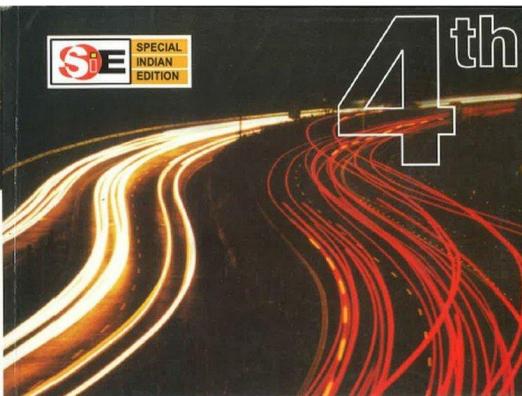


# Computer networks

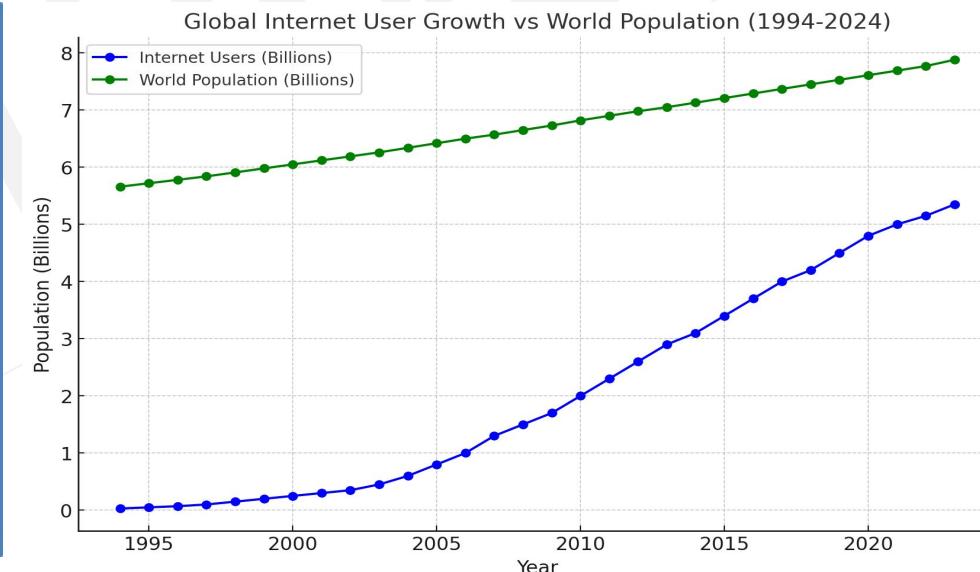
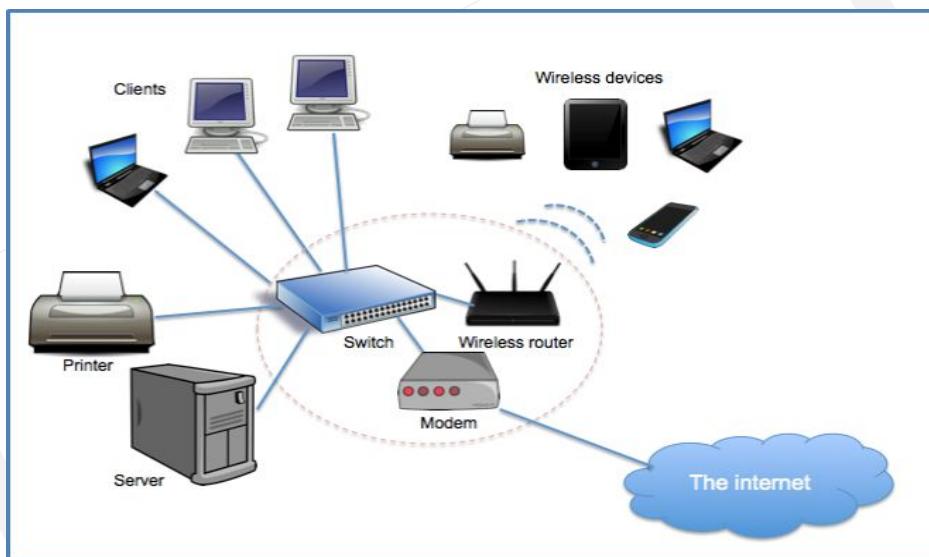
- **Core Subject** for CS/IT Students. **GATE Importance:** 7-8 marks out of 100, with 5-6 questions on average. **Question Type:** Primarily numerical problems. **Time Investment:** Requires moderate preparation but offers high scoring potential. **Application:** Directly applied in specific industries like Networking, Telecom, and IT Infrastructure.

Concept of layering: OSI and TCP/IP Protocol Stacks; Basics of packet, circuit and virtual circuit-switching; Data link layer: framing, error detection, Medium Access Control, Ethernet bridging; Routing protocols: shortest path, flooding, distance vector and link state routing; Fragmentation and IP addressing, IPv4, CIDR notation, Basics of IP support protocols (ARP, DHCP, ICMP), Network Address Translation (NAT); Transport layer: flow control and congestion control, UDP, TCP, sockets; Application layer protocols: DNS, SMTP, HTTP, FTP, Email.

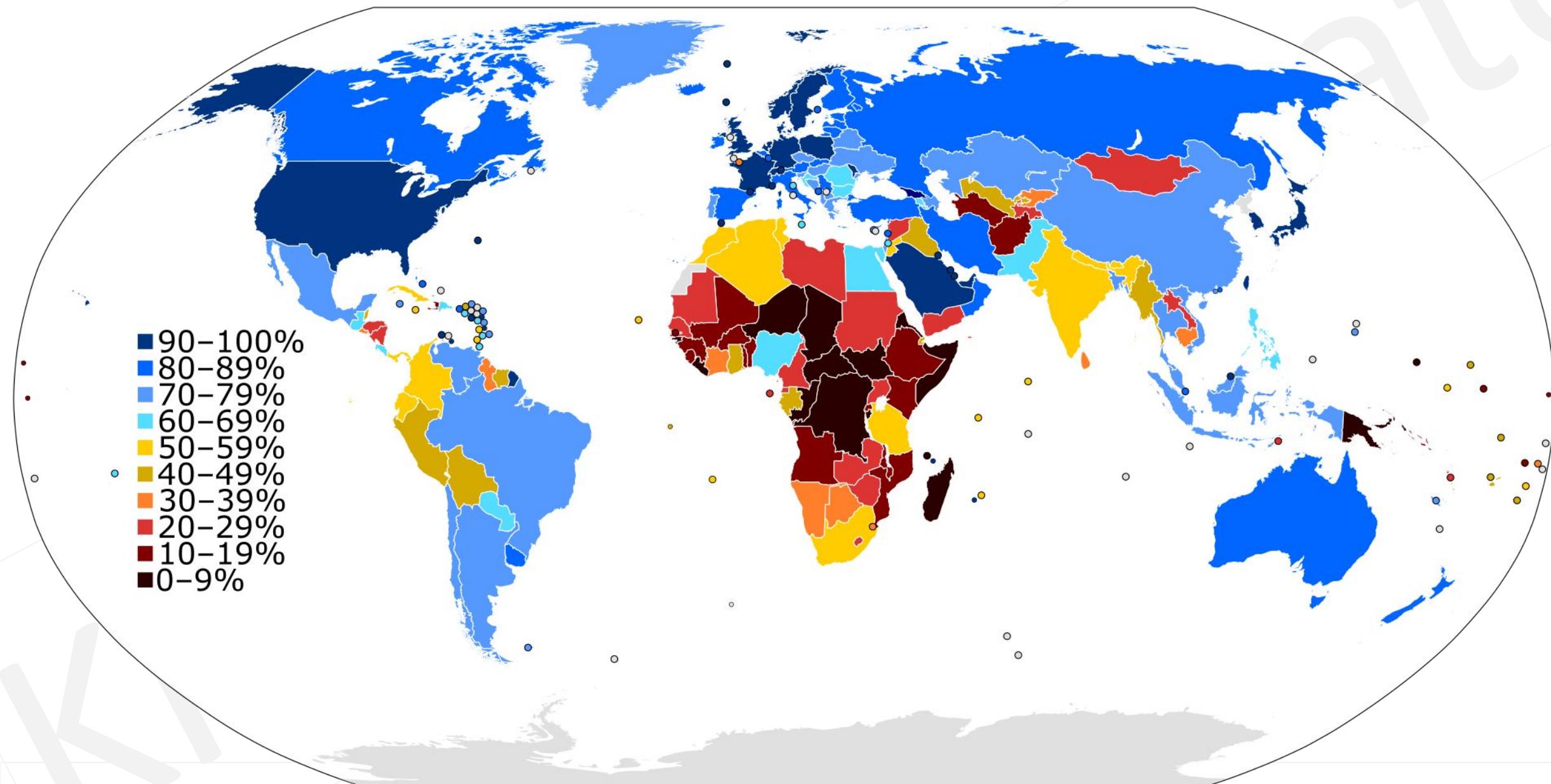


# Basics of Computer Networks

- A **computer network** is a telecommunications framework that allows digital devices (nodes) to exchange data via **wired or wireless connections** to share resources, such as hardware or software (e.g., the internet). It is a collection of **autonomous computers** interconnected by a single technology. Networks vary in **sizes, shapes, and forms**, and they are often connected to create larger networks, with the **Internet** being the most prominent example—a network of networks.



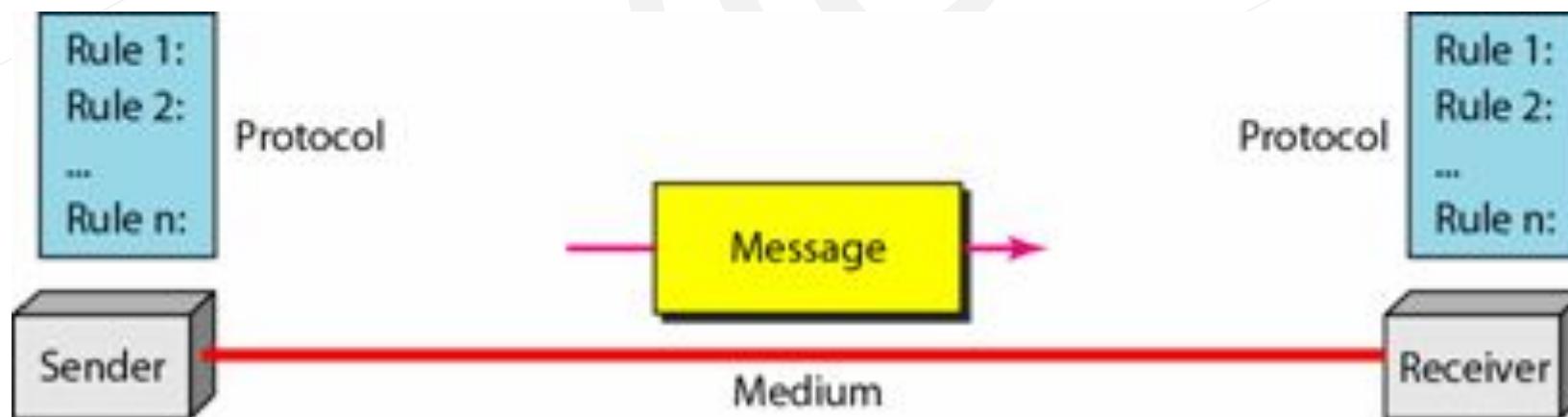
# Internet means empowerment



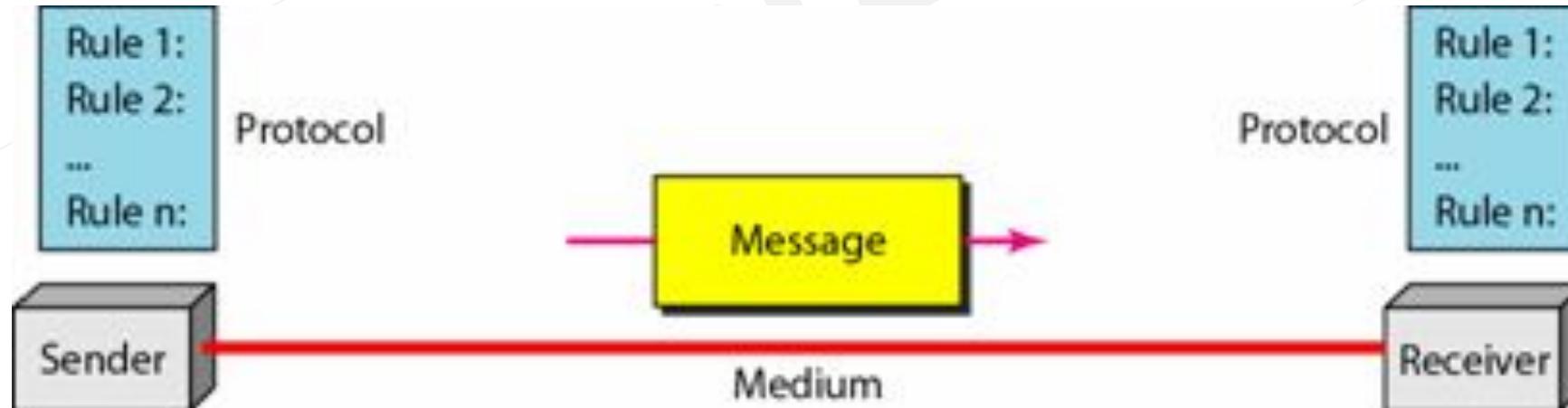
# Data communication

Data communications are the exchange of data between two devices via some transmission medium. Data communication system has five components

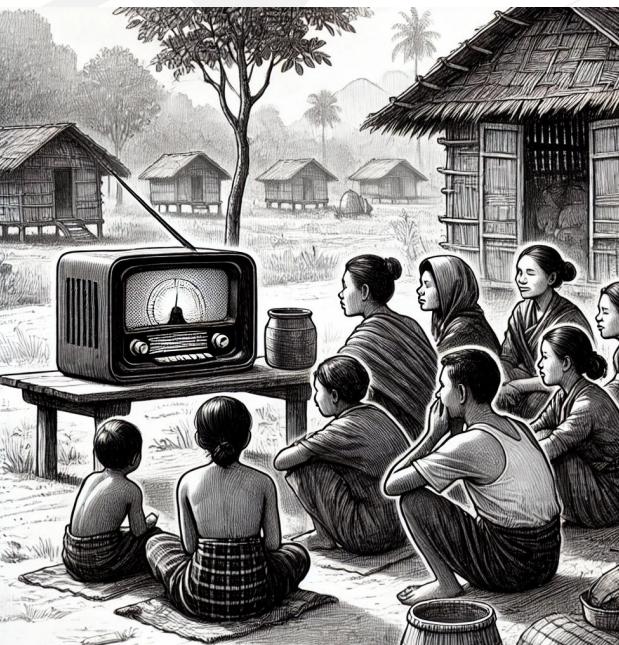
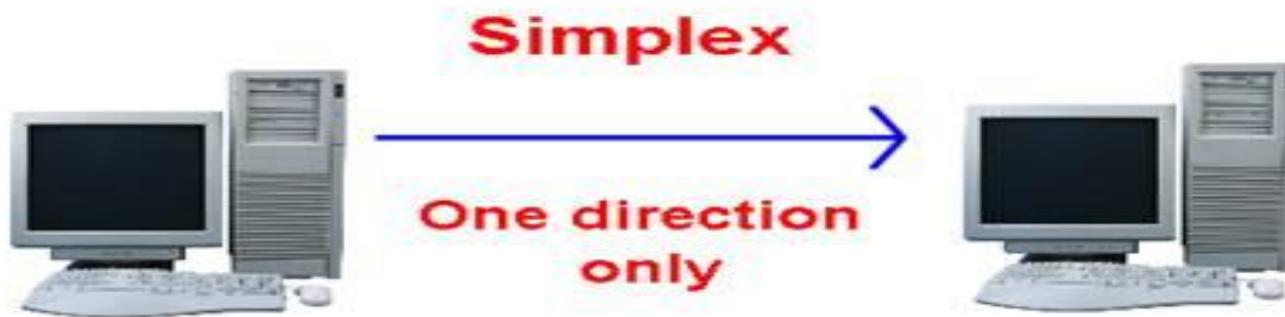
1. **Message**- information (data) to be communicated e.g. text, audio, video.
2. **Sender**- device how sends the message (computer, phone, camera etc.)
3. **Receiver**- device how receives the message (computer, phone, television etc.)
4. **Transmission medium** – is the physical path by which a message travels from sender to receiver.
5. **Protocol** – the set of rules that governs the data communication.



- Effectiveness of the data communications system depends on four fundamental characteristics
  1. **Delivery**- Must deliver the data to correct destination.
  2. **Accuracy**- Must be delivered accurately without any error
  3. **Timeliness**- Must deliver the data in a timely manner, sometime time in real time applications data delivered after time is useless.
  4. **Jitter**- Refers to variation in the packet arrival time i.e. the uneven delay between the packets (mismatch in audio and picture in a video)



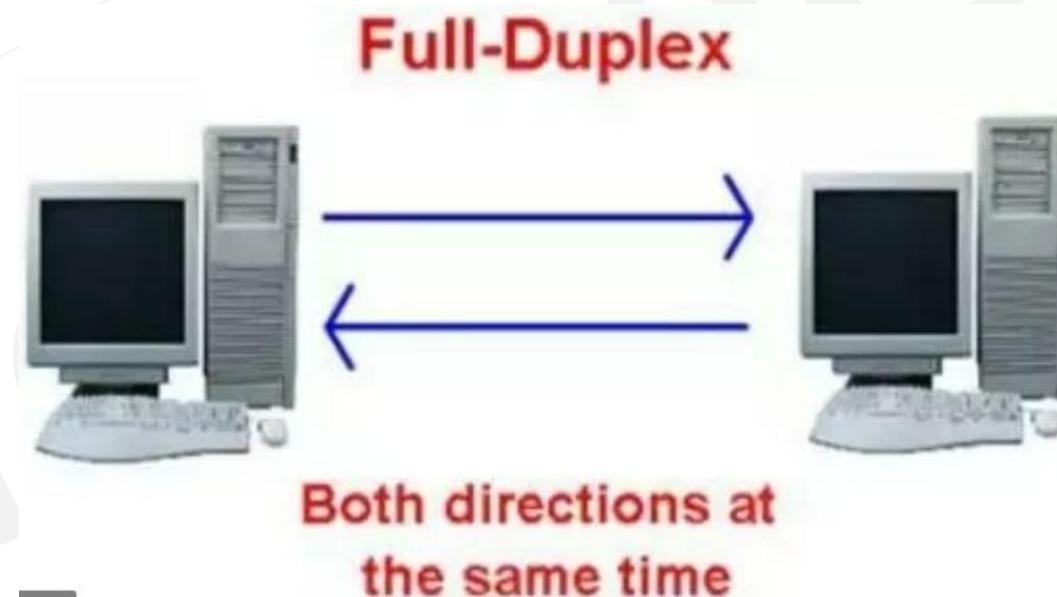
- Transmission mode between two systems can be categorised into three types
  - **Simplex** – The communication is unidirectional, as a one-way street. one device always sends can always send other can always receive. E.g. radio, mouse.
  - The simplex mode can use the entire capacity of the channel to send data in one direction.



- **Half duplex** – each station can both transmit and receive, but not at the same time. E.g. like a one lane road, walkie-talkie etc.
  - When one device is sending, the other can only receive, and vice versa.
  - In a half-duplex transmission, the entire capacity of a channel is taken over by whichever of the two devices is transmitting at the time.
  - Walkie-talkies are half-duplex systems.



- **Full duplex** – both stations can transmit and receive at the same time. Actually, it is two half duplex connections.
- Telephone network is an example of full-duplex mode, when two people are communicating by a telephone line, both can talk and listen at the same time.
- The capacity of the channel, must be divided between the two directions.

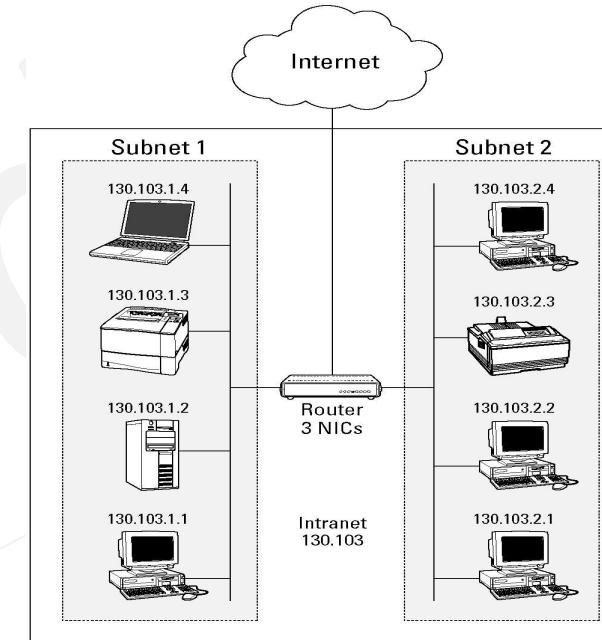


- Network criteria- a network must be able to meet a certain number of criteria. The most important of these are performance, reliability and security.
  1. **Performance** – Can be measured in many ways including transit time, response time, number of users, type of transmission medium, capabilities of connected hardware's and efficiency of software.
  2. **Reliability** – Is a measure of frequency of failure and the time taken to resolve from the failure.
  3. **Security** – Includes protecting data from unauthorised access, protecting data from damage and development.



## Types of connection-

- **Point to point**- A point-to-point connection provides a dedicated link between two devices. Most point-to-point connections use an actual length of wire or cable to connect the two ends, but other options, such as microwave or satellite links, are also possible.
- **Multipoint** - A multipoint (also called multidrop) connection is one in which more than two specific devices share a single link.



**Physical topology** - Refers to the physical layout of a network and how devices are interconnected. It is the **geometric representation** of the relationships between links and networking devices.



Point to Point



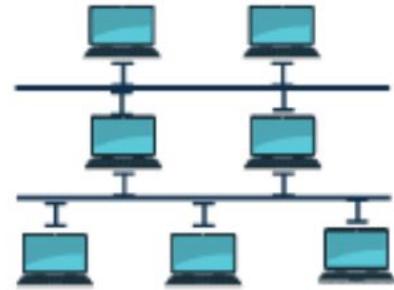
Bus



Ring



Star



Tree



Mesh



Hybrid

# Mesh Topology

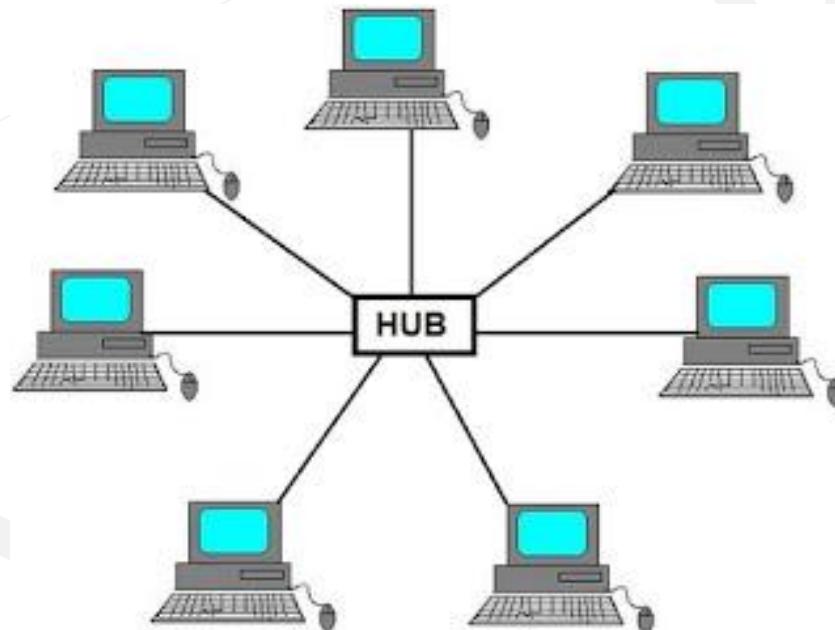
- **Mesh Topology**
  - In a mesh topology, each device has a **dedicated point-to-point link** to every other device.
  - Requires  $n(n-1)/2$  duplex-mode links for  $n$  nodes.
- **Advantage:**
  - Dedicated links eliminate traffic issues, provide robustness, ensure privacy, and make fault identification easy.
- **Disadvantage:**
  - Difficult installation, high wiring bulk, and expensive hardware requirements.

**Mesh Topology**



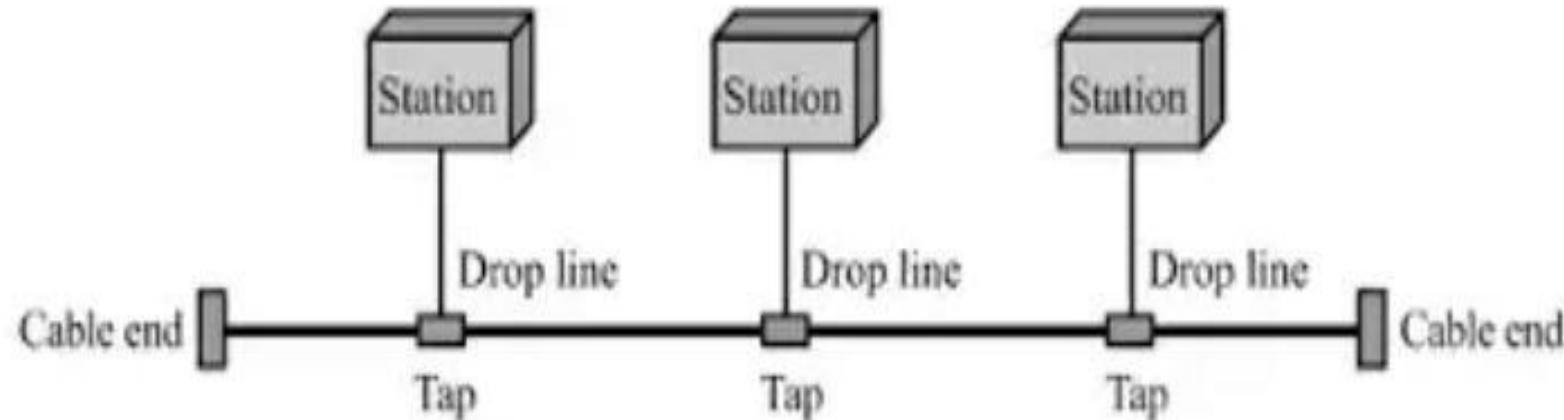
# Star Topology

- Each device has a **dedicated point-to-point link** to a central hub that manages communication between devices.
- **Advantage:**
  - Easy to install, reconfigure, and identify faults. It is robust (failure of one link doesn't affect the rest), and less expensive than mesh topology.
- **Disadvantage:**
  - Entire network depends on the central hub, and if it fails, the whole system is affected. Requires more cabling than other topologies.



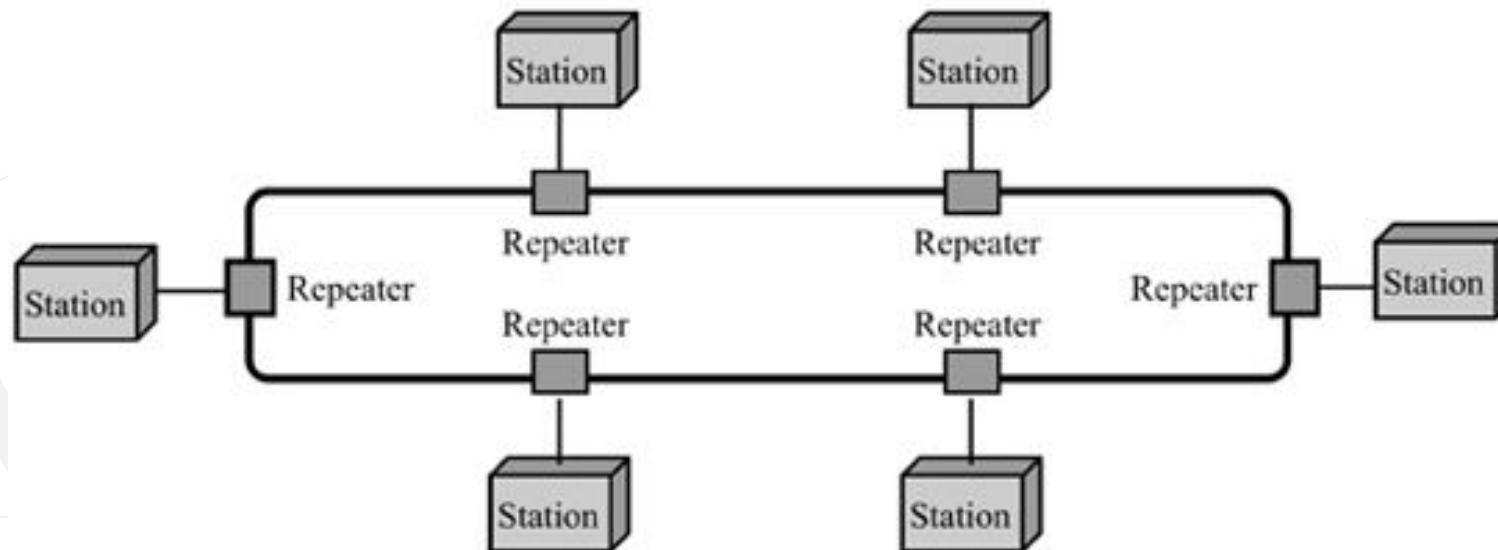
# Bus Topology

- A multipoint topology where one long cable acts as a **backbone** connecting all devices.
- Devices connect to the backbone via **drop lines** and **taps**.
- **Advantage:**
  - Easy to install and requires less cabling compared to mesh or star topologies.
- **Disadvantage:**
  - Faults are difficult to isolate, and a break in the bus cable halts all transmissions. It is also challenging to add new devices.



# Ring Topology

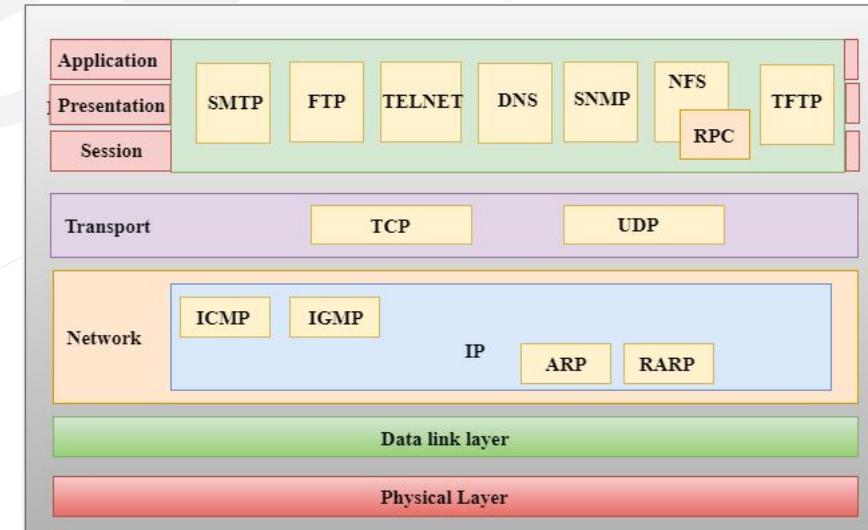
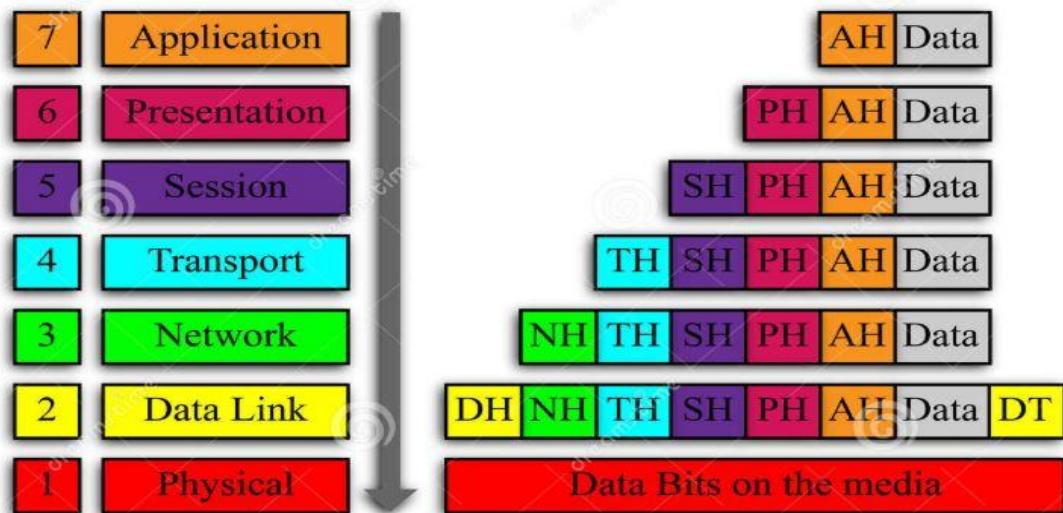
- In a ring topology, each device connects directly to two adjacent devices, forming a closed loop. Data signals travel in one direction through each device until they reach their destination. Each device functions as a repeater, regenerating the signal before passing it to the next.
- **Advantages:**
  - Simple installation and reconfiguration. Easier fault isolation.
- **Disadvantages:**
  - A single point of failure (like a disabled device) can disrupt the entire network.



# Network Models

- The OSI model, proposed by the International Standards Organization (ISO), enables communication between different systems irrespective of their underlying architecture. It serves as a conceptual framework for creating flexible, robust, and interoperable network architectures, not as a protocol itself.
  - Open System:** A set of protocols allowing different systems to communicate.
  - Purpose:** To facilitate system communication without modifying hardware or software logic.
  - Structure:** Comprises seven interrelated layers, each defining a specific part of the information transfer process across a network.

OSI Model

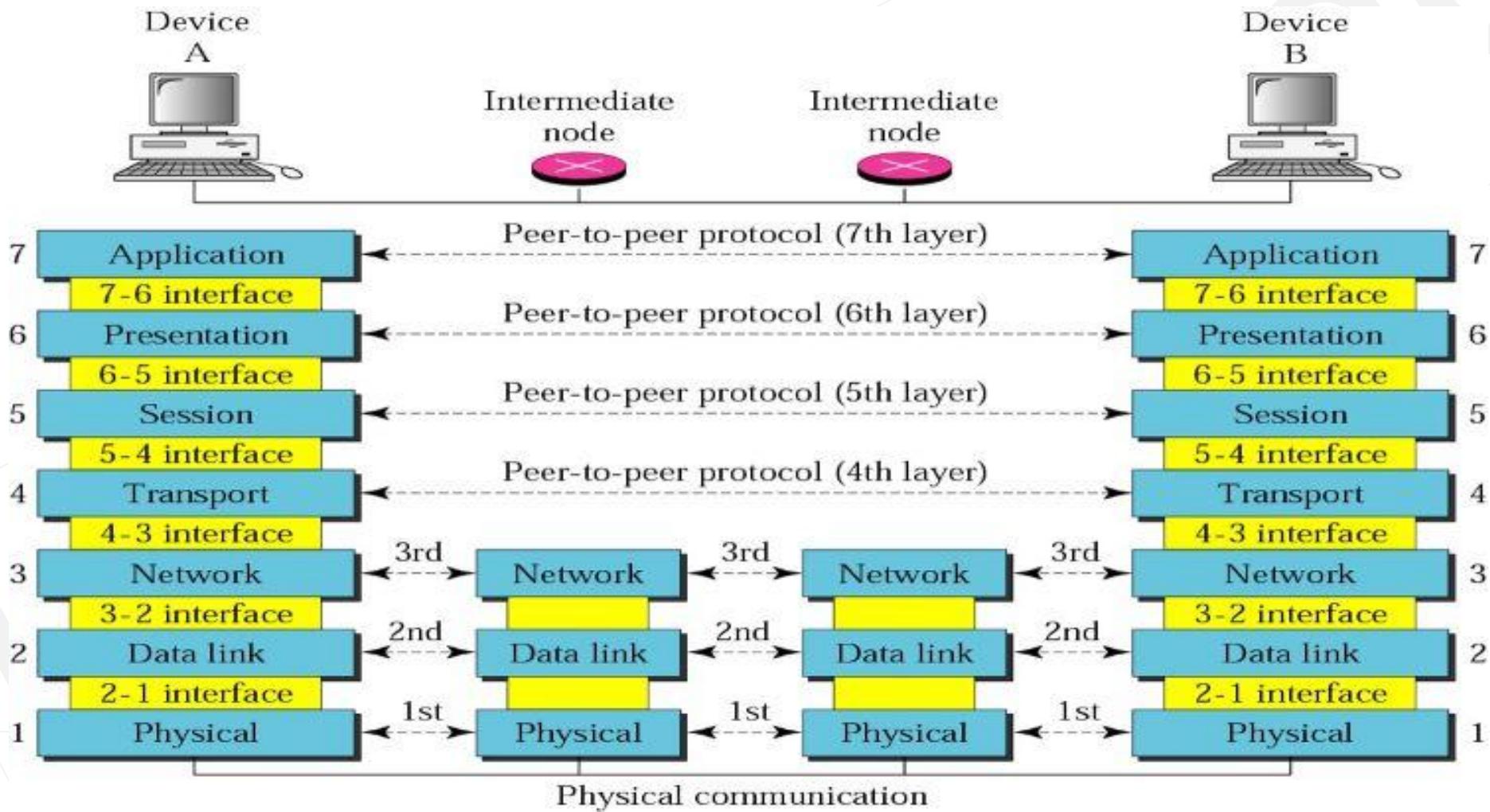


Directive Principles

Constitution  
[www.knowledgegate.in](http://www.knowledgegate.in)

## Layered Architecture

- Each layer on one machine communicates with the same layer on another machine, following a set of rules called protocols. Within a single machine, each layer relies on the services of the layer below it and provides services to the layer above. The processes that communicate between layers are known as **peer-to-peer processes**.



# Physical Layer Overview

The physical layer defines the basic interface between devices and the transmission medium.

## **Key Responsibilities:**

- **Bit Representation**: Manages the raw bit stream (0s and 1s) and converts them into signals (electrical or optical) for transmission.
- **Data Rate**: Determines the number of bits transmitted per second.
- **Line Configuration**: Defines the connection setup between devices and the transmission media.
- **Physical Topology**: Specifies how devices are physically interconnected to form a network.
- **Transmission Mode**: Controls the direction of data transmission, supporting simplex, half-duplex, and full-duplex modes.

# Data Link Layer Overview

The data link layer is responsible for reliable transmission of data across a physical network.

## **Key Functions:**

- **Framing:** Breaks data from the network layer into smaller, manageable frames.
- **Physical Addressing:** Adds a header to identify the sender and/or receiver.
- **Access Control:** Ensures that when multiple devices share a link, only one has control at a time.
- **Flow Control:** Manages the rate of data transmission to prevent the receiver from being overwhelmed.
- **Error Control:** Detects and retransmits damaged or lost frames while eliminating duplicates using a trailer mechanism.

# Network Layer Overview

The network layer ensures end-to-end packet delivery from the source to the destination, possibly across multiple networks.

## **Key Functions:**

- **Logical Addressing**: Adds a header containing the logical addresses (e.g., IP addresses) of the sender and receiver, allowing packets to traverse network boundaries.
- **Routing**: Determines the best path for data packets through interconnected networks (internetworks) using routers or switches.

## Transport Layer Overview

The transport layer is responsible for reliable, process-to-process communication.

### **Key Functions:**

- **Service-Point Addressing**: Adds a port address to ensure data reaches the correct process on the destination computer.
- **Segmentation and Reassembly**: Divides messages into smaller segments, assigns sequence numbers, and reassembles them at the destination.
- **Connection Control**: Supports both connection-oriented (establishes a connection before data transfer) and connectionless communication (treats each segment independently).
- **Flow Control**: Manages data flow end-to-end to prevent the sender from overwhelming the receiver.
- **Error Control**: Ensures entire message integrity by detecting and correcting errors through retransmission.

## Session Layer Overview

The session layer manages and controls dialog between two systems by establishing, maintaining, and synchronizing communication.

### **Key Functions:**

- **Dialog Control**: Supports communication in half-duplex (one-way) or full-duplex (two-way) modes.
- **Synchronization**: Adds checkpoints (synchronization points) to data streams to ensure proper communication flow.

## Presentation Layer Overview

The presentation layer handles the syntax and semantics of the data exchanged between systems, ensuring that information is in a format understandable by both.

### **Key Functions:**

- **Translation**: Converts data from one system's format to a common format for transmission and then back to the destination's format for interpretation.
- **Encryption/Decryption**: Protects sensitive information by converting it to a different form before transmission and restoring it to its original form at the destination.
- **Compression**: Reduces the data size, which is especially important for multimedia transmissions (e.g., text, audio, video).

## Application Layer Overview

The application layer provides an interface for users (human or software) to access network services. It supports functions like email, file transfer, and distributed database management.

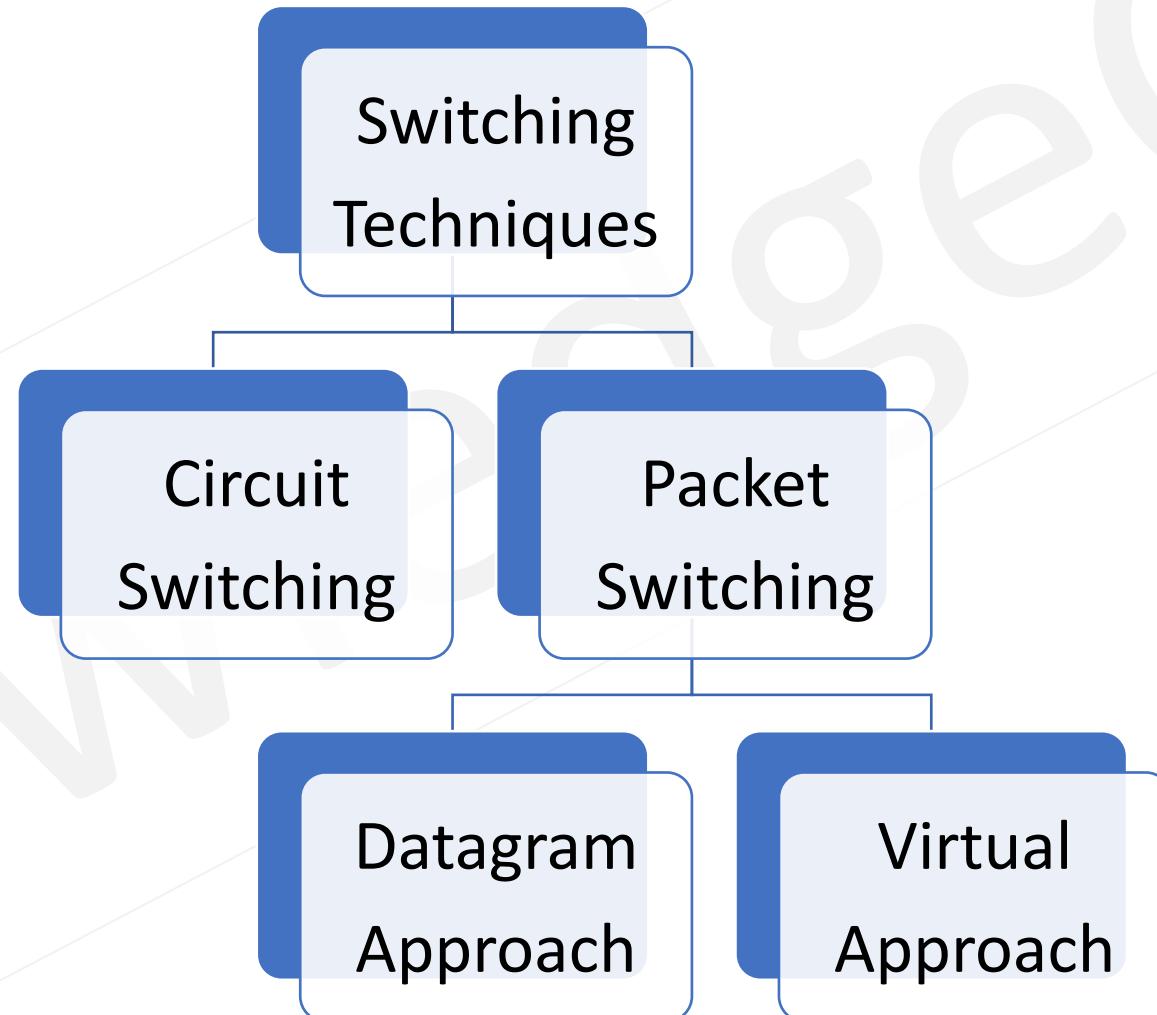
### **Key Services:**

- **Network Virtual Terminal**: Emulates a physical terminal to allow users to log on to remote hosts.
- **File Transfer, Access, and Management**: Enables users to access, modify, retrieve, and manage files on remote hosts.
- **Mail Services**: Provides the foundation for email storage and forwarding.
- **Directory Services**: Offers access to distributed databases containing global information about various objects and services.

## Connectionless and Connection-Oriented

- **Connectionless Protocol:** Frames are sent from one node to the next without any relationship between the frames; each frame is independent. Connectionless means that there is no connection between frames, it does not imply that there is no physical link between nodes.
- **Connection-Oriented Protocol:** A logical connection should first be established between the two nodes (setup phase). After all frames that are somehow related to each other are transmitted (transfer phase), the logical connection is terminated (teardown phase). The frames are numbered and sent in order. If they are not received in order, the receiver needs to wait until all frames belonging to the same set are received and then deliver them in order to the network layer.

- Switching is the technique by which nodes control or switch data to transmit it between specific points on a network.

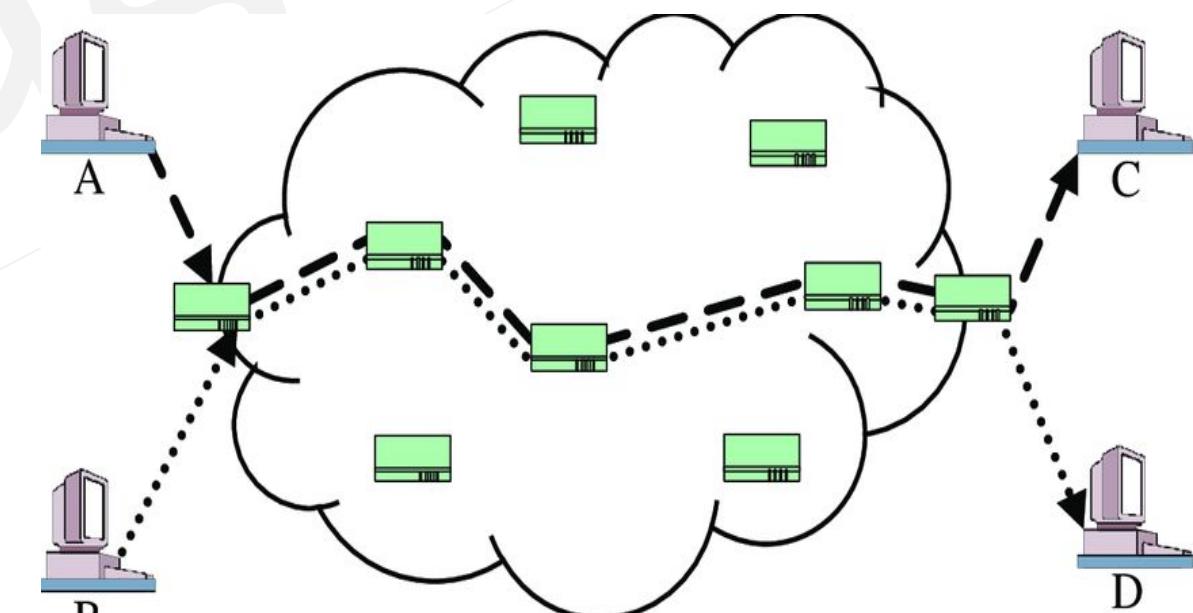
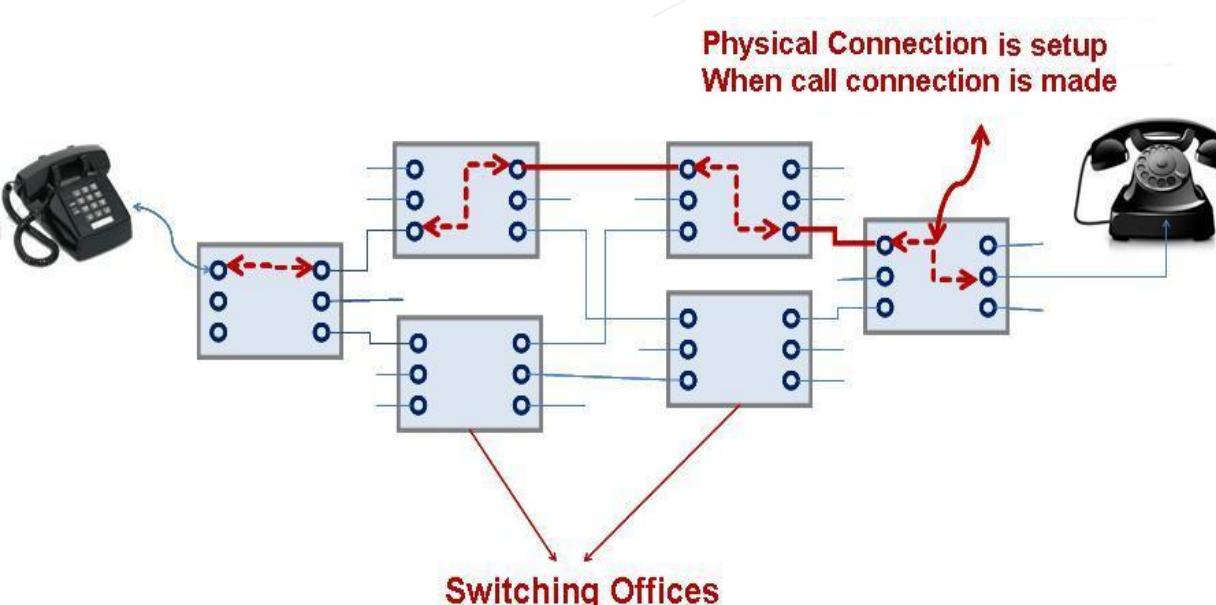


# Circuit Switching

- **Definition:** Circuit switching involves dividing network resources (like bandwidth) into segments and establishing a dedicated path between the sender and receiver. This dedicated path ensures a constant bit delay during the connection and guarantees a specific data rate.
- **Operation:** Once a circuit is established, data can flow without delays. Circuit switching is commonly used in traditional telephone networks.



- **Time Division Multiplexing (TDM):**
  - Divides communication into time slots or frames.
  - Transmits independent signals over a shared path using synchronized switches.
  - Used in digital circuit-switched systems, especially for long-distance links with heavy data loads.
- **Frequency Division Multiplexing (FDM):**
  - Divides bandwidth into non-overlapping frequency sub-bands.
  - Each sub-band carries a different signal.
  - Practical applications include radio spectrum and optical fiber communication.



- **Advantages of Circuit Switching**
  - **Guaranteed Data Rate:** A dedicated transmission path ensures a consistent rate.
  - **No Delays:** Data flow is uninterrupted once the path is established.
  - **No Header Required:** No need to add headers to each packet.
  - **Data Order:** Reordering of data is not needed due to the established path.
- **Disadvantages of Circuit Switching**
  - **Connection Setup Time:** Establishing a dedicated path can take a long time.
  - **Higher Bandwidth Requirement:** Dedicated channels consume more bandwidth.
  - **Lack of Flexibility:** Channels remain dedicated even if data is not being transmitted.
  - **Outdated Technology:** Circuit switching is less commonly used in modern networks.

## Packet Switching

- **Definition:** In a packet-switched network, messages are divided into small packets of variable or fixed sizes, determined by the network protocols.
- **Resource Allocation:** There's no reserved bandwidth or dedicated scheduling for each packet. Resources are allocated dynamically based on demand and processed on a **first-come, first-served basis**.
- **Processing & Delays:** If a switch is busy with other packets, incoming packets must wait, which may introduce delays. This lack of reservation means potential packet loss or drop if there's insufficient processing capacity.

- **Datagram Network:**
  - Treats each packet as independent, regardless of its source or destination.
  - Requires upper-layer protocols to handle tasks like **reordering packets** and **requesting lost packets** before passing data to applications.
- **Virtual Network:**
  - Acts as a combination of **circuit-switched** and **datagram networks**, providing a middle ground.
  - Commonly used in modern telephone networks for efficient data transmission.

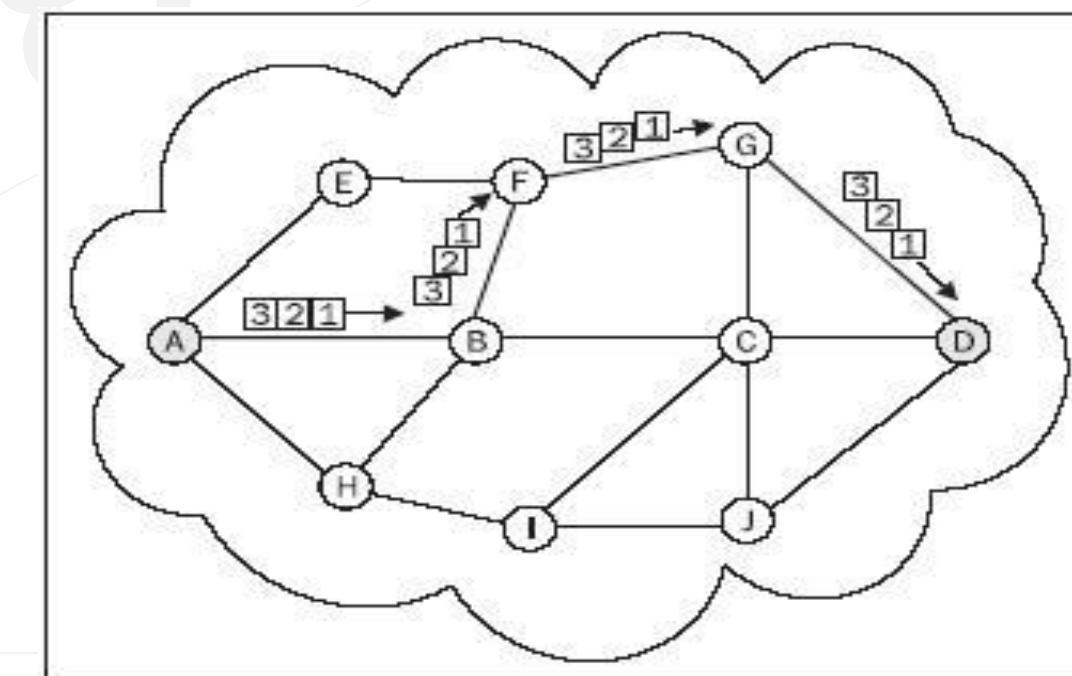
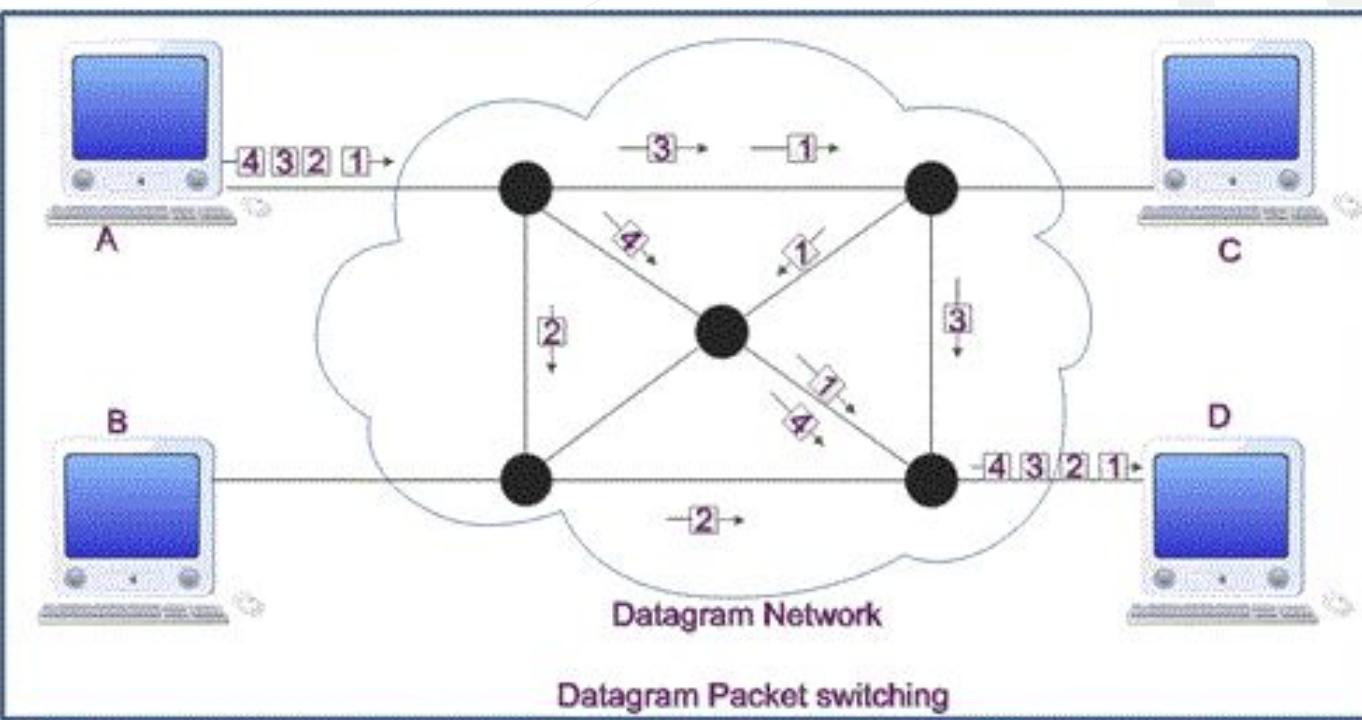
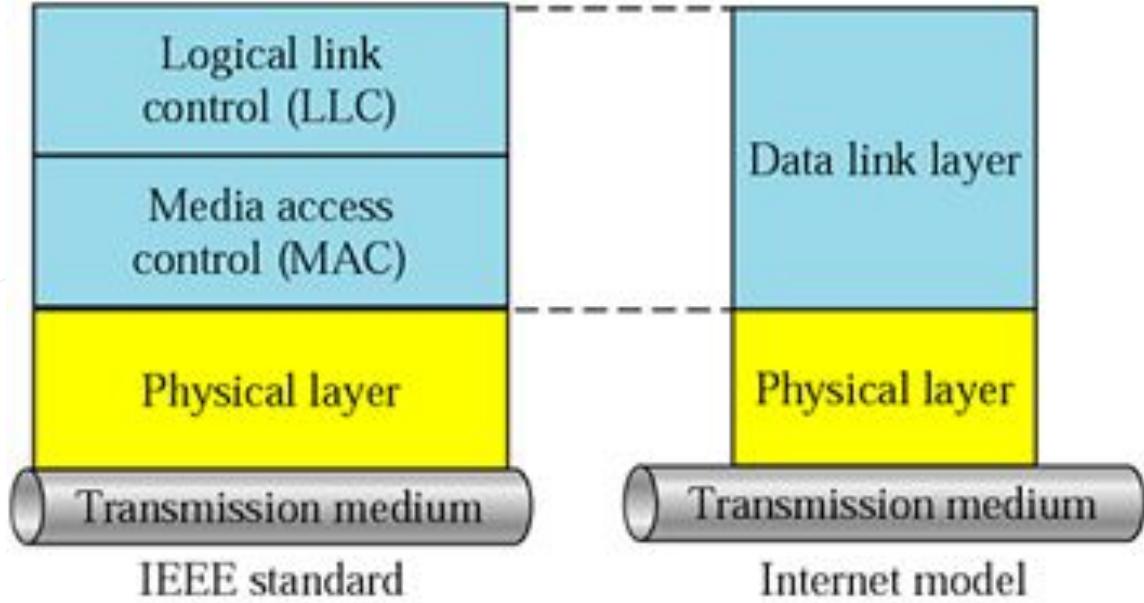


Figure 1.16: Virtual circuit

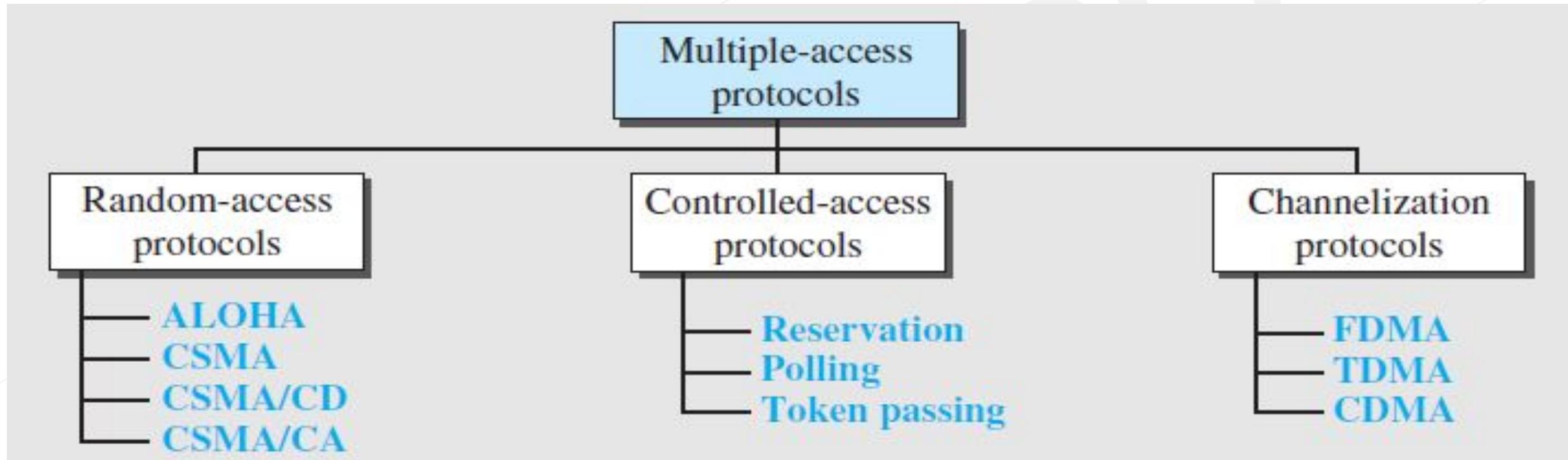
## Two Sublayers

1. The IEEE has subdivided the data-link layer into two sublayers: **logical link control (LLC)** (TOP) and **media access control (MAC)** (BOTTOM).
2. **Media Access Control (MAC):** It defines the specific access method for each LAN. For example, it defines CSMA/CD as the media access method for Ethernet LANs. Take care of Addressing at the level(Lan technology).
3. Flow control, error control, and part of the framing duties are collected into one sublayer called the *logical link control (LLC)*.
4. Framing is handled in both the LLC sublayer and the MAC sublayer.



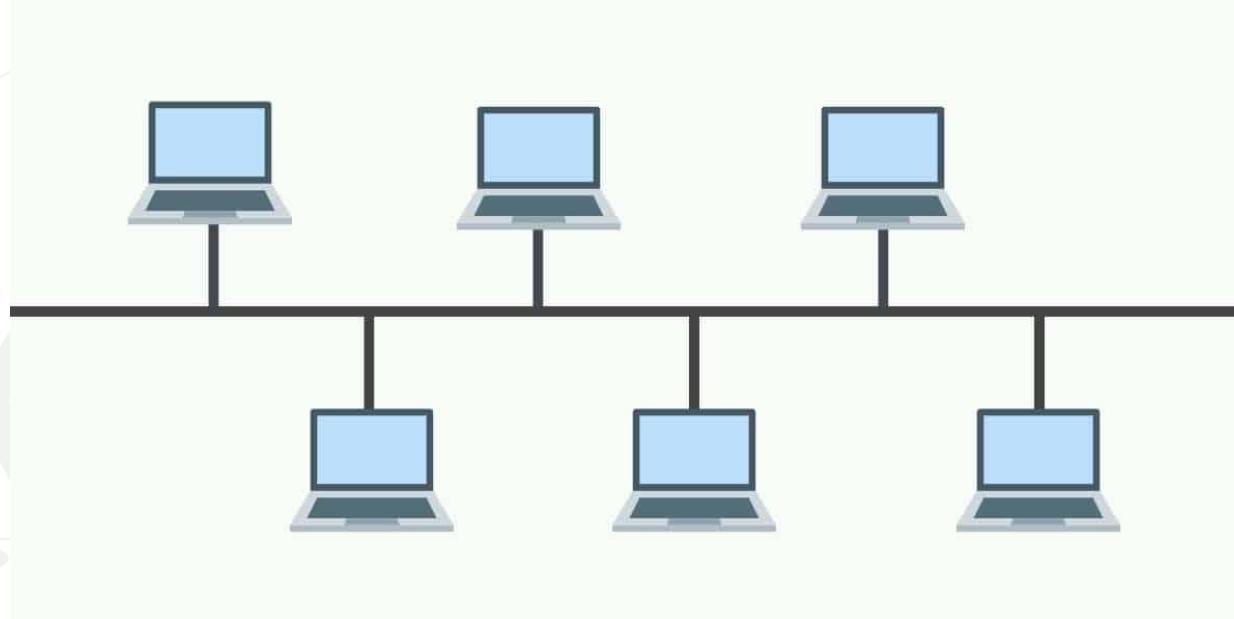
## Media Access Control

- When nodes or stations are connected and use a common link, called a *multipoint* or *broadcast link*, we need a multiple-access protocol to coordinate access to the link. Many protocols have been devised to handle access to a shared link. All of these protocols belong to a sublayer in the data-link layer called media access control (MAC).



# **RANDOM ACCESS**

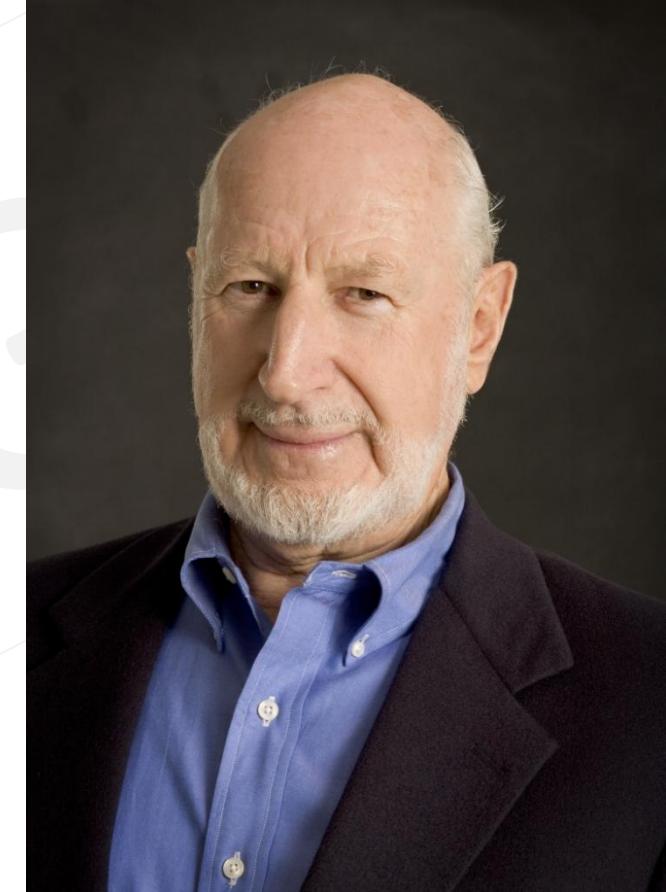
1. In random access methods, no station is superior to another station and none is assigned the control over another.
2. No station permits, or does not permit, another station to send.
3. Two features give this method its name.
  - First, there is no scheduled time for a station to transmit. Transmission is random among the stations. That is why these methods are called random access.
  - Second, no rules specify which station should send next. Stations compete with one another to access the medium. That is why these methods are also called contention methods.



- However, if more than one station tries to send, there is an access conflict-collision-and the frames will be either destroyed or modified.
- All the protocols in Random access approach will answer the following questions
  1. When can the station access the medium?
  2. What can the station do if the medium is busy?
  3. How can the station determine the success or failure of the transmission?
  4. What can the station do if there is an access conflict?

# Aloha

- Earliest random-access method, was developed at the University of Hawaii around 1970.
- It was designed for a radio (wireless) LAN, but it can be used on any shared medium.
- The original ALOHA protocol is called pure ALOHA. This is a simple, but elegant protocol.
- The idea is that each station sends a frame whenever it has a frame to send. However, there is the possibility of collision between frames from different stations.



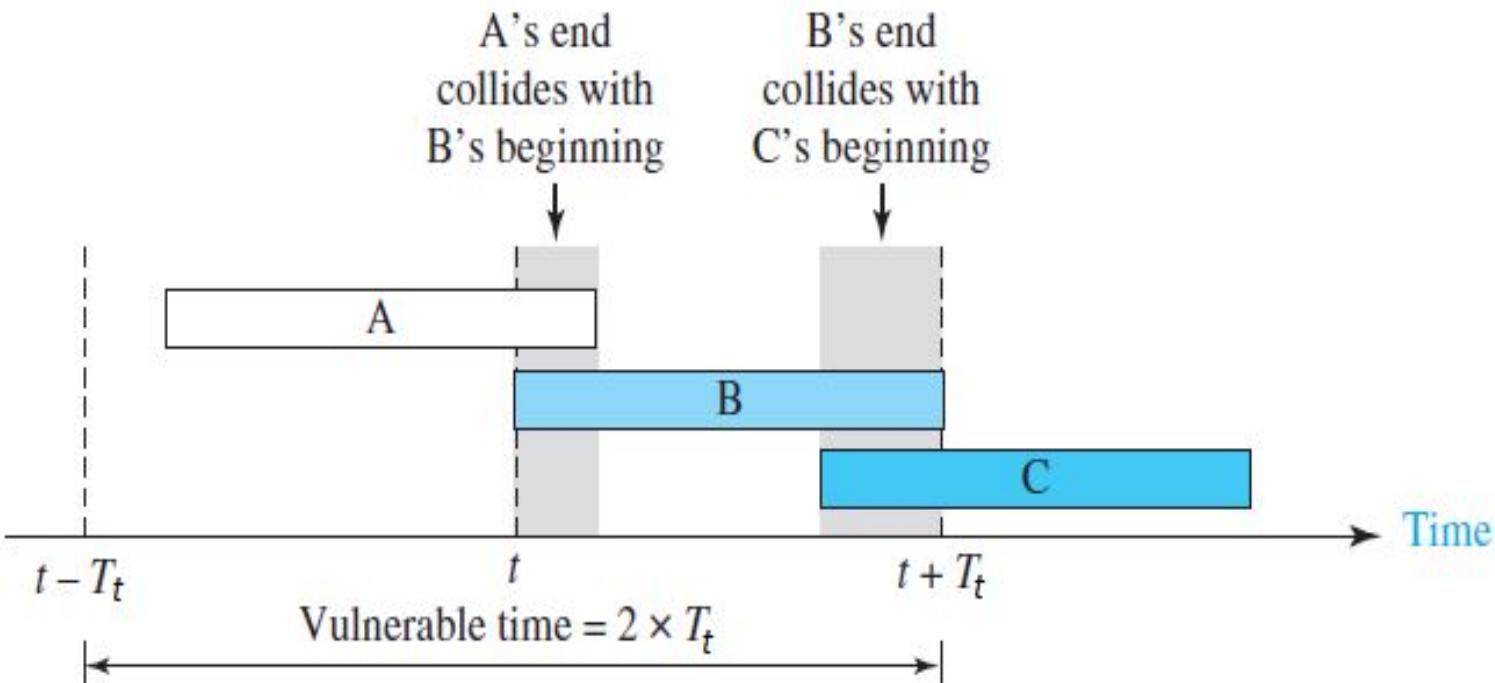
**Norman Manuel Abramson**

- **Propagation Delay:** Propagation delay is the time it takes for a bit to travel from point A to point B in the transmission media.

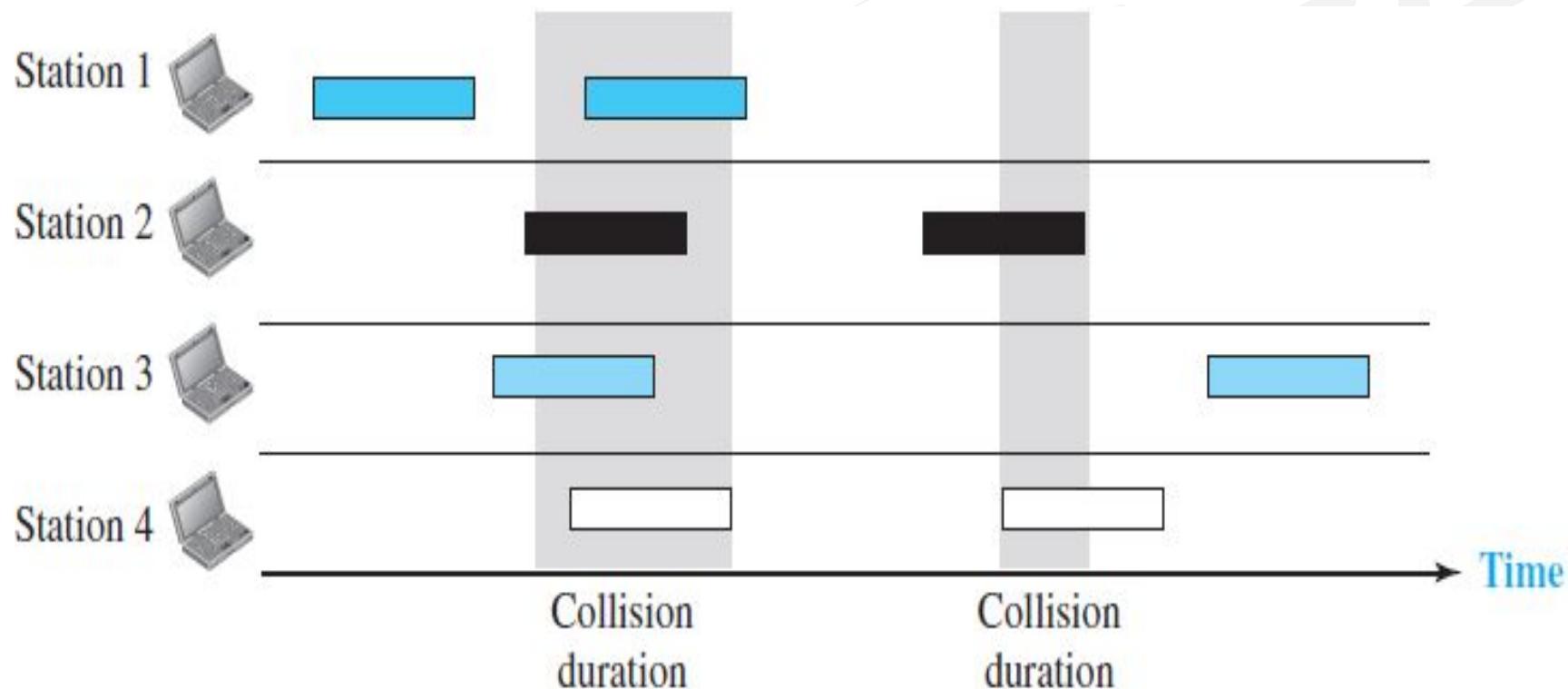
$$T_p = (\text{Distance}) / (\text{Propagation speed})$$

- **Transmission Delay (TT)** : A sender needs to put the bits in a packet on the line one by one. If the first bit of the packet is put on the line at time  $t_1$  and the last bit is put on the line at time  $t_2$ , transmission delay of the packet is  $(t_2 - t_1)$ .  
 $T_t = (\text{Packet length (L)}) / (\text{Transmission rate or Bandwidth (B)}) = L / B$

- Vulnerable time in which there is a possibility of collision. We assume that the stations send fixed-length frames with each frame taking  $T_{fr}$  S to send.
- Station A sends a frame at time t. Now imagine station B has already sent a frame between  $t - T_{fr}$  and t. This leads to a collision between the frames from station A and station B. The end of B's frame collides with the beginning of A's frame.
- On the other hand, suppose that station C sends a frame between t and  $t + T_{fr}$ . Here, there is a collision between frames from station A and station C. The beginning of C's frame collides with the end of A's frame. we see that the vulnerable time, during which a collision may occur in pure ALOHA, is 2 times the frame transmission time.
- Pure ALOHA vulnerable time=  $2 \times T_{fr}$



- The pure ALOHA protocol relies on acknowledgments from the receiver. When a station sends a frame, it expects the receiver to send an acknowledgment.
- If the acknowledgment does not arrive in time-out period, the station assumes that the frame (or the acknowledgment) has been destroyed and resends the frame.



# Procedure for Pure ALOHA protocol

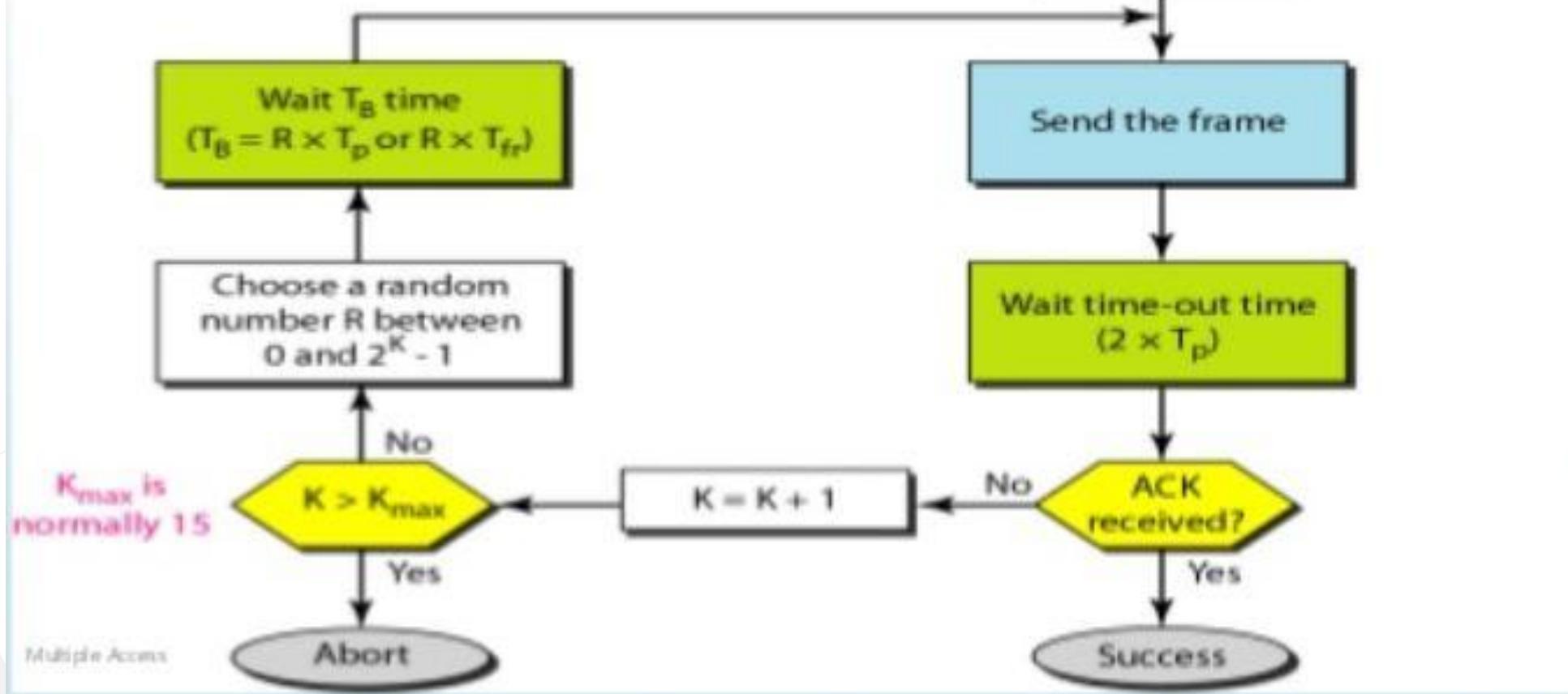
K: Number of attempts

$T_p$ : Maximum propagation time

$T_{fr}$ : Average transmission time for a frame

$T_B$ : Back-off time

Station has  
a frame to send



Multiple Access

- If all these stations try to resend their frames after the time-out, the frames will collide again.
- Pure ALOHA dictates that when the time-out period passes, each station waits a random amount of time before resending its frame. The randomness will help avoid more collisions. We call this time the back-off time  $T_B$ .
- Pure ALOHA has a second method to prevent congesting the channel with retransmitted frames. After a maximum number of retransmissions attempts  $K_{\max}$  a station must give up and try later.

**Example:** The stations on a wireless ALOHA network are a maximum of 600 km apart. If we assume that signals propagate at  $3 \times 10^8$  m/s . Find back off time possibility after two consecutive collision ?

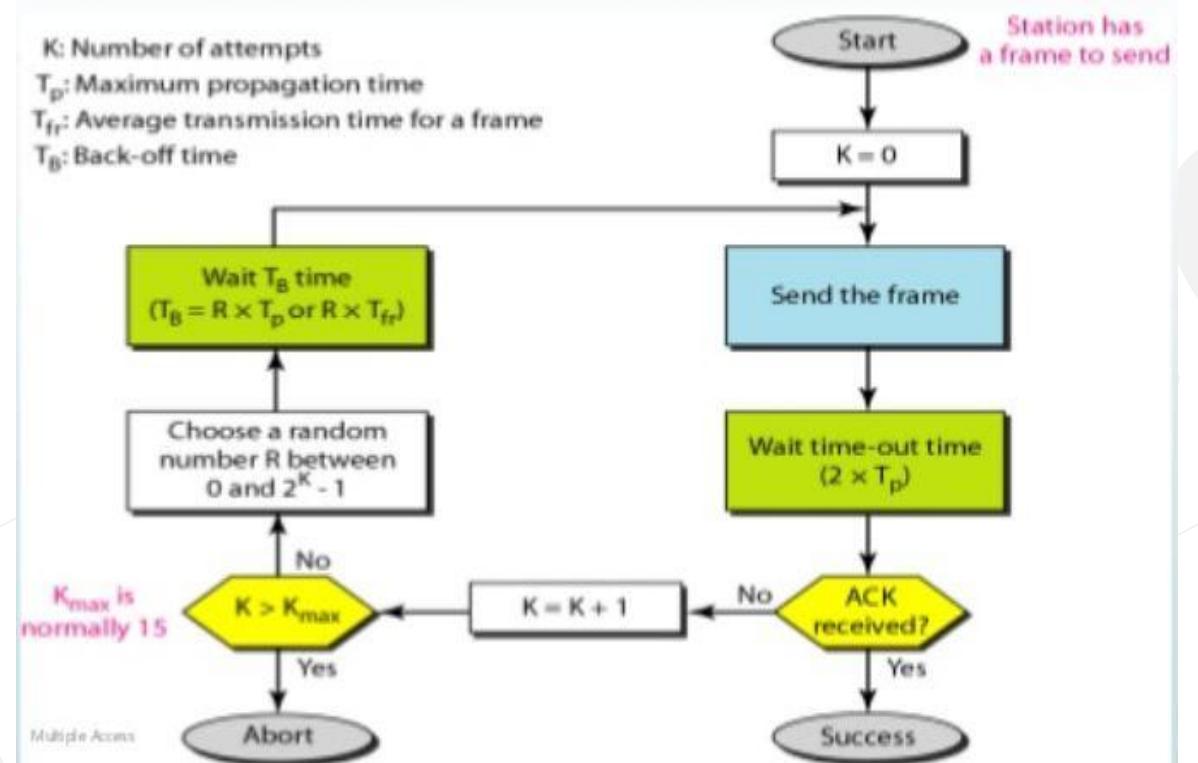
## Procedure for Pure ALOHA protocol

K: Number of attempts

$T_p$ : Maximum propagation time

$T_{fr}$ : Average transmission time for a frame

$T_B$ : Back-off time



**Example:** A pure ALOHA network transmits 200-bit frames on a shared channel of 200 kbps. What is the requirement to make this frame collision-free?

## Example 12.2

A pure ALOHA network transmits 200-bit frames on a shared channel of 200 kbps. What is the requirement to make this frame collision-free?

### Solution

Average frame transmission time  $T_{fr}$  is 200 bits/200 kbps or 1 ms. The vulnerable time is  $2 \times 1 \text{ ms} = 2 \text{ ms}$ . This means no station should send later than 1 ms before this station starts transmission and no station should start sending during the period (1 ms) that this station is sending.

### Throughput

Let us call  $G$  the average number of frames generated by the system during one frame transmission time. Then it can be proven that the average number of successfully transmitted frames for pure ALOHA is  $S = G \times e^{-2G}$ . The maximum throughput  $S_{max}$  is 0.184, for  $G = 1/2$ . (We can find it by setting the derivative of  $S$  with respect to  $G$  to 0; see Exercises.) In other words, if one-half a frame is generated during one frame transmission time (one frame during two frame transmission times), then 18.4 percent of these frames reach their destination successfully. We expect  $G = 1/2$  to produce the maximum throughput because the vulnerable time is 2 times the frame transmission time. Therefore, if a station generates only one frame in this vulnerable time (and no other stations generate a frame during this time), the frame will reach its destination successfully.

The throughput for pure ALOHA is  $S = G \times e^{-2G}$ .

The maximum throughput  $S_{max} = 1/(2e) = 0.184$  when  $G = (1/2)$ .

### Example 12.3

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?

### Solution

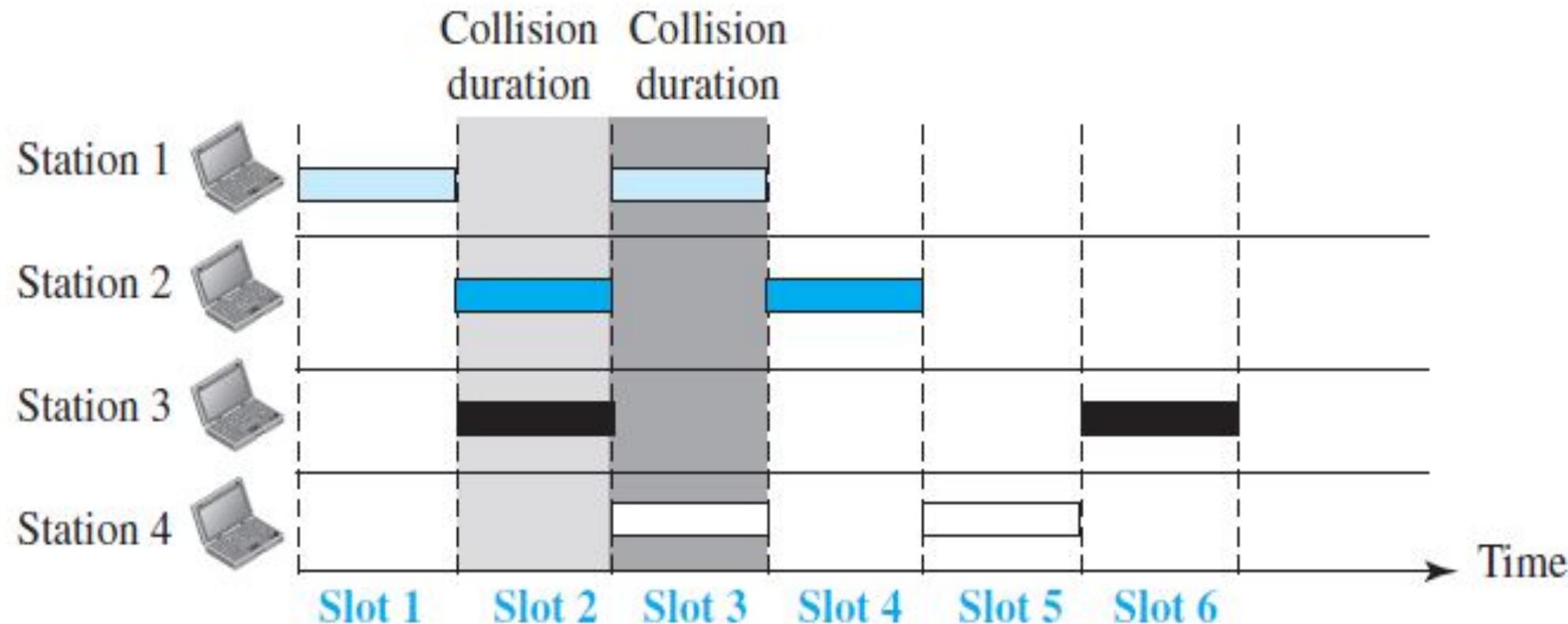
The frame transmission time is  $200/200$  kbps or 1 ms.

- a. If the system creates 1000 frames per second, or 1 frame per millisecond, then  $G = 1$ . In this case  $S = G \times e^{-2G} = 0.135$  (13.5 percent). This means that the throughput is  $1000 \times 0.135 = 135$  frames. Only 135 frames out of 1000 will probably survive.
- b. If the system creates 500 frames per second, or  $1/2$  frames per millisecond, then  $G = 1/2$ . In this case  $S = G \times e^{-2G} = 0.184$  (18.4 percent). This means that the throughput is  $500 \times 0.184 = 92$  and that only 92 frames out of 500 will probably survive. Note that this is the *maximum throughput* case, percentagewise.
- c. If the system creates 250 frames per second, or  $1/4$  frames per millisecond, then  $G = 1/4$ . In this case  $S = G \times e^{-2G} = 0.152$  (15.2 percent). This means that the throughput is  $250 \times 0.152 = 38$ . Only 38 frames out of 250 will probably survive.

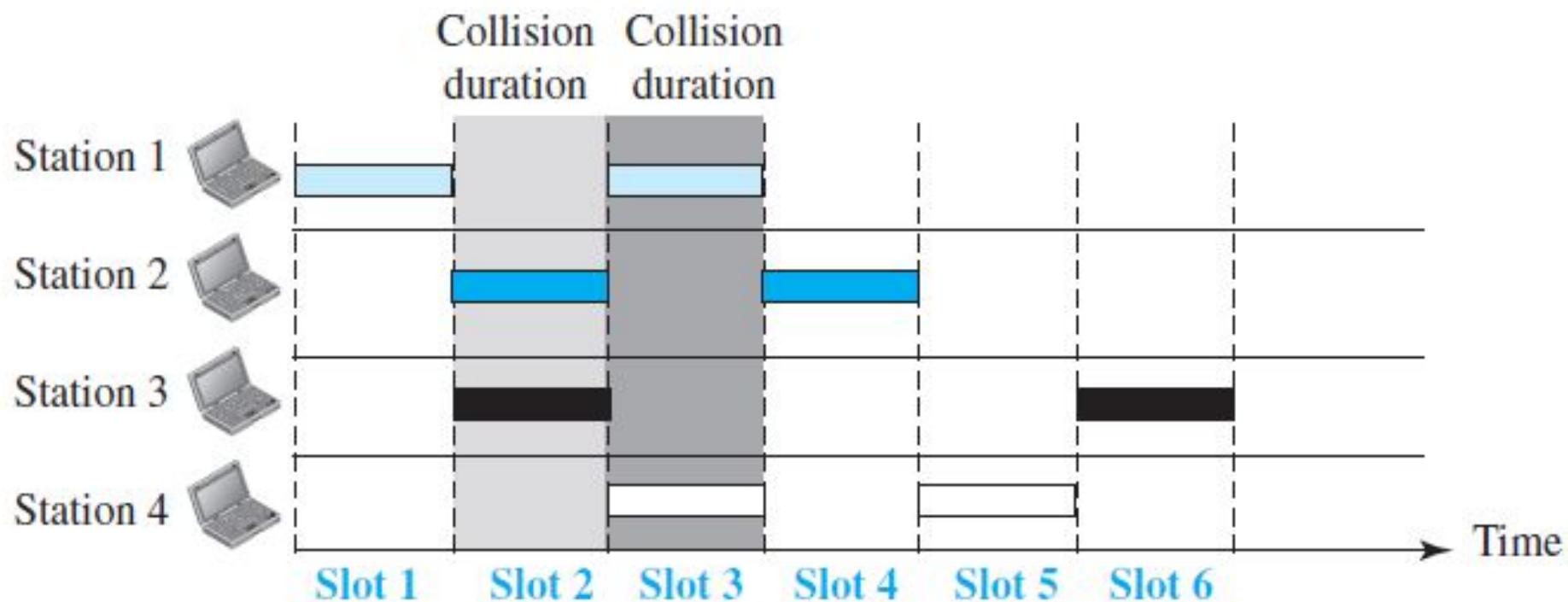
**Q** Consider a network using the pure ALOHA medium access control protocol, where each frame is of length 1,000 bits. The channel transmission rate is 1 Mbps (=106 bits per second). The aggregate number of transmissions across all the nodes (including new frame transmissions and retransmitted frames due to collisions) is modelled as a Poisson process with a rate of 1,000 frames per second. Throughput is defined as the average number of frames successfully transmitted per second. The throughput of the network (rounded to the nearest integer) is \_\_\_\_\_ . **(GATE 2021) (2 MARKS)**

## Slotted ALOHA

- Pure ALOHA has a vulnerable time of  $2 \times T_{fr}$ . This is so because there is no rule that defines when the station can send. A station may send soon after another station has started or soon before another station has finished.
- Slotted ALOHA was invented to improve the efficiency of pure ALOHA. In slotted ALOHA we divide the time into slots of  $T_{fr}$  s and force the station to send only at the beginning of the time slot.



- Because a station is allowed to send only at the beginning of the synchronized time slot, if a station misses this moment, it must wait until the beginning of the next time slot. This means that the station which started at the beginning of this slot has already finished sending its frame.
- Of course, there is still the possibility of collision if two stations try to send at the beginning of the same time slot. However, the vulnerable time is now reduced to one-half, equal to  $T_{fr}$



**Q** Consider a LAN with four nodes  $S_1$ ,  $S_2$ ,  $S_3$  and  $S_4$ . Time is divided into fixed-size slots, and a node can begin its transmission only at the beginning of a slot. A collision is said to have occurred if more than one node transmits in the same slot. The probabilities of generation of a frame in a time slot by  $S_1$ ,  $S_2$ ,  $S_3$  and  $S_4$  are 0.1, 0.2, 0.3 and 0.4, respectively. The probability of sending a frame in the first slot without any collision by any of these four stations is \_\_\_\_\_. (Gate-2015) (2 Marks)

- (A) 0.462    (B) 0.711    (C) 0.5    (D) 0.652

**Q** There are  $n$  stations in a slotted LAN. Each station attempts to transmit with a probability  $p$  in each time slot. What is the probability that ONLY one station transmits in a given time slot?  
**(Gate-2007) (2 Marks)**

- a)  $np(1-p)^{n-1}$
- b)  $(1-p)^{n-1}$
- c)  $p(1-p)^{n-1}$
- d)  $1-(1-p)^{n-1}$

**Q** A and B are the only two stations on an Ethernet. Each has a steady queue of frames to send. Both A and B attempt to transmit a frame, collide, and A wins the first backoff race. At the end of this successful transmission by A, both A and B attempt to transmit and collide. The probability that A wins the second backoff race is: **(Gate-2004) (2 Marks)**

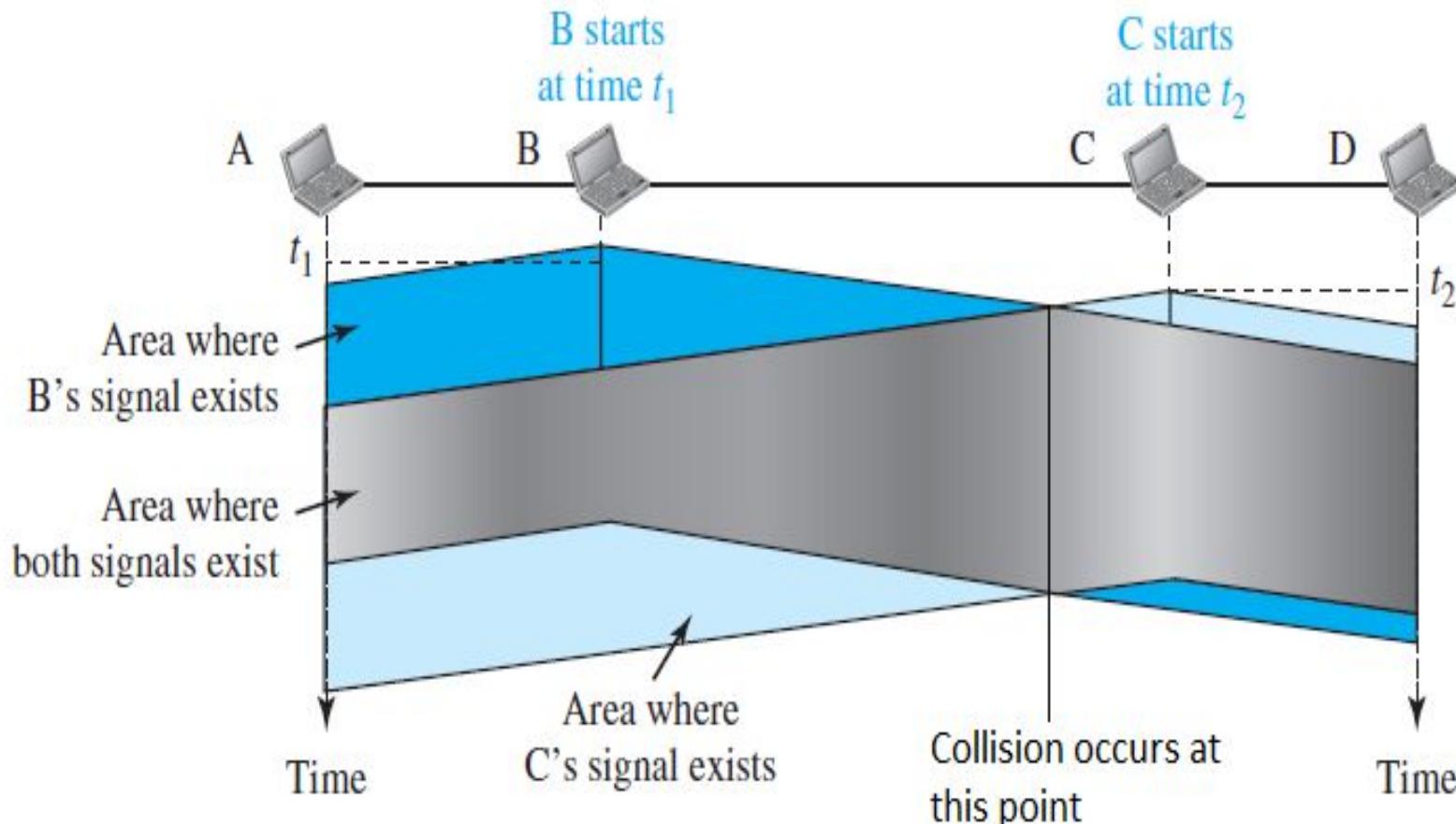
- (A) 0.5      (B) 0.625      (C) 0.75      (D) 1.0**

## Carrier Sense Multiple Access (CSMA)

- To minimize the chance of collision and, therefore, increase the performance, the CSMA method was developed. The chance of collision can be reduced if a station senses the medium before trying to use it.
- Carrier sense multiple access (CSMA) requires that each station first listen to the medium (or check the state of the medium) before sending, so "sense before transmit" or "listen before talk." CSMA can reduce the possibility of collision, but it cannot eliminate it.
- The possibility of collision still exists because of propagation delay; when a station sends a frame, it still takes time (although very short) for the first bit to reach every station and for every station to sense it.

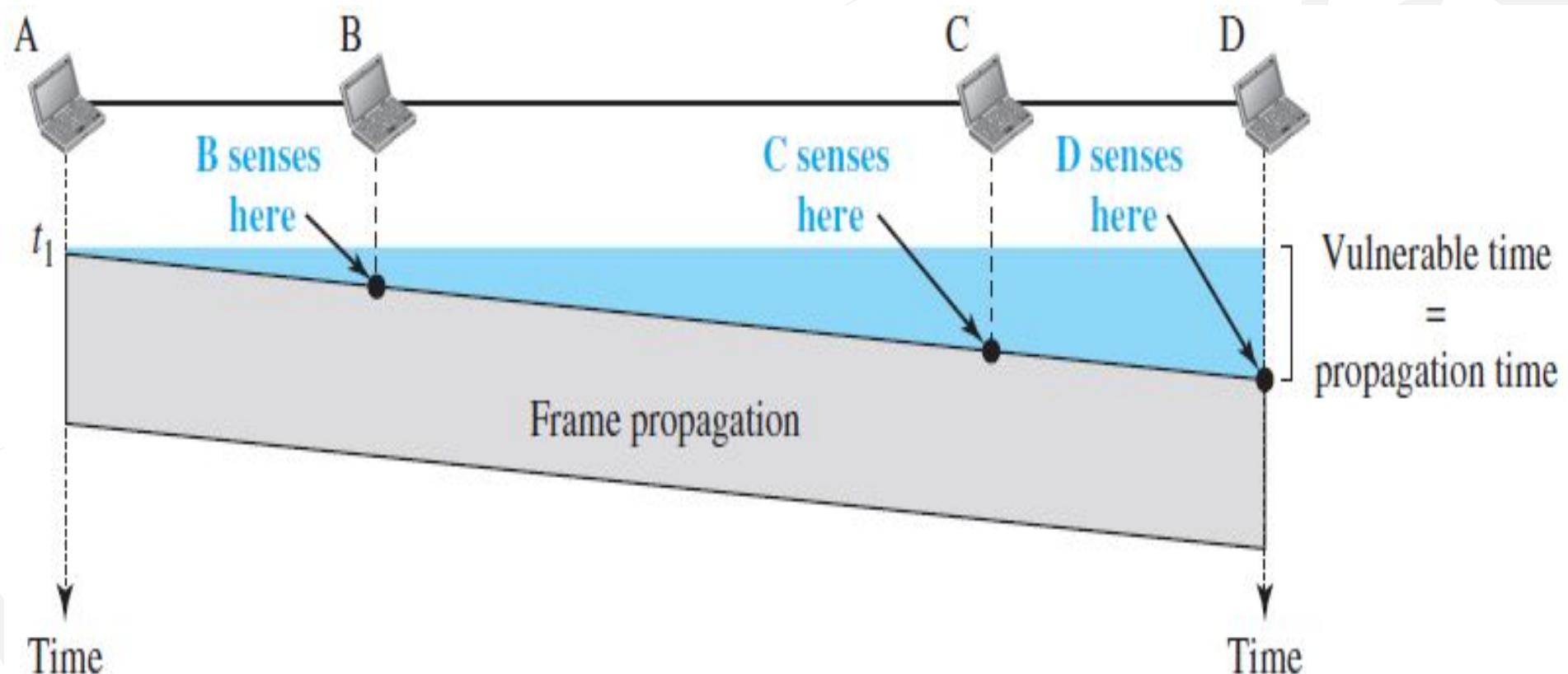


- In other words, a station may sense the medium and find it idle, only because the first bit sent by another station has not yet been received.
- At time  $t_1$ , station B senses the medium and finds it idle, so it sends a frame. At time  $t_2$  ( $t_2 > t_1$ ) station C senses the medium and finds it idle because, at this time, the first bits from station B have not reached station C. Station C also sends a frame. The two signals collide and both frames are destroyed.



## Vulnerable Time

- The vulnerable time for CSMA is the ***propagation time***  $T_p$ .
- When a station sends a frame and any other station tries to send a frame during this time, a collision will result.
- But if the first bit of the frame reaches the end of the medium, every station will already have heard the bit and will refrain from sending.
- Station A has sent a frame at time  $t_1$ , which reaches the rightmost station, D, at time  $t_1 + T_p$ .

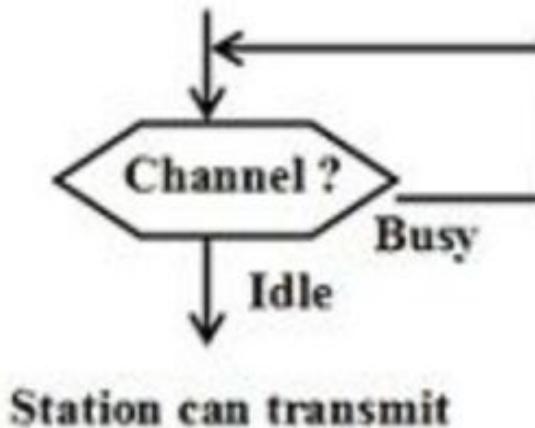
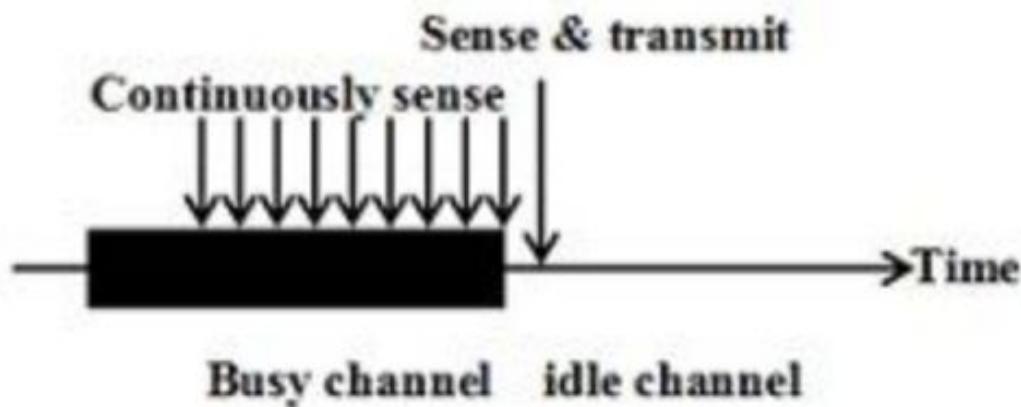


## Persistence Methods

- What should a station do if the channel is busy? What should a station do if the channel is idle?
- Three methods have been devised to answer these questions:
  - 1-persistent method
  - Non-persistent method
  - P-persistent method.

- **1-Persistent**

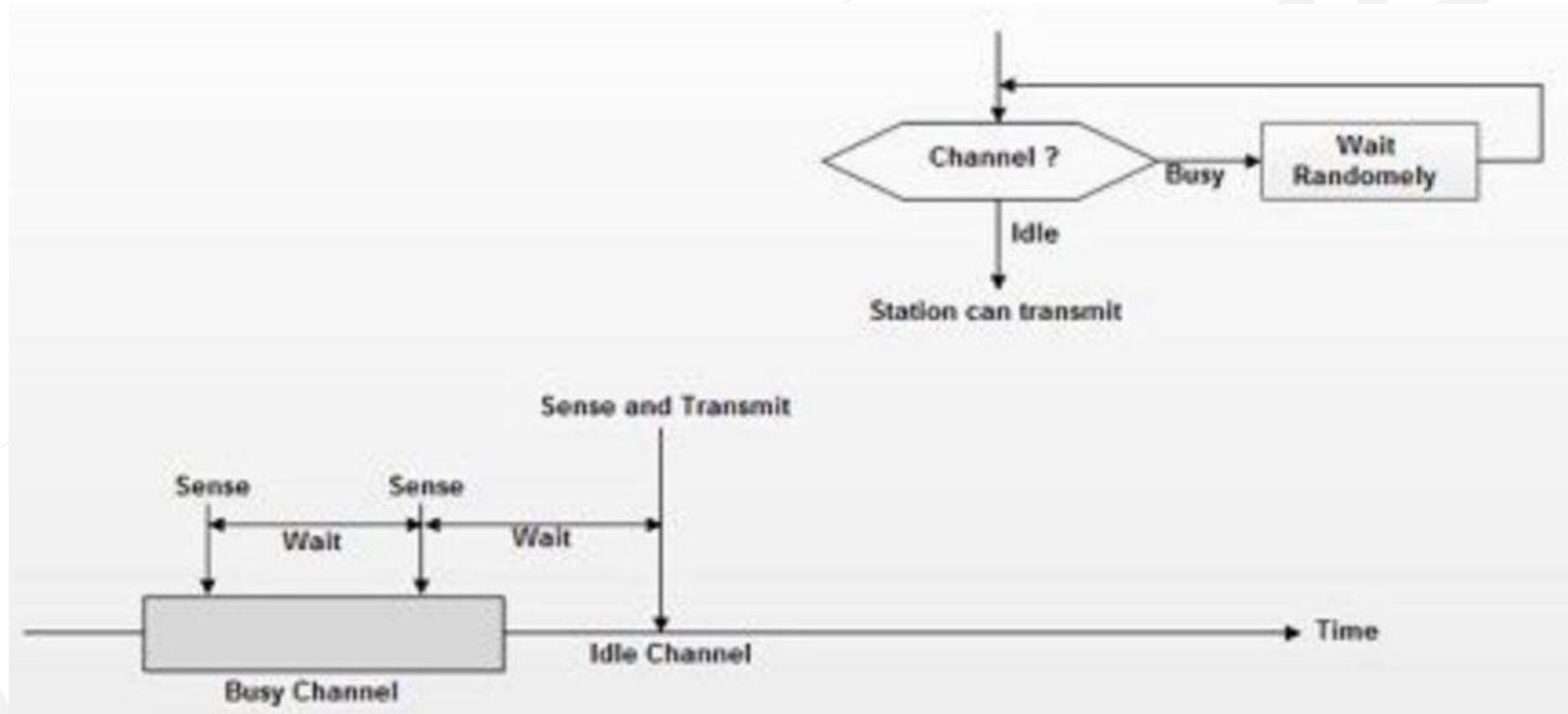
- The 1-persistent method is simple and straightforward. In this method, after the station finds the line idle, it sends its frame immediately (with probability 1). This method has the highest chance of collision because two or more stations may find the line idle and send their frames immediately.



1-persistent CSMA

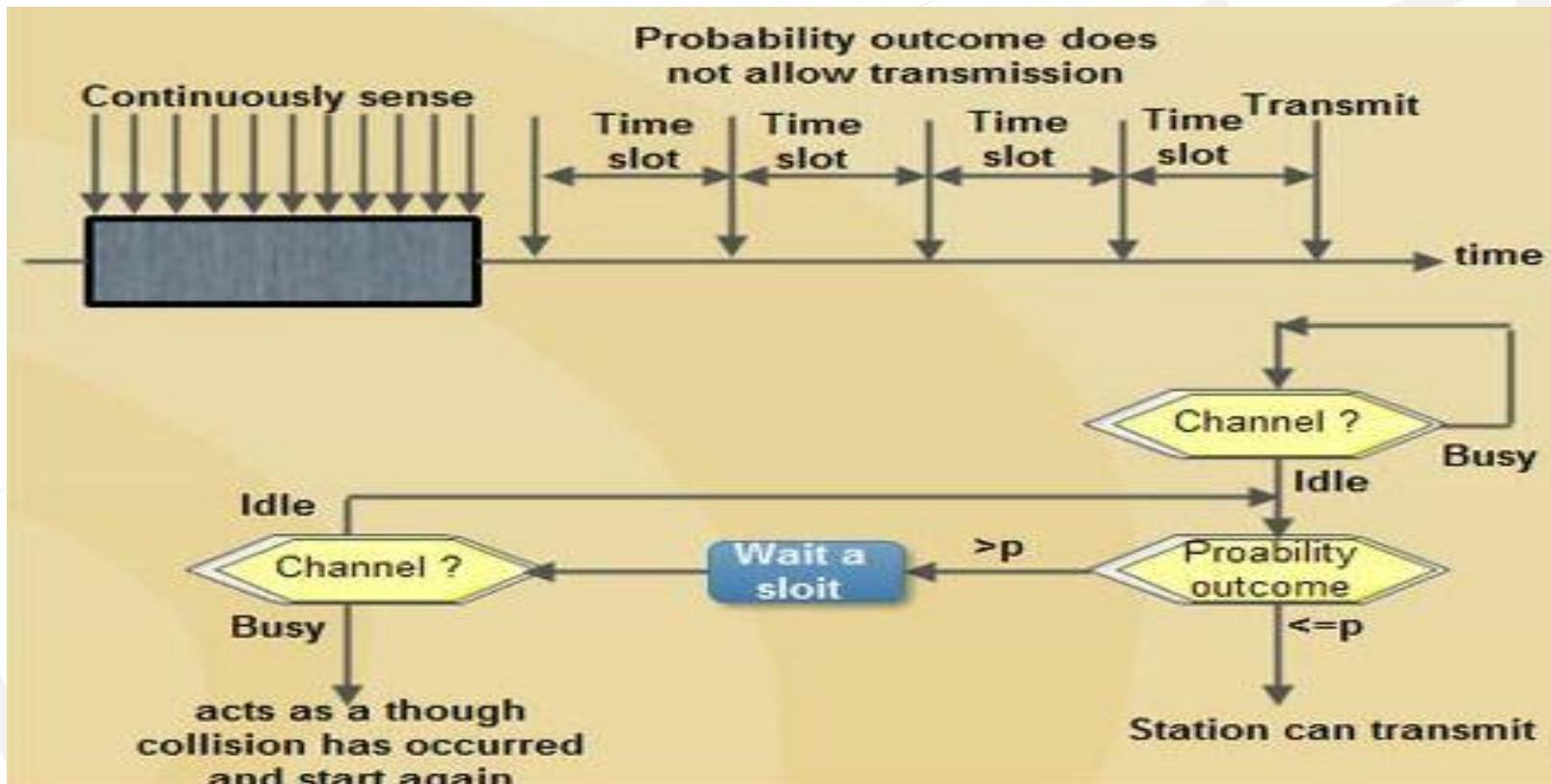
## • Nonpersistent

- In the nonpersistent method, a station that has a frame to send senses the line. If the line is idle, it sends immediately. If the line is not idle, it waits a random amount of time and then senses the line again.
- The nonpersistent approach reduces the chance of collision because it is unlikely that two or more stations will wait the same amount of time and retry to send simultaneously. However, this method reduces the efficiency of the network because the medium remains idle when there may be stations with frames to send.



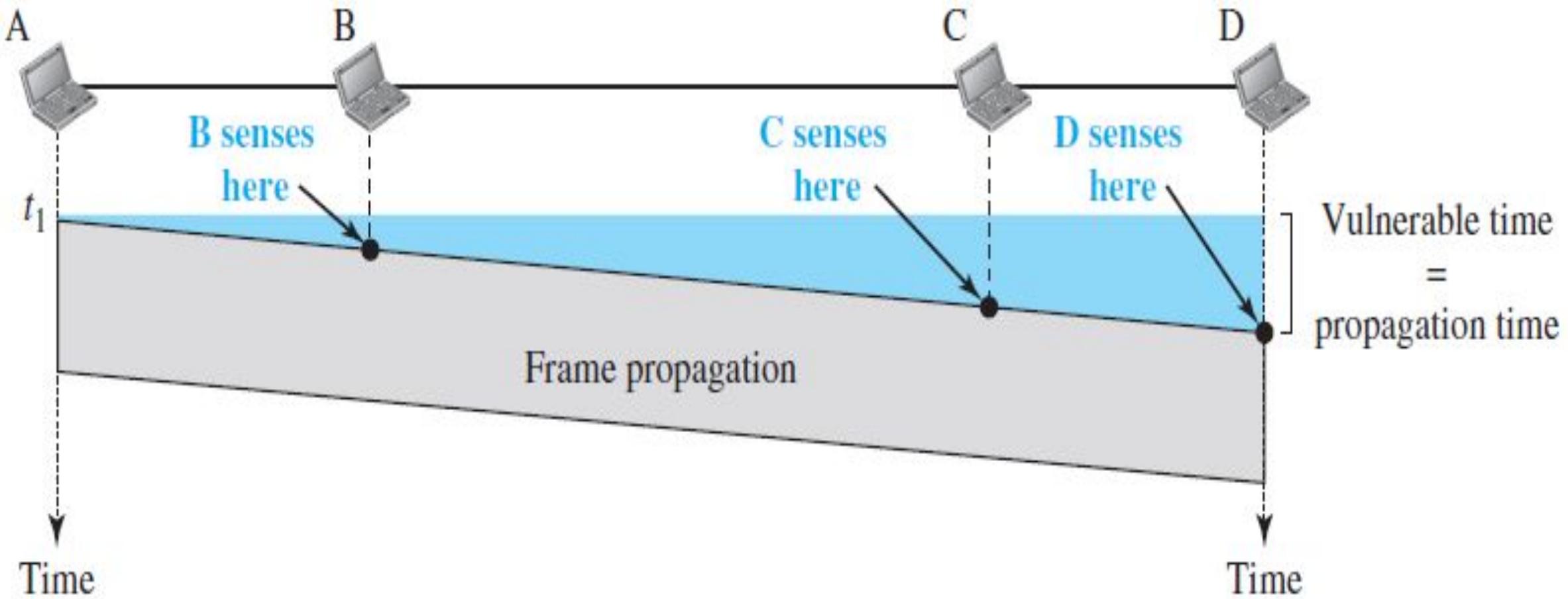
## • P-Persistent

- The p-persistent approach combines the advantages of the other two strategies. It reduces the chance of collision and improves efficiency. In this method, after the station finds the line idle it follows these steps:
- With probability  $p$ , the station sends its frame.
- With probability  $q = 1 - p$ , the station waits for the beginning of the next time slot and checks the line again.
  - a. If the line is idle, it goes to step 1.
  - b. If the line is busy, it acts as though a collision has occurred and uses the backoff procedure.



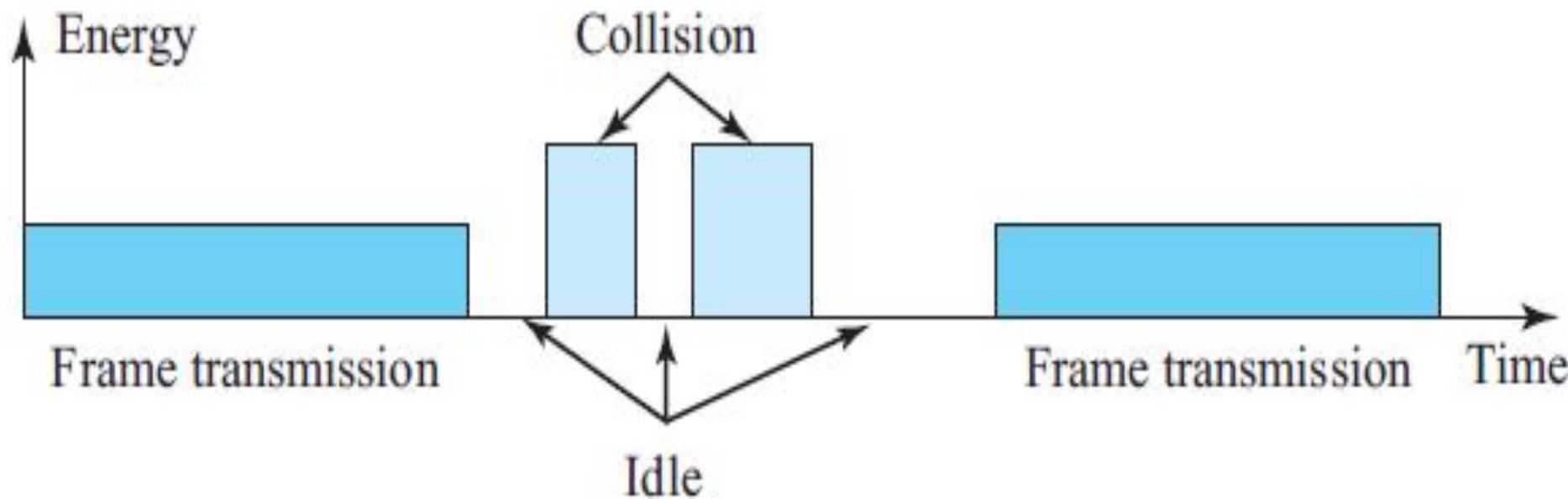
## Carrier Sense Multiple Access with Collision Detection (CSMA/CD)

- In this method, a station monitors the medium after it sends a frame to see if the transmission was successful. If so, the station is finished. If, however, there is a collision, the frame is sent again.
- Minimum Frame Size - For CSMA / CD to work, we need a restriction on the minimum frame size. Before sending the last bit of the frame, the sending station must detect a collision, if any, and abort the transmission.
- This is so because the station, once the entire frame is sent, does not monitor the line for collision detection. Therefore, the frame transmission time  $T_{fr}$  must be at least two times the maximum propagation time  $T_p$ .
- To understand the reason, let us think about the worst-case scenario. If the two stations involved in a collision are the maximum distance apart, the signal from the first takes time  $T_p$  to reach the second, and the effect of the collision takes another time  $T_p$  to reach the first. So the requirement is that the first station must still be transmitting after  $2T_p$ .



## • Energy Level

- We can say that the level of energy in a channel can have three values: zero, normal, and abnormal. At the zero level, the channel is idle. At the normal level, a station has successfully captured the channel and is sending its frame.
- At the abnormal level, there is a collision and the level of the energy is twice the normal level. A station that has a frame to send or is sending a frame needs to monitor the energy level to determine if the channel is idle, busy, or in collision mode.

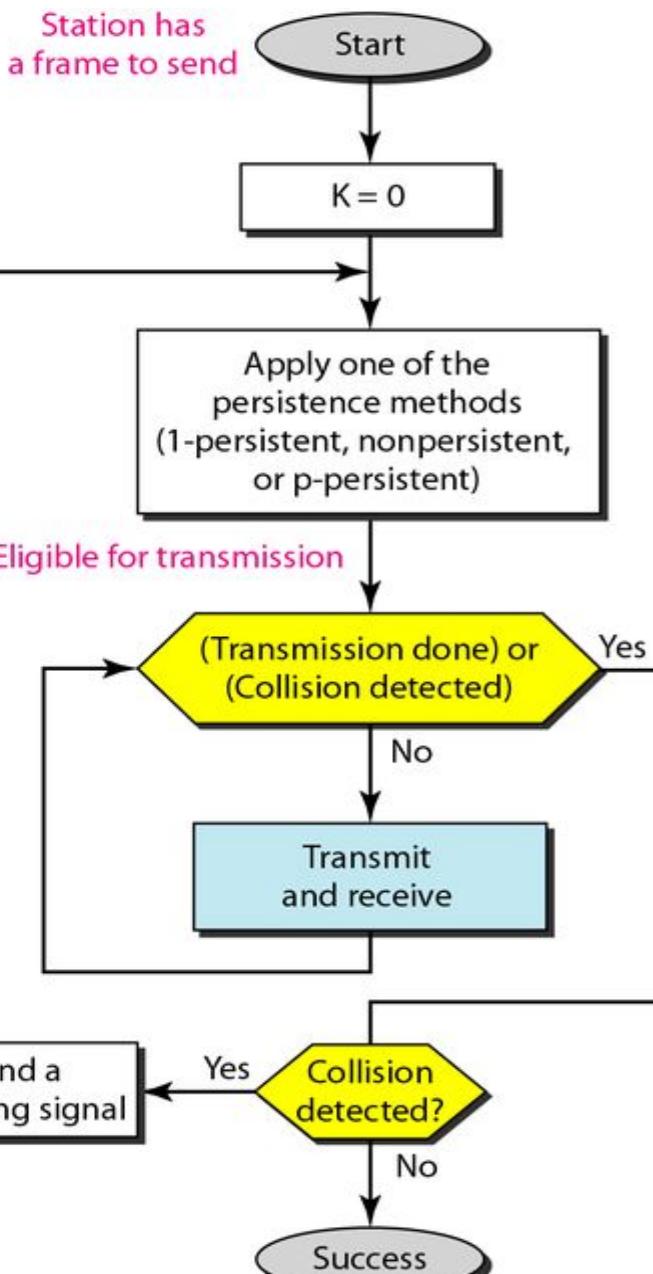


K: Number of attempts

$T_p$ : Maximum propagation time

$T_{fr}$ : Average transmission time for a frame

$T_B$ : Back-off time



$K_{max}$  is normally 15

**Q** A network has a data transmission bandwidth of  $20 \times 10^6$  bits per second. It uses CSMA/CD in the MAC layer. The maximum signal propagation time from one node to another node is 40 microseconds. The minimum size of a frame in the network is \_\_\_\_\_ bytes. **(Gate-2016) (2 Marks)**

**Q** Consider a CSMA/CD network that transmits data at a rate of 100 Mbps ( $10^8$  bits per second) over a 1 km (kilometre) cable with no repeaters. If the minimum frame size required for this network is 1250 bytes, what is the signal speed (km/sec) in the cable? **(Gate-2015) (1 Marks)**

- (A) 8000      (B) 10000      (C) 16000      (D) 20000

**Q** A network with CSMA/CD protocol in the MAC layer is running at 1 Gbps over a 1 km cable with no repeaters. The signal speed in the cable is  $2 \times 10^8$  m/sec. The minimum frame size for this network should be **(Gate-2005) (2 Marks)**

- (A) 10000 bits    (B) 10000 bytes    (C) 5000 bits    (D) 5000 bytes

**Q** A 2 km long broadcast LAN has  $10^7$  bps bandwidth and uses CSMA/CD. The signal travels along the wire at  $2 \times 10^8$  m/s. What is the minimum packet size that can be used on this network?

**(Gate-2003) (2 Marks)**

- (A) 50 bytes
- (B) 100 bytes
- (C) 200 bytes
- (D) None of these

**Q** The minimum frame size required for a CSMA/CD based computer network running at 1 Gbps on a 200m cable with a link speed of  $2 \times 10^8$ m/s is **(Gate-2008) (2 Marks)**

- (A) 125 bytes    (B) 250 bytes    (C) 500 bytes    (D) None of these

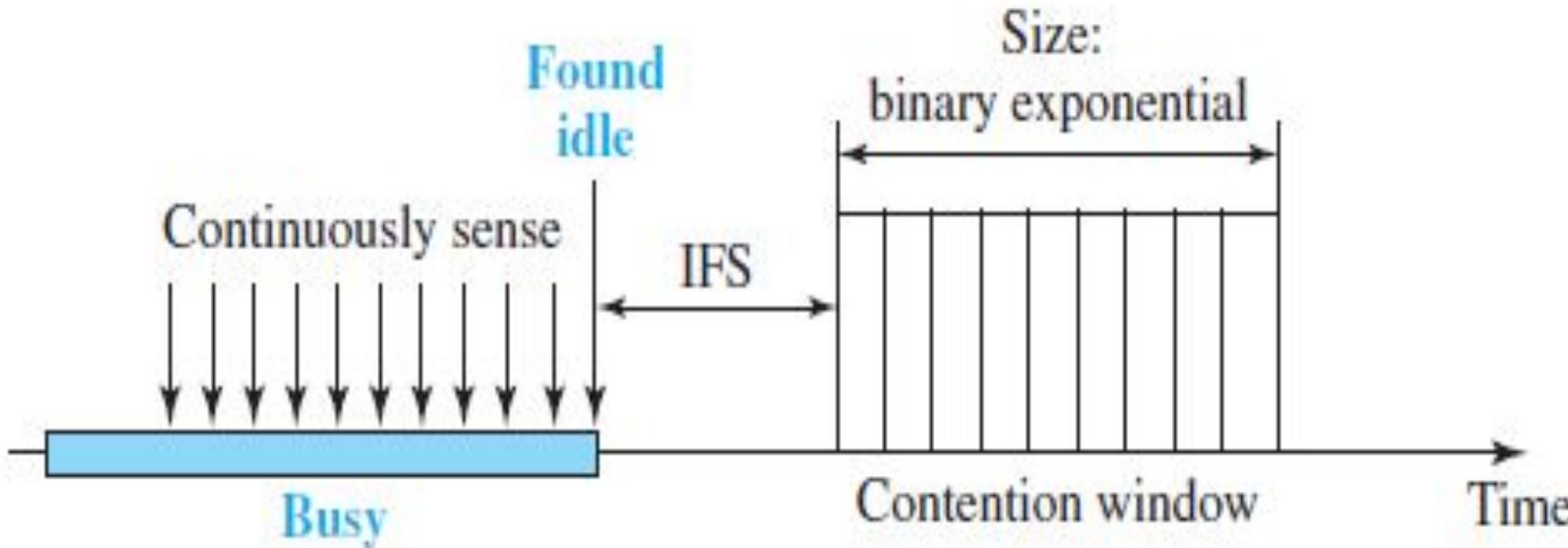
**Q** Consider a simple communication system where multiple nodes are connected by a shared broadcast medium (like Ethernet or wireless). The nodes in the system use the following carrier-sense based medium access protocol. A node that receives a packet to transmit will carrier-sense the medium for 5 units of time. If the node does not detect any other transmission in this duration, it starts transmitting its packet in the next time unit. If the node detects another transmission, it waits until this other transmission finishes, and then begins to carrier-sense for 5 time units again. Once they start to transmit, nodes do not perform any collision detection and continue transmission even if a collision occurs. All transmissions last for 20 units of time. Assume that the transmission signal travels at the speed of 10 meters per unit time in the medium. Assume that the system has two nodes P and Q, located at a distance  $d$  meters from each other. P starts transmitting a packet at time  $t=0$  after successfully completing its carrier-sense phase. Node Q has a packet to transmit at time  $t=0$  and begins to carrier-sense the medium. The maximum distance  $d$  (in meters, rounded to the closest integer) that allows Q to successfully avoid a collision between its proposed transmission and P's ongoing transmission is \_\_\_\_\_. (Gate-2018) (2 Marks)

**Q.** Consider an Ethernet segment with a transmission speed of  $10^8$  bits/sec and a maximum segment length of 500 meters. If the speed of propagation of the signal in the medium is  $2 \times 10^8$  meters/sec, then the minimum frame size (in bits) required for collision detection is \_\_\_\_\_ **(Gate 2024 CS) (2 Marks) (NAT)**

**Q** Consider a 100 Mbps link between an earth station (sender) and a satellite (receiver) at an altitude of 2100 km. The signal propagates at a speed of  $3 \times 10^8$  m/s. The time taken (in milliseconds, rounded off to two decimal places) for the receiver to completely receive a packet of 1000 bytes transmitted by the sender is \_\_\_\_\_. **(GATE 2022) (2 MARKS)**

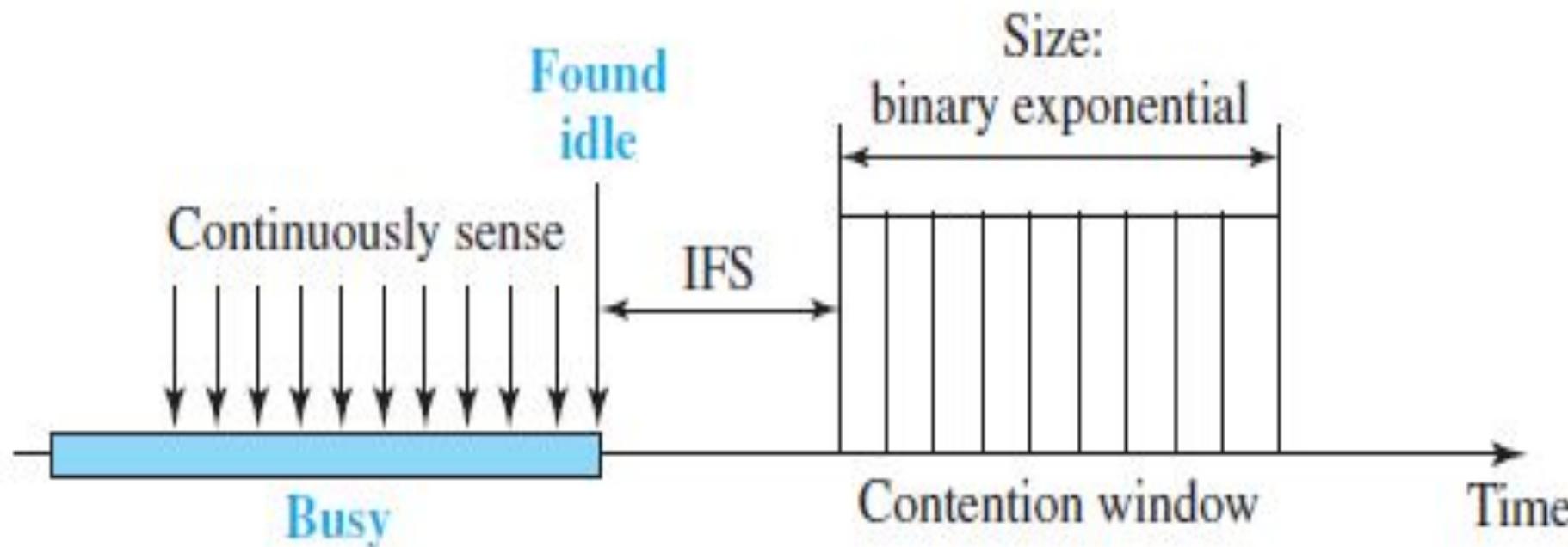
## Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA)

- In a wireless network, much of the sent energy is lost in transmission. The received signal has very little energy. Therefore, a collision may add only 5 to 10 percent additional energy. This is not useful for effective collision detection.
- We need to avoid collisions on wireless networks because they cannot be detected. Carrier sense multiple access with collision avoidance (CSMA / CA) was invented for this network. Collisions are avoided through the use of CSMA / CA
- three strategies: the interframe space, the contention window, and acknowledgment



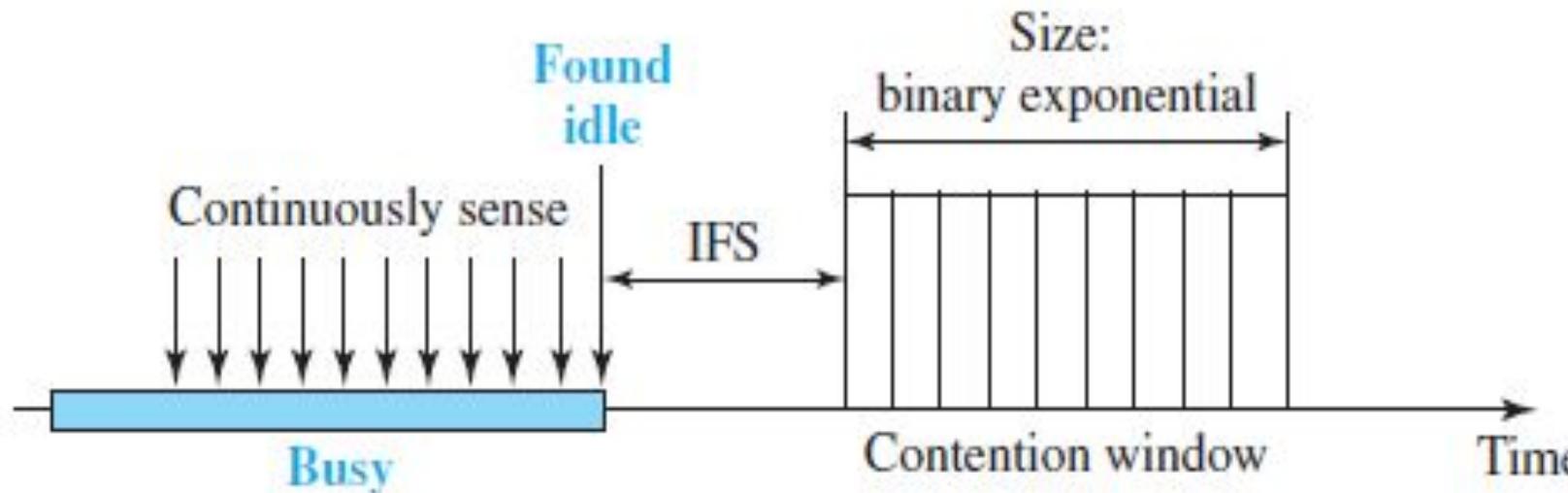
## • Interframe Space (IFS)

- First, collisions are avoided by deferring transmission even if the channel is found idle. When an idle channel is found, the station does not send immediately. It waits for a period of time called the interframe space or IFS.
- The IFS time allows the front of the transmitted signal by the distant station to reach this station. If after the IFS time the channel is still idle, the station can send, but it still needs to wait a time equal to the contention time. The IFS variable can also be used to prioritize stations or frame types.



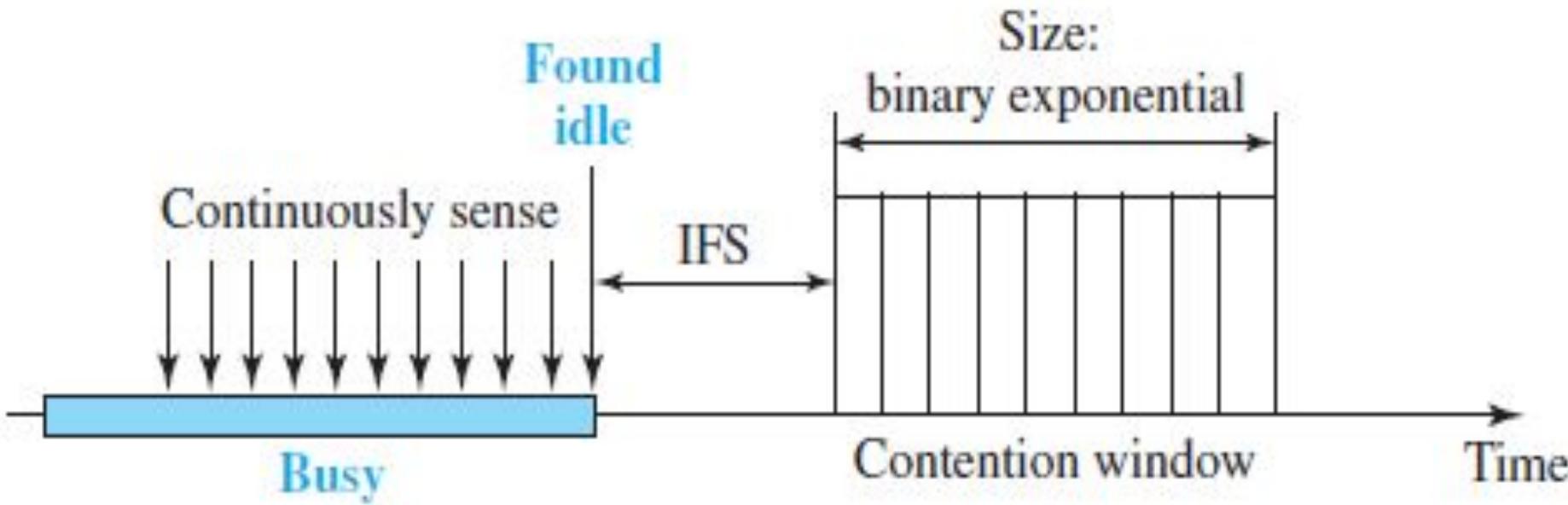
## Contention Window

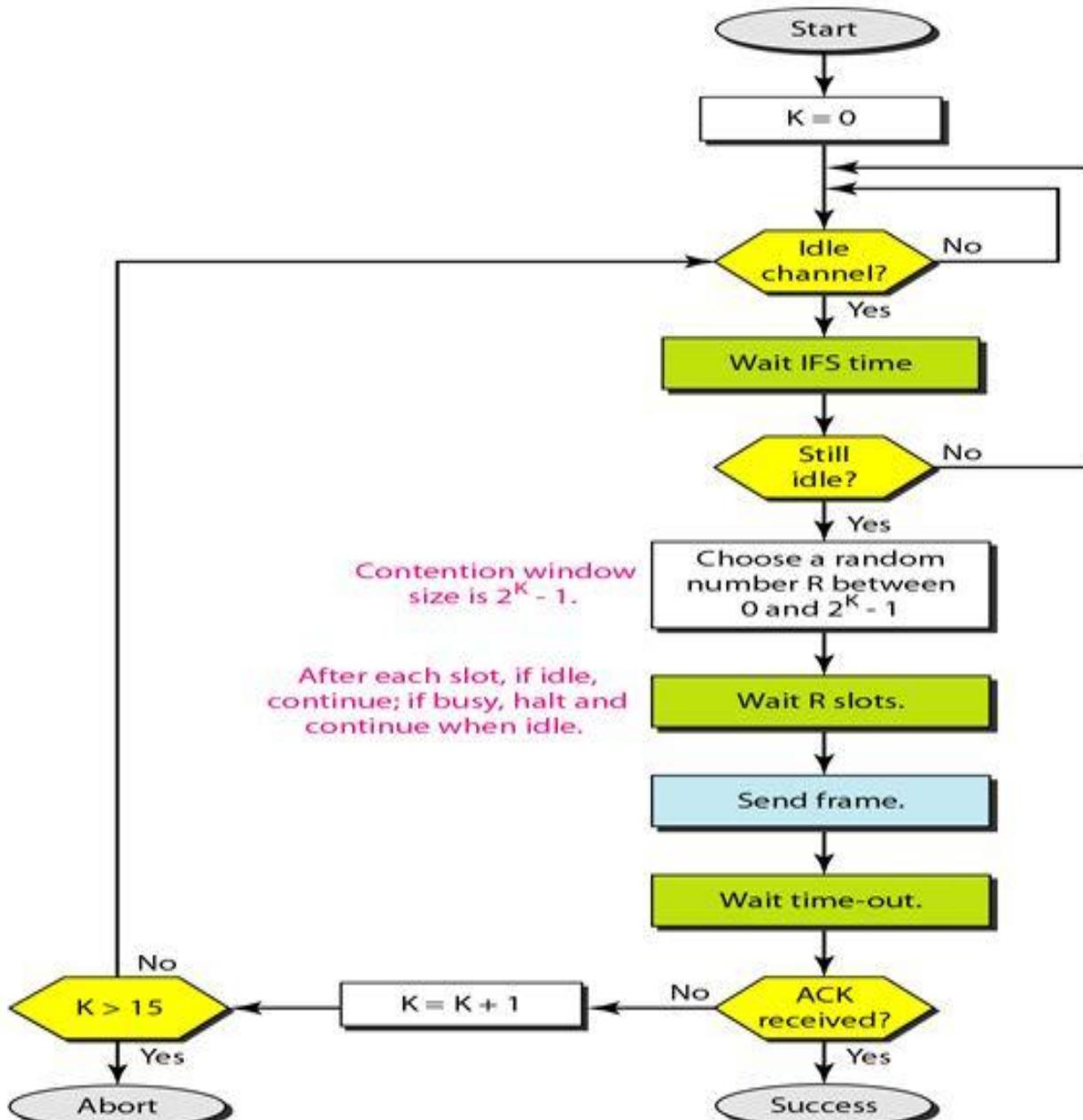
- The contention window is an amount of time divided into slots. A station that is ready to send chooses a random number of slots as its wait time.
- The number of slots in the window changes according to the binary exponential back-off strategy.
- This means that it is set to one slot the first time and then doubles each time.
- One interesting point about the contention window is that the station needs to sense the channel after each time slot.
- However, if the station finds the channel busy, it does not restart the process; it just stops the timer and restarts it when the channel is sensed as idle. This gives priority to the station with the longest waiting time.



## • Acknowledgment

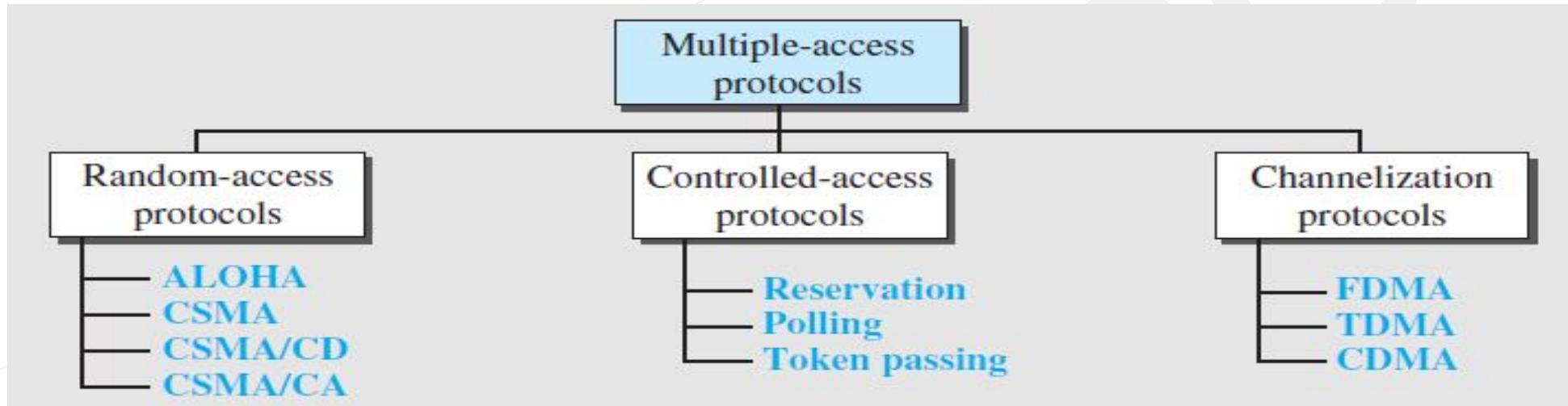
- With all these precautions, there still may be a collision resulting in destroyed data. In addition, the data may be corrupted during the transmission. The positive acknowledgment and the time-out timer can help guarantee that the receiver has received the frame.





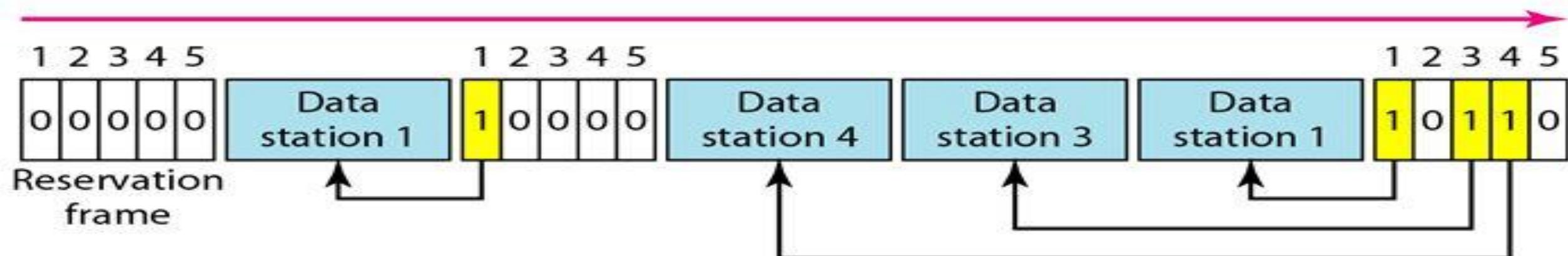
# CONTROLLED ACCESS

- In controlled access, the stations consult one another to find which station has the right to send. A station cannot send unless it has been authorized by other stations. We discuss three popular controlled-access methods.



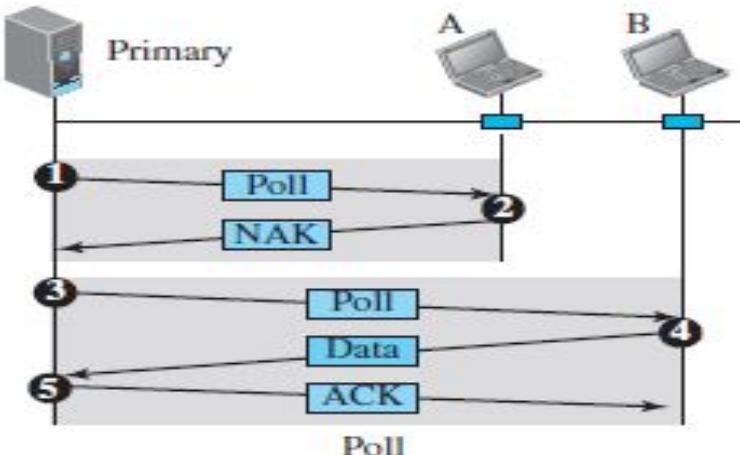
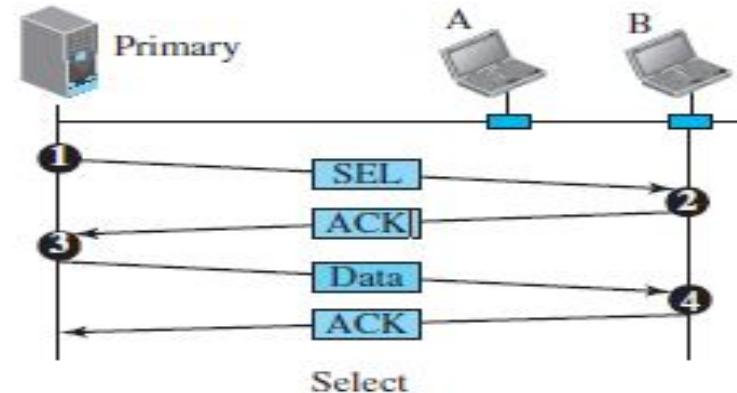
# Reservation

- **Purpose:** Stations make a reservation before sending data, which helps in organizing transmissions and avoiding collisions.
- **Timeline Structure: Reservation Interval:** A fixed time length dedicated to reserving slots. **Data Transmission Period:** A variable period for transmitting data frames.
- **Slot Allocation:** If there are **M stations**, the reservation interval is divided into **M slots**, with each station having one slot. If a station (e.g., Station 1) has a frame to send, it indicates this by transmitting a 1-bit signal during its respective slot. No other station is allowed to transmit during another station's slot.
- **Announcement & Data Transfer:** Any station that wants to send data announces it by marking its respective slot. Once all slots have been checked, each station knows which other stations intend to transmit. Stations that have reserved slots transmit their frames in the agreed order, avoiding collisions.
- **Process Continuation:** After the data transmission period, a new reservation interval begins. This method ensures an orderly data transfer, preventing any collisions by having a clear agreement on which station transmits next.



# Polling

- Polling works with topologies in which one device is designated as a primary station and the other devices are secondary stations. All data exchanges must be made through the primary device even when the ultimate destination is a secondary device. The primary device controls the link; the secondary devices follow its instructions. It is up to the primary device to determine which device is allowed to use the channel at a given time.
- **Select**
  - The select function is used whenever the primary device has something to send. Remember that the primary controls the link. If the primary is neither sending nor receiving data, it knows the link is available. If it has something to send, the primary device sends it. What it does not know, however, is whether the target device is prepared to receive. So the primary must alert the secondary to the upcoming transmission and wait for an acknowledgment of the secondary's ready status. Before sending data, the primary creates and transmits a select (SEL) frame, one field of which includes the address of the intended secondary.
- **Poll**
  - The poll function is used by the primary device to solicit transmissions from the secondary devices. When the primary is ready to receive data, it must ask (poll) each device in turn if it has anything to send. When the first secondary is approached, it responds either with a NAK frame if it has nothing to send or with data (in the form of a data frame) if it does. If the response is negative (a NAK frame), then the primary polls the next secondary in the same manner until it finds one with data to send. When the response is positive (a data frame), the primary reads the frame and returns an acknowledgment (ACK frame), verifying its receipt.

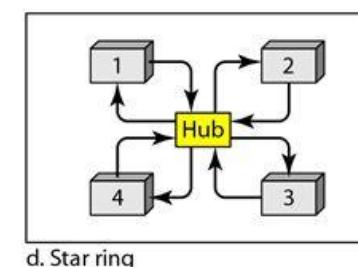
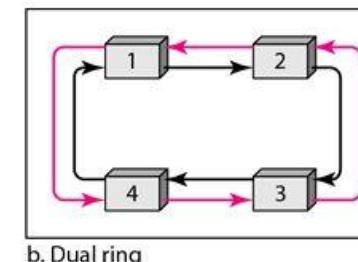
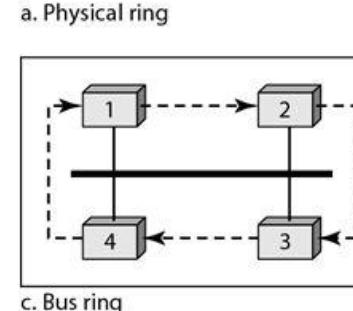
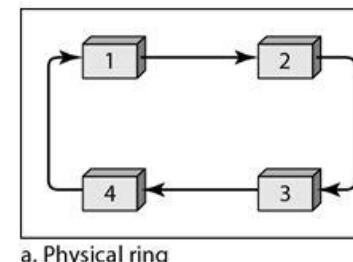


**Q** A broadcast channel has 10 nodes and total capacity of 10 Mbps. It uses polling for medium access. Once a node finishes transmission, there is a polling delay of  $80 \mu\text{s}$  to poll the next node. Whenever a node is polled, it is allowed to transmit a maximum of 1000 bytes. The maximum throughput of the broadcast channel is **(Gate-2007) (2 Marks)**

- (A) 1 Mbps      (B)  $100/11$  Mbps      (C) 10 Mbps      (D) 100 Mbps

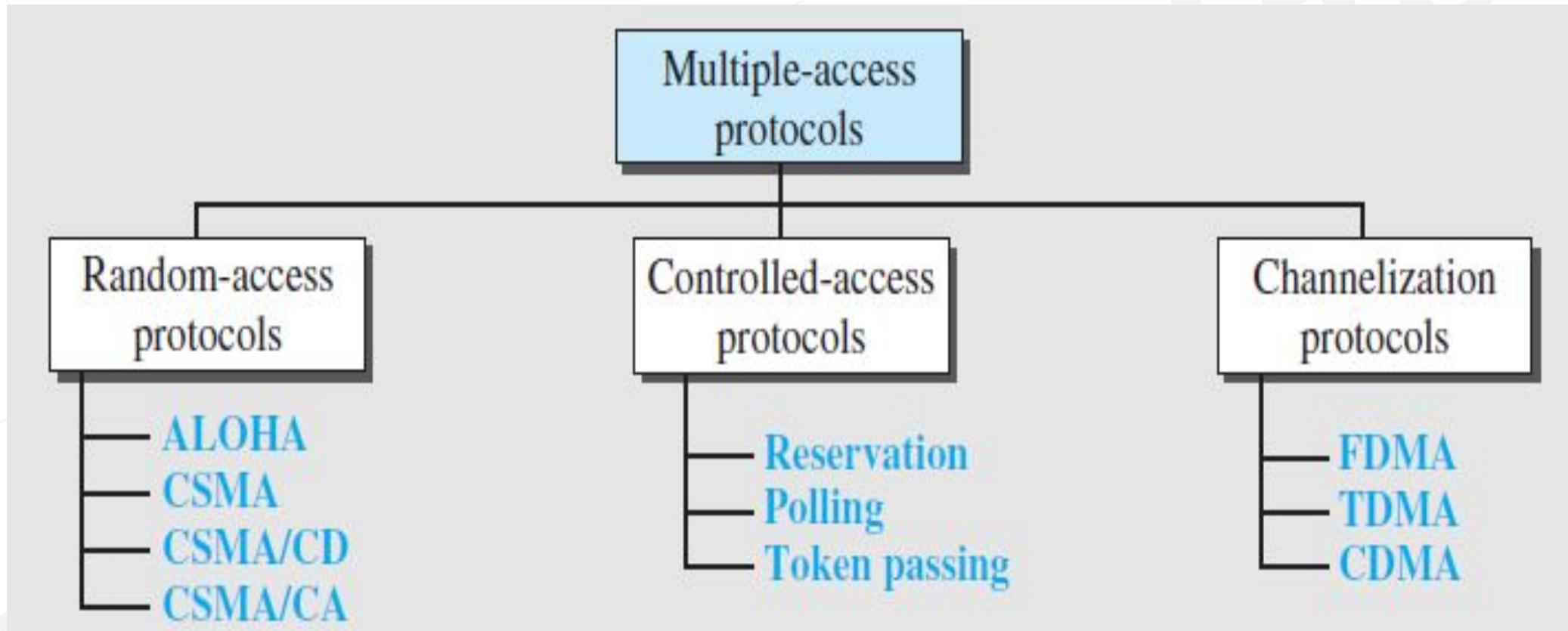
# Token Passing

- Token passing is a method used to control access to the network in a logical ring topology. In this approach, a special packet called a *token* circulates among the stations, allowing them to transmit data only when they hold the token.
- **Logical Ring Topology:** Stations are organized in a logical ring, where each station has a *predecessor* and a *successor*. The predecessor is the station logically before, and the successor is the one after in the ring.
- **Token Circulation:** The token is a special packet that circulates through the ring, granting permission to a station to access the communication channel. Only the station holding the token can transmit data. When a station has data to send, it waits to receive the token from its predecessor. After sending its data, the station releases the token to its successor in the ring.
- **Token Management: Time Control:** The time a station can hold the token must be limited to prevent monopolization. **Token Monitoring:** To ensure network integrity, the token must be tracked, especially to detect if it is lost or if a station holding it fails. **Priority Assignment:** Token management allows assigning priorities to stations and the types of data transmitted, ensuring critical information gets transmitted first.
- **Handling Failures:** When a station holding the token fails, it is essential to regenerate or reassign the token to maintain network functionality.
- **Data Transmission and Token Passing:** When a station receives the token but has no data to send, it immediately passes the token to the next station. A station must wait for the next round to transmit if it has already passed the token in the current round.



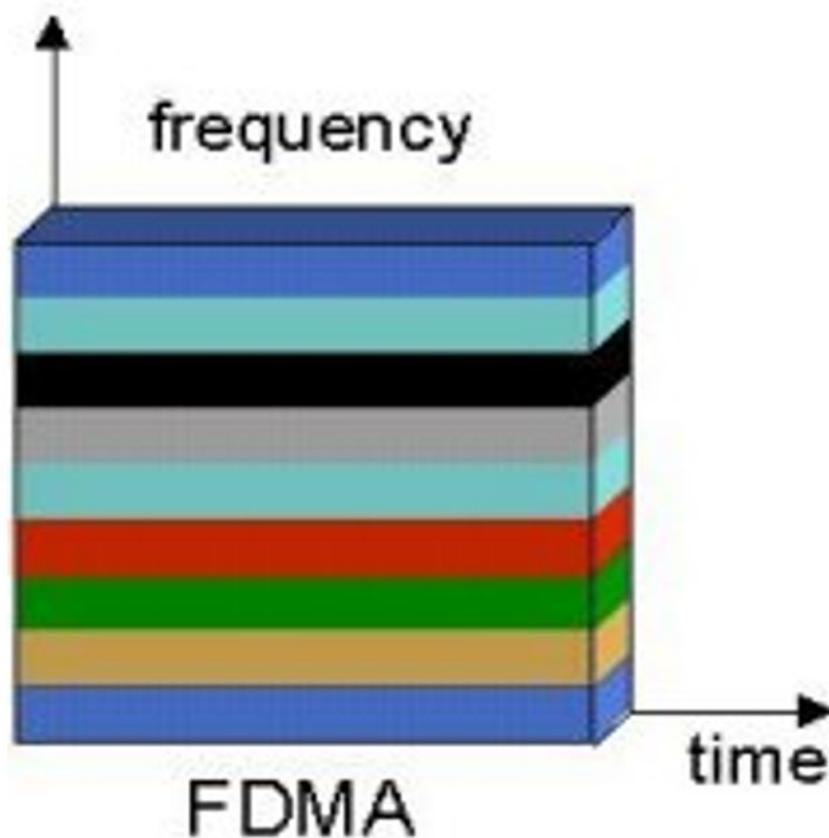
# CHANNELIZATION

- Channelization is a multiple-access method in which the available bandwidth of a link is shared in time, frequency, or through code, between different stations. In this section, we discuss three channelization protocols: FDMA, TDMA, and CDMA.



## Frequency-Division Multiple Access (FDMA)

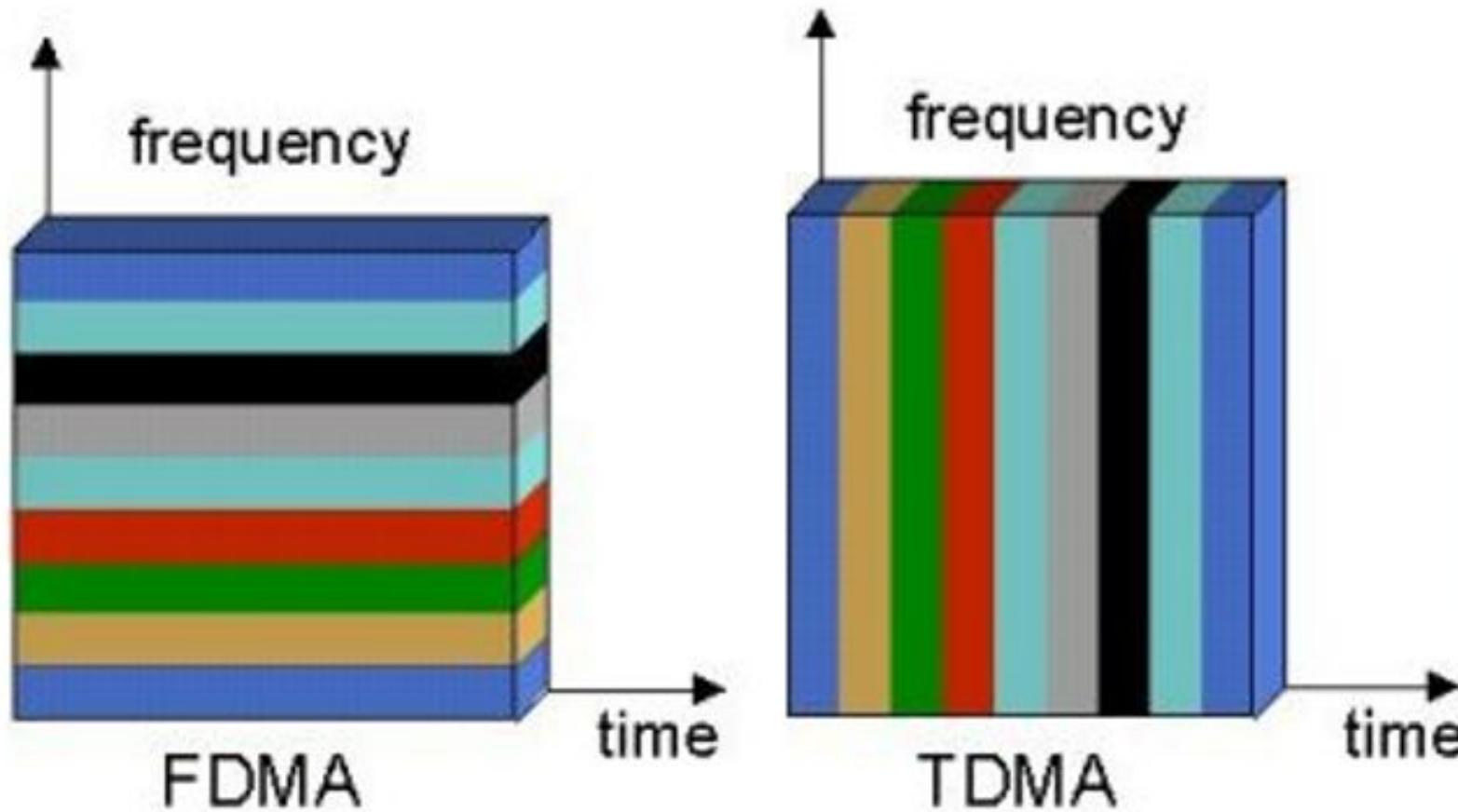
- In frequency-division multiple access (FDMA), the available bandwidth is divided into frequency bands. Each station is allocated a band to send its data.
- In other words, each band is reserved for a specific station, and it belongs to the station all the time.



Frequency	Station Name
90.4	Bol Radio
<b>91.1</b>	<b>Radio City</b>
92.7	BIG FM 92.7
<b>93.5</b>	<b>RED FM</b>
94.3	Fever
<b>95</b>	<b>Mirchi95</b>
98.3	Radio Mirchi
<b>101.9</b>	<b>AIR Rainbow</b>
102.8	AIR Vividh Bharati
<b>104</b>	<b>KOOL FM</b>
105.6	IGNOU Gyan Vani
<b>106.4</b>	<b>Magic FM</b>
107.8	Deccan Radio
<b>107.8</b>	<b>Radio Charminar</b>

## Time-Division Multiple Access (TDMA)

- In time-division multiple access (TDMA), the stations share the bandwidth of the channel in time. Each station is allocated a time slot during which it can send data. Each station transmits its data in its assigned time slot.

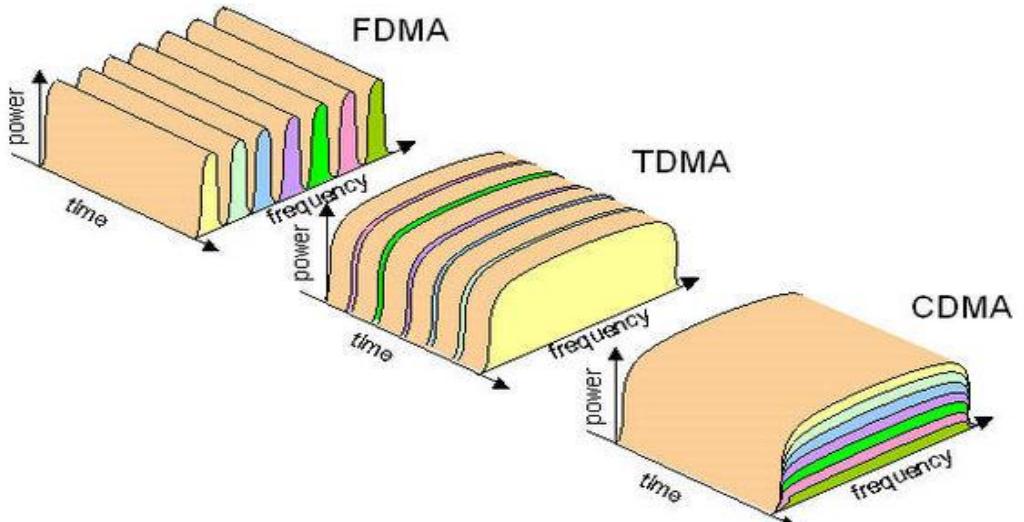


**Q** In a TDM medium access control bus LAN, each station is assigned one time slot per cycle for transmission. Assume that the length of each time slot is the time to transmit 100 bits plus the end-to-end propagation delay. Assume a propagation speed of  $2 \times 10^8$  m/sec. The length of the LAN is 1 km with a bandwidth of 10 Mbps. The maximum number of stations that can be allowed in the LAN so that the throughput of each station can be  $2/3$  Mbps is (Gate-2005) (2 Marks)

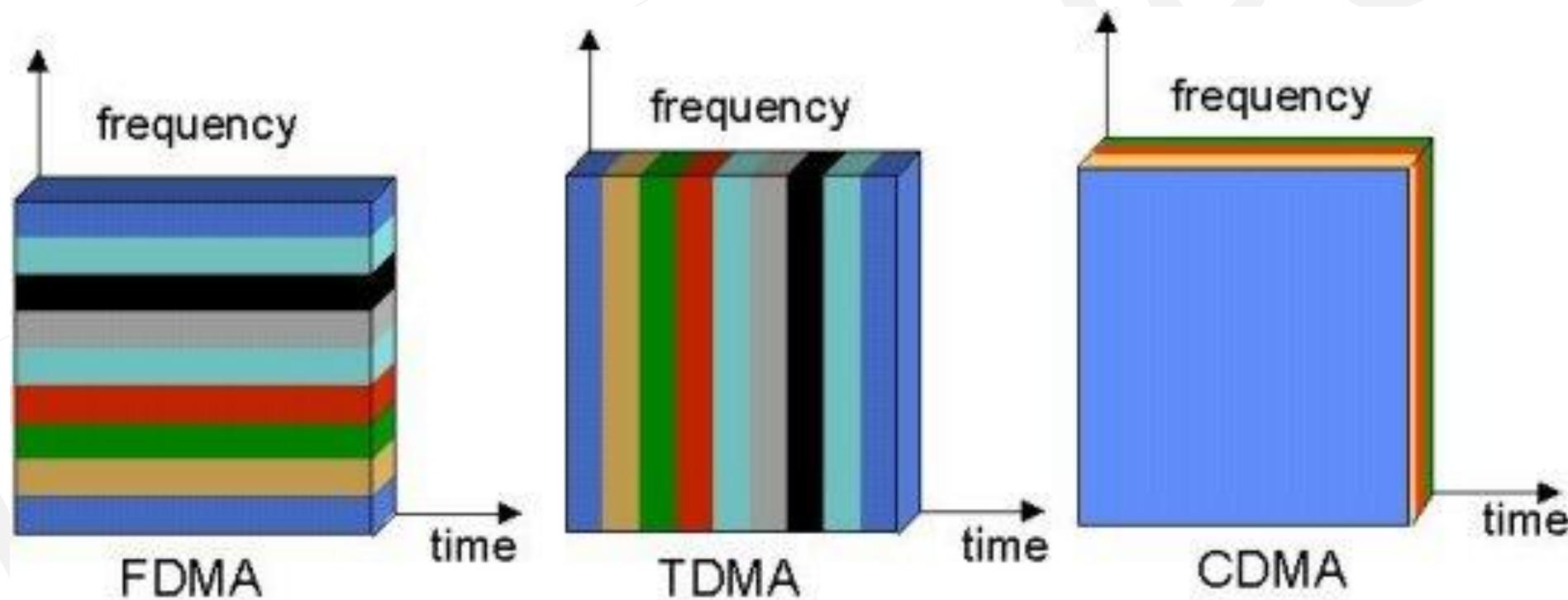
- (A) 3                    (B) 5                    (C) 10                    (D) 20

## Code-Division Multiple Access (CDMA)

- Code-division multiple access (CDMA) was conceived several decades ago. Recent advances in electronic technology have finally made its implementation possible. CDMA differs from FDMA because only one channel occupies the entire bandwidth of the link. It differs from TDMA because all stations can send data simultaneously; there is no timesharing.



- Let us first give an analogy. CDMA simply means communication with different codes. For example, in a large room with many people, two people can talk in English if nobody else understands English.
- Another two people can talk in Chinese if they are the only ones who understand Chinese, and so on. In other words, the common channel, the space of the room in this case, can easily allow communication between several couples, but in different languages (codes).



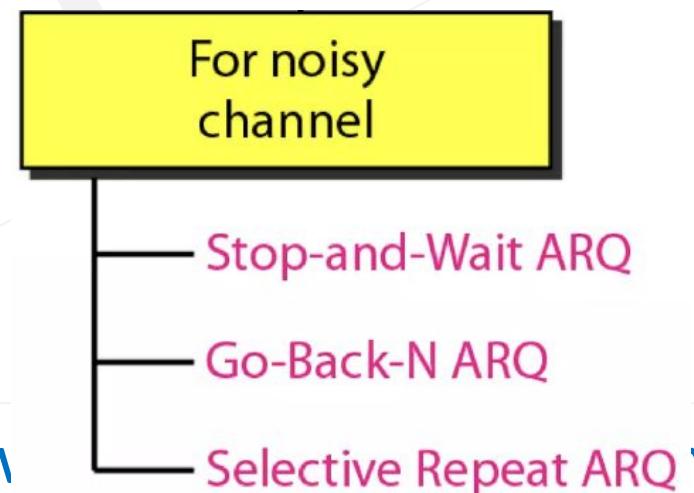
# FLOW AND ERROR CONTROL

- **Flow Control:**

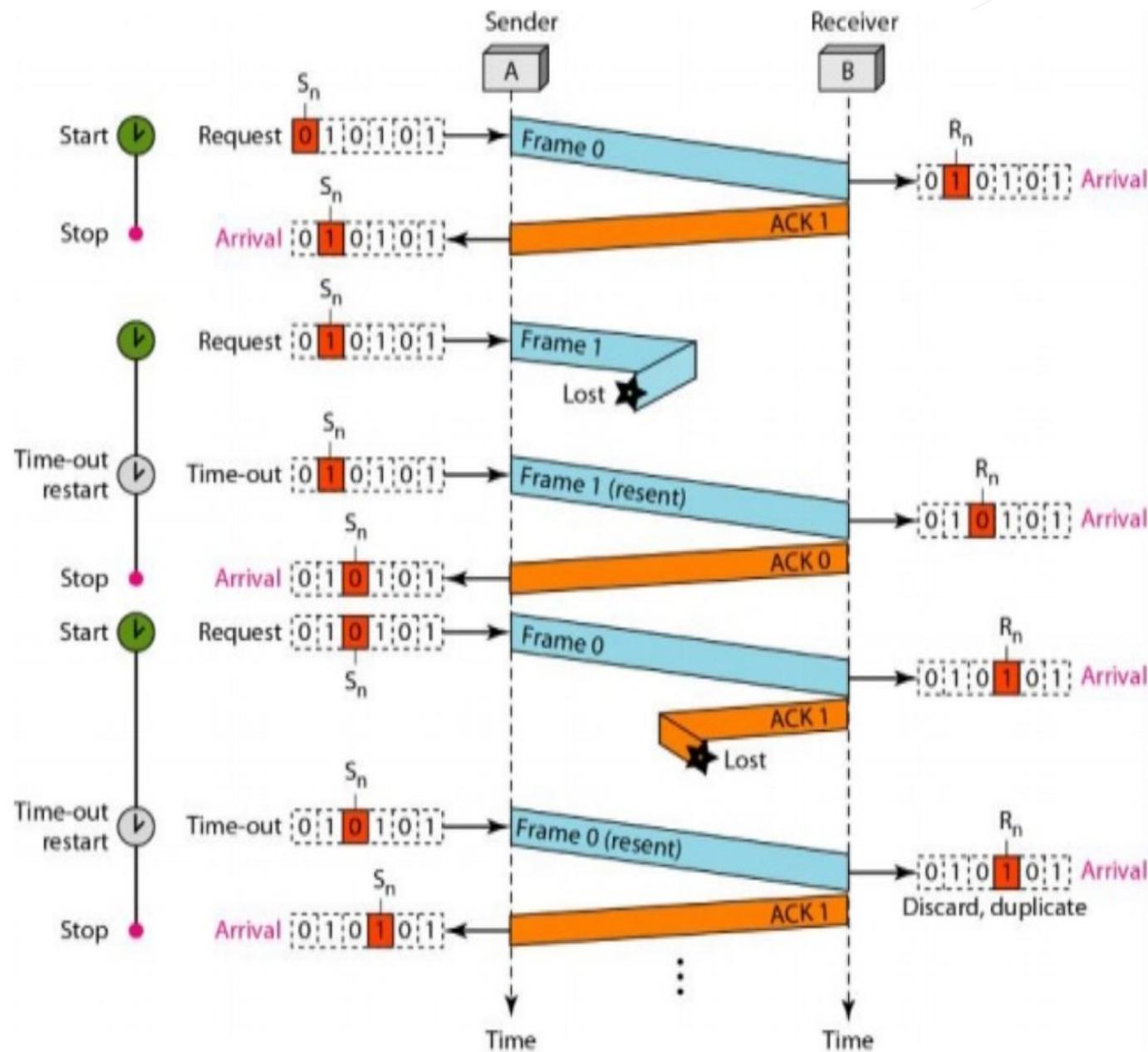
- **Purpose:** To prevent the receiver from being overwhelmed by incoming data due to its limited processing speed and memory.
- **Buffer Mechanism:** Each receiver has a buffer to store incoming data temporarily until they are processed.
- **Handling Overflow:** If the buffer nears its capacity, the receiver signals the sender to pause or slow down transmission until it can handle more data.

- **Error Control:**

- **Purpose:** To detect and correct errors in transmitted data.
- **Error Detection:** Identifies lost, out-of-order, or corrupted frames during transmission.
- **Retransmission (ARQ):** If errors are detected, the receiver requests the sender to retransmit the affected frames using Automatic Repeat Request (ARQ) protocols.



# Stop-and-Wait Automatic Repeat Request



- Stop-and-Wait ARQ is a basic error control protocol that uses numbered frames and a simple acknowledgment mechanism to ensure reliable data transmission.
- **Error Handling:**
  - **Corrupted Frames:** Detected by the receiver's silence; corrupted frames are discarded silently.
  - **Lost Frames:** To detect and handle lost frames, frames are numbered sequentially.
- **Sequence Numbers:**
  - Frames are assigned sequence numbers to identify them uniquely.
  - A minimal range of sequence numbers is used to reduce frame size, cycling between 0 and 1.
  - The receiver uses these numbers to detect duplicates and lost frames.
- **Acknowledgment and Timer Mechanism:**
  - After sending a frame, the sender keeps a copy and starts a timer.
  - If the acknowledgment (ACK) does not arrive within the timer period, the sender resends the frame.
  - This helps in identifying whether the sent frame was received correctly or needs retransmission.
- **Why Use Sequence Numbers:**
  - To differentiate between new frames and retransmitted duplicates, sequence numbers are essential.
  - The range of sequence numbers is chosen to be small (e.g., 0, 1) for simplicity and efficient communication.
- **Summary:**
  - Stop-and-Wait ARQ ensures reliable communication by numbering frames, using acknowledgments, and employing a timer-based retransmission strategy. This allows the protocol to manage corrupted and lost frames effectively with minimal ambiguity.

- **Transmission Delay (TT)**: A sender needs to put the bits in a packet on the line one by one. If the first bit of the packet is put on the line at time  $t_1$  and the last bit is put on the line at time  $t_2$ , transmission delay of the packet is  $(t_2 - t_1)$ .

$$T_t = (\text{Packet length (L)}) / (\text{Transmission rate or Bandwidth (B)}) = L / B$$

- **Propagation Delay**: Propagation delay is the time it takes for a bit to travel from point A to point B in the transmission media.

$$T_p = (\text{Distance}) / (\text{Propagation speed})$$

- **Processing Delay** - It is the time required for a destination host to receive a packet from its input port, remove the header, perform an error detection procedure, and deliver the packet to the output port or deliver the packet to the upper-layer protocol (in the case of the destination host).
  - $\text{Delay}_{\text{pr}}$  = Time required to process a packet in a destination host
- **Queuing Delay** - It is measured as the time a packet waits in the input queue and output queue of a router.
  - $\text{Delay}_{\text{qu}}$  = The time a packet waits in input and output queues in a router

## Measuring Performance for Stop and Wait

1. The total time is measured as =  $T_{t(\text{data})} + T_{p(\text{data})} + \text{Delay}_{\text{que}} + \text{Delay}_{\text{pro}} + T_{t(\text{ack})} + T_{p(\text{ack})}$
2. Queuing delay and processing delays are generally kept 0.
  - Total Time =  $T_{t(\text{data})} + T_{p(\text{data})} + T_{t(\text{ack})} + T_{p(\text{ack})}$
3. In general we have taken  $T_{t(\text{ack})}$  as negligible as the ack size is generally very less
  - Total Time =  $T_{t(\text{data})} + T_{p(\text{data})} + T_{p(\text{ack})}$
4. The  $T_p$  for data and ack are almost going to be same
  - Total Time =  $T_{t(\text{data})} + 2 * T_p$
  - Note:- Some times  $2 * T_p$  time is also called Round Trip Time (RTT)

- **Efficiency( $\eta$ ):-**

- Useful Time / Total Cycle time =  $T_t / T_t + 2 * T_p$
- Here, Useful time in the entire cycle time is  $T_t$  and for the rest  $2 * T_p$  time we are waiting for the processing, whereas instead of waiting we could have sent more packets.
- Dividing numerator and denominator with  $T_t$ , we get:  $\eta = 1 / 1 + (2 * T_p / T_t)$
- So,  $\eta = 1 / 1 + 2a$ , (where  $a = T_p / T_t$ )
- Effective Bandwidth / Throughput / Bandwidth Utilization is calculated as:
- Throughput =  $\eta * B$  (efficiency \* bandwidth)

**Q** A sender uses the Stop-and-Wait ARQ protocol for reliable transmission of frames. Frames are of size 1000 bytes and the transmission rate at the sender is 80 Kbps ( $1\text{Kbps} = 1000 \text{ bits/second}$ ). Size of an acknowledgement is 100 bytes and the transmission rate at the receiver is 8 Kbps. The one-way propagation delay is 100 milliseconds. Assuming no frame is lost, the sender throughput is \_\_\_\_\_ bytes/second. **(Gate-2016) (2 Marks)**

**Q** Suppose that the stop-and-wait protocol is used on a link with a bit rate of 64 kilobits per second and 20 milliseconds propagation delay. Assume that the transmission time for the acknowledgment and the processing time at nodes are negligible. Then the minimum frame size in bytes to achieve a link utilization of at least 50% is \_\_\_\_\_. (Gate-2015) (2 Marks)

- (A) 160      (B) 320      (C) 640      (D) 220

**Q** A link has a transmission speed of  $10^6$  bits/sec. It uses data packets of size 1000 bytes each. Assume that the acknowledgment has negligible transmission delay, and that its propagation delay is the same as the data propagation delay. Also assume that the processing delays at nodes are negligible. The efficiency of the stop-and-wait protocol in this setup is exactly 25%. The value of the one-way propagation delay (in milliseconds) is \_\_\_\_\_.

**(Gate-2015) (1 Marks)**

**Q** On a wireless link, the probability of packet error is 0.2. A stop-and-wait protocol is used to transfer data across the link. The channel condition is assumed to be independent from transmission to transmission. What is the average number of transmission attempts required to transfer 100 packets? **(Gate-2006)(2 Marks)**

- (A) 100**
- (B) 125**
- (C) 150**
- (D) 200**

**Q** A channel has a bit rate of 4 kbps and one-way propagation delay of 20 ms. The channel uses stop and wait protocol. The transmission time of the acknowledgement frame is negligible. To get a channel efficiency of at least 50%, the minimum frame size should be **(Gate-2005) (2 Marks)**

- (A) 80 bytes      (B) 80 bits      (C) 160 bytes      (D) 160 bits

**Q** The values of parameters for the Stop-and-Wait ARQ protocol are as given below:

- Bit rate of the transmission channel = 1 Mbps.
- Propagation delay from sender to receiver = 0.75 ms.
- Time to process a frame = 0.25 ms.
- Number of bytes in the information frame = 1980.
- Number of bytes in the acknowledge frame = 20.
- Number of overhead bytes in the information frame = 20.

Assume there are no transmission errors. Then, the transmission efficiency (expressed in percentage) of the Stop-and-Wait ARQ protocol for the above parameters is \_\_\_\_\_ (correct to 2 decimal places). **(Gate-2017) (2 Marks)**

**Q.** Suppose we are transmitting frames between two nodes using Stop-and-Wait protocol.

The frame size is 3000 bits. The transmission rate of the channel is 2000 bps (bits/second),

and the propagation delay between the two nodes is 100 milliseconds.

Assume that the processing times at the source and destination are negligible.

Also, assume that the size of the acknowledgement packet is negligible.

Which ONE of the following most accurately gives the channel utilization for the above scenario in percentage? (GATE 2025)

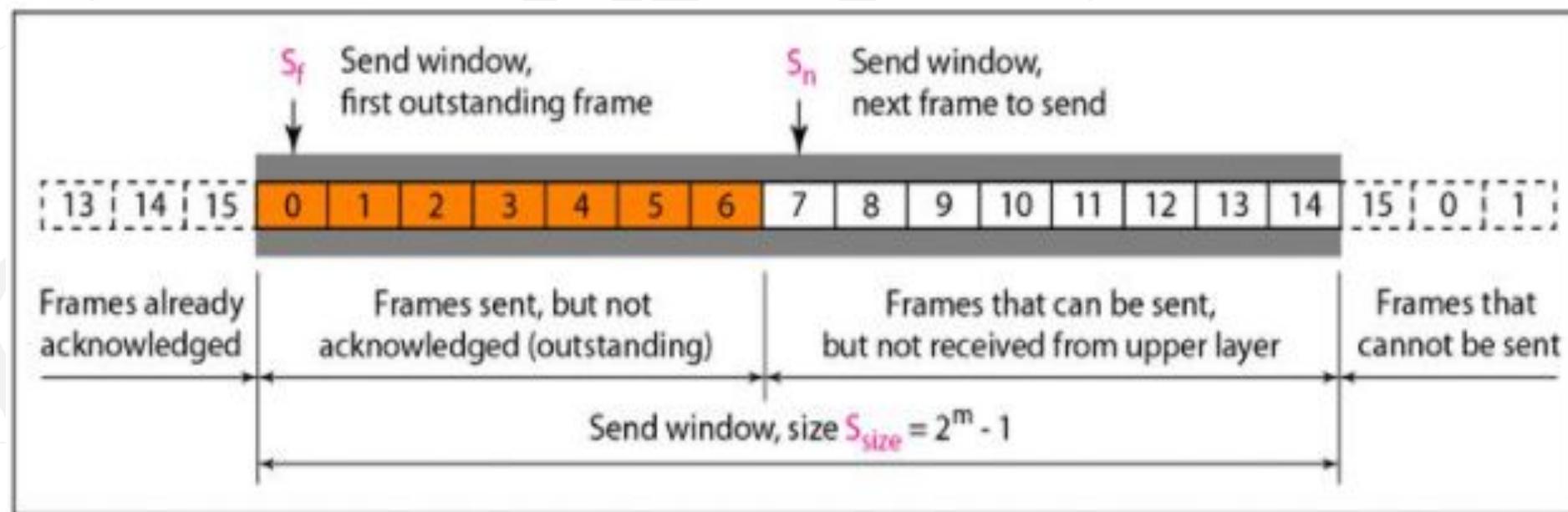
- A) 88.23
- B) 93.75
- C) 85.44
- D) 66.67

## Efficiency Limitation of Stop-and-Wait ARQ

- **Issue with Large Bandwidth and Delay:**
  - Stop-and-Wait ARQ becomes inefficient on channels with large bandwidth (thick) and long round-trip delays (long).
  - The combination of bandwidth and delay is termed the *bandwidth-delay product*, which represents the amount of data (in bits) that can be in transit while waiting for an acknowledgment.
- **Conceptual Comparison:**
  - Think of the channel as a pipe. The bandwidth-delay product measures the pipe's capacity in bits. If the protocol does not utilize this capacity effectively, data transfer efficiency drops.
- **Summary:** Stop-and-Wait ARQ struggles with high bandwidth and long delay channels, as it fails to utilize the full capacity of the communication channel, leading to inefficiency.

# Go-Back-N Automatic Repeat Request

- **Purpose:** Go-Back-N ARQ enhances transmission efficiency by allowing multiple frames to be sent without waiting for individual acknowledgments, unlike Stop-and-Wait ARQ.
- **Sending Multiple Frames:**
  - Several frames can be in transit before receiving their acknowledgments. The sender keeps copies of all sent frames until their acknowledgments arrive.
- **Window Concept:**
  - **Send Window:** Divides sequence numbers into four regions:  
**Acknowledged Frames:** Sequence numbers of frames already acknowledged.  
**Outstanding Frames:** Sent frames whose acknowledgments are still awaited.  
**Frames Ready to Send:** Sequence numbers for frames that are ready but not yet sent.  
**Future Frames:** Sequence numbers that cannot be used until the window slides.
  - **Receive Window:** Size is always 1, ensuring the receiver only accepts the expected frame. Out-of-order frames are discarded and need to be resent.



- **Variables:**
  - **S<sub>f</sub>:** Sequence number of the first outstanding frame.
  - **S<sub>n</sub>:** Sequence number to be assigned to the next frame.
  - **Ssize:** Fixed size of the send window.
- **Timer Management:**
  - A single timer is used for the first outstanding frame. This timer always expires first, prompting the sender to resend all outstanding frames when needed.
- **Acknowledgment Strategy:**
  - **Positive Acknowledgment:** Sent if a frame arrives correctly and in order.
  - **Receiver Silence:** If a frame is out of order or damaged, the receiver stays silent and discards any subsequent frames until the expected frame is received.
  - **Cumulative Acknowledgment:** Instead of acknowledging each frame individually, the receiver can send a single acknowledgment for multiple frames.
- **Resending Frames:**
  - When the timer expires, indicating an unacknowledged frame, the sender resends all outstanding frames starting from the frame with the expired timer.
  - Example: If the sender has sent frames 3, 4, 5, and 6, and the timer for frame 3 expires, the sender resends frames 3 to 6.
- Go-Back-N ARQ improves transmission by allowing multiple frames in transit at once, reducing idle times. The sender manages these frames using a sliding window mechanism, while the receiver ensures orderly delivery by maintaining a window size of one, discarding any out-of-order frames.

## Sequence and Acknowledgement Numbers

- To improve the efficiency of transmission (to fill the pipe), multiple packets must be in transition while the sender is waiting for acknowledgment
- In order to maximize the efficiency, the window size ( $W_s$ ) =  $(1 + 2a)$
- Number of bits required for sequence numbers =  $\text{ceil} (\log_2 (1 + 2a))$

**Q** Consider a network connecting two systems located 8000 kilometres apart. The bandwidth of the network is  $500 \times 10^6$  bits per second. The propagation speed of the media is  $4 \times 10^6$  meters per second. It is needed to design a Go-Back-N sliding window protocol for this network. The average packet size is  $10^7$  bits. The network is to be used to its full capacity. Assume that processing delays at nodes are negligible. Then, the minimum size in bits of The sequence number field has to be \_\_\_\_\_. **(Gate-2015) (2 Marks)**

**Q** A 1Mbps satellite link connects two ground stations. The altitude of the satellite is 36,504 km and speed of the signal is  $3 \times 10^8$  m/s. What should be the packet size for a channel utilization of 25% for a satellite link using go-back-127 sliding window protocol? Assume that the acknowledgment packets are negligible in size and that there are no errors during communication. **(Gate-2008) (2 Marks)**

- (A)** 120 bytes      **(B)** 60 bytes      **(C)** 240 bytes      **(D)** 90 bytes

**Q** Station A needs to send a message consisting of 9 packets to Station B using a sliding window (window size 3) and go-back-n error control strategy. All packets are ready and immediately available for transmission. If every 5th packet that A transmits gets lost (but no acks from B ever get lost), then what is the number of packets that A will transmit for sending the message to B?

**(Gate-2006) (2 Marks)**

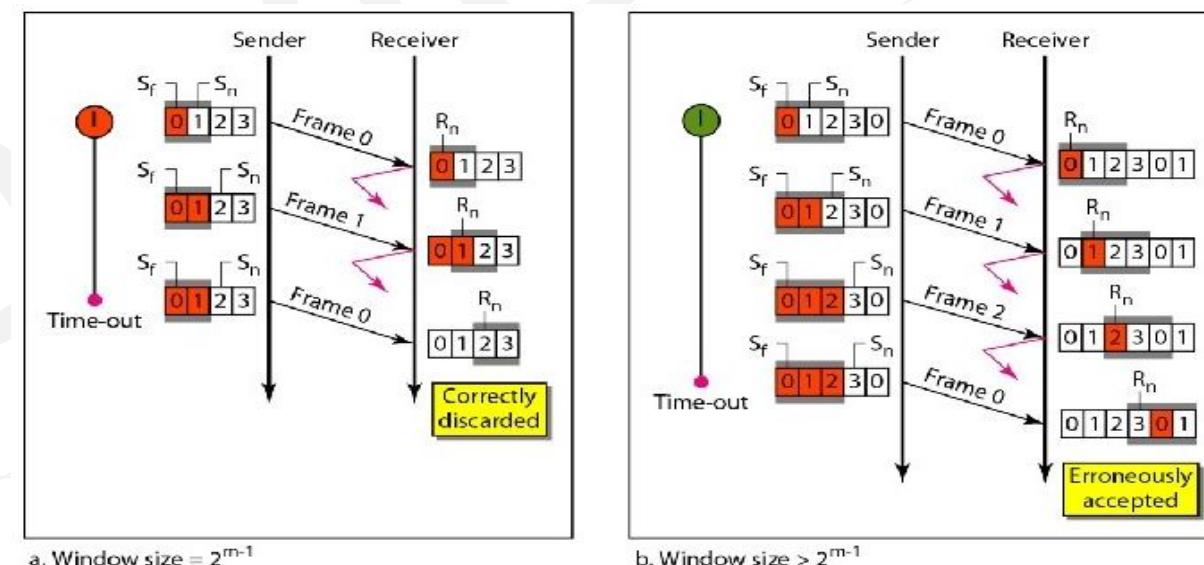
- (A) 12
- (B) 14
- (C) 16
- (D) 18

**Q** A 20 Kbps satellite link has a propagation delay of 400 ms. The transmitter employs the “go back n ARQ” scheme with n set to 10. Assuming that each frame is 100 bytes long, what is the maximum data rate possible? **(Gate-2004) (2 Marks)**

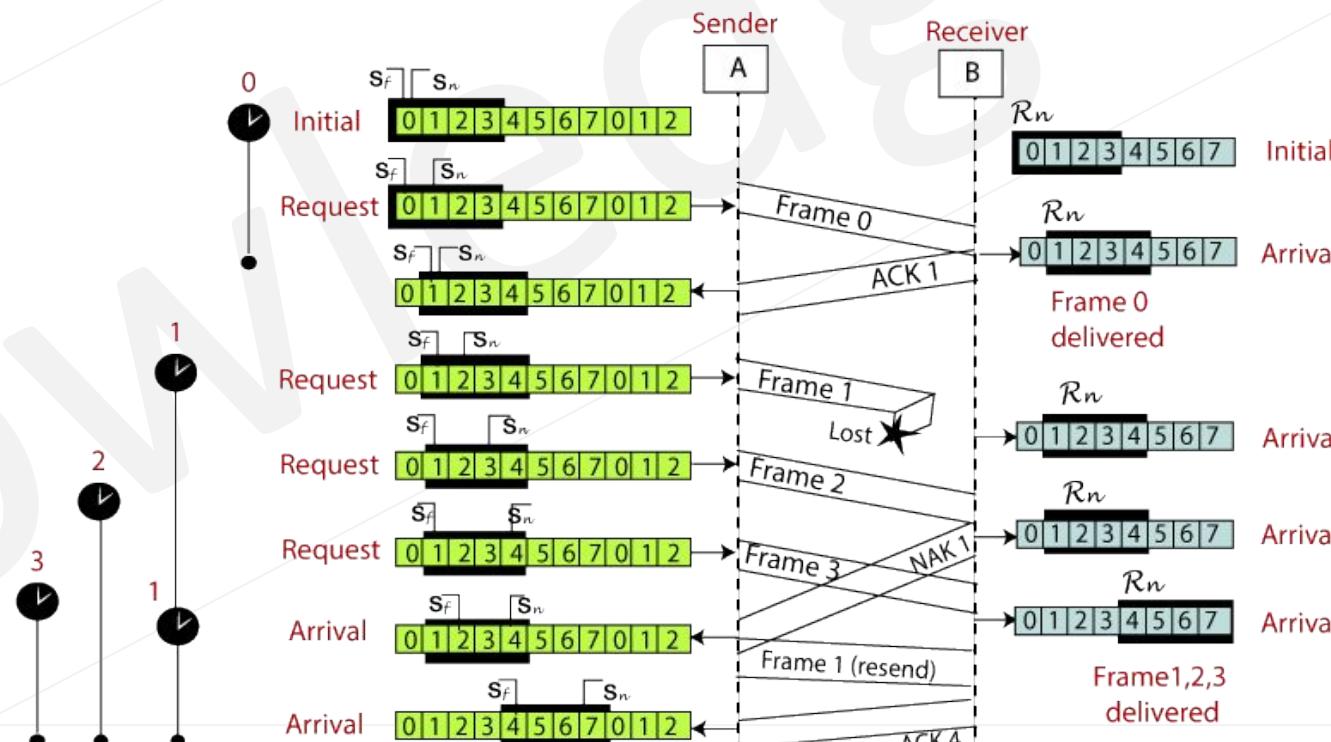
- (A) 5Kbps**
- (B) 10Kbps**
- (C) 15Kbps**
- (D) 20Kbps**

# Selective Repeat Automatic Repeat Request

- Selective Repeat ARQ improves efficiency on noisy links by retransmitting only the damaged frames, unlike Go-Back-N ARQ which resends all subsequent frames when an error occurs.
- **Handling Out-of-Order Frames:**
  - Out-of-order frames are buffered and stored until all frames in a sequence are received correctly.
  - Frames are delivered to the network layer only when a set of in-order frames is complete.
- **Send and Receive Windows:**
  - The size of both send and receive windows must be at most half of the maximum sequence number range ( $2^m$ ), ensuring reliable frame handling.
  - All frames within the window size can be received out of order and temporarily stored.
- **Acknowledgments and Negative Acknowledgments (NAKs):**
  - The receiver sends an *ACK* for correctly received frames and a *NAK* for damaged ones.
  - A timer is maintained for each sent frame, and only the frame with an expired timer is retransmitted.
  - When the receiver gets a *NAK*, it only resends the specific damaged frame.



- **Efficient Error Handling:**
  - Reduces unnecessary retransmissions by only resending the problematic frames.
  - This reduces bandwidth usage and increases efficiency, especially on noisy links.
- **Sliding Window Mechanism:**
  - The window slides forward when contiguous frames are acknowledged and delivered.
  - If contiguous frames starting from  $R_n$  are marked as received, the window slides, and their data is delivered.
- Selective Repeat ARQ improves efficiency by retransmitting only specific damaged frames rather than all subsequent ones. It uses buffer storage for out-of-order frames and maintains individual timers for each frame, making it ideal for noisy links where errors are more frequent.



**Q** Consider two hosts X and Y, connected by a single direct link of rate  $10^6$  bits/sec. The distance between the two hosts is 10,000 km and the propagation speed along the link is  $2 \times 10^8$  m/s. Hosts X send a file of 50,000 bytes as one large message to hosts Y continuously. Let the transmission and propagation delays be p milliseconds and q milliseconds, respectively. Then the values of p and q are: **(Gate-2017) (2 Marks)**

- (A) p = 50 and q = 100
- (B) p = 50 and q = 400
- (C) p = 100 and q = 50
- (D) p = 400 and q = 50

**Q** Consider a  $128 \times 10^3$  bits / second satellite communication link with one way propagation delay of 150 milliseconds. Selective retransmission (repeat) protocol is used on this link to send data with a frame size of 1 kilobyte. Neglect the transmission time of acknowledgement. The minimum number of bits required for the sequence number field to achieve 100% utilization is \_\_\_\_\_. **(Gate-2016) (2 Marks)**

**Q** Consider a selective repeat sliding window protocol that uses a frame size of 1 KB to send data on a 1.5 Mbps link with a one-way latency of 50 msec. To achieve a link utilization of 60%, the minimum number of bits required to represent the sequence number field is \_\_\_\_\_.

**(Gate-2014) (2 Marks)**

**Q** Consider a source computer(S) transmitting a file of size  $10^6$  bits to a destination computer(D) over a network of two routers ( $R_1$  and  $R_2$ ) and three links( $L_1$ ,  $L_2$ , and  $L_3$ ).  $L_1$  connects S to  $R_1$ ;  $L_2$  connects  $R_1$  to  $R_2$ ; and  $L_3$  connects  $R_2$  to D. Let each link be of length 100 km. Assume signals travel over each link at a speed of  $10^8$  meters per second. Assume that the link bandwidth on each link is 1Mbps. Let the file be broken down into 1000 packets each of size 1000 bits. Find the total sum of transmission and propagation delays in transmitting the file from S to D?

**(Gate-2012) (2 Marks)**

- (A) 1005 ms      (B) 1010 ms      (C) 3000 ms      (D) 3003 ms**

**Q** The distance between two stations  $M$  and  $N$  is  $L$  kilometres. All frames are  $K$  bits long. The propagation delay per kilometre is  $t$  seconds. Let  $R$  bits/second be the channel capacity. Assuming that processing delay is negligible, the *minimum* number of bits for the sequence number field in a frame for maximum utilization, when the *sliding window protocol* is used, is: **(Gate-2007) (2 Marks)**

- a)  $\lceil \log_2(2LtR+2K/K) \rceil$
- b)  $\lceil \log_2(2LtR/K) \rceil$
- c)  $\lceil \log_2(2LtR+K/K) \rceil$
- d)  $\lceil \log_2(2LtR+K/2K) \rceil$

**Q** Station A uses 32-byte packets to transmit messages to Station B using a sliding window protocol. The round-trip delay between A and B is 80 milliseconds and the bottleneck bandwidth on the path between A and B is 128 kbps. What is the optimal window size that A should use?

**(Gate-2006) (2 Marks)**

- (A) 20
- (B) 40
- (C) 160
- (D) 320

**Q** The maximum window size for data transmission using the selective reject protocol with n-bit frame sequence numbers is: **(Gate-2005) (1 Marks)**

- (A)  $2^n$       (B)  $2^{n-1}$       (C)  $2^n - 1$       (D)  $2^{n-2}$

**Q** In a sliding window ARQ scheme, the transmitter's window size is N and the receiver's window size is M. The minimum number of distinct sequence numbers required to ensure correct operation of the ARQ scheme is **(Gate-2004) (2 Marks)**

- (A) min (M, N)
- (B) max (M, N)
- (C) M + N
- (D) MN

**Q** Host A is sending data to host B over a full duplex link. A and B are using the sliding window protocol for flow control. The send and receive window sizes are 5 packets each. Data packets (sent only from A to B) are all 1000 bytes long and the transmission time for such a packet is 50  $\mu$ s. Acknowledgement packets (sent only from B to A) are very small and require negligible transmission time. The propagation delay over the link is 200  $\mu$ s. What is the maximum achievable throughput in this communication? **(Gate-2003)**

**(2 Marks)**

- (A)  $7.69 \times 10^6$  bytes per second
- (B)  $11.11 \times 10^6$  bytes per second
- (C)  $12.33 \times 10^6$  bytes per second
- (D)  $15.00 \times 10^6$  bytes per second

**Q** Consider the sliding window flow-control protocol operating between a sender and a receiver over a full-duplex error-free link. Assume the following:

- The time taken for processing the data frame by the receiver is negligible.
- The time taken for processing the acknowledgement frame by the sender is negligible.
- The sender has infinite number of frames available for transmission.
- The size of the data frame is 2,000 bits and the size of the acknowledgement frame is 10 bits.
- The link data rate in each direction is 1 Mbps (= 10<sup>6</sup> bits per second).
- One way propagation delay of the link is 100 milliseconds.

The minimum value of the sender's window size in terms of the number of frames, (rounded to the nearest integer) needed to achieve a link utilization of 50% is \_\_\_\_\_. **(GATE 2021) (2 MARKS)**

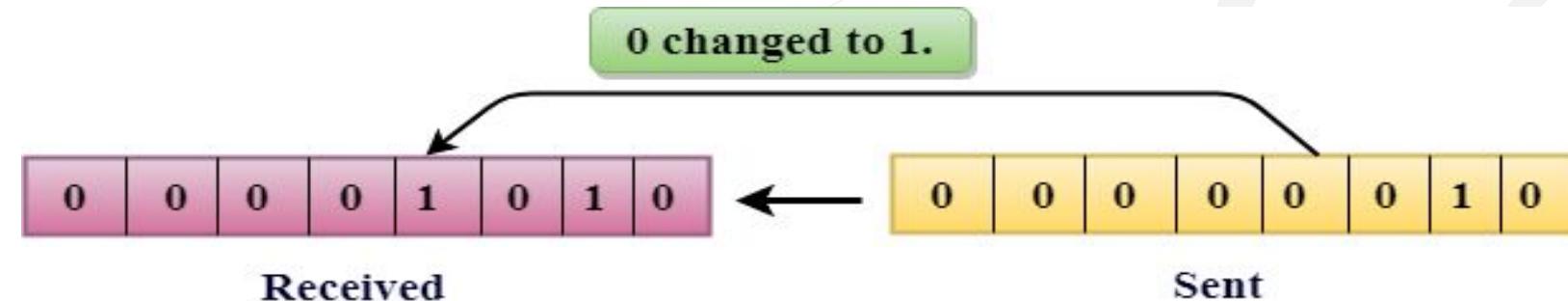
**Q.** Consider a network path P—Q—R between nodes P and R via router Q .Node P sends a file of size  $10^6$  bytes to R via this path by splitting the file into chunks  $10^3$  bytes each. Node P sends these chunks one after the other without any wait time between the successive chunk transmissions. Assume that the size of extra headers added to these chunks is negligible, and that the chunk size is less than the MTU. Each of the links P—Q and Q—R has a bandwidth of  $10^6$  bitwise, and negligible propagation latency. Router Q immediately transmits every packet it receives from P to R. with negligible processing and queueing delays. Router Q can simultaneously receive on link P—Q and transmit on link Q—R. Assume P starts transmitting the chunks at time t=0. Which of the following options gives the time (in seconds, rounded off to 3 decimal places) at which R receives all the chunks of the file? **(Gate 2024,CS) (2 Marks) (MCQ)**

- (a) 8.000
- (b) 8.008
- (c) 15.992
- (d) 16.000

## Types of Errors

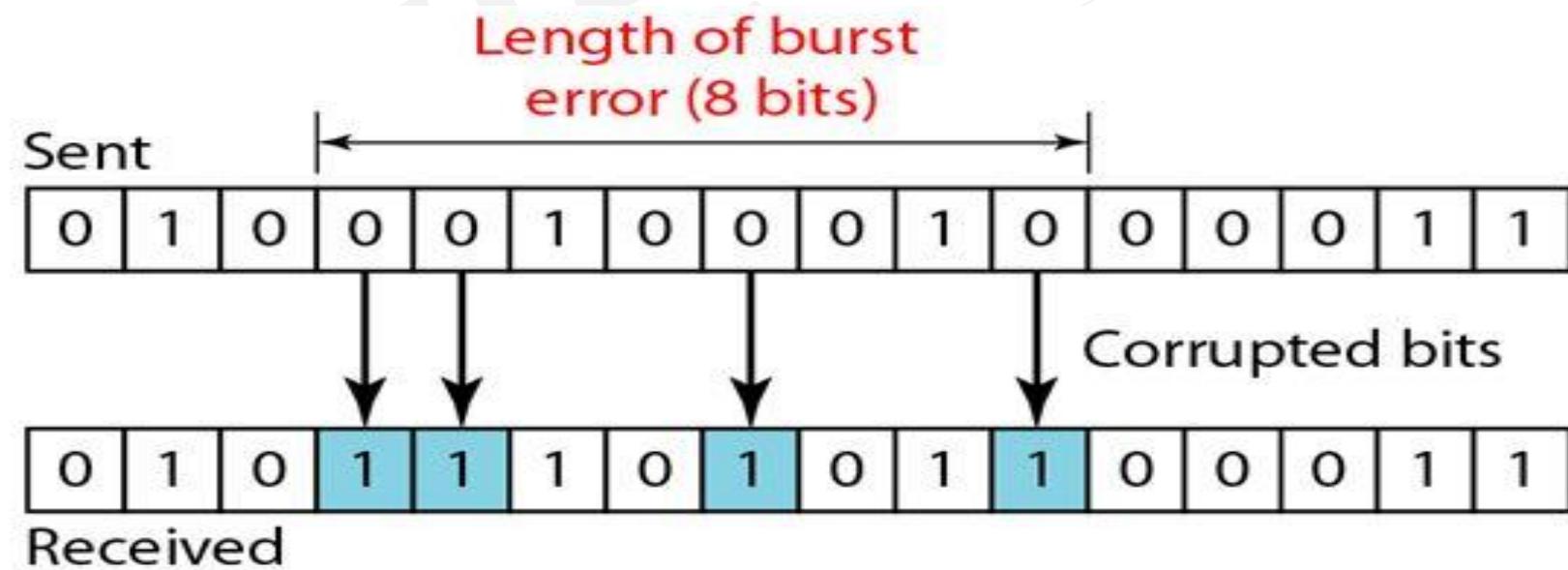
- **Single-Bit Error:**

A single-bit error occurs when only one bit within a data unit (such as a byte or packet) is altered from 0 to 1 or vice versa. This type of error is typically less common and can be easily detected and corrected.



- **Burst Error:**

A burst error involves two or more bits changing from 0 to 1 or vice versa within a data unit. These errors are more complex as they can affect multiple bits, though not necessarily in consecutive order. The length of a burst error is measured from the first corrupted bit to the last, regardless of whether all bits in between are affected.

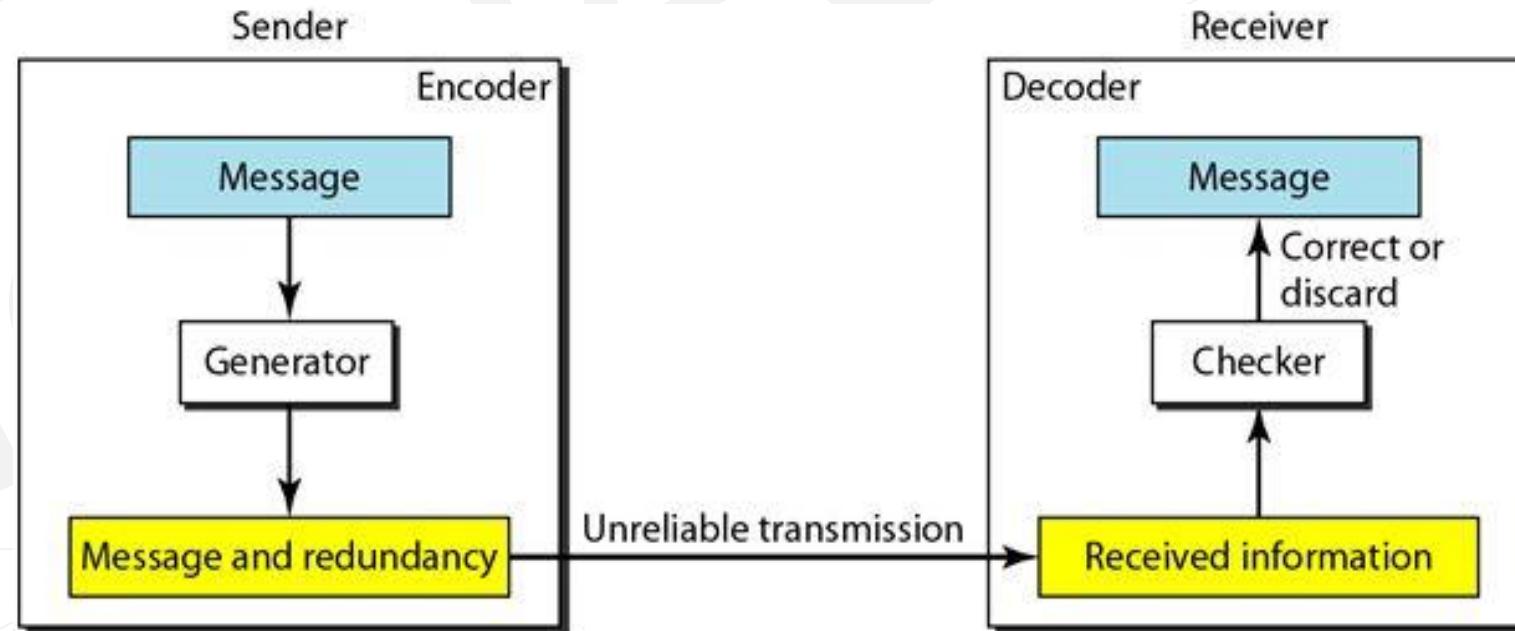


- **Redundancy in Error Detection and Correction:**

- **Redundancy** refers to sending extra bits with data to help detect or correct errors. These redundant bits are added by the sender and removed by the receiver.
- **Error Detection** focuses on identifying if an error has occurred, without needing to know the specific location or number of corrupted bits.
- **Error Correction** goes beyond detection to identify the location and number of errors, which is more complex.

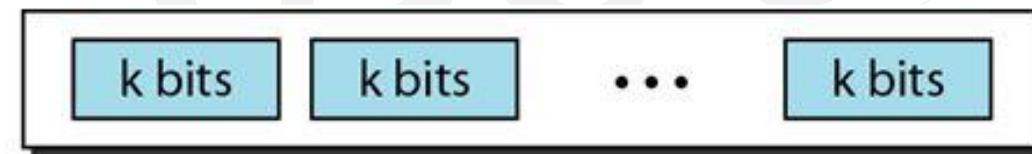
- **Forward Error Correction (FEC) vs. Retransmission:**

- **Forward Error Correction:** The receiver tries to guess the correct data using redundant bits, mainly when errors are small and manageable.
- **Retransmission:** If errors are detected, the receiver requests the sender to resend the message. This process continues until the message is deemed error-free.

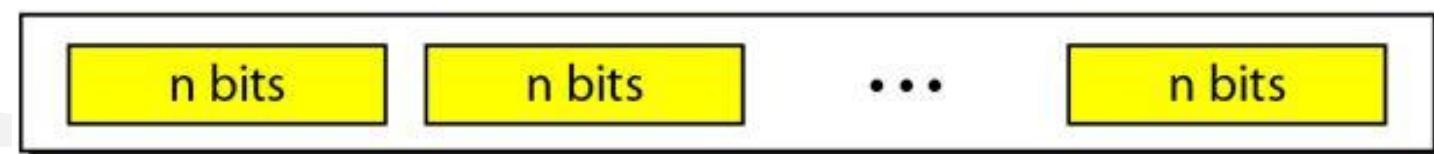


## BLOCK CODING

- In block coding, we divide our message into blocks, each of  $k$  bits, called datawords. We add  $r$  redundant bits to each block to make the length  $n = k + r$ . The resulting  $n$ -bit blocks are called codewords.
- It is important to know that we have a set of datawords, each of size  $k$ , and a set of code words, each of size of  $n$ . With  $k$  bits, we can create a combination of  $2^k$  datawords; with  $n$  bits, we can create a combination of  $2^n$  codewords.
- Since  $n > k$ , the number of possible codewords is larger than the number of possible datawords. The block coding process is one-to-one; the same dataword is always encoded as the same codeword. This means that we have  $2^n - 2^k$  codewords that are not used.



$2^k$  Datawords, each of  $k$  bits



$2^n$  Codewords, each of  $n$  bits (only  $2^k$  of them are valid)

## Error Detection

- How can errors be detected by using block coding? If the following two conditions are met, the receiver can detect a change in the original codeword.
  - The receiver has (or can find) a list of valid codewords.
  - The original codeword has changed to an invalid one.

Datawords	Codewords
00	000
01	011
10	101
11	110

Datawords	Codewords
00	00000
01	01011
10	10101
11	11110

## Hamming Distance

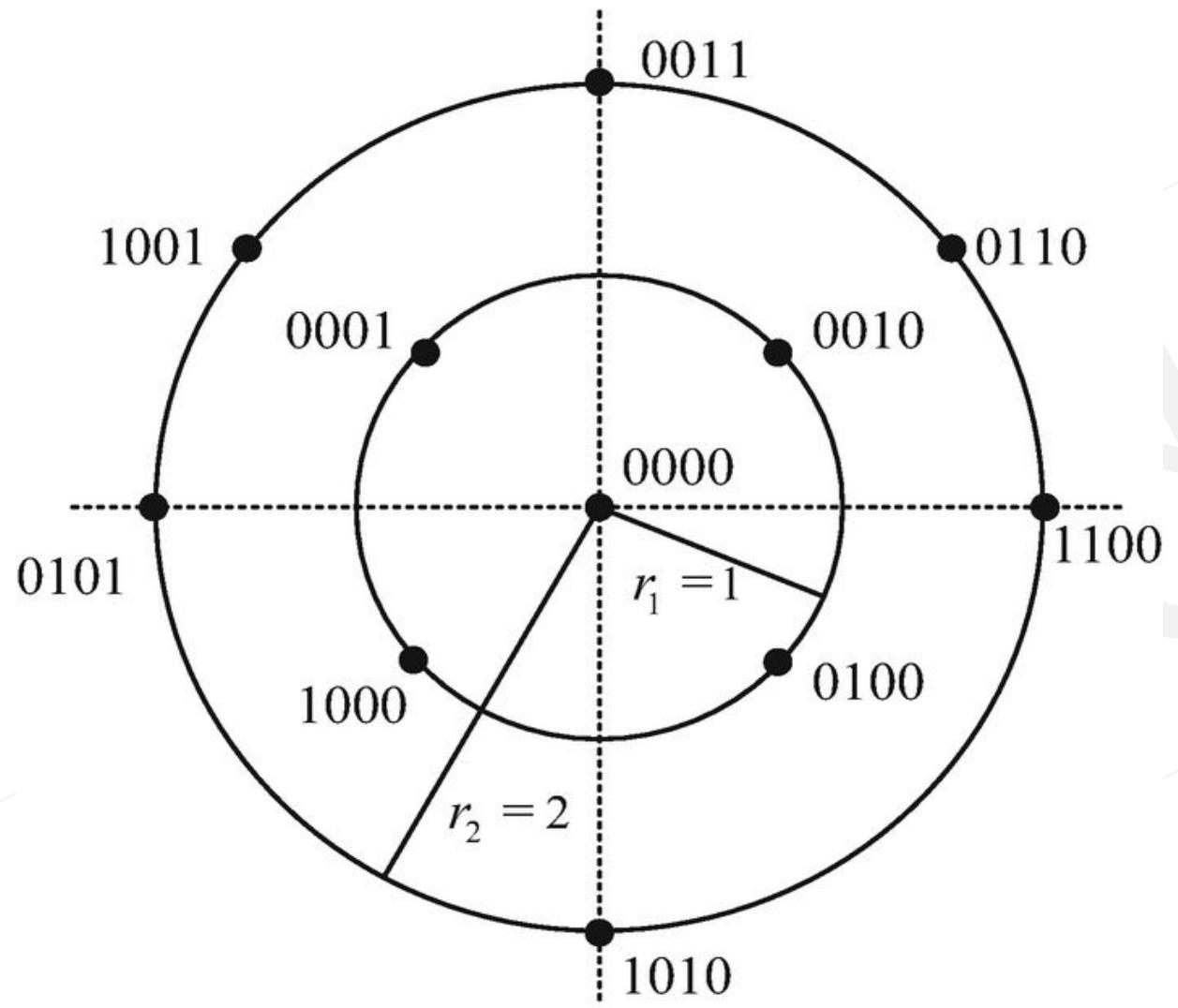
- The Hamming distance between two words (of the same size) is the number of differences between the corresponding bits. We show the Hamming distance between two words  $x$  and  $y$  as  $d(x, y)$ . The Hamming distance can easily be found if we apply the XOR operation on the two words and count the number of 1s in the result.
- The Hamming distance  $d(000, 011)$  is 2 because 000 XOR 011 is 011 (two 1's).
- The Hamming distance  $d(10101, 11110)$  is \_\_\_\_\_ ?
- The minimum Hamming distance is the smallest Hamming distance between all possible pairs.
  - $C_1 = 00000$
  - $C_2 = 01011$
  - $C_3 = 10101$
  - $C_4 = 11110$
- $d(00000, 01011) = 3$      $d(01011, 10101) = 4$      $d(00000, 10101) = 3$
- $d(01011, 11110) = 3$      $d(00000, 11110) = 4$      $d(10101, 11110) = 3$
- The  $d_{\min}$  in this case is 3.

## Coding Scheme Parameters

- Any coding scheme (used in error correction/detection) needs to have at least three essential parameters:
  - **Codeword Size (n)** - The total length of the codeword generated from the data.
  - **Dataword Size (k)** - The length of the original data before adding redundant bits.
  - **Minimum Hamming Distance ( $d_{min}$ )** - The smallest number of bits that need to change to transform one valid codeword into another.
- The coding scheme is represented as  $C(n,k)$ , with an associated minimum Hamming distance  $d_{min}$ . For example, one scheme might be denoted as  $C(3,2)$  with  $d_{min}=2$  while another is  $C(5,2)$  with  $d_{min}=3$ .

Datawords	Codewords
00	000
01	011
10	101
11	110

Datawords	Codewords
00	00000
01	01011
10	10101
11	00110

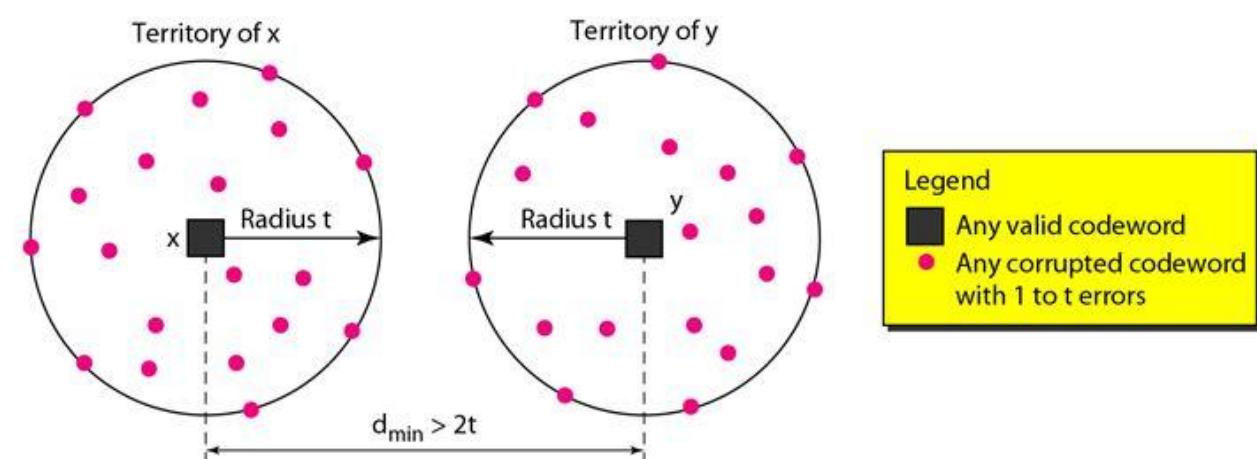
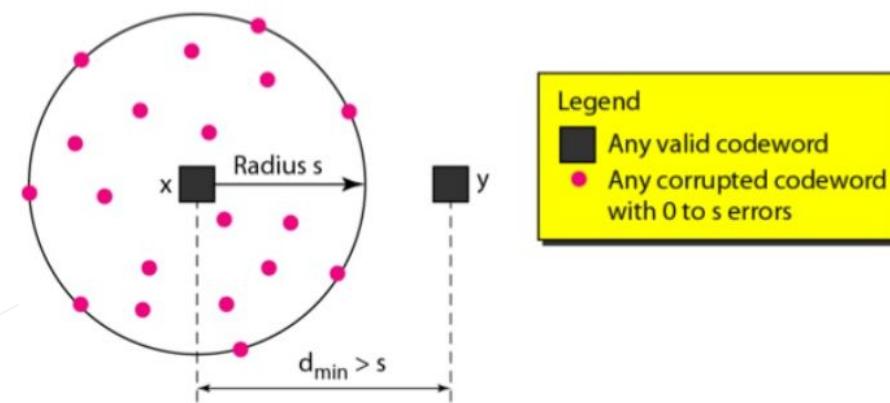


- **Minimum Hamming Distance for Error Detection:**

- The minimum distance  $d_{min}$  needed to detect up to ' $s$ ' errors in transmission should be  $s+1$ . This means that if the distance between valid codewords is at least  $s+1$ , any received codeword with up to ' $s$ ' errors will not match another valid codeword mistakenly.

- **Minimum Distance for Error Correction:**

- To correct up to ' $t$ ' errors, the minimum distance  $d_{min}$  must be at least  $2t+1$ . The concept of territory is used, where each valid codeword is surrounded by a region of radius ' $t$ '. If the received codeword falls within a valid codeword's territory, the receiver can correctly determine the original codeword. This provides reliable correction for up to ' $t$ ' errors.



**Q** Consider a binary code that consists only four valid codewords as given below. a

00000, 01011, 10101, 11110

Lets minimum Hamming distance of code be p and maximum number of erroneous bits that can be corrected by the code be q. The value of p and q are: **(Gate-2017) (2 Marks)**

**(A)**  $p = 3$  and  $q = 1$

**(B)**  $p = 3$  and  $q = 2$

**(C)**  $p = 4$  and  $q = 1$

**(D)**  $p = 4$  and  $q = 2$

**Q** An error correcting code has the following code words:

00000000, 00001111, 01010101, 10101010, 11110000.

What is the maximum number of bit errors that can be corrected? **(Gate-2007) (2 Marks)**

**(A) 0**

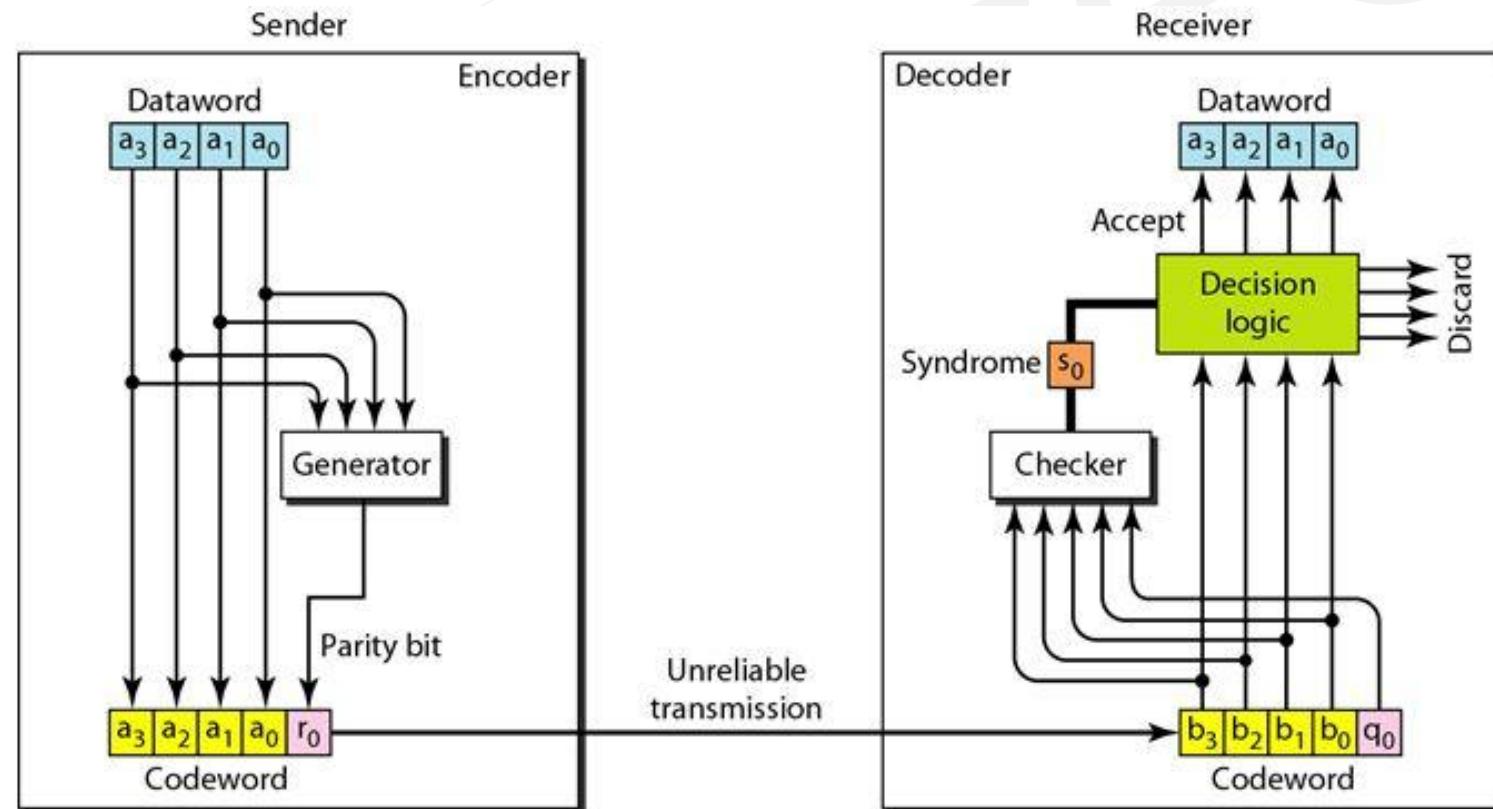
**(B) 1**

**(C) 2**

**(D) 3**

# Simple Parity-Check Code

- A simple parity-check code adds a single parity bit to a dataword of 'k' bits to create an 'n' bit codeword, where  $n=k+1$ . The parity bit is chosen to make the total number of 1's in the codeword even (or odd in some cases). This technique is designed to detect single-bit errors and can only detect an odd number of errors. The minimum Hamming distance ( $d_{min}$ ) for this code is 2.



- A better approach is the two-dimensional parity check. In this method, the dataword is organized in a table (rows and columns). The data to be sent, five 7-bit bytes, are put in separate rows. For each row and each column, 1 parity-check bit is calculated.
- The whole table is then sent to the receiver, which finds the syndrome for each row and each column., the two-dimensional parity check can detect up to three errors that occur anywhere in the table (arrows point to the locations of the created nonzero syndromes). However, errors affecting 4 bits may not be detected.

Original data	11001110	10111010	01110010	01010010	
Column Parities	0 1 0 1 0 1 0				
	1	1	0	1	
Row Parities					
1 1 0 0 1 1 1 0	1				
1 0 1 1 1 0 1 0		1			
0 1 1 1 0 0 1 0			0		
0 1 0 1 0 0 1 0				1	
0 1 0 1 0 1 0					1

**Q** Data transmitted on a link uses the following 2D parity scheme for error detection:

Each sequence of 28 bits is arranged in a  $4 \times 7$  matrix (rows  $r_0$  through  $r_3$ , and columns  $d_7$  through  $d_1$ ) and is padded with a column  $d_0$  and row  $r_4$  of parity bits computed using the Even parity scheme. Each bit of column  $d_0$  (respectively, row  $r_4$ ) gives the parity of the corresponding row (respectively, column). These 40 bits are transmitted over the data link. The table shows data received by a receiver and has  $n$  corrupted bits. What is the minimum possible value of  $n$ ? (Gate-2008) (2 Marks)

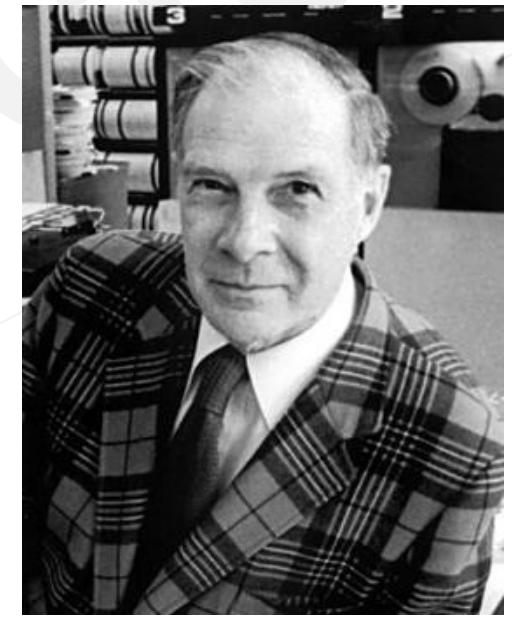
- (A) 1                    (B) 2                    (C) 3                    (D) 4

	$d_7$	$d_6$	$d_5$	$d_4$	$d_3$	$d_2$	$d_1$	$d_0$
$r_0$	0	1	0	1	0	0	1	1
$r_1$	1	1	0	0	1	1	1	0
$r_2$	0	0	0	1	0	1	0	0
$r_3$	0	1	1	0	1	0	1	0
$r_4$	1	1	0	0	0	1	1	0

## Hamming Codes

- Now let us discuss a category of error-correcting codes called Hamming codes. These codes were originally designed with  $d_{min} = 3$ , which means that they can detect up to two errors or correct one single error.
- Although there are some Hamming codes that can correct more than one error, our discussion focuses on the single-bit error-correcting code.

Bit position		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Encoded data bits		p1	p2	d1	p4	d2	d3	d4	p8	d5	d6	d7	d8	d9	d10	d11	p16	d12	d13	d14	d15
Parity bit coverage	p1	X		X		X		X		X		X		X		X		X		X	
	p2		X	X		X	X		X	X		X	X		X		X	X			
	p4			X	X	X	X			X	X	X	X						X		
	p8					X	X	X	X	X	X	X	X								
	p16																X	X	X	X	
																				X	



Richard Hamming

- First let us find the relationship between  $n$  and  $k$  in a Hamming code. The values of  $n$  and  $k$  are then calculated from  $n = k + 2^r - r - 1$ .

- For example, if  $m = 3$ , then  $n = 7$  and  $k = 4$ . This is a Hamming code  $C(7, 4)$  with  $d_{min} = 3$ . shows the datawords and codewords for this code.
- Position of parity bits  $2^0, 2^1, 2^2, \dots, 2^n$
- If we use even parity then
  - For parity bit  $P_1$  we check position 1, 3, 5, 7 (take 1, leave 1)(which have 1 at  $2^0$ )
  - For parity bit  $P_2$  we check position 2, 3, 6, 7 (take 2, leave 2)(which have 1 at  $2^1$ )
  - For parity bit  $P_4$  we check position 4, 5, 6, 7 (take 4, leave 4)(which have 1 at  $2^2$ )

7	6	5	4	3	2	1
111	110	101	100	011	010	001
$D_4$	$D_3$	$D_2$	$P_4$	$D_1$	$P_2$	$P_1$

**Q** Assume that a 12-bit Hamming codeword consisting of 8-bit data and 4 check bits is  $d_8 d_7 d_6 d_5 c_8 d_4 d_3 d_2 c_4 d_1 c_2 c_1$ , where the data bits and the check bits are given in the following tables:

Data bits							
$d_8$	$d_7$	$d_6$	$d_5$	$d_4$	$d_3$	$d_2$	$d_1$
1	1	0	x	0	1	0	1

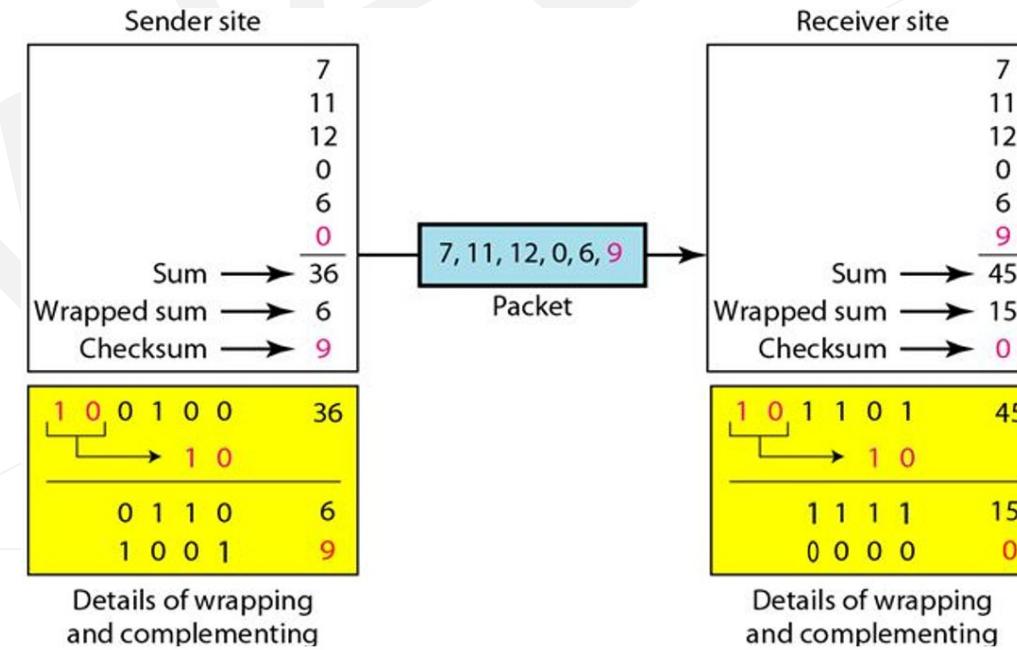
Check bits			
$c_8$	$c_4$	$c_2$	$c_1$
y	0	1	0

Which one of the following choices gives the correct values of x and y? (GATE 2021) (2 MARKS)

- (a) x is 0 and y is 0
- (b) x is 0 and y is 1
- (c) x is 1 and y is 0
- (d) x is 1 and y is 1

# CHECKSUM

- The checksum is a technique used for error detection. It involves adding a sum of data items and sending it along with the data. At the receiver's end, the sum is recalculated and compared with the sent sum.
- Process:** Suppose we send five 4-bit numbers like (7, 11, 12, 0, 6). The checksum is the sum of these numbers (in this case, 36). The sender transmits (7, 11, 12, 0, 6, 36) to the receiver.
- Verification:** The receiver adds the received numbers. If the calculated sum matches the received checksum, the data is accepted as error-free. Otherwise, the receiver identifies an error.
- Complement Approach**
  - Using Negative (Complement) of Sum:** To simplify detection, the sender can transmit the complement of the sum (e.g., -36). The receiver sums all received values, and if the total is zero, the data is considered correct.
  - One's Complement:** One's complement arithmetic is used to manage negative values. The technique involves inverting all bits and adding extra bits that exceed the data length.



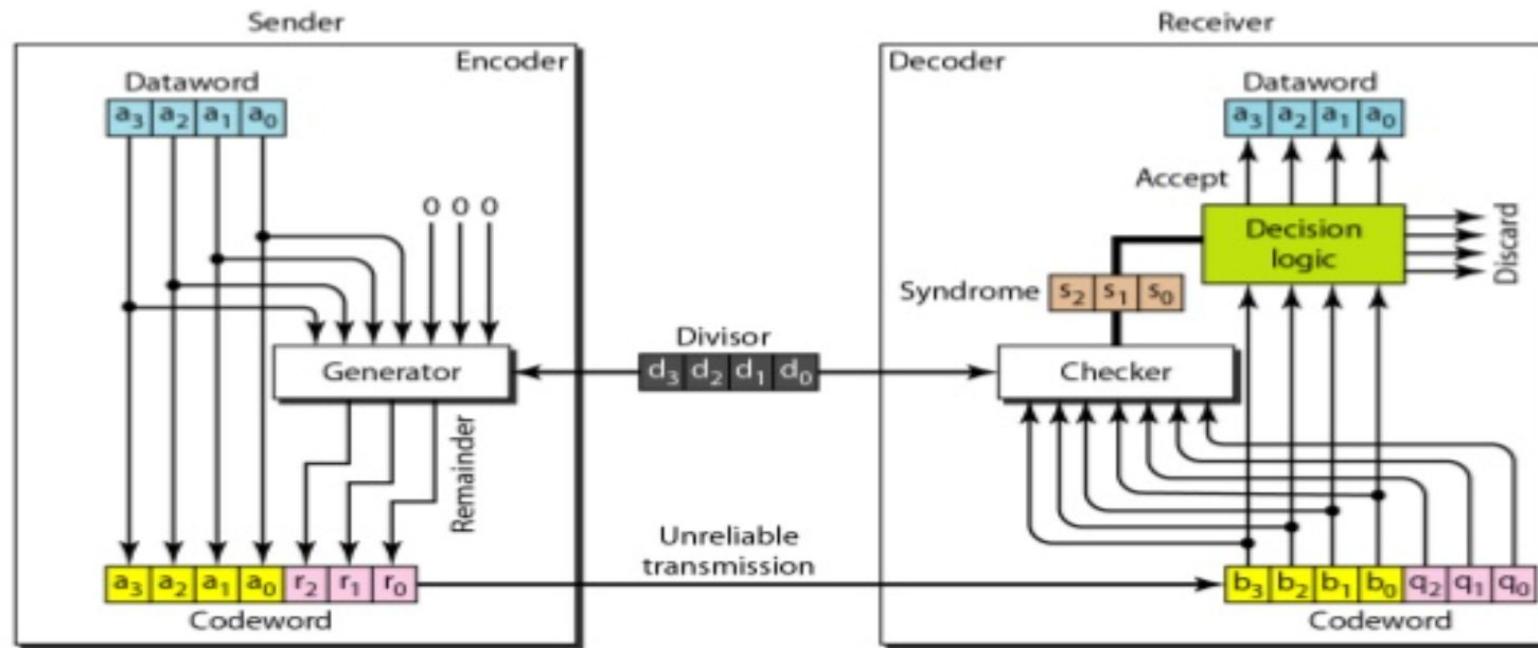
## Cyclic Codes and Cyclic Redundancy Check (CRC)

- **Cyclic Codes:**

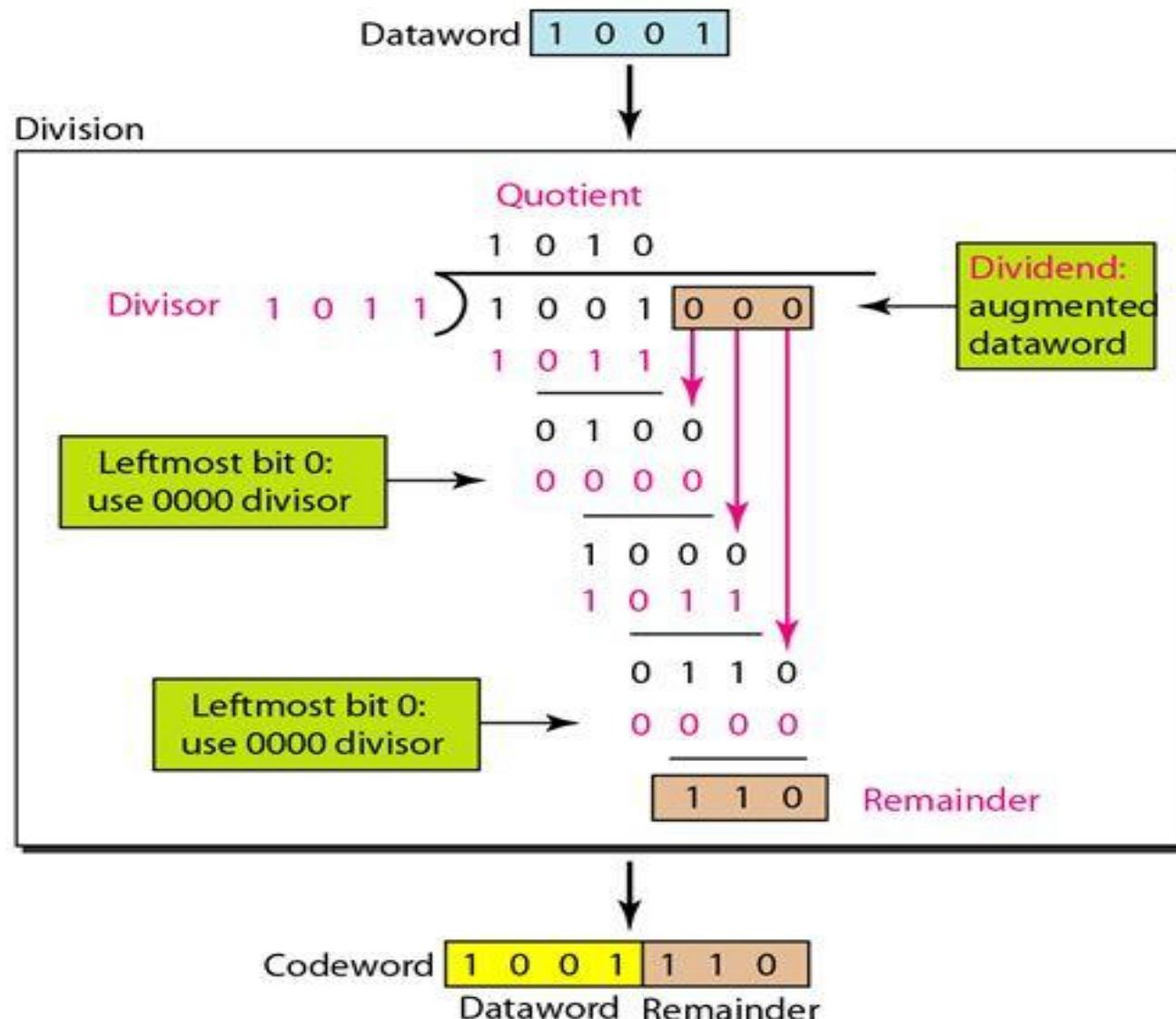
- A type of linear block code with a special property where if a codeword is cyclically shifted (rotated), the resulting word remains a valid codeword. **Example:** If the codeword is 1011000, a cyclic left shift would yield 0110001, which is also a valid codeword.

- **Cyclic Redundancy Check (CRC):**

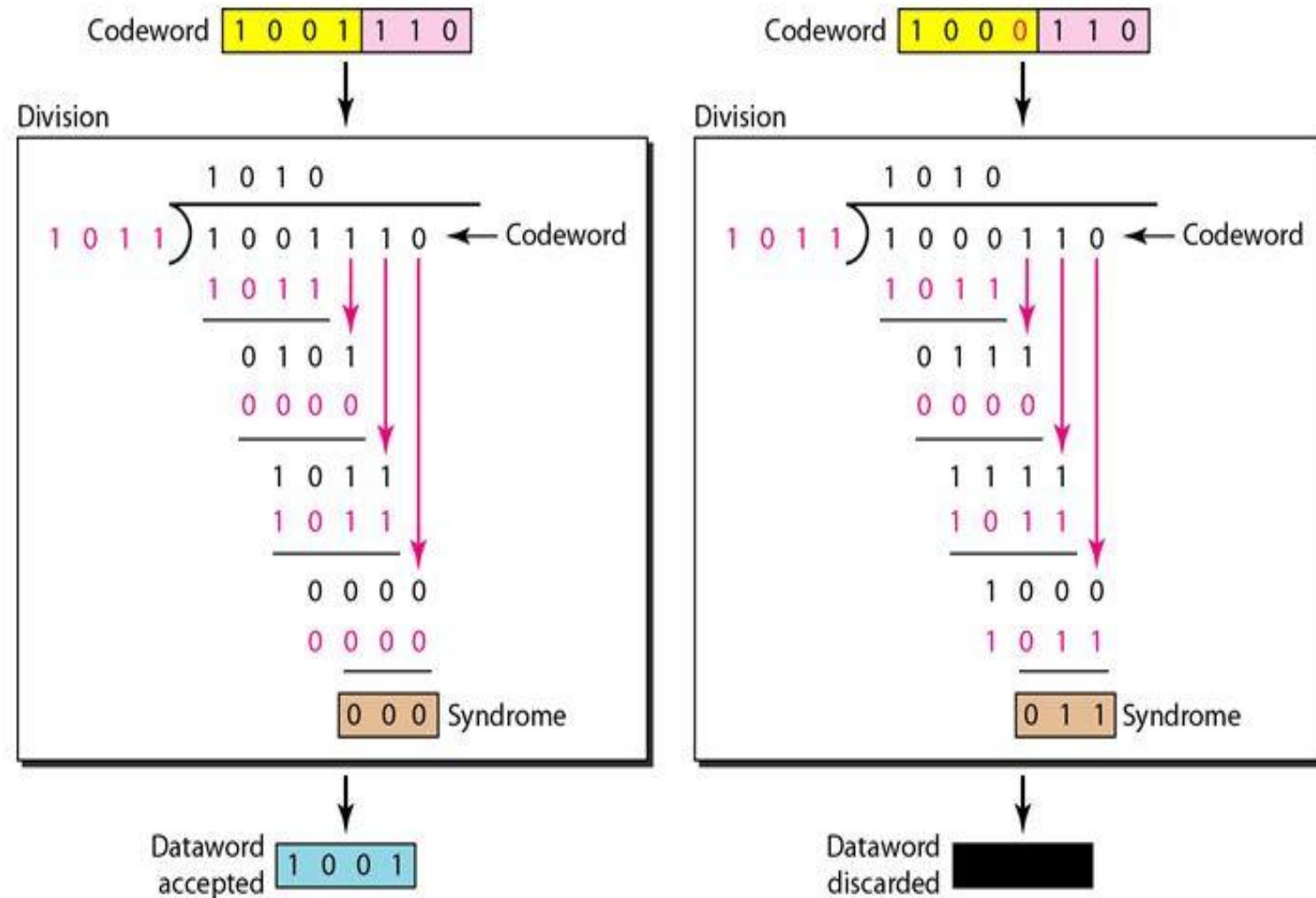
- **Purpose:** CRC is used to detect errors in data transmission by creating special codewords.
- **Process:**
  - The original dataword has  $k$  bits, and the resulting codeword has  $n$  bits ( $n > k$ ).
  - Extra  $n - k$  bits (usually zeros) are added to the dataword to form an extended  $n$ -bit word.
  - A **generator** with a predefined divisor of size  $n - k + 1$  bits is used to divide the extended dataword. This is done using **modulo-2 division**.
  - The **remainder** from the division is appended to the original dataword to create the final codeword.



# Encoder

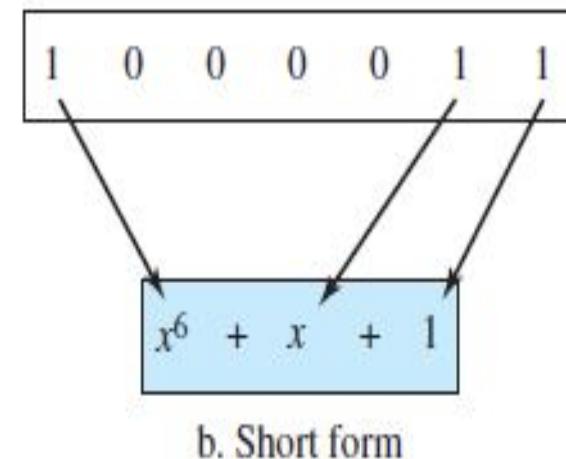
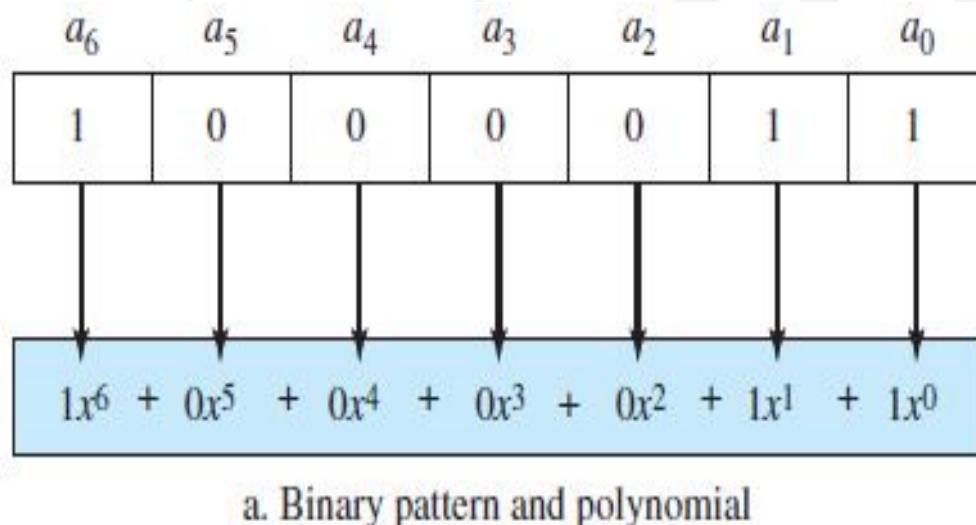


- The decoder receives the possibly corrupted codeword. A copy of all n bits is fed to the checker which is a replica of the generator.
- The remainder produced by the checker is a syndrome of  $n - k$  (3 here) bits, which is fed to the decision logic analyzer. The analyzer has a simple function. If the syndrome bits are all as, the 4 leftmost bits of the codeword are accepted as the dataword (interpreted as no error); otherwise, the 4 bits are discarded (error).



# Polynomials

- **Polynomials Representation:** A sequence of 0s and 1s can be represented as a polynomial. The power in each polynomial term indicates the bit's position, and the coefficient (either 0 or 1) reflects the value of the bit.
- **Advantages:** This representation allows large binary patterns to be simplified into shorter polynomial terms.
- **Degree of a Polynomial:** The degree is the highest power in the polynomial. For example, the degree of  $x^6+x+1$  is 6. The degree is always one less than the number of bits in the binary pattern, so a 7-bit pattern would have a polynomial with a degree of 6.

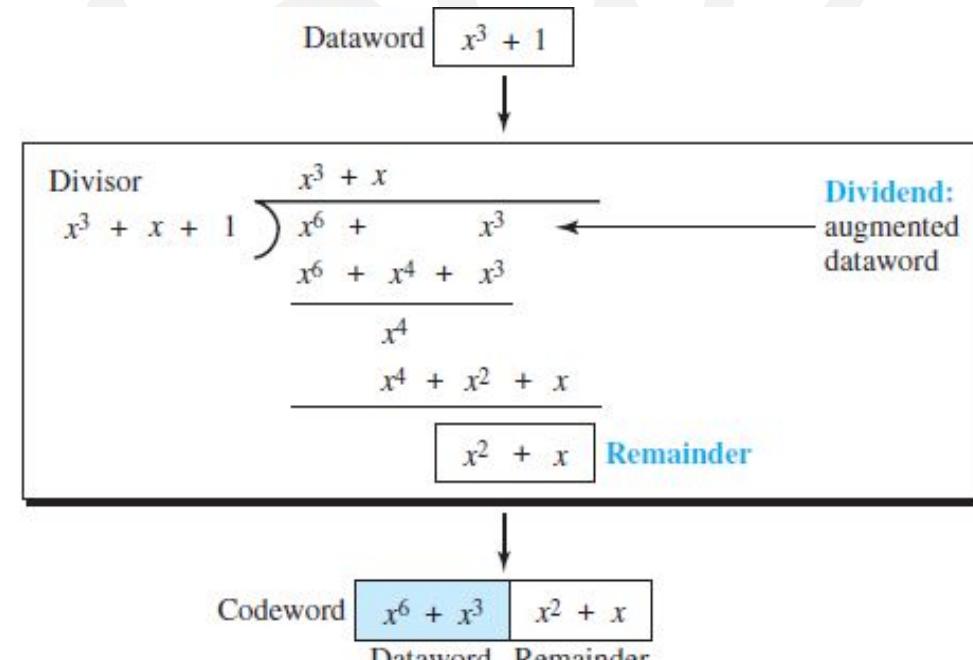


## Adding and Subtracting Polynomials

- **Adding and Subtracting Polynomials:** When adding or subtracting polynomials, identical terms cancel out. For example, adding  $x^5+x^4+x^2$  and  $x^6+x^4+x^2$  results in  $x^6+x^5$ , as identical terms cancel out. Similarly, if a term appears an odd number of times, it remains in the result.
- **Multiplying Terms:** The powers are simply added. For example,  $x^3 \times x^4$  becomes  $x^7$ .
- **Dividing Terms:** The power of the denominator is subtracted from the numerator. For example,  $x^5/x^2$  results in  $x^3$ .
- **Shifting:**
  - **Left Shift:** Multiplying by  $2^n$  (or  $x^n$  in polynomial notation), e.g., shifting 10011 left by 3 bits becomes 10011000.
  - **Right Shift:** Dividing by  $2^n$  (or  $x^n$ ), e.g., shifting 10011 right by 3 bits becomes 10.

## Cyclic Code Encoder Using Polynomials

- The dataword 1001 is represented as  $x^3 + 1$ . The divisor 1011 is represented as  $x^3 + x + 1$ .
- To find the augmented dataword, we have left-shifted the dataword 3 bits (multiplying by  $x^3$ ).
- The result is  $x^6 + x^3$ . Division is straightforward.
- We divide the first term of the dividend,  $x^6$ , by the first term of the divisor,  $x^3$ .
- The first term of the quotient is then  $x^6/x^3$ , or  $x^3$ . Then we multiply  $x^3$  by the divisor and subtract (according to our previous definition of subtraction) the result from the dividend.
- The result is  $x^4$ , with a degree greater than the divisor's degree; we continue to divide until the degree of the remainder is less than the degree of the divisor.



**Q** Consider the cyclic redundancy check (CRC) based error detecting scheme having the generator polynomial  $X^3 + X + 1$ . Suppose the message  $m_4m_3m_2m_1m_0 = 11000$  is to be transmitted. Check bits  $c_2c_1c_0$  are appended at the end of the message by the transmitter using the above CRC scheme. The transmitted bit string is denoted by  $m_4m_3m_3m_1m_0c_2c_1c_0$ . The value of the checkbit sequence  $c_2c_1c_0$  is **(GATE 2021) (2 MARKS)**

- (a) 101
- (b) 110
- (c) 100
- (d) 111

**Q** A computer network uses polynomials for error checking with 8 bits as information bits and uses  $x^3 + x + 1$  as the generator polynomial to generate the check bits. In this network, the message 01011011 is transmitted as **(Gate-2017) (2 Marks)**

**(a)** 01011011010

**(b)** 01011011011

**(c)** 01011011101

**(d)** 01011011100

**Q** The message 11001001 is to be transmitted using the CRC polynomial  $x^3+1$  to protect it from errors. The message that should be transmitted is: **(Gate-2007) (2 Marks)**

- a) 11001001000
- b) 11001001011
- c) 11001010
- d) 110010010011

**Q** Consider the following message  $M = 1010001101$ . The cyclic redundancy check (CRC) for this message using the divisor polynomial  $x^5 + x^4 + x^2 + 1$  is: **(Gate-2005) (2 Marks)**

**(A) 01110**

**(B) 0101**

**(C) 10101**

**(D) 10110**

**Q.** Suppose a 5-bit message is transmitted from a source to a destination through a noisy channel.

The probability that a bit of the message gets flipped during transmission is 0.01.

Flipping of each bit is independent of one another.

The probability that the message is delivered error-free to the destination is \_\_\_\_\_

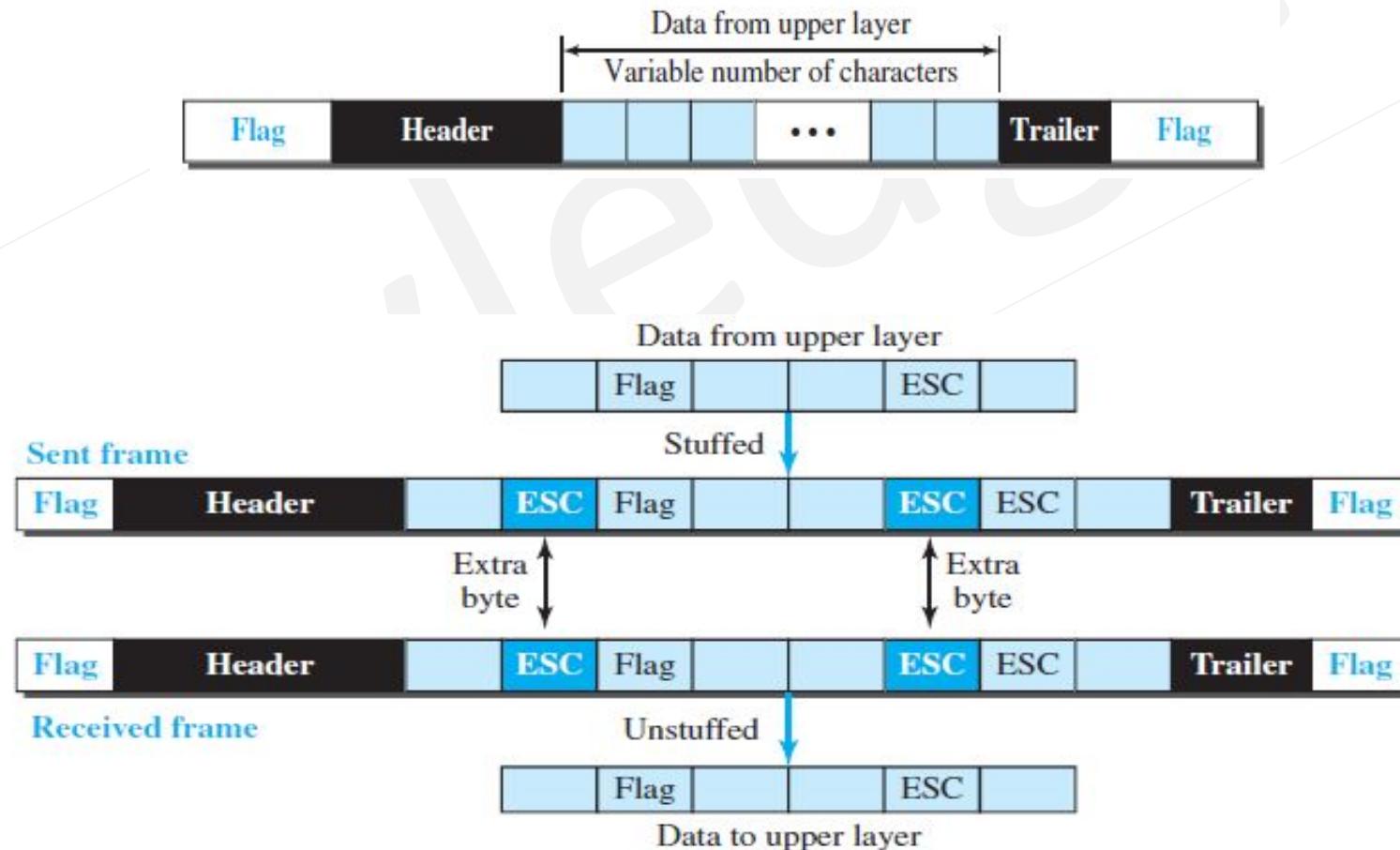
(rounded off to three decimal places)(Gate 2025)

# FRAMING

- In the Data Link Layer, data is packed into **frames** to distinguish one from another. Rather than packing an entire message into a single large frame, it is divided into smaller frames. This improves error control since a single-bit error in a large frame would require retransmitting the entire message, while in smaller frames, only the affected frame needs retransmission.
- **Types of Framing:**
  - **Fixed-Size Framing:**
    - No need for explicit frame boundaries as the frame size itself acts as a delimiter.
  - **Variable-Size Framing:**
    - Boundaries of frames need definition.
    - Two approaches:
      - **Character-Oriented Approach:** Uses special characters to indicate the start and end of frames.
      - **Bit-Oriented Approach:** Utilizes specific bit patterns to mark frame boundaries.

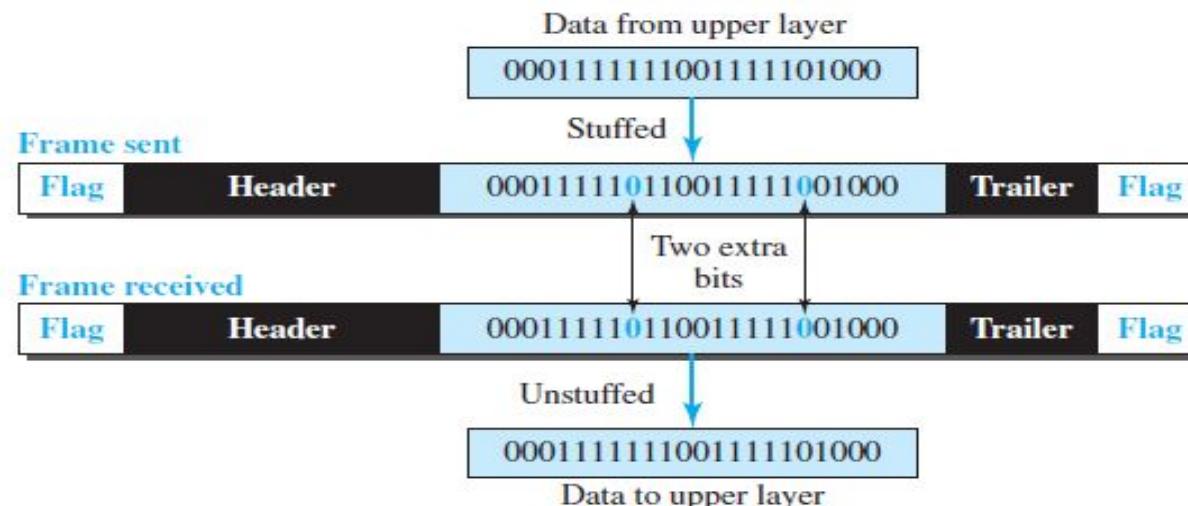
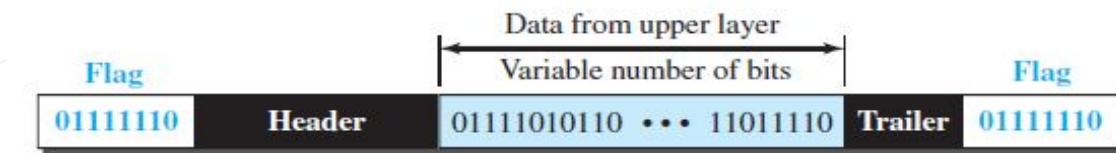
## Character-Oriented Protocols

- **Data Composition:** Carries 8-bit characters (like ASCII). It includes a header, trailer, and a frame-delimiting flag (8 bits) to indicate the start and end of a frame.
- **Issue with Flags:** If the flag pattern appears within the data, the receiver might mistakenly consider it as the frame boundary.
- **Solution - Byte Stuffing:** Adds an escape character (ESC) before any data pattern that matches the flag. The ESC character is also added to genuine ESC sequences to differentiate them.
- **Challenge with Unicode:** As modern data uses wider characters (16-bit, 32-bit), character-oriented protocols face compatibility issues. The industry is shifting towards *bit-oriented protocols* to address this.



## Bit-Oriented Protocols

- **Bit-Oriented Protocol Definition:** Data is treated as a sequence of bits, which can represent various types of information like text, graphics, or video. A special 8-bit pattern (01111110) is used as a delimiter to mark the start and end of frames.
- **Challenge with Delimiters:** If this delimiter pattern (01111110) appears in the data, it could be misinterpreted as a frame boundary.
- **Bit Stuffing Solution:** To avoid confusion, a technique called "bit stuffing" is used. Whenever five consecutive 1s are detected in the data, a 0 is inserted. This ensures that the delimiter pattern does not accidentally appear within the data. Upon receiving, the extra bit is removed to restore the original data.



**Q** A bit-stuffing based framing protocol uses an 8-bit delimiter pattern of 01111110. If the output bit-string after stuffing is 01111100101, then the input bit-string is **(Gate-2014) (1 Marks)**

**A) 0111110100**

**B) 0111110101**

**C) 0111111101**

**D) 0111111111**

**Q** In a data link protocol, the frame delimiter flag is given by 0111. Assuming that bit stuffing is employed, the transmitter sends the data sequence 01110110 as **(Gate-2004) (2 Marks)**

- (A)** 01101011
- (B)** 011010110
- (C)** 011101100
- (D)** 0110101100

# Ethernet

- **Ethernet Overview:** A family of networking technologies mainly used in Local Area Networks (LANs) and Metropolitan Area Networks (MANs).
- **Introduction and Standardization:** First introduced commercially in 1980 and standardized in 1983 as IEEE 802.3. It has evolved to support higher speeds and longer distances.
- **Original and Newer Variants:** Initially, 10BASE5 Ethernet used coaxial cables as a shared medium. Modern variants use twisted pair and fiber optic cables with hubs or switches.
- **Speed Evolution:** Data rates have increased from 2.94 Mbps to 100 Gbps.
- **Frames:** Ethernet divides data into frames that include source and destination addresses and error-checking data. Damaged frames can be detected and discarded, with higher-layer protocols handling retransmissions.
- **Services in the OSI Model:** Ethernet provides services up to and including the data link layer in the OSI model.
- **Compatibility and Influence:** Retains good backward compatibility and has influenced other protocols with its 48-bit MAC addresses and frame formats.
- **Topology and Encoding:** Uses Bus Topology and Manchester encoding. Acknowledgements are not built-in, but can be sent as data packets if needed.
- **Alternatives:** Competes with wireless protocols like Wi-Fi (standardized as IEEE 802.11).

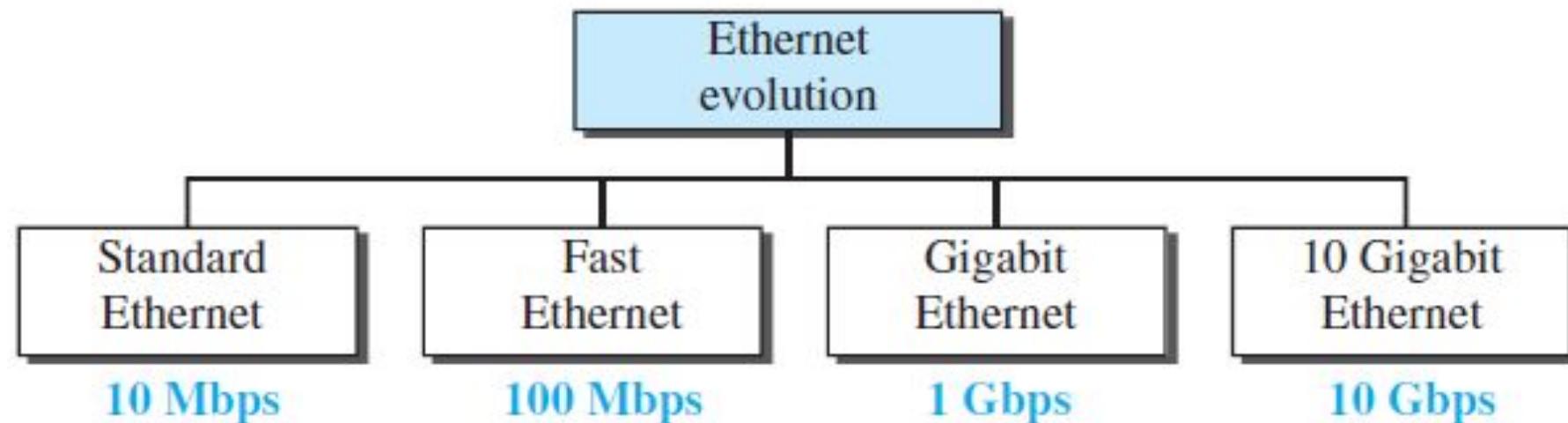


Protocol	MTU
Hyperchannel	65,535
Token Ring (16 Mbps)	17,914
Token Ring (4 Mbps)	4,464
FDDI	4,352
Ethernet	1,500
X.25	576
PPP	296

## STANDARD ETHERNET

### Connectionless and Unreliable Service:

- Each frame sent is independent of the previous or next frame. Ethernet has no connection establishment or connection termination phases.
- Ethernet is also unreliable, if a frame is corrupted during transmission and the receiver finds out about the corruption, the receiver drops the frame silently.
- In case of requirement ack can be sent separately at data packets.



- **Preamble**

- It is a 7-byte field that contains a pattern of alternating 0's and 1's.
- It alerts the stations that a frame is going to start.
- It also enables the sender and receiver to establish bit synchronization.
- The Preamble field is added at the physical layer.

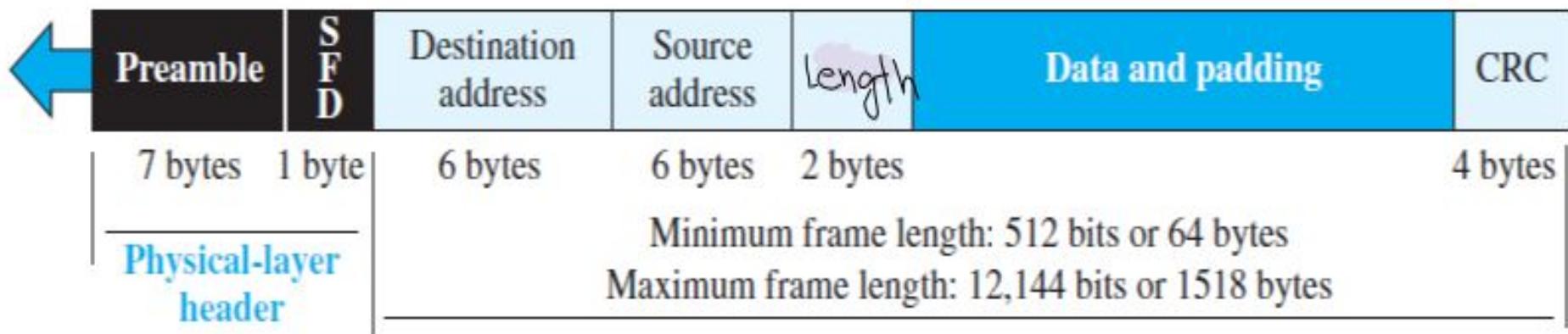
- **Start Frame Delimiter (SFD)**

- It is a 1-byte field which is always set to 10101011.
- The last two bits "11" indicate the end of Start Frame Delimiter and marks the beginning of the frame.
- The SFD field is also added at the physical layer.
- Initially only SFD was there Preamble was added later

**Preamble:** 56 bits of alternating 1s and 0s

**SFD:** Start frame delimiter, flag (10101011)

Minimum payload length: 46 bytes  
Maximum payload length: 1500 bytes

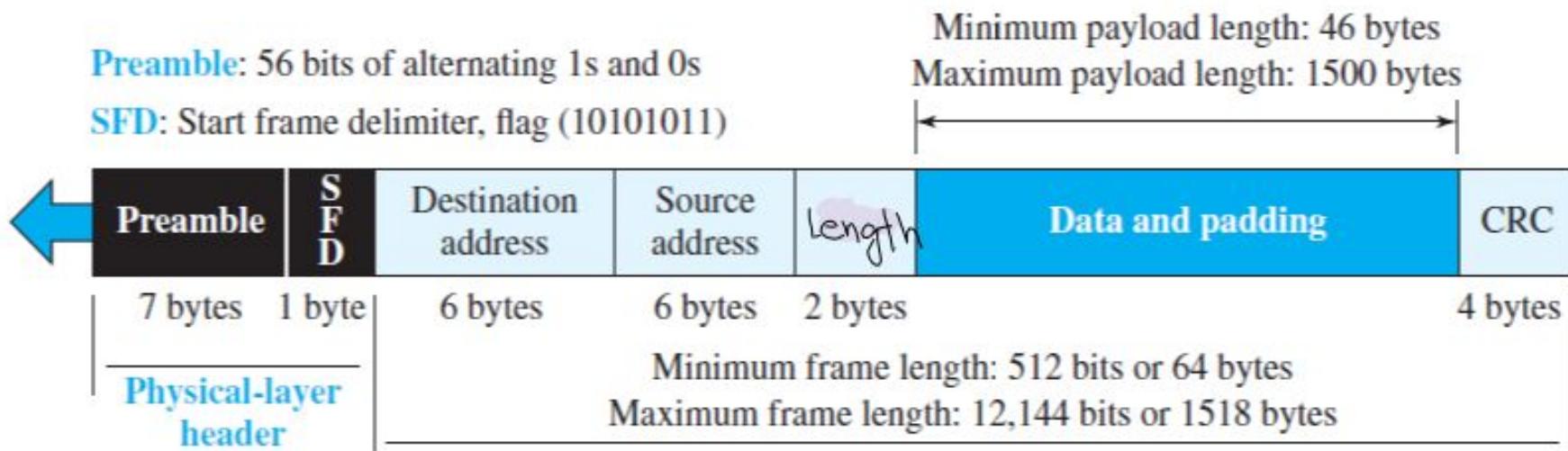


## • Destination Address

- It is a 6-byte field that contains the MAC address of the destination for which the data is destined. e.g. 2D : 8A : 7B : C5
- MAC address is present on NIC card.
- MAC address can be of three types
  - Unicast-LSB of the first byte is 0 (Source address will always be unicast)
  - Multicast- LSB of the first byte is 1, if we want to send, repeated messages to a group of station on the network then we can group these stations together and can assign a Multicast address to the group.
  - Broadcast-all bit are assigned 1's

## • Source Address

- It is a 6-byte field that contains the MAC address of the source which is sending the data.
- Using some protocol, we can broadcast a request message asking MAC address of every other station in the network.

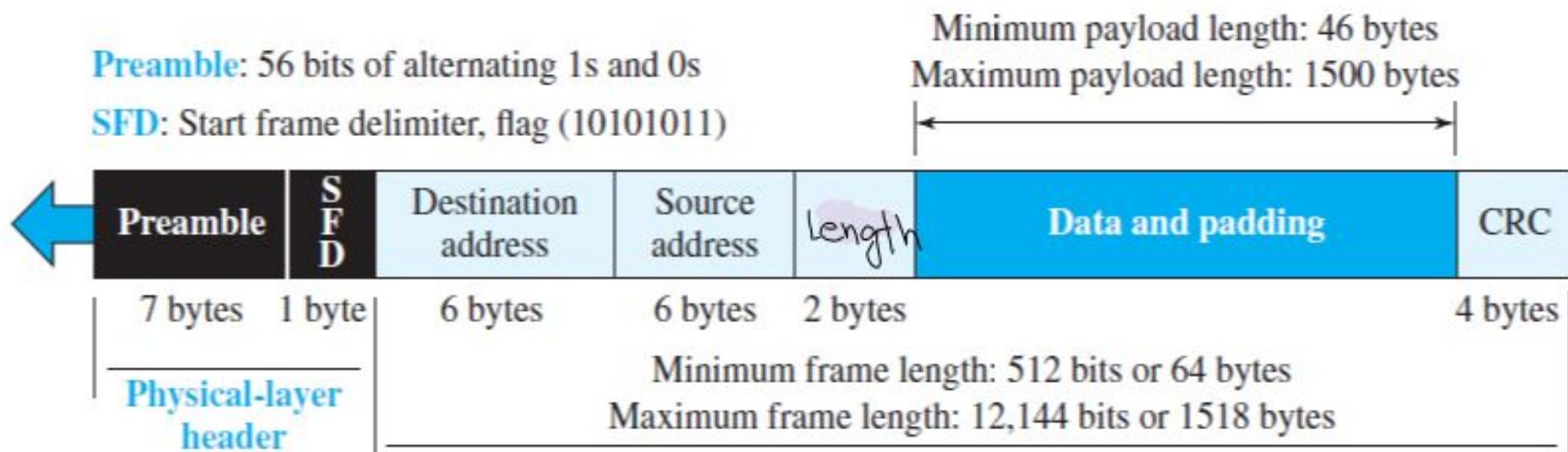


- **Length**

- As ethernet use variable size frames therefore we need Length field
- It is a 16-bit field.

- **Data**

- It is a variable length field which contains the actual data, also called as a payload field.
- The length of this field lies in the range [46 bytes, 1500 bytes], i.e. in an Ethernet frame, minimum data has to be 46 bytes and maximum data can be 1500 bytes.
- If it is less than 46 bytes, it needs to be padded with extra 0s.
- If more than 1500 bytes, it should be fragmented and encapsulated in more than one frame.



- Data

- The minimum length restriction is required for the correct operation of CSMA/CD, value in general come to be 64B,  $64-6-6-2-4 = 46B$ .
- The maximum length restriction has two historical reasons:
  - Memory was very expensive when Ethernet was designed; a maximum length restriction helped to reduce the size of the buffer.
  - The maximum length restriction prevents one station from monopolizing the shared medium, blocking other stations that have data to send.

Preamble: 56 bits of alternating 1s and 0s

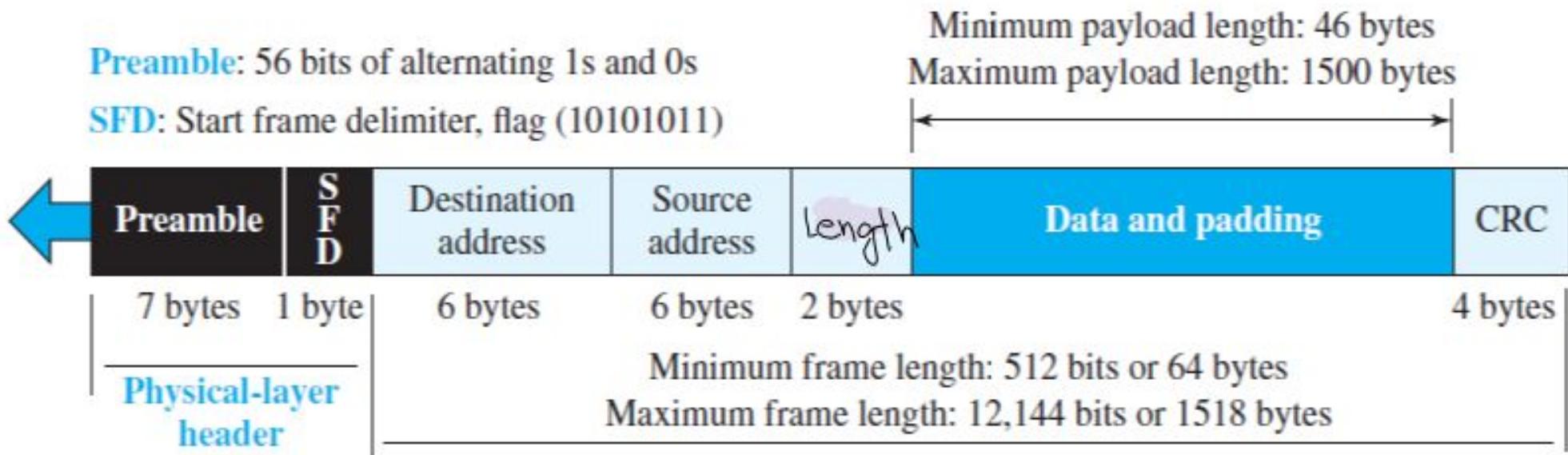
SFD: Start frame delimiter, flag (10101011)

Minimum payload length: 46 bytes  
Maximum payload length: 1500 bytes



- **CRC**

- The last field contains error detection information.
- At the time of transmission CRC is calculated so it is in the last.
- It is a 4-Byte field



- Ethernet is very simple easy to install and reconfigure.
- Should not be used with real time applications, because of collision possibility.
- if amount of data is very less then also should not be used.
- No idea of priority (Server suffer).

**Q In an Ethernet local area network, which one of the following statements is TRUE? (Gate-2016) (2 Marks)**

- (A) A station stops to sense the channel once it starts transmitting a frame
- (B) The purpose of the jamming signal is to pad the frames that are smaller than the minimum frame size
- (c) A station continues to transmit the packet even after the collision is detected
- (D) The exponential back off mechanism reduces the probability of collision on retransmission

**Q** Determine the maximum length of the cable (in km) for transmitting data at a rate of 500 Mbps in an Ethernet LAN with frames of size 10,000 bits. Assume the signal speed in the cable to be 2,00,000 km/s. **(Gate-2013) (2 Marks)**

- (A) 1                    (B) 2                    (C) 2.5                    (D) 5

**Q** Suppose the round-trip propagation delay for a 10 Mbps Ethernet having 48-bit jamming signal is 46.4 ms. The minimum frame size is **(Gate-2005) (2 Marks)**

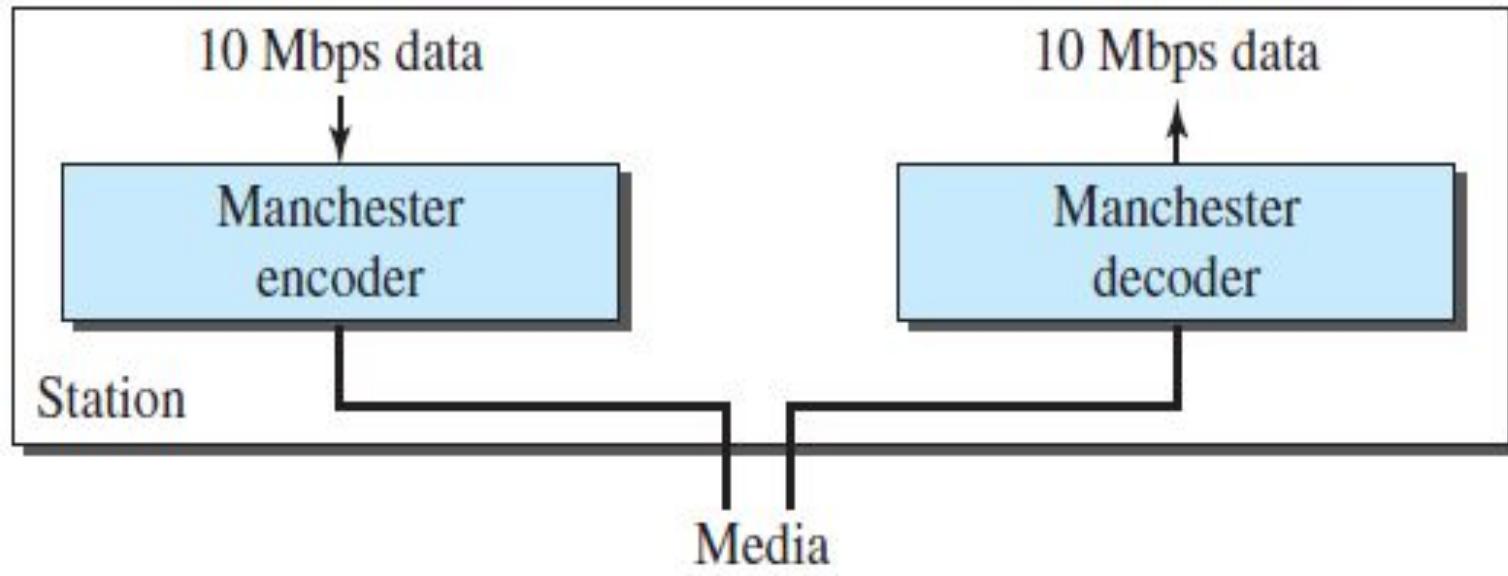
- (A) 94
- (B) 416
- (C) 464
- (D) 512

**Q** A host is connected to a Department network which is part of a University network. The University network, in turn, is part of the Internet. The largest network in which the Ethernet address of the host is unique is: **(Gate-2004) (1 Marks)**

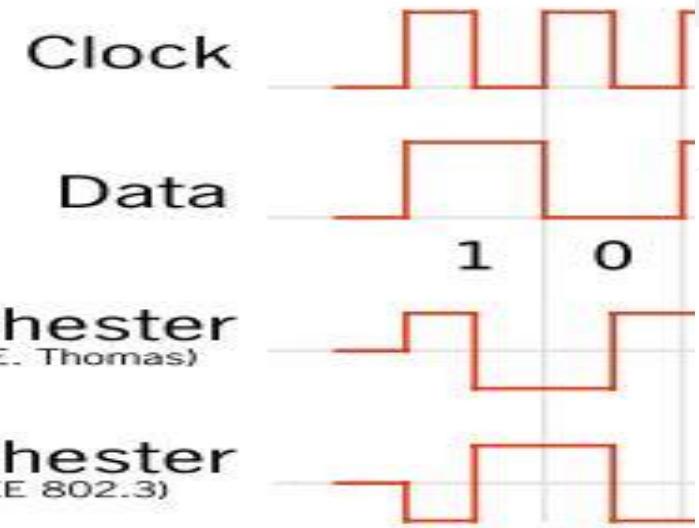
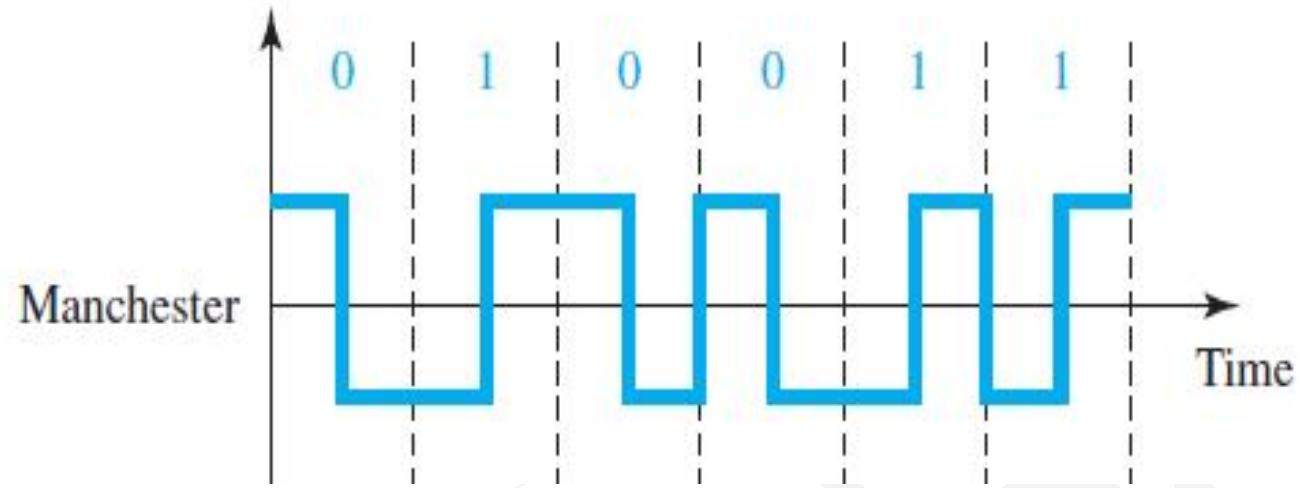
- (A) the subnet to which the host belongs      (B) the Department network
- (C) the University network                        (D) the Internet

# Implementation

- At the sender, data are converted to a digital signal using the Manchester scheme; at the receiver, the received signal is interpreted as Manchester and decoded into data.

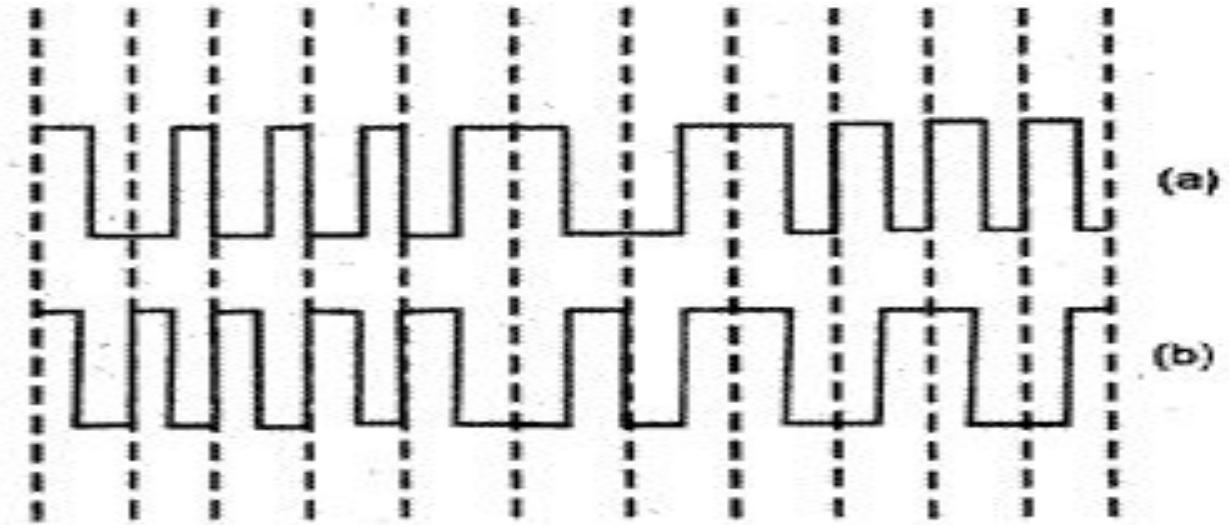


- In Manchester encoding, the duration of the bit is divided into two halves. The voltage remains at one level during the first half and moves to the other level in the second half. The transition at the middle of the bit provides synchronization



**Q** In the waveform (a) given below, a bit stream is encoded by Manchester encoding scheme. The same bit stream is encoded in a different coding scheme in wave form (b). The bit stream and the coding scheme are (Gate-2007) (2 Marks)

- (A) 1000010111 and Differential Manchester respectively
- (B) 0111101000 and Differential Manchester respectively
- (C) 1000010111 and Integral Manchester respectively
- (D) 0111101000 and Integral Manchester respectively



**Q** In Ethernet when Manchester encoding is used, the bit rate is: **(Gate-2007) (1 Marks)**

- a) Half the baud rate.
- b) Twice the baud rate.
- c) Same as the baud rate
- d) None of the above.

Bit rate is related to the speed of the transmission of the digital bit, while baudrate is related to the speed of change of symbols, which are significancies in analog signal. These can be either in amplitude, frequency or phase or more complex modulation methods. In manchester encoding, one bit is represented by two different levels of voltage. Therefore, lets say if you want to transfer 1Mbit digital data in one second, then you will need to make ~ 2 million changes in the level of the analogous signal. That is why, your bit rate will be 1Mbps, while your baud rate will be 2M bauds.

**Q** How many 8-bit characters can be transmitted per second over a 9600 baud serial communication link using asynchronous mode of transmission with one start bit, eight data bits, two stop bits, and one parity bit ? **(Gate-2004) (1 Marks)**

- (A) 600      (B) 800      (C) 876      (D) 1200

**Q** Consider that 15 machines need to be connected in a LAN using 8-port Ethernet switches. Assume that these switches do not have any separate uplink ports. The minimum number of switches needed is \_\_\_\_\_. **(Gate-2019) (1 Marks).**

**Q** You are given the following four bytes:

10100011

00110111

11101001

10101011

Which of the following are substrings of the base 64 encoding of the above four bytes? **(Gate-2007) (2 Marks)**

(A) zdp

(B) fpq

(C) qwA

(D) oze

**Q** You are given the following four bytes:

10100011

00110111

11101001

10101011

Which of the following are substrings of the base 64 encoding of the above four bytes? (Gate-2007) (2 Marks)

(A) zdp

(B) fpq

(C) qwA

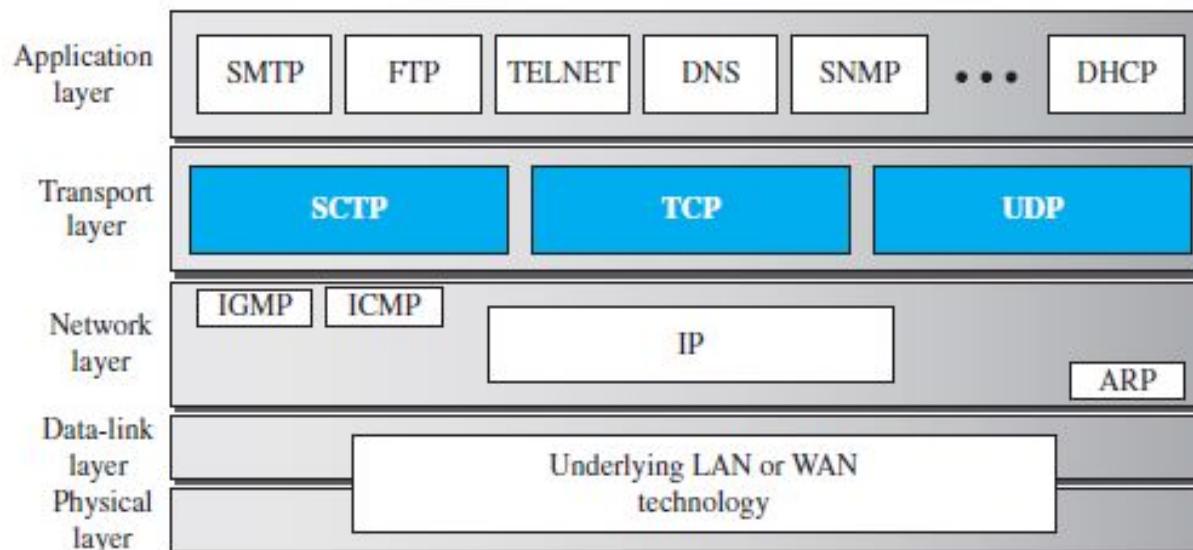
(D) oze

10100011001101111110100110101011

Index	Binary	Char	Index	Binary	Char	Index	Binary	Char	Index	Binary	Char
0	000000	A	16	010000	Q	32	100000	g	48	110000	w
1	000001	B	17	010001	R	33	100001	h	49	110001	x
2	000010	C	18	010010	S	34	100010	i	50	110010	y
3	000011	D	19	010011	T	35	100011	j	51	110011	z
4	000100	E	20	010100	U	36	100100	k	52	110100	ø
5	000101	F	21	010101	V	37	100101	l	53	110101	1
6	000110	G	22	010110	W	38	100110	m	54	110110	2
7	000111	H	23	010111	X	39	100111	n	55	110111	3
8	001000	I	24	011000	Y	40	101000	o	56	111000	4
9	001001	J	25	011001	Z	41	101001	p	57	111001	5
10	001010	K	26	011010	a	42	101010	q	58	111010	6
11	001011	L	27	011011	b	43	101011	r	59	111011	7
12	001100	M	28	011100	c	44	101100	s	60	111100	8
13	001101	N	29	011101	d	45	101101	t	61	111101	9
14	001110	O	30	011110	e	46	101110	u	62	111110	+
15	001111	P	31	011111	f	47	101111	v	63	111111	/
Padding		=									

## Network layer

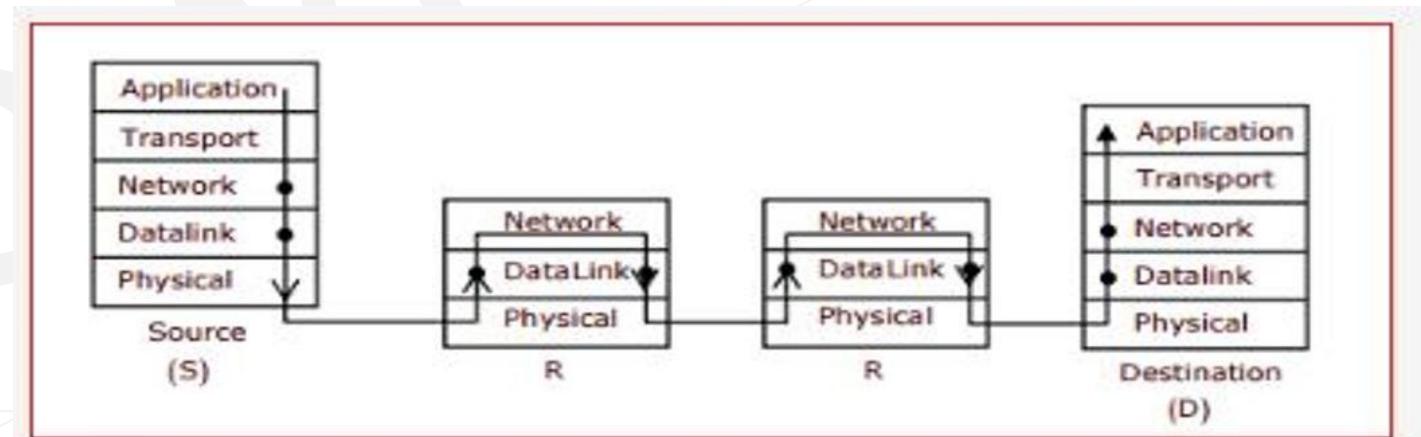
- **Purpose:** Manages source-to-destination packet delivery, often across multiple networks (links).
- **Services:**
  - **Logical Addressing:** Provides unique addresses (logical) to distinguish between source and destination systems.
  - **Routing:** Determines the best route for packets among multiple networks using routers or switches.
  - **Packetizing:** Encapsulates the payload into a packet with headers containing source, destination addresses, and other necessary information.
  - **Error Control:** Checks for header corruption using checksums in the datagram; utilizes ICMP for error control.
  - **Flow Control:** Not provided directly as the network layer's task at the receiver is usually straightforward.
  - **Congestion Control:** Manages network congestion, particularly when data flow exceeds network or router capacity.



**Q** Assume that source S and destination D are connected through two intermediate routers labelled R. Determine how many times each packet has to visit the network layer and the data link layer during a transmission from S to D. **(GATE-2013) (1 Marks)**



- (A) Network layer – 4 times and Data link layer – 4 times
- (B) Network layer – 4 times and Data link layer – 3 times
- (C) Network layer – 4 times and Data link layer – 6 times
- (D) Network layer – 2 times and Data link layer – 6 times



**Q Which one of the following statements is FALSE? (GATE-2004) (1 Marks)**

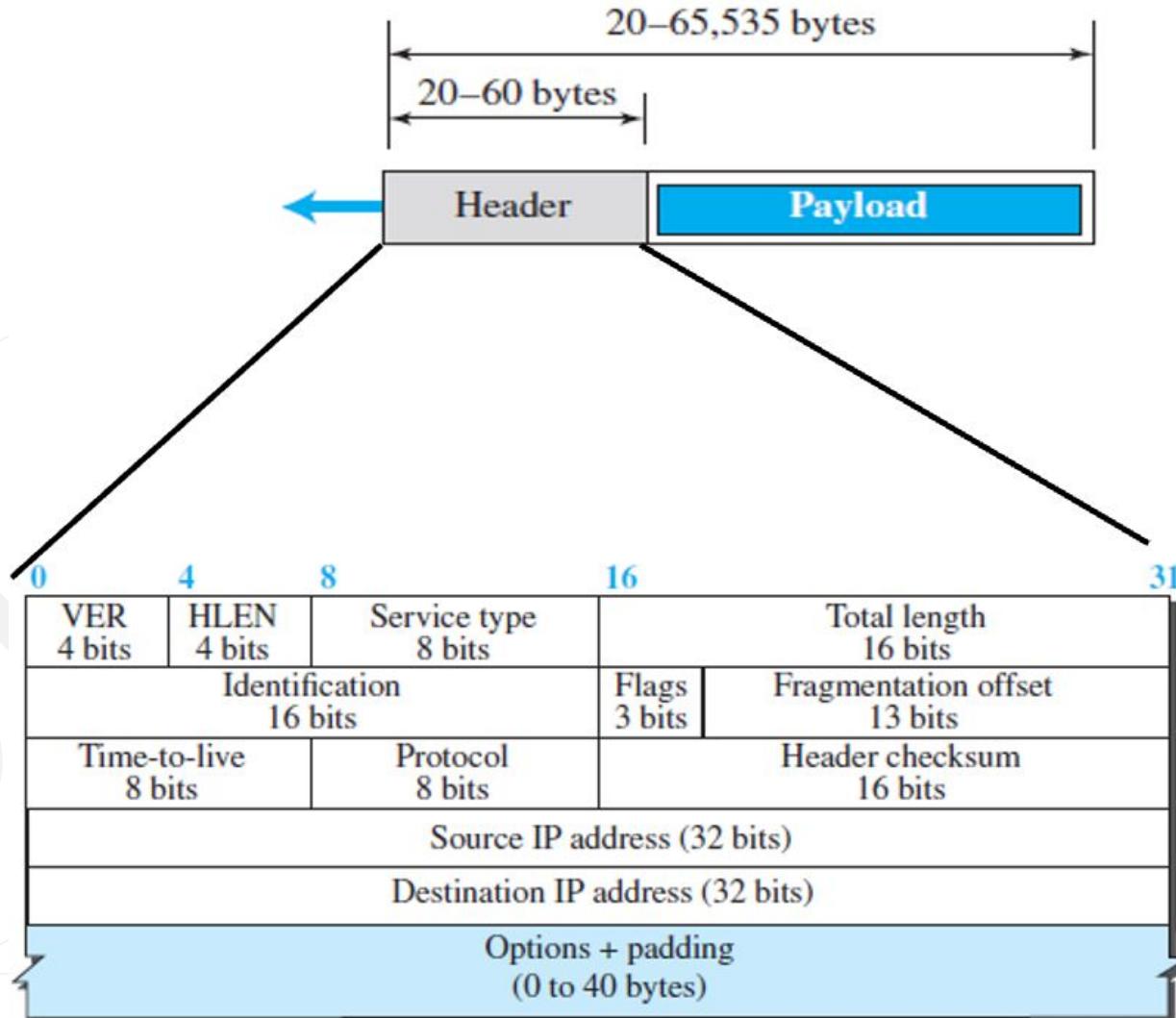
- (A) Packet switching leads to better utilization of bandwidth resources than circuit switching.
- (B) Packet switching results in less variation in delay than circuit switching.
- (C) Packet switching requires more per packet processing than circuit switching
- (D) Packet switching can lead to reordering unlike in circuit switching

## IPv4

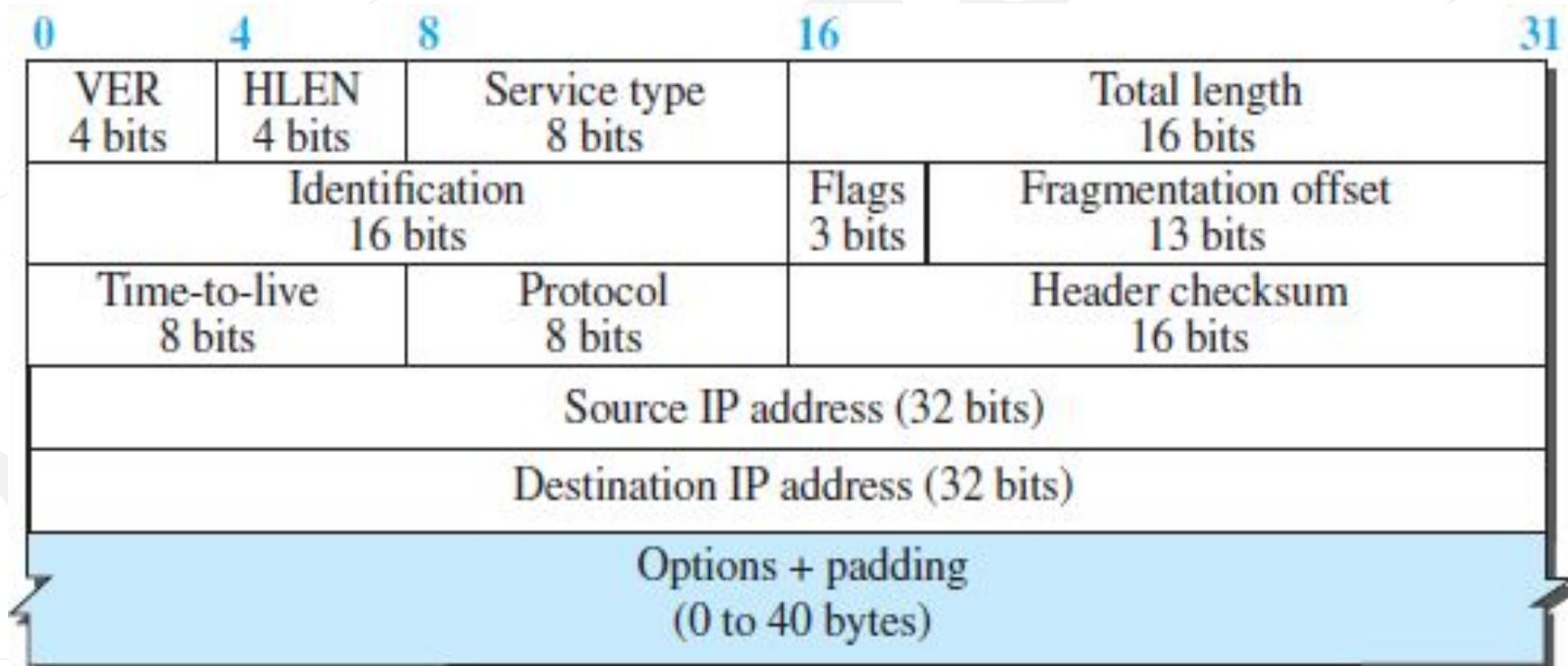
- IPv4 is an ***unreliable connectionless datagram protocol***—a best-effort delivery service.
- The term *best-effort* means that IPv4 packets can be corrupted, be lost, arrive out of order, or be delayed, and may create congestion for the network.
- ***datagram*** approach means Each datagram is handled independently, and each datagram can follow a different route to the destination.
- If reliability is important, IPv4 must be paired with a reliable protocol such as TCP, so the delivery mechanism used is TCP/IP protocols.

## Datagram Format

- Packets used by the IP are called ***datagrams***.
- A datagram is a variable-length packet consisting of two parts: header and payload (data).
- The header is 20 to 60 bytes in length and contains information essential to routing and delivery.

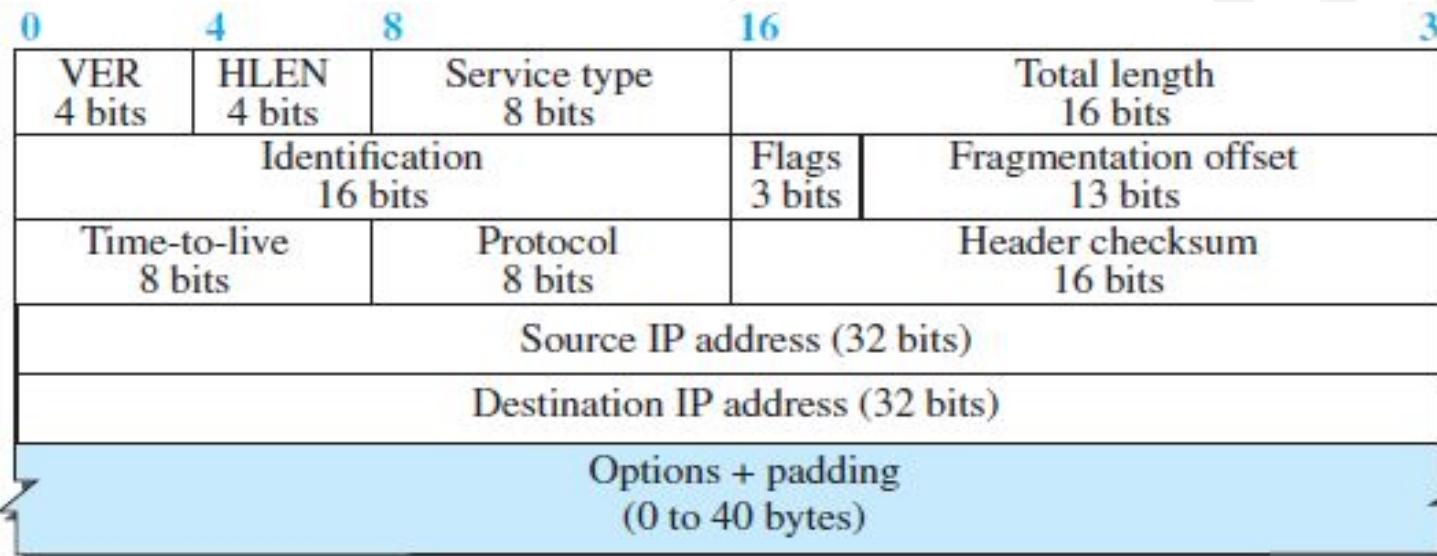


- **Version Number.** The 4-bit version number (VER) field defines the version of the IPv4 protocol, which, has the value of 4.
- **Header Length.** The 4-bit header length (HLEN) field defines the total length of the datagram header in 4-byte words. The IPv4 datagram has a variable-length header.
- **Scaling Factor:**
  - To make the value of the header length (number of bytes) fit in a 4-bit header length, the total length of the header is calculated as 4-byte words.
  - The total length is divided by 4 and the value is inserted in the field.
  - The receiver needs to multiply the value of this field by 4 to find the total length.
  - Example: If header length field contains decimal value 5 (represented as 0101), then Header length =  $5 \times 4 = 20$  bytes

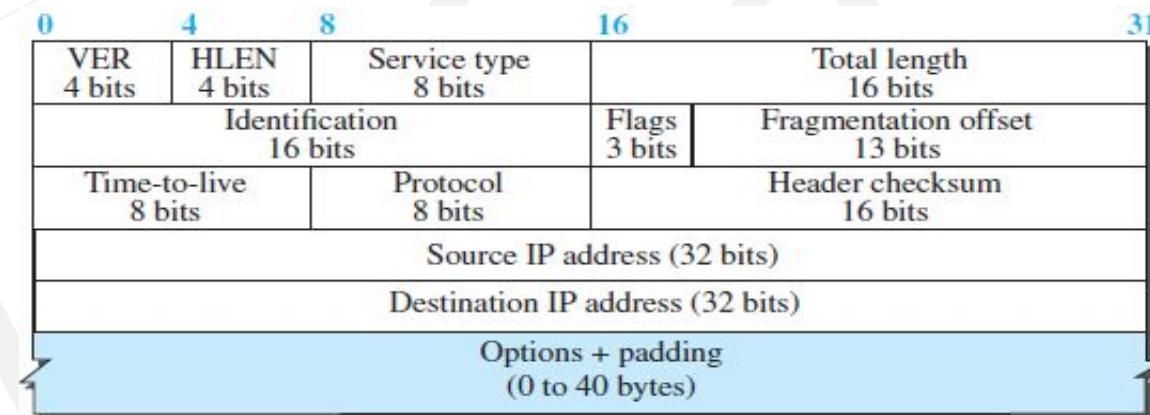
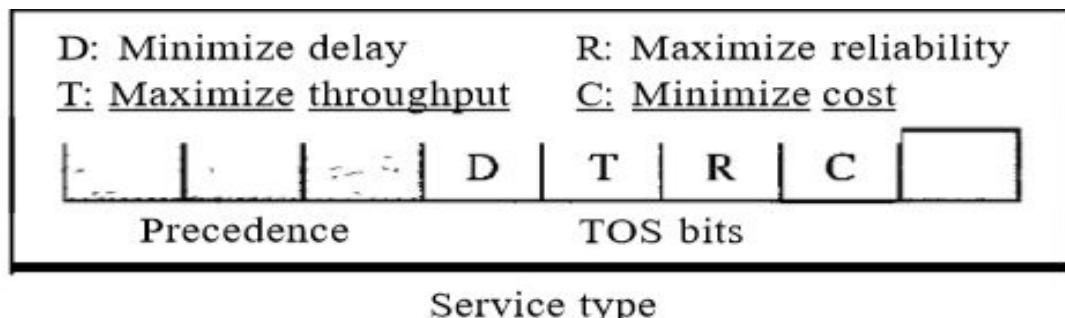


- Point to Note

- The length of IP header always lies in the range of [20 bytes, 60 bytes]
- The initial 5 rows of the IP header are always used. So, **minimum length of IP header** =  $5 \times 4$  bytes = 20 bytes.
- The size of Options field can go up to 40 bytes. So, **maximum length of IP header** = 20 bytes + 40 bytes = 60 bytes.
- The range of header length field value is always [5, 15] as  $[20/4 = 5, 60/4 = 15]$
- The range of header length is always [20, 60].



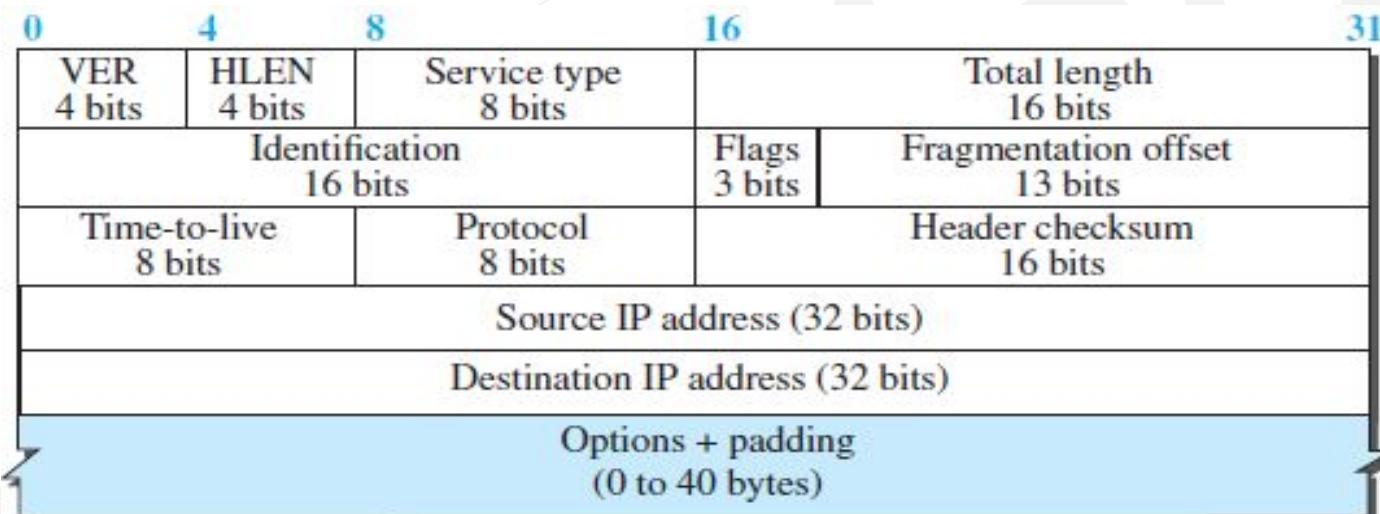
- **Precedence:** A 3-bit subfield with values ranging from 0 to 7, indicating the priority of a datagram. In case of network congestion, datagrams with lower precedence are discarded first.
- **Service Type (Quality of Service):** An 8-bit field used to specify how the datagram should be managed. It contains a 4-bit TOS subfield, where each bit represents a specific service type. Only one bit can be set to 1 at a time for each datagram to indicate the required service.



Protocol	TOS Bits	Description
ICMP	0000	Normal
BOOTP	0000	Normal
NNTP	0001	Minimize cost
IGP	0010	Maximize reliability
SNMP	0010	Maximize reliability
TELNET	1000	Minimize delay
FTP (data)	0100	Maximize throughput
FTP (control)	1000	Minimize delay
TFTP	1000	Minimize delay
SMTP (command)	1000	Minimize delay
SMTP (data)	0100	Maximize throughput
DNS (UDP query)	1000	Minimize delay
DNS (TCP query)	0000	Normal
DNS (zone)	0100	Maximize throughput

TOS Bits	Description
0000	Normal (default)
0001	Minimize cost
0010	Maximize reliability
0100	Maximize throughput
1000	Minimize delay

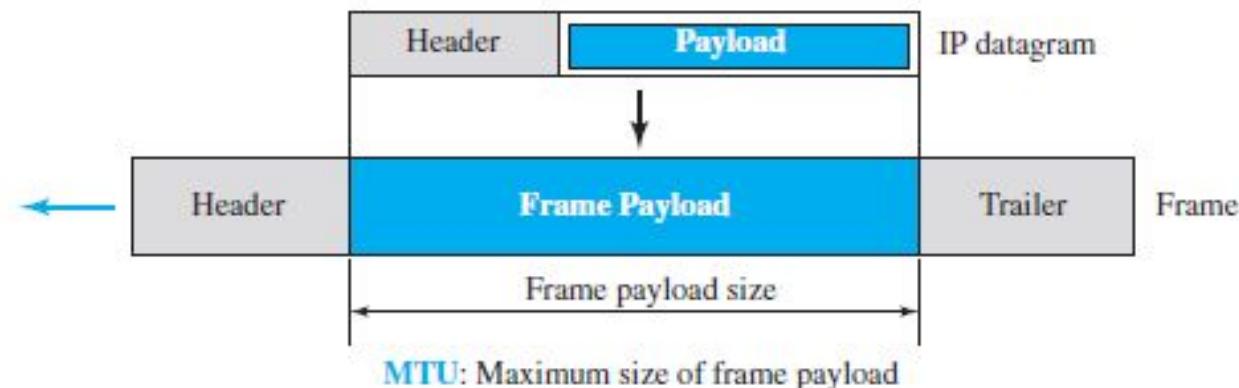
- **Total Length.** It defines the total length (header plus data) of the IP datagram in bytes. This field helps the receiving device to know when the packet has completely arrived.
- **Minimum total length of datagram** = 20 bytes (20 bytes header + 0 bytes data)
- **Maximum total length of datagram** = Maximum value of 16-bit word = 65535 bytes
- To find the length of the data coming from the upper layer, subtract the header length from the total length.
- **Length of data = total length – (HLEN) × 4**



## Maximum Transfer Unit (MTU)

- **MTU Definition:** The Maximum Transfer Unit (MTU) is the maximum size of the payload (data portion) that a link-layer protocol can encapsulate within its frame format. Each link-layer protocol defines its MTU size.
- **MTU Variation:** The value of the MTU differs among physical network protocols. For instance, a LAN typically has an MTU of 1500 bytes, while a WAN's MTU can be larger or smaller.
- **Datagram Fragmentation:** When a datagram exceeds the MTU size, it is fragmented. This means that the payload is split, and each fragment carries its own header. Some header fields like flags, fragmentation offset, total length, and checksum are adjusted during this process.
- **Multiple Fragmentation:** A datagram can undergo multiple fragmentations as it traverses networks with varying MTUs until it reaches the final destination. Each network adjusts the fragmentation as necessary.

Protocol	MTU
Hyperchannel	65,535
Token Ring (16 Mbps)	17,914
Token Ring (4 Mbps)	4,464
FDDI	4,352
Ethernet	1,500
X.25	576
PPP	296

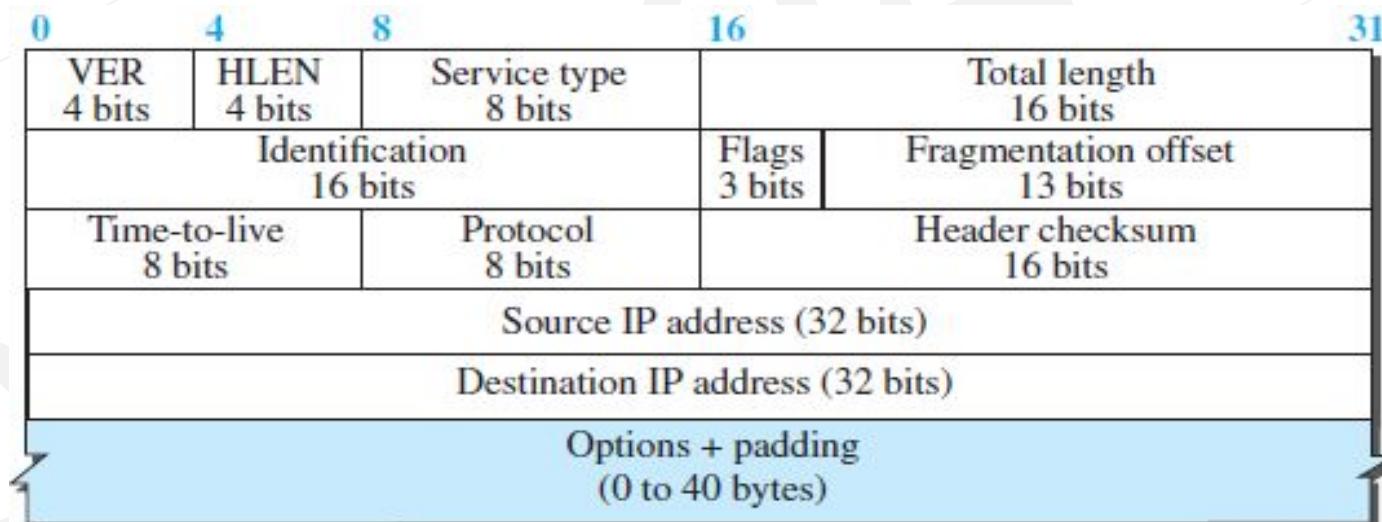


- **Identification:**

- A 16-bit field in an IP datagram uniquely identifies the datagram.
- The IP protocol uses a counter to label each datagram, initializing it to a positive number and incrementing it by one for each new datagram sent.
- When fragmentation occurs, the identification value is copied to all fragments to help in reassembly.

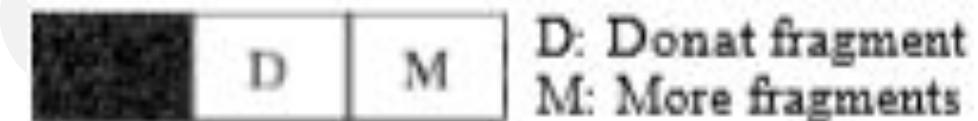
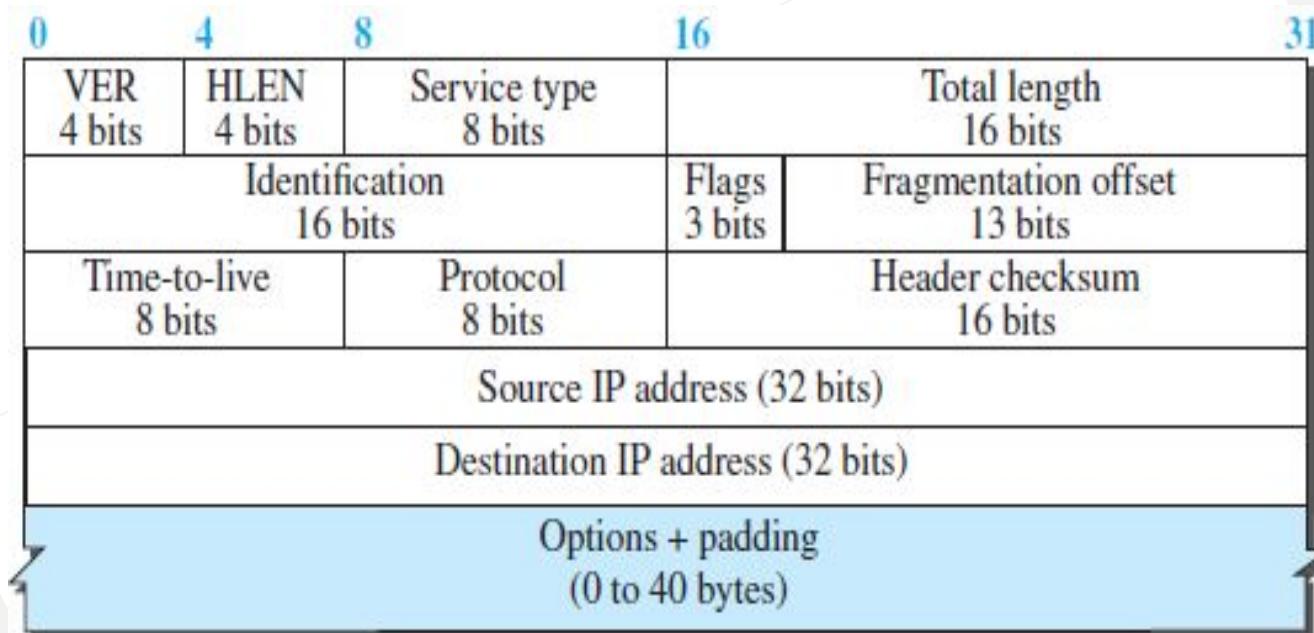
- **Fragmentation:**

- Fragmentation involves breaking the datagram into smaller fragments during transmission if necessary.
- Fragmentation can occur at the source or any router in the path.
- The destination host is responsible for reassembling the fragmented datagram.
- Each fragment can travel through different routes to reach the destination.

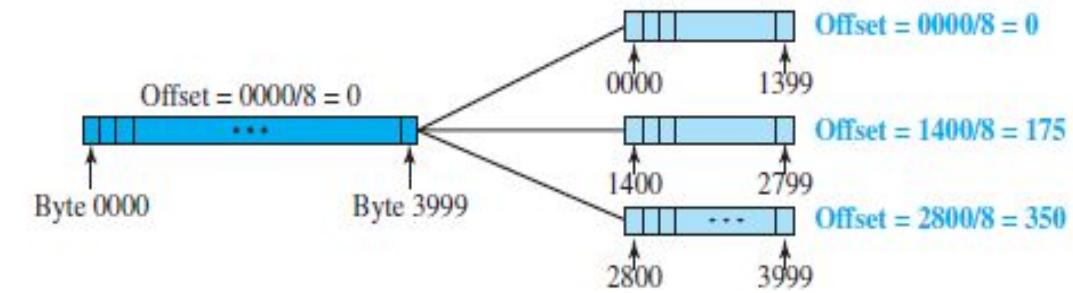


- **Flag Field:** The 3-bit *flags* field defines three flags.

- The leftmost bit is reserved (not used).
- The second bit (D bit) is called the *do not fragment* bit.
  - If its value is 1, the machine must not fragment the datagram.
  - If its value is 0, the datagram can be fragmented if necessary.
- The third bit (M bit) is called the *more fragment* bit.
  - If its value is 1, it means the datagram is not the last fragment; there are more fragments after this one.
  - If its value is 0, it means this is the last or only fragment.



0	4	8	16	31
VER 4 bits	HLEN 4 bits	Service type 8 bits	Total length 16 bits	
		Identification 16 bits	Flags 3 bits	Fragmentation offset 13 bits
Time-to-live 8 bits	Protocol 8 bits			Header checksum 16 bits
		Source IP address (32 bits)		
		Destination IP address (32 bits)		
		Options + padding (0 to 40 bytes)		



- **Fragmentation Offset:** The 13-bit *fragmentation offset* field shows the relative position of this fragment with respect to the whole datagram.
- It is the offset of the data in the original datagram measured in units of 8 bytes.
- The bytes in the original datagram are numbered 0 to 3999.
- The first fragment carries bytes 0 to 1399. The offset value =>  $0/8 = 0$ .
- The second fragment carries bytes 1400 to 2799; the offset value =>  $1400/8 = 175$ .
- The third fragment carries bytes 2800 to 3999. The offset value =>  $2800/8 = 350$ .

**Example:** Consider host A is present in network X having MTU = 520 bytes. There is another host B present in network Y having MTU = 200 bytes. Now, host A wants to send a message to host B.

**Q.12** Consider two hosts  $P$  and  $Q$  connected through a router  $R$ . The maximum transfer unit (MTU) value of the link between  $P$  and  $R$  is 1500 bytes, and between  $R$  and  $Q$  is 820 bytes. A TCP segment of size 1400 bytes was transferred from  $P$  to  $Q$  through  $R$ , with IP identification value as 0x1234. Assume that the IP header size is 20 bytes. Further, the packet is allowed to be fragmented, i.e., Don't Fragment (DF) flag in the IP header is not set by  $P$ .

Which of the following statements is/are correct?

- (a) Two fragments are created at  $R$  and the IP datagram size carrying the second fragment is 620 bytes.
- (b) If the second fragment is lost,  $P$  is required to resend the whole TCP segment.
- (c) TCP destination port can be determined by analysing only the second fragment.
- (d) If the second fragment is lost,  $R$  will resend the fragment with the IP identification value 0x1234

**Q** Consider an IP packet with a length of 4,500 bytes that includes a 20-byte IPv4 header and a 40-byte TCP header. The packet is forwarded to an IPv4 router that supports a Maximum Transmission Unit (MTU) of 600 bytes. Assume that the length of the IP header in all the outgoing fragments of this packet is 20 bytes. Assume that the fragmentation offset value stored in the first fragment is 0. The fragmentation offset value stored in the third fragment is \_\_\_\_\_. **(Gate-2018) (2 Marks)**

**Q** An IP datagram of size 1000 bytes arrives at a router. The router has to forward this packet on a link whose MTU (maximum transmission unit) is 100 bytes. Assume that the size of the IP header is 20 bytes. The number of fragments that the IP datagram will be divided into for transmission is \_\_\_\_\_.

**(Gate-2016) (2 Marks)**

**Q** Host A sends a UDP datagram containing 8880 bytes of user data to host B over an Ethernet LAN. Ethernet frames may carry data up to 1500 bytes (i.e. MTU = 1500 bytes). Size of UDP header is 8 bytes and size of IP header is 20 bytes. There is no option field in IP header. How many total number of IP fragments will be transmitted and what will be the contents of offset field in the last fragment? **(Gate-2015) (2 Marks)**

- (A)** 6 and 925
- (B)** 6 and 7400
- (C)** 7 and 1110
- (D)** 7 and 8880

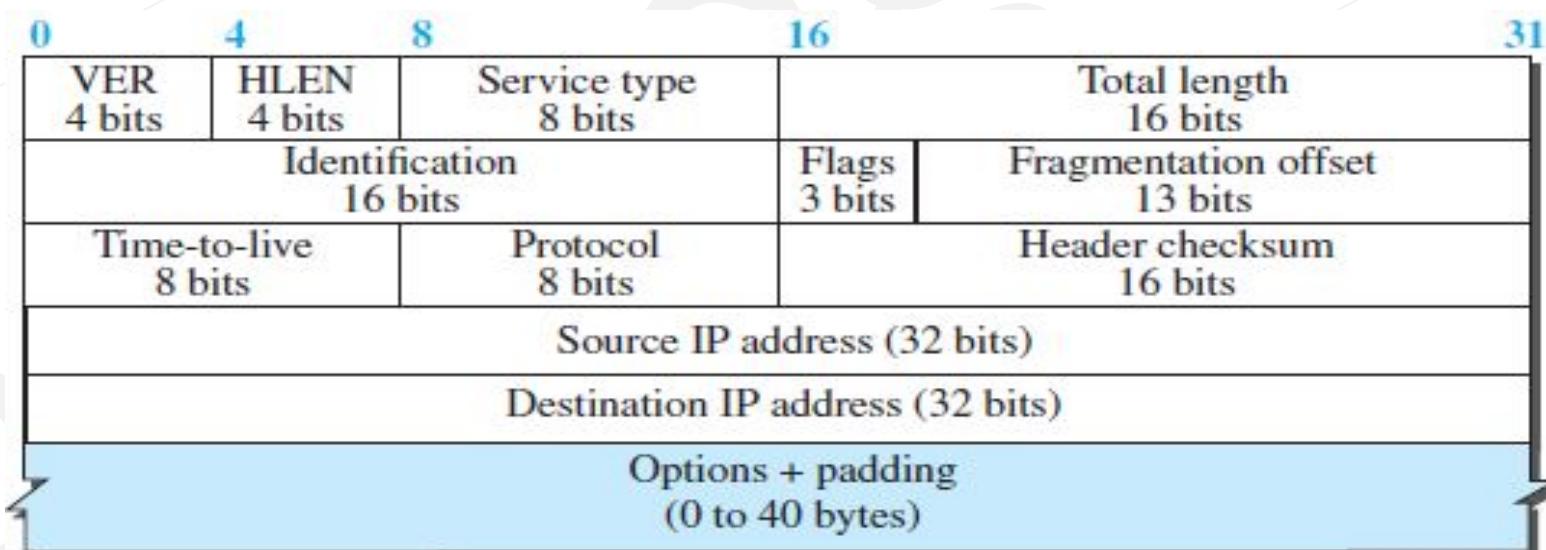
**Q** An IP router with a Maximum Transmission Unit (MTU) of 1500 bytes has received an IP packet of size 4404 bytes with an IP header of length 20 bytes. The values of the relevant fields in the header of the third IP fragment generated by the router for this packet are **(Gate-2014) (2 Marks)**

- (A) MF bit: 0, Datagram Length: 1444; Offset: 370
- (B) MF bit: 1, Datagram Length: 1424; Offset: 185
- (C) MF bit: 1, Datagram Length: 1500; Offset: 37
- (D) MF bit: 0, Datagram Length: 1424; Offset: 2960

**Q** In an IPv4 datagram, the M bit is 0, the value of HLEN is 10, the value of total length is 400 and the fragment offset value is 300. The position of the datagram, the sequence numbers of the first and the last bytes of the payload, respectively are **(Gate-2013) (2 Marks)**

- (A)** Last fragment, 2400 and 2789
- (B)** First fragment, 2400 and 2759
- (C)** Last fragment, 2400 and 2759
- (D)** Middle fragment, 300 and 689

Q. Consider sending an IP datagram of size 1420 bytes (including 20 bytes of IP header) from a sender to a receiver over a path of two links with a router between them. The first link (sender to router) has an MTU (Maximum Transmission Unit) size of 542 bytes, while the second link (router to receiver) has an MTU size of 360 bytes. The number of fragments the world be delivered at the receiver is \_\_\_\_\_ (Gate 2024,CS) (1 Marks) (NAT)

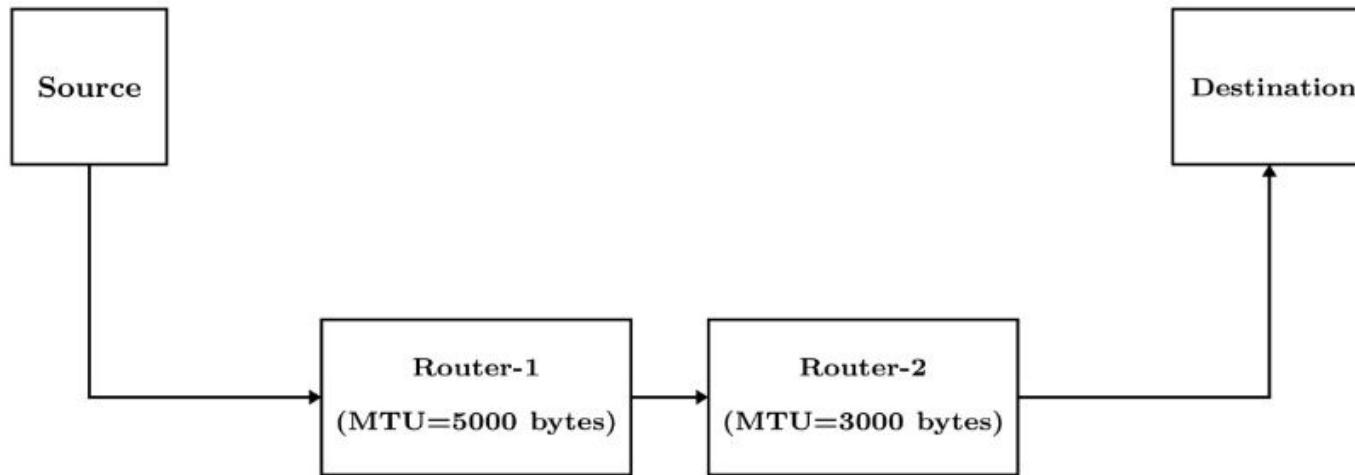


**Q. Which of the following statements about IPv4 fragmentation is/are TRUE? (Gate 2024 CS) (1 Mark) (MSQ)**

- (a) The fragmentation of an IP datagram is performed only at the source of the datagram
- (b) The fragmentation of an IP datagram is performed at any IP router which finds that the size of the datagram to be transmitted exceeds the MTU
- (c) The reassembly of fragments is performed only at the destination of the datagram
- (d) The reassembly of fragments is performed at all intermediate routers along the path from the source to the destination

**Q.** Suppose a message of size 15000 bytes is transmitted from a source to a destination using IPv4 protocol via two routers as shown in the figure. Each router has a defined maximum transmission unit (MTU) as shown in the figure, including IP header.

The number of fragments that will be delivered to the destination is \_\_\_\_\_  
(Answer in integer)(Gate 2025)



**Q.** Consider a network that uses Ethernet and IPv4. Assume that IPv4 headers do not use any options field.

Each Ethernet frame can carry a maximum of 1500 bytes in its data field.

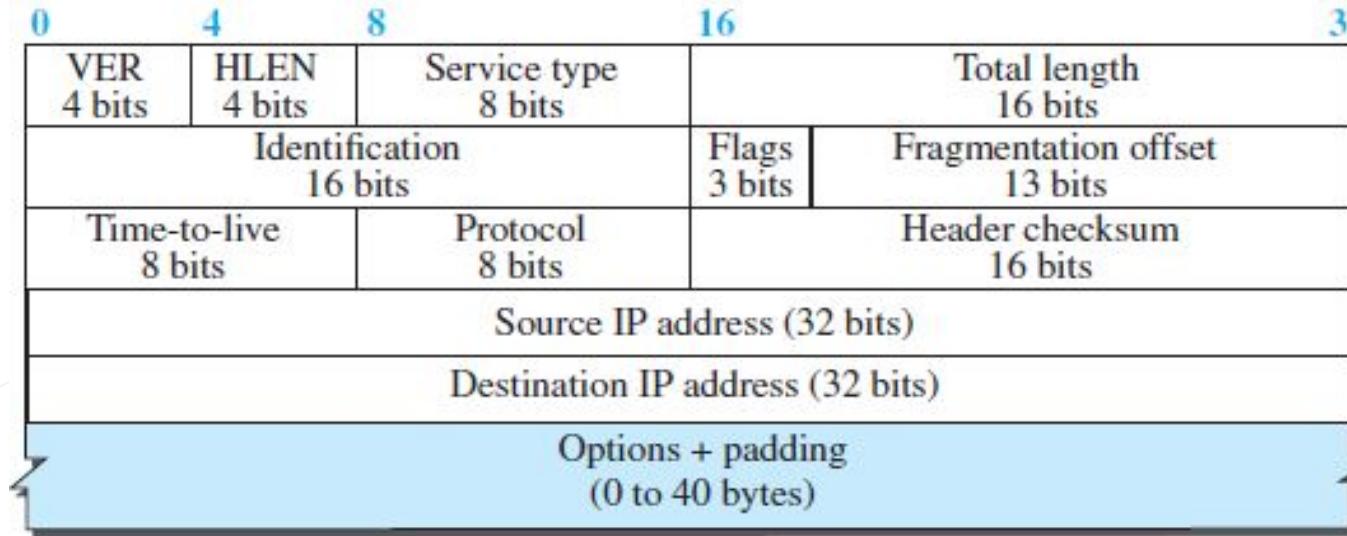
A UDP segment is transmitted. The payload (data) in the UDP segment is 7488 bytes.

Which ONE of the following choices has the CORRECT total number of fragments transmitted and the size of the last fragment including IPv4 header? **(GATE 2025)**

- A)** 5 fragments 1488 bytes
- B)** 6 fragments 88 bytes
- C)** 6 fragments 108 bytes
- D)** 6 fragments 116 bytes

## Time-to-Live (TTL) Field

- **Purpose:** Controls the maximum number of hops (routers) a datagram can traverse.
- **Function:**
  - Each router decreases the TTL value by one.
  - If the value reaches zero, the router discards the datagram.
- **Why Needed:**
  - Prevents datagrams from looping indefinitely between routers due to corrupted routing tables.
- **Use Case:**
  - Limiting packet journeys intentionally. For example, setting the TTL to 1 confines the packet to the local network, ensuring it gets discarded after the first router.



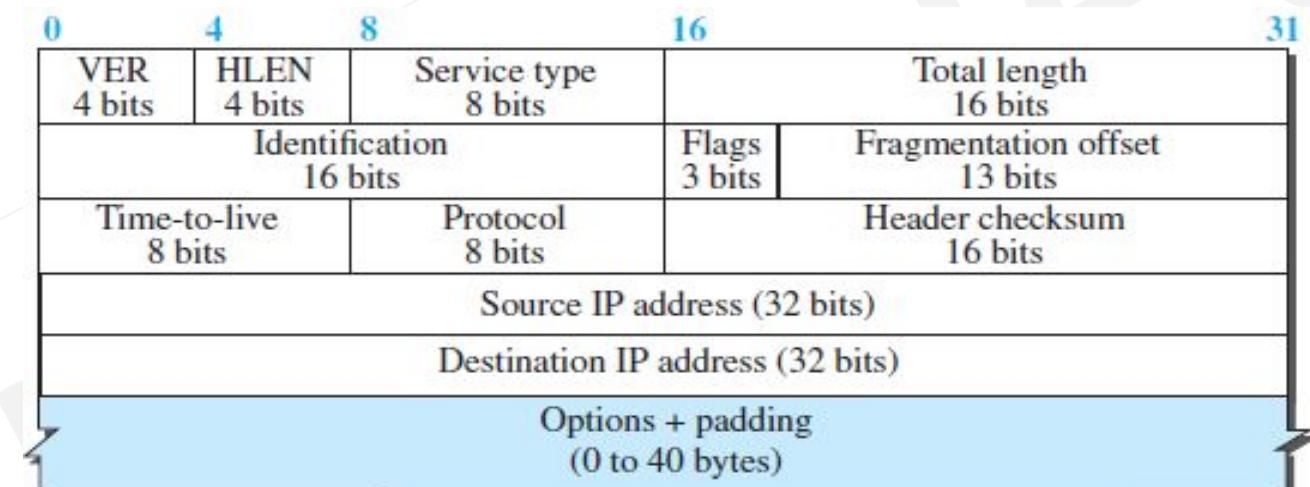
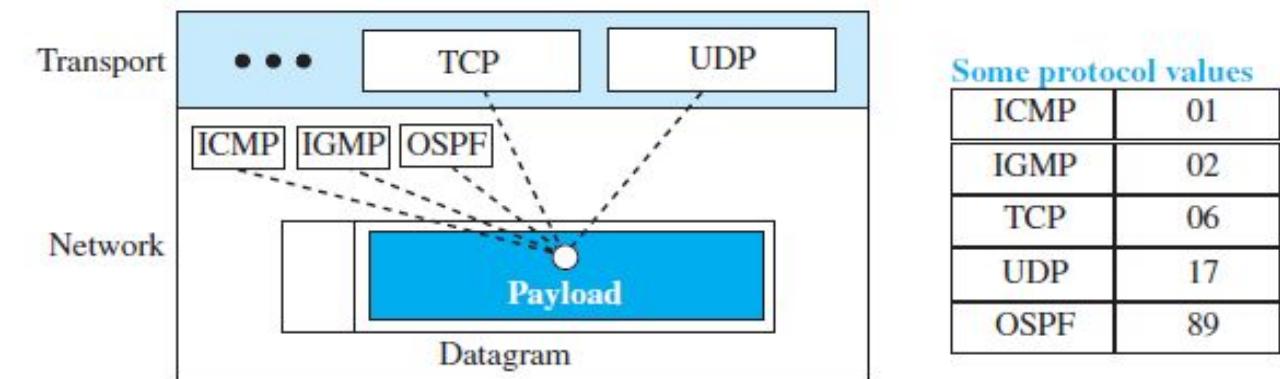
**Q** One of the header fields in an IP datagram is the Time to Live (TTL) field. Which of the following statements best explains the need for this field? **(Gate-2010) (1 Marks)**

- (A)** It can be used to prioritize packets
- (B)** It can be used to reduce delays
- (C)** It can be used to optimize throughput
- (D)** It can be used to prevent packet looping

**Q** For which one of the following reasons does Internet Protocol (IP) use the time-to-live (TTL) field in the IP datagram header (**Gate-2006**) (1 Marks)

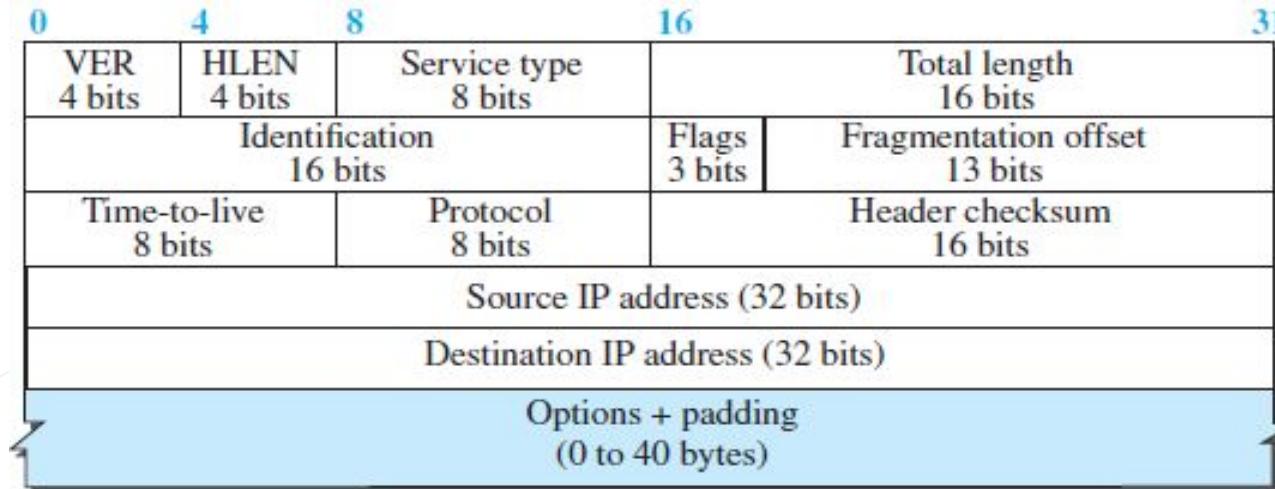
- (A) Ensure packets reach destination within that time
- (B) Discard packets that reach later than that time
- (C) Prevent packets from looping indefinitely
- (D) Limit the time for which a packet gets queued in intermediate routers.

- **Protocol.** In TCP/IP, the data section of a packet, called the payload, carries the whole packet from another protocol. A datagram, for example, can carry a packet belonging to any transport-layer protocol such as UDP or TCP.
- When the datagram arrives at the destination, the value of this field helps to define to which protocol the payload should be delivered.



- **Header Checksum (IPv4):**

- **Purpose:** The header checksum is used to verify the integrity of the IP header but not the payload.
- **Unreliability of IP:** IP is not responsible for ensuring the integrity of the payload, only the header.
- **Changing Header:** Fields like TTL change at each router, so the checksum must be recalculated at every hop.
- **Higher-level Check:** Higher-level protocols (like TCP, UDP) already include a checksum that covers the entire packet, so IPv4 only needs to check the header.
- **Efficiency:** Recalculating the checksum for the entire datagram at every router would slow down processing. By focusing only on the header, the IPv4 protocol maintains efficiency while ensuring header integrity.



**Q** Host A (on TCP/IP v4 network A) sends an IP datagram D to host B (also on TCP/IP v4 network B). Assume that no error occurred during the transmission of D. When D reaches B, which of the following IP header field(s) may be different from that of the original datagram D? **(Gate-2014) (1 Marks)**

- (i) TTL
- (ii) Checksum
- (iii) Fragment Offset

**(A)** (i) only

**(B)** (i) and (ii) only

**(C)** (ii) and (iii) only

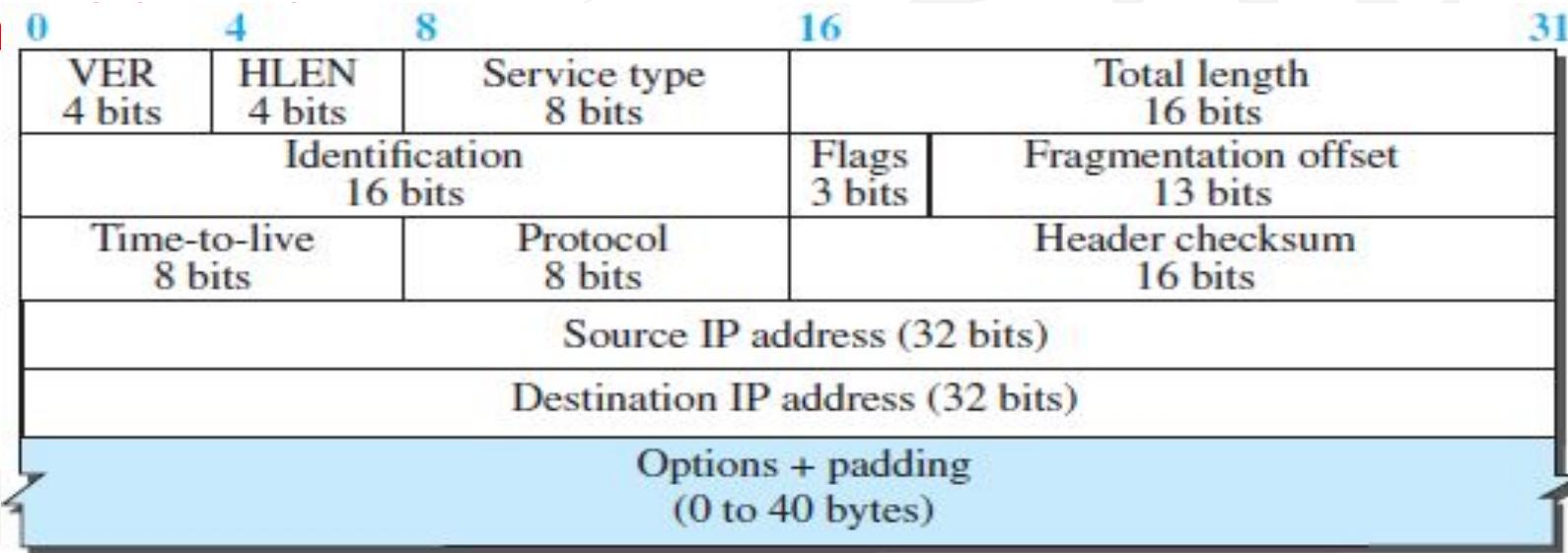
**(D)** (i), (ii) and (iii)

**Q Which of the following statements is TRUE? (Gate-2006) (1 Marks)**

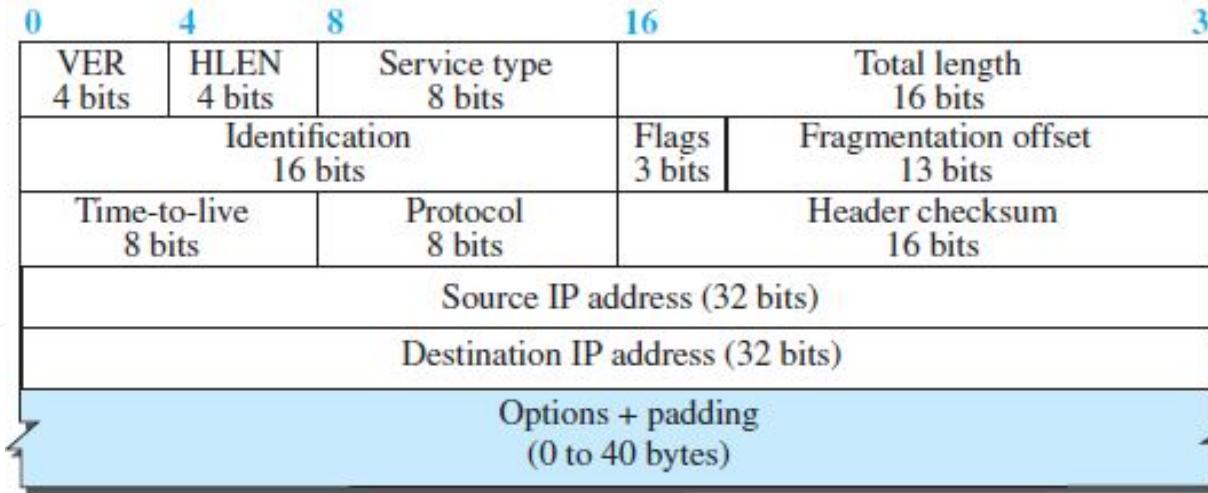
- (A) Both Ethernet frame and IP packet include checksum fields
- (B) Ethernet frame includes a checksum field and IP packet includes a CRC field
- (C) Ethernet frame includes a CRC field and IP packet includes a checksum field
- (D) Both Ethernet frame and IP packet include CRC fields

Q.Which of the following fields is/are modified in the IP header of a packet going out of a network address transaction (NAT) device from an internal network to an external network? (Gate 2024,CS) (1 Ma

- (a) Source IP
- (b) Destination IP
- (c) Header Checksum
- (d) Total Length

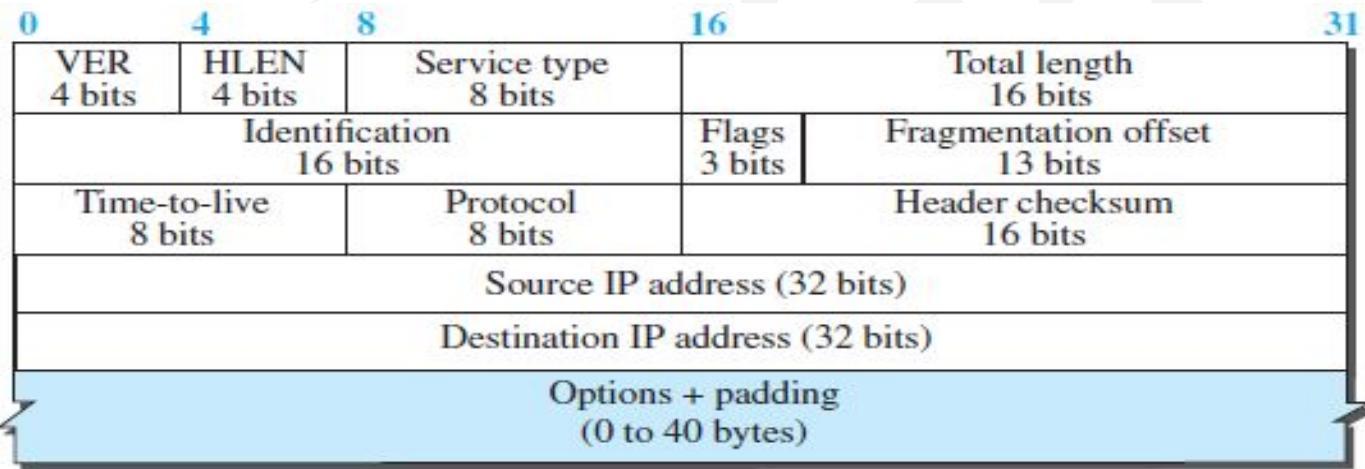


- Source and Destination Addresses:**
  - These are 32-bit fields specifying the IP addresses of the source and destination.
- Options:**
  - The header can include up to 40 bytes of options used for network testing and debugging.
  - Options are not mandatory for the IP header.
- Payload:**
  - The payload is the actual data carried by the datagram.
  - It represents the main content, while the header serves as the informational part of the package for handling and routing.



**Q. Which of the following fields of an IP header is/are always modified by any router before it forwards the IP packet? (Gate 2024 CS) (1 Mark) (MSQ)**

- (a) Source IP Address
- (b) Protocol
- (c) Time to Live (TTL)
- (d) Header Checksum



**Example:** An IPv4 packet has arrived with the first 8 bits as  $(01000010)_2$ . The receiver discards the packet. Why?

**Example:** In an IPv4 packet, the value of HLEN is 5, and the value of the total length field is  $(0028)_{16}$ . How many bytes of data are being carried by this packet?

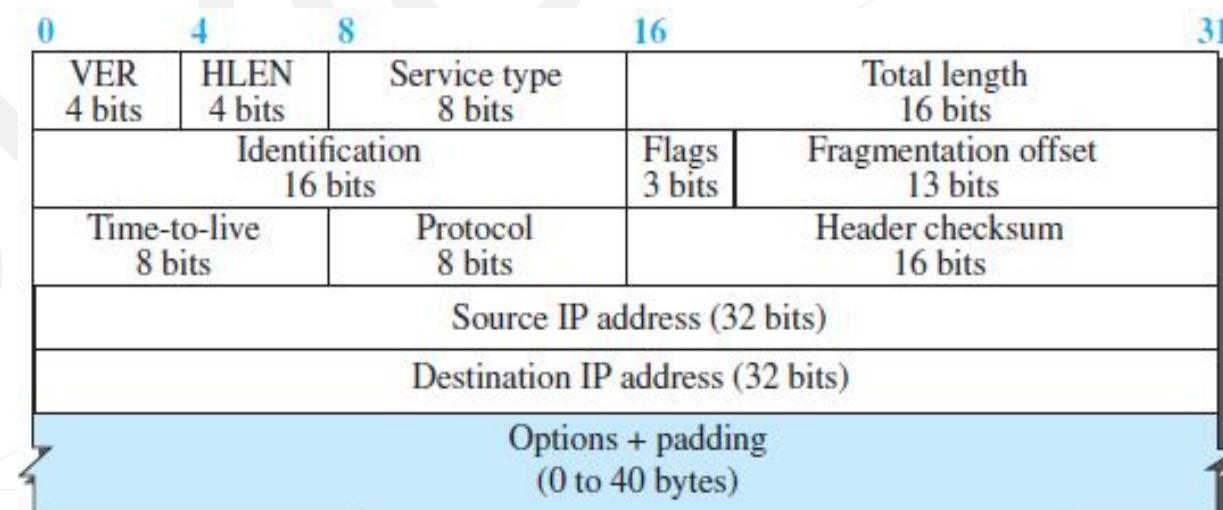
**Example:** An IPv4 packet has arrived with the first few hexadecimal digits as shown.

$(45000028000100000102\dots)_{16}$

How many hops can this packet travel before being dropped?

## Variable part

- **Variable Part Overview:** The IPv4 datagram header has two parts: a fixed part (20 bytes) and a variable part (up to 40 bytes). The variable part includes options that are not mandatory but must be supported by IPv4 implementations.
- **End of Option:** A 1-byte option used to indicate the end of the option field, primarily used for padding.
- **Record Route Option:** Records up to nine router addresses as the datagram traverses the network. Useful for network debugging and management.
- **Strict Source Route:** Allows the sender to predetermine an exact route that the datagram must follow. If the datagram visits a router not in the specified list, it is discarded with an error message. This can be useful for controlling the quality, security, or reliability of the route.
- **Loose Source Route:** Similar to strict source route but with more flexibility. All listed routers must be visited, but additional routers can also be traversed.
- **Timestamp Option:** Records the time (in milliseconds) when a router processes the datagram. This option helps track the route's behavior and estimate transit times between routers, though router clocks may not be perfectly synchronized.



**Q** The maximum number of IPv4 router addresses that can be listed in the record route (RR) option field of an IPv4 header is \_\_\_\_\_ (Gate-2017) (1 Marks)

**Q** Which one of the following fields of an IP header is NOT modified by a typical IP router? **(Gate-2015) (1 Marks)**

- (A)** Checksum
- (B)** Source address
- (C)** Time to Live (TTL)
- (D)** Length

**Q Which of the following assertions is FALSE about the Internet Protocol (IP)? (Gate-2003) (1 Marks)**

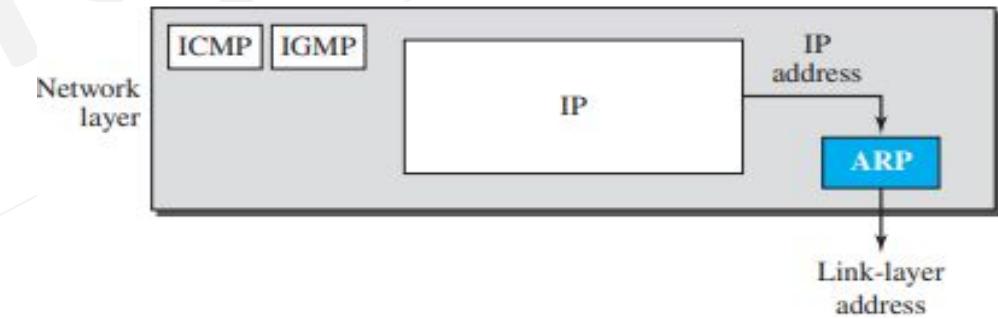
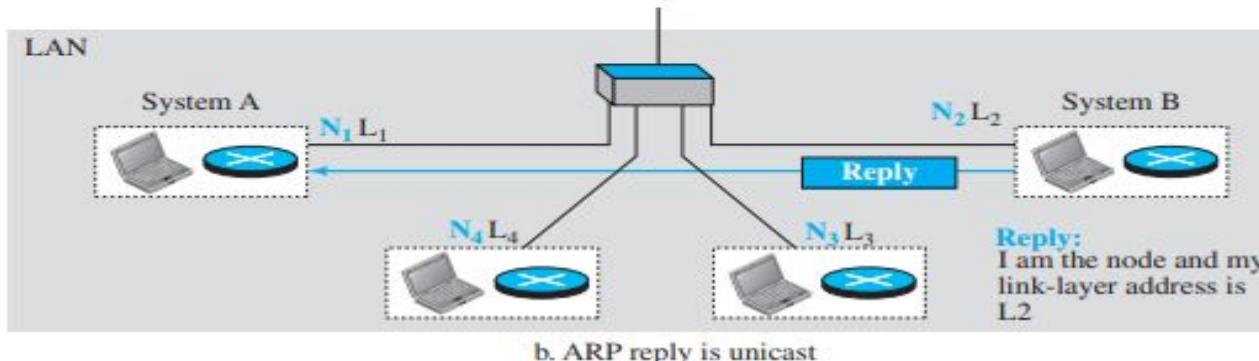
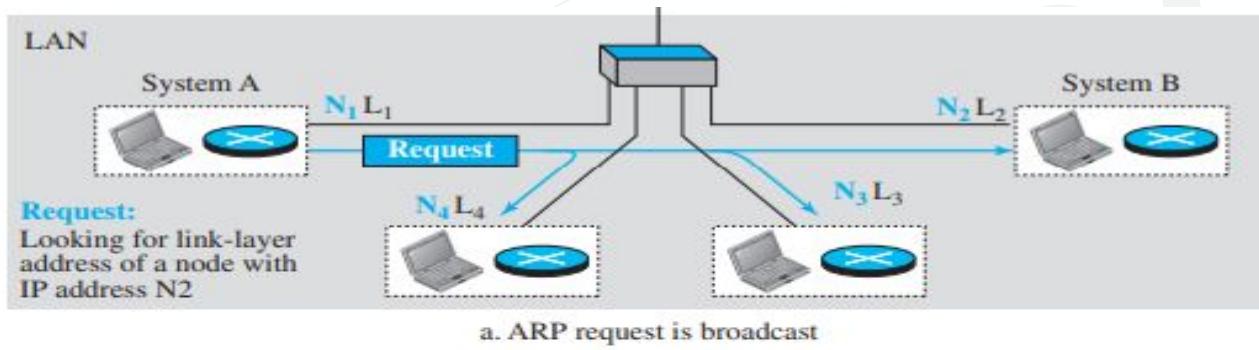
- (A) It is possible for a computer to have multiple IP addresses**
- (B) IP packets from the same source to the same destination can take different routes in the network**
- (C) IP ensures that a packet is discarded if it is unable to reach its destination within a given number of hops**
- (D) The packet source cannot set the route of an outgoing packets; the route is determined only by the routing tables in the routers on the way**

## Additional protocols

- **ARP (Address Resolution Protocol):** Used for mapping a logical IP address to a physical address (node-to-node communication).
- **RARP (Reverse Address Resolution Protocol):** Performs reverse mapping, translating a physical address back to a logical address, commonly used in booting diskless systems or leasing IP addresses.
- **ICMP (Internet Control Message Protocol):** Addresses flow and error control limitations in IP. It reports network congestion and certain types of errors at the destination or along the network.
- **IGMP (Internet Group Management Protocol):** Extends IP's capabilities to handle multicast delivery (one source to many destinations), addressing the growing need for group communication on the Internet.

# Address Resolution Protocol (ARP)

- ARP is used to map an IP address to a link-layer address (such as MAC addresses) to enable communication over a network.
- It works by accepting an IP address from the IP protocol and finding the corresponding link-layer address, which is then provided to the data-link layer for communication.
- **ARP Process:**
  - A sender (host or router) sends an ARP request packet to find the link-layer address of another device. This request contains the IP and link-layer addresses of the sender and the IP address of the target device.
  - The ARP request is broadcasted to all devices in the network. Every device processes this request.
  - Only the target device with the matching IP address recognizes the request and responds with an ARP reply. This reply contains both the IP address and link-layer address of the target device.
  - The response is sent directly (unicast) back to the sender.



**Q** Consider the following two statements.

- $S_1$ : Destination MAC address of an ARP reply is a broadcast address.
- $S_2$ : Destination MAC address of an ARP request is a broadcast address.

Which one of the following choices is correct? **(GATE 2021)**

(a) Both  $S_1$  and  $S_2$  are true

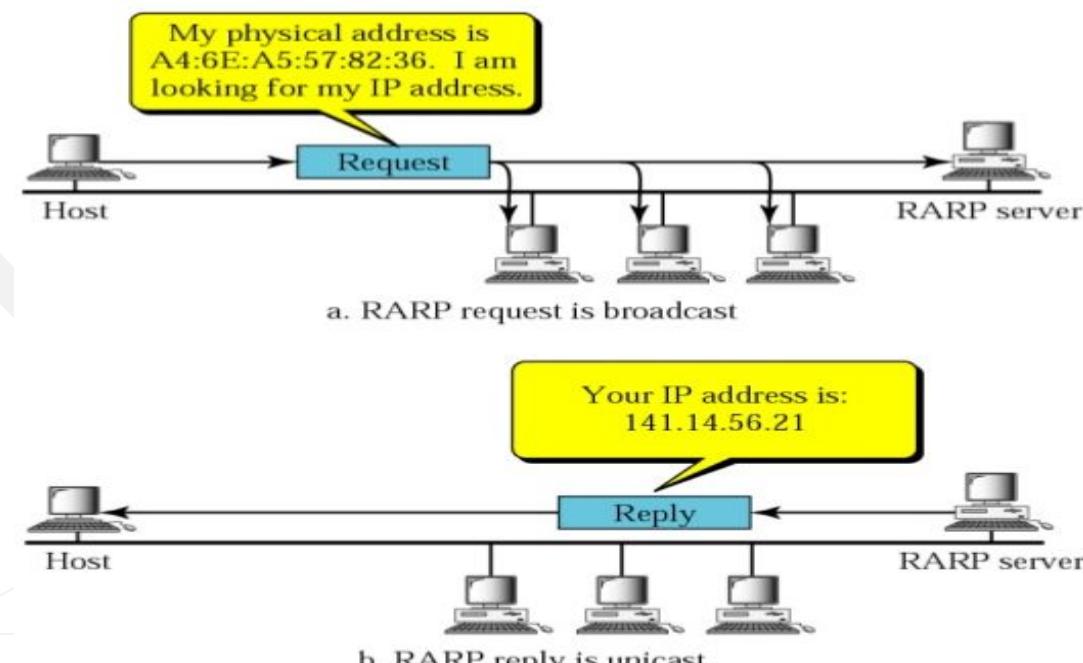
(b)  $S_1$  is true and  $S_2$  is false

(c)  $S_1$  is false and  $S_2$  is true

(d)  $S_1$  is false and  $S_2$  is true

# RARP

- **Purpose:** RARP is used to find the logical (IP) address for a device that only knows its physical (MAC) address.
- **Functionality:** Each host or router is assigned a unique logical address (IP). A device, especially a diskless one, might not have its IP address stored but can read its MAC address.
- **Usage:** A diskless machine, booted with minimum information from ROM, uses RARP to obtain its IP address from a network server.
- **Process:**
  - The device broadcasts a RARP request containing its physical address.
  - A RARP server on the same local network recognizes the request and responds with the correct IP address.
- **Requirements:** The device must have a RARP client, and the responding system must run a RARP server.

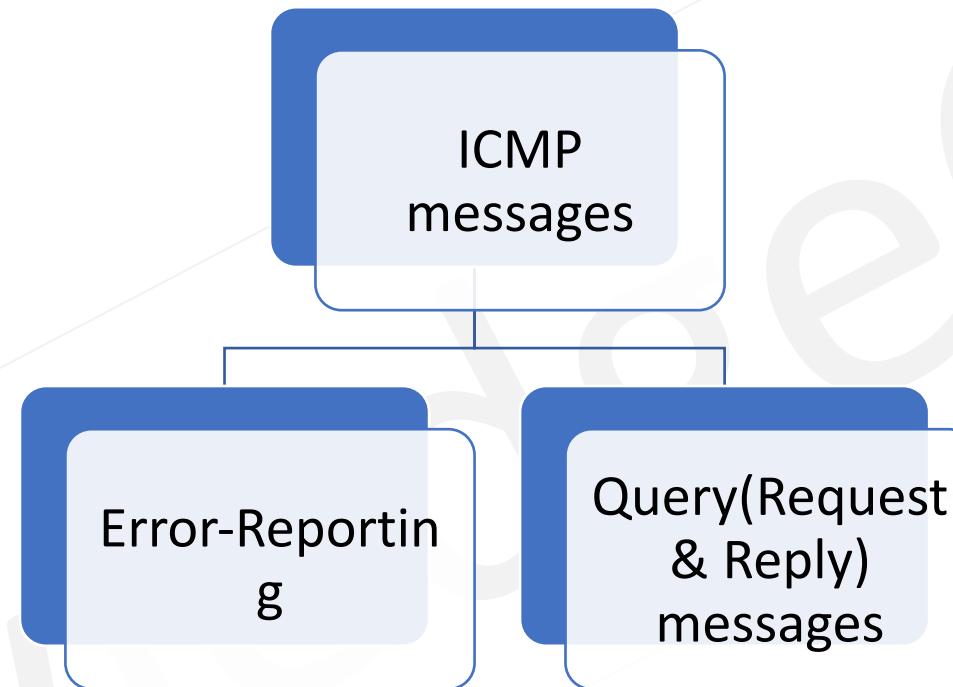


# ICMP

- The **Internet Protocol (IP)** has two primary deficiencies:
  - **Lack of Error Control:** IP does not include error-reporting or error-correction mechanisms.
  - **Lack of Assistance Mechanisms:** IP lacks the capability to assist hosts and routers with status queries or management tasks.
- **Common Scenarios Highlighting IP Deficiencies**
  - **Routing Issues:** A router discards a datagram if it cannot find a route to the destination or if the **Time-To-Live (TTL)** field reaches zero.
  - **Fragmentation Issues:** If a destination host cannot receive all fragments of a datagram within a certain time limit, it discards the incomplete datagram.
- **Missing Error and Query Mechanisms in IP**
  - **Notify the Original Host:** When an error occurs, there is no mechanism to inform the sender.
  - **Query Hosts and Routers:** Hosts may need to check if other hosts or routers are operational, and network administrators may need information for troubleshooting.
- **Introduction of ICMP**
  - To address these gaps, the **Internet Control Message Protocol (ICMP)** was introduced as a companion protocol to IP. ICMP provides:
    - **Error Reporting:** Notifies the original sender when issues occur.
    - **Assistance Mechanisms:** Allows for host and router status checks, improving network management and troubleshooting.

# Types of Messages

- ICMP messages are divided into two broad categories: Error-Reporting messages and query(request & reply) messages.



- **Purpose:** ICMP was introduced to address IP's deficiency in error reporting. However, it is important to note that ICMP does **not correct errors**; it merely reports them. Error correction is the responsibility of higher-level protocols such as TCP.
- **Sending Error Messages**
  - **Message Destination:** ICMP error messages are always sent to the **original source** of the datagram. This is because the IP datagram only contains information about the **source and destination IP addresses**.
  - **Process:** ICMP uses the **source IP address** to route error messages back to the originator of the datagram that encountered an issue.
- **Types of Problems Reported**
  - **Error Conditions:** ICMP error-reporting messages inform the original source of any issues a router or a destination host faces while processing an IP packet. These issues might include unreachable destinations, TTL expiration, and fragmentation problems.

- **Purpose of Query Messages**
  - **Diagnostic Role:** ICMP query messages are used to obtain specific information about the network or its components.
  - **Paired Messages:** These messages work in pairs—a node sends a query, and the destination node responds with the required information.
- **Encapsulation in IP Packets**
  - **Transmission:** A query message is **encapsulated** within an IP packet, which is then placed inside a data link layer frame for transmission.
- **Functionality**
  - **Network Information Gathering:** Query messages help a host or network manager gain specific details from another router or host, aiding in troubleshooting and network management.

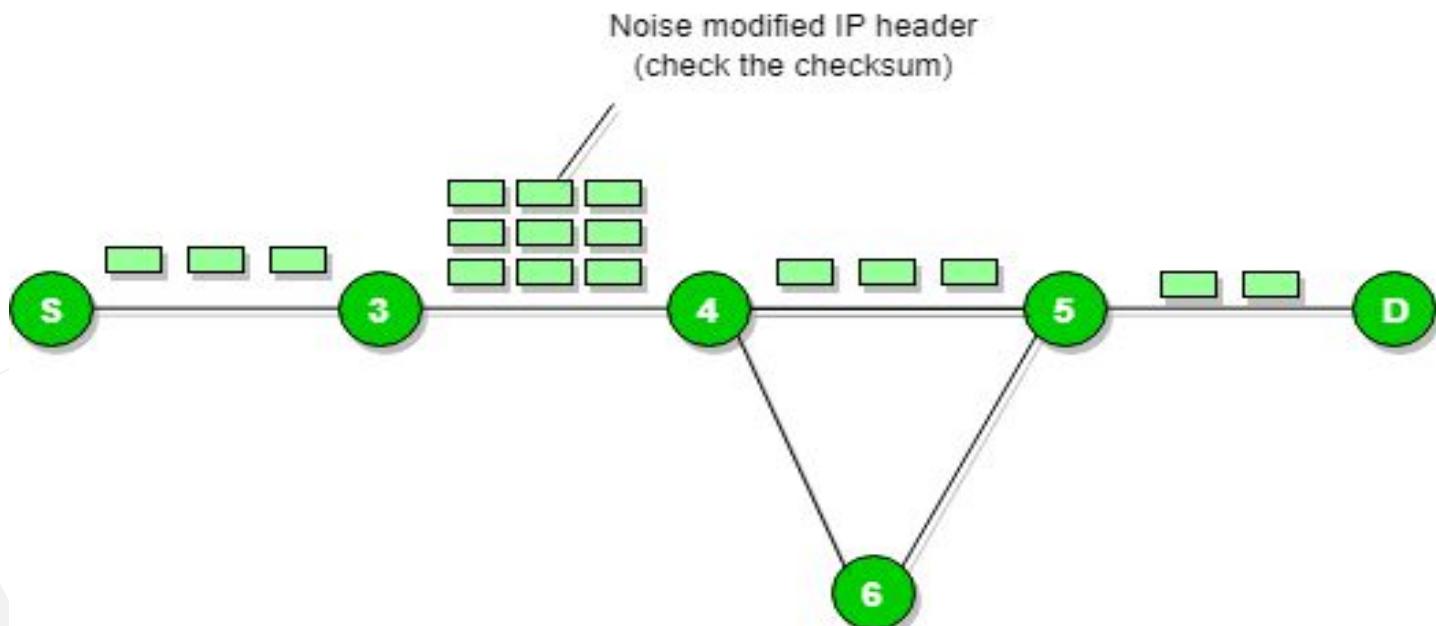
## Time exceeded message

- When a datagram's **Time-To-Live (TTL)** field reaches zero during transmission, the packet is discarded by the router. In such cases, **ICMP** takes the following actions:
  - Source IP Identification:** ICMP retrieves the **source IP** address from the discarded packet.
  - Sending Error Notification:** It informs the original sender about the discarded datagram by sending a **Time Exceeded** message.



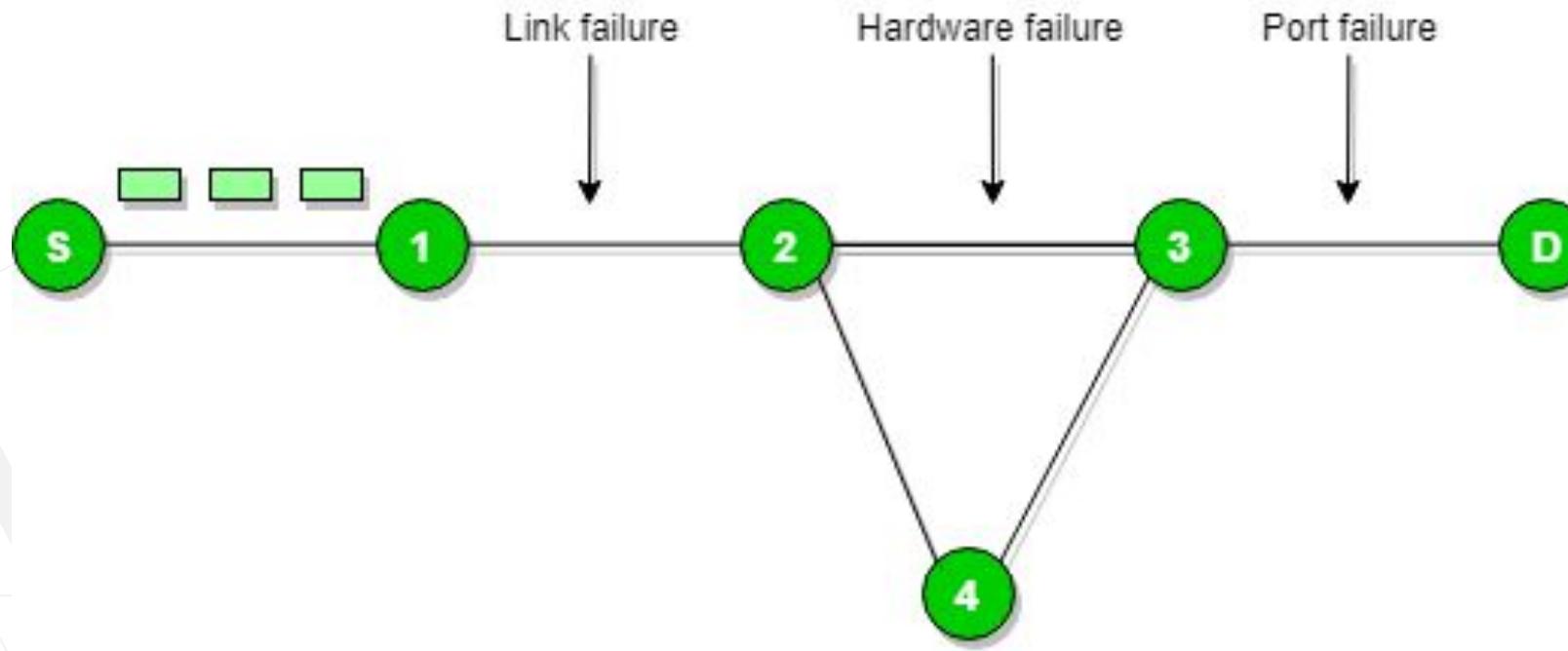
## Parameter Problem

- When a router receives a packet, it performs a checksum calculation to verify the integrity of the **header**. If there is a discrepancy between the **calculated header checksum** and the **received checksum**, the router discards the packet. In such cases, ICMP takes the following actions:
  - Source IP Identification:** ICMP extracts the **source IP** from the discarded packet.
  - Sending Notification:** It informs the original sender by sending a **Parameter Problem** message.



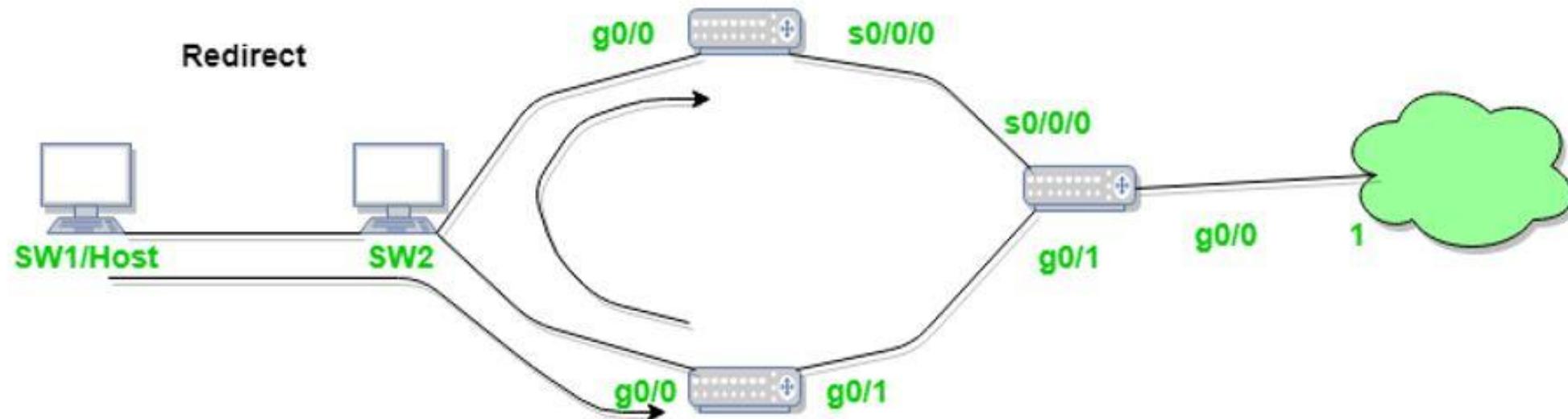
## Destination Un-reachable

- The **Destination Unreachable** message is generated by either a **router** or a **destination host** to inform the sender that the destination is not reachable due to various reasons. Here are key points:
- **Purpose:** This message informs the sender that the destination cannot be reached.
- **Issuers:** Not just routers, but also destination hosts can generate this message when specific failures occur, such as:
  - **Link failures:** When the network link is down.
  - **Hardware failures:** Malfunctioning network hardware.
  - **Port failures:** When the required port is not accessible or not open.



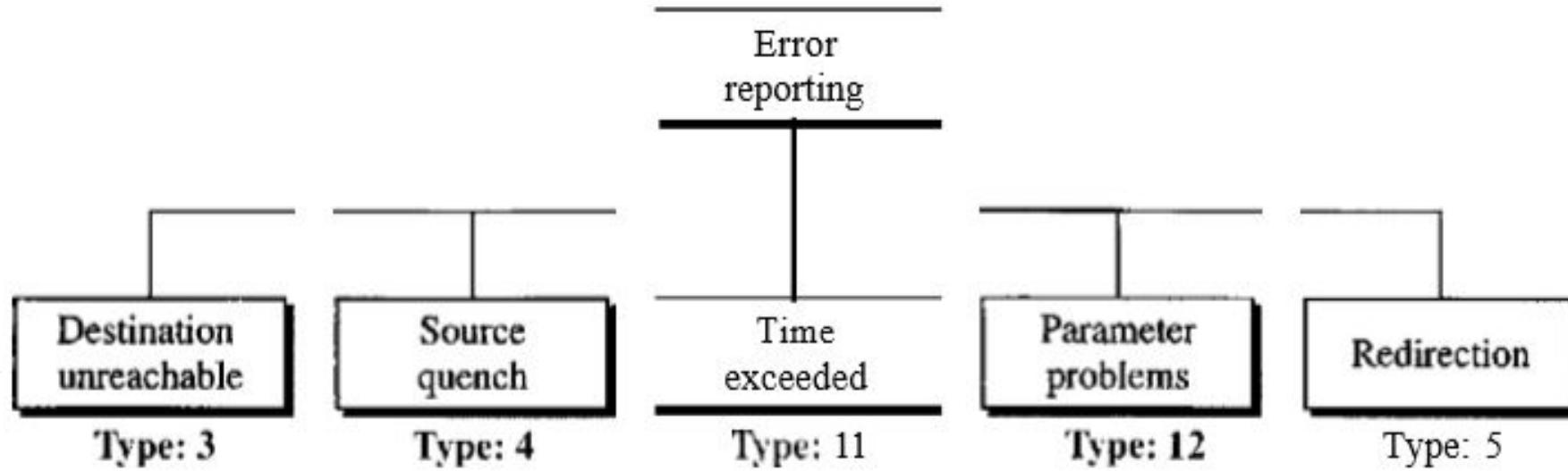
## Redirection message

- The **Redirect message** is used to inform a host that there is a better route available to reach the intended destination. Key points include:
  - Purpose:** The redirect message requests the host to **update its routing information**.
  - Action:** It advises the sender to send data packets through an **alternate route** for more efficient delivery.



- Key Rules for ICMP Error Messages

- No ICMP error message is generated in response to a datagram carrying another ICMP error message.
- No ICMP error message is generated for a fragmented datagram that is not the first fragment.
- No ICMP error message is generated for a datagram with a multicast address.
- No ICMP error message is generated for a datagram with special addresses like 127.0.0.0 or 0.0.0.0.



# Echo Request and Reply

- The **echo-request** and **echo-reply** messages are part of ICMP, primarily used for diagnosing network issues.
- **Purpose and Function**
  - **Network Diagnostics:** These messages help network managers and users verify if two systems (hosts or routers) can communicate at the **IP level**.
  - **Proof of Communication:** When a machine receives an echo-reply in response to its echo-request, it indicates successful communication between sender and receiver using the IP protocol.
- **Additional Benefits**
  - **Router Path Verification:** It also confirms that intermediate routers are properly processing and forwarding IP datagrams.
  - **Ping Command:** Most systems have a **ping** command that uses these messages to send a series of requests and receive replies, providing **statistical data** on response times and packet loss.

## Router Solicitation and Advertisement

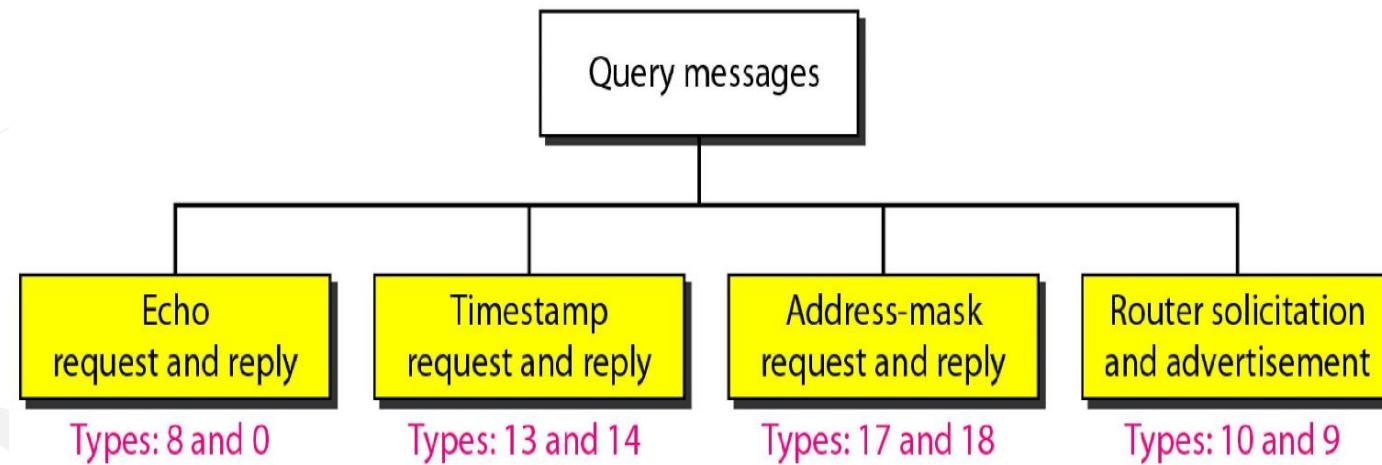
- The **Router Solicitation** and **Router Advertisement** messages help hosts discover and verify routers on their network. Here's how they function:
- **Purpose of Router Solicitation**
  - **Need for Router Information:** A host requires the address of routers within its own network to send data to other networks.
  - **Router Verification:** The host also needs to check if the routers are active and functioning properly.
- **How It Works**
  - **Router Solicitation:** A host sends a **broadcast** (or **multicast**) router-solicitation message to discover available routers.
  - **Router Advertisement:** Routers receiving this solicitation reply with a **router-advertisement** message, providing routing information.
- **Periodic Advertisements**
  - **Proactive Advertisement:** Routers can periodically send **unsolicited advertisements** to announce their presence and the presence of other routers on the network.

## Address-Mask Request and Reply

- The **Address-Mask Request and Reply** messages help a host obtain its **subnet mask**. Here's how they work:
- **Purpose**
  - **Need for Subnet Mask:** A host may know its IP address but might not know the corresponding subnet mask, such as **/24** for the address **159.31.17.24**.
- **How It Works**
  - **Request to Router:** To get the mask, the host sends an **Address-Mask Request** message to a router.
    - If the host knows the router's IP address, it sends the request directly.
    - If it doesn't know, the host **broadcasts** the request.
  - **Router Reply:** The router that receives the request responds with an **Address-Mask Reply** message, providing the necessary subnet mask.
- **Application**
  - The host can then apply the received subnet mask to its IP address to determine its **subnet address**.

## Timestamp Request and Reply

- The **Timestamp Request and Reply** messages are used for time-related diagnostics between two machines (hosts or routers). Here's their purpose:
- **Purpose**
  - **Round-Trip Time Calculation:** These messages help determine the **round-trip time** (RTT) required for an IP datagram to travel between two devices.
  - **Clock Synchronization:** They can also be used to **synchronize the clocks** of two machines.



# IGMP

- The **Internet Group Management Protocol (IGMP)** supports different types of IP communication:
- **Types of Communication**
  - **Unicasting:** A **one-to-one** communication between a single sender and a single receiver.
  - **Multicasting:** A **one-to-many** communication where a sender transmits a message to multiple receivers simultaneously.
- **Purpose of Multicasting**
  - **Efficient Broadcasting:** Multicasting allows sending the same message to multiple recipients without redundant transmissions.
  - **Applications:** Used in scenarios like updating stock prices to multiple brokers or notifying travel agents about flight cancellations.

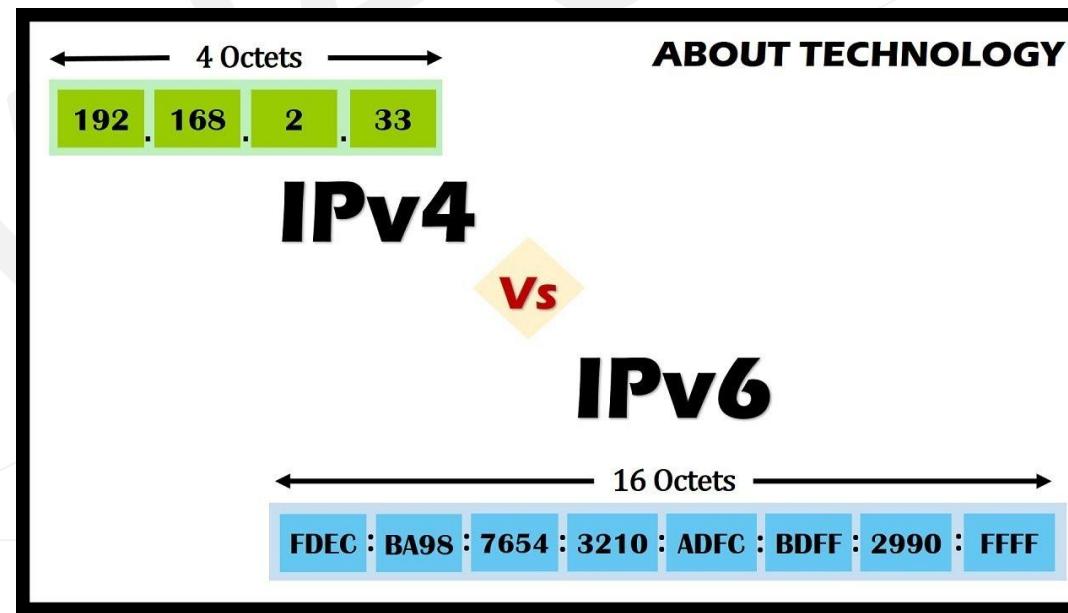


**Video-on-Demand**

**Distance Learning**  
[www.knowledgegate.in](http://www.knowledgegate.in)

# IPV4 ADDRESSES

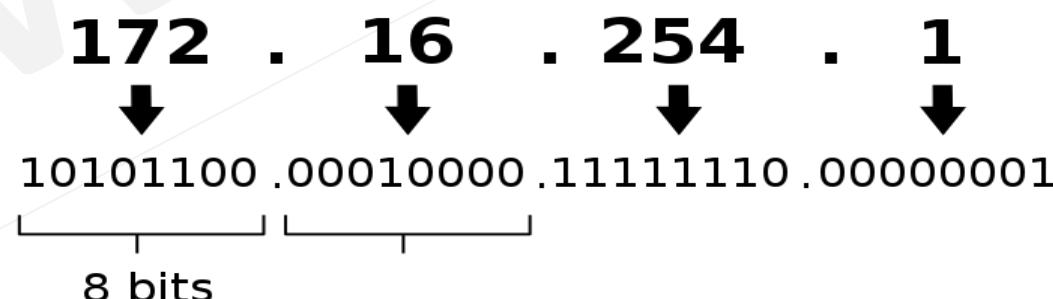
- The Internet Protocol addresses are **32 bits** in length; this gives us a maximum of  **$2^{32}$  addresses**. These addresses are referred to as **IPv4 (IP version 4)** addresses or simply **IP addresses** if there is no confusion.
- This means that, theoretically, if there were no restrictions, more than 4 billion (4,29,49,67,296) devices could be connected to the Internet. The actual number is much less because of the restrictions imposed on the addresses.
- World population** is often used to refer to the total number of humans currently living, and was estimated to have exceeded **7.9 billion as of November 2021**
- The need for more addresses, in addition to other concerns about the IP layer, motivated a new design of the IP layer called the new generation of IP or IPv6 (IP version 6). In this version, the Internet uses **128-bit addresses** that give much greater flexibility in address allocation ( **$3.4 * 10^{38}$** ). These addresses are referred to as **IPv6 (IP version 6)** addresses.



# Unique and Universal

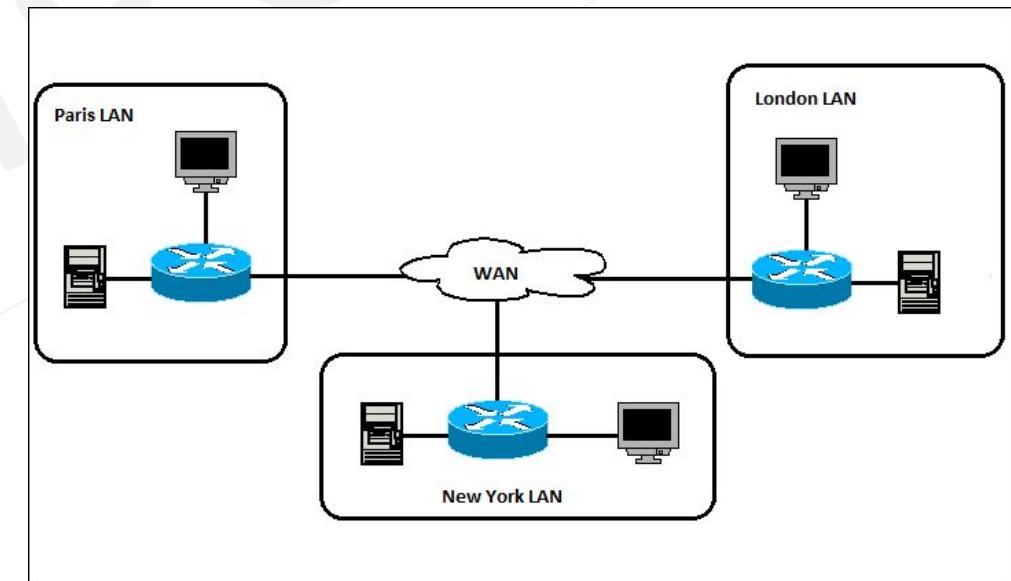
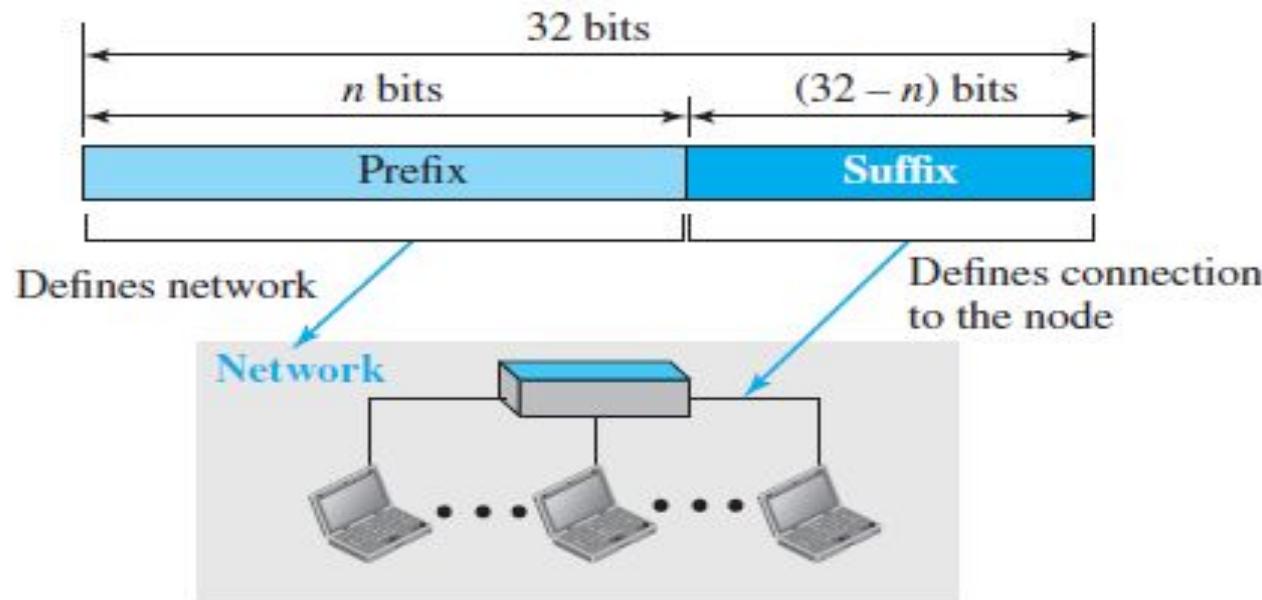
- An **IP address** serves as a unique and universal identifier for the connection of a host or router to the Internet. Here's a breakdown:
  - **Uniqueness:** Each IP address defines one and only one connection to the Internet, ensuring that no two devices can have the same IP address simultaneously.
  - **Universality:** The **IPv4 addressing system** must be universally accepted by all hosts that wish to connect to the Internet.
- There are two prevalent notations to show an IPv4 address: binary notation and dotted decimal notation.
  - **Binary Notation** - In binary notation, the **IPv4 address is displayed as 32 bits**. Each octet is often referred to as a **byte**. So, it is common to hear an IPv4 address referred to as a 32-bit address or a 4-byte address. An example of an IPv4 address in binary notation:  
**01110101 10010101 00011101 00000010**
  - **Dotted-Decimal Notation** - To make the **IPv4 address more compact and easier to read**, **Internet addresses are usually written in decimal form with a decimal point (dot) separating the bytes**. The following is the dotted decimal notation of the above address: **117.149.29.2**

IPv4 address in dotted-decimal notation

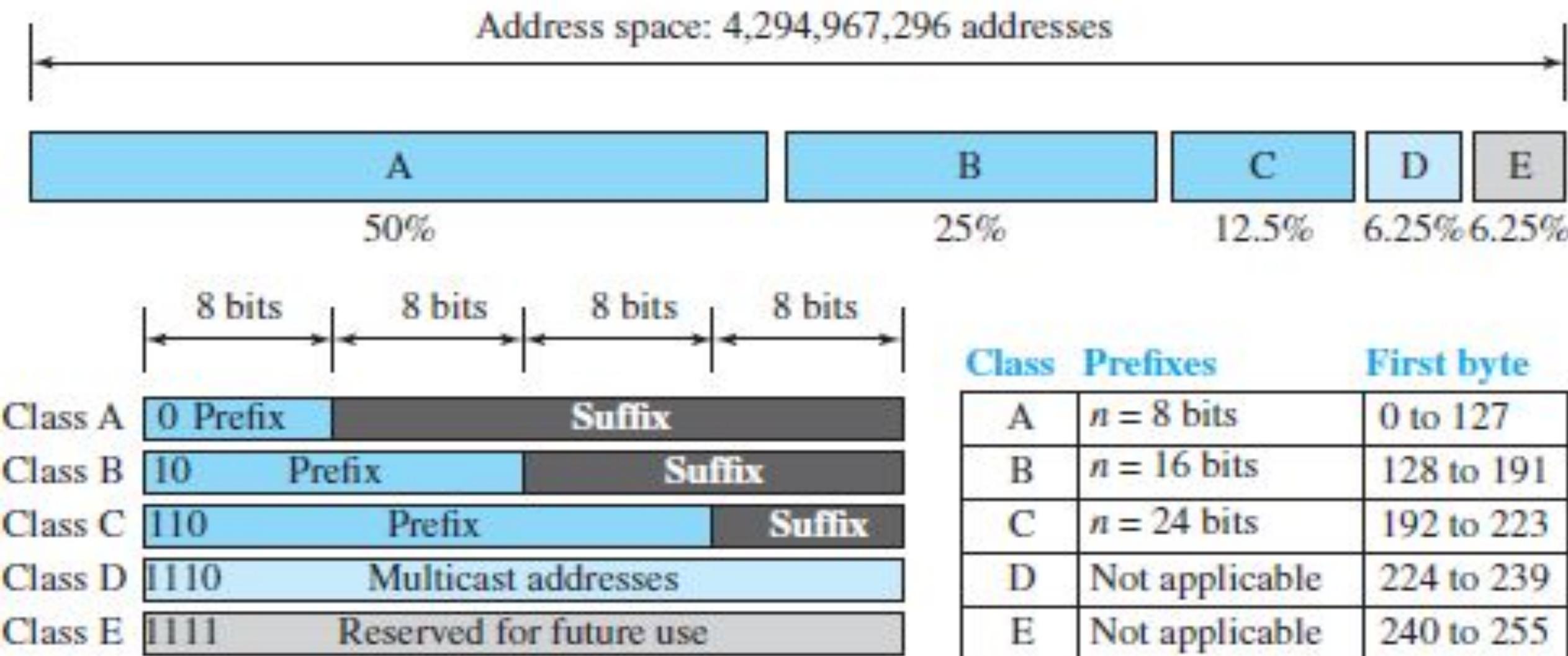


# Classful Addressing

- IPv4 addressing, at its inception, used the concept of classes. This architecture is called classful addressing.
- A 32-bit IPv4 address is hierarchical and divided only into two parts:
  - The first part of the address, called the *prefix*, defines the network (NetworkID).
  - The second part of the address, called the *suffix*, defines the node (connection of a device to the Internet (HostID)).

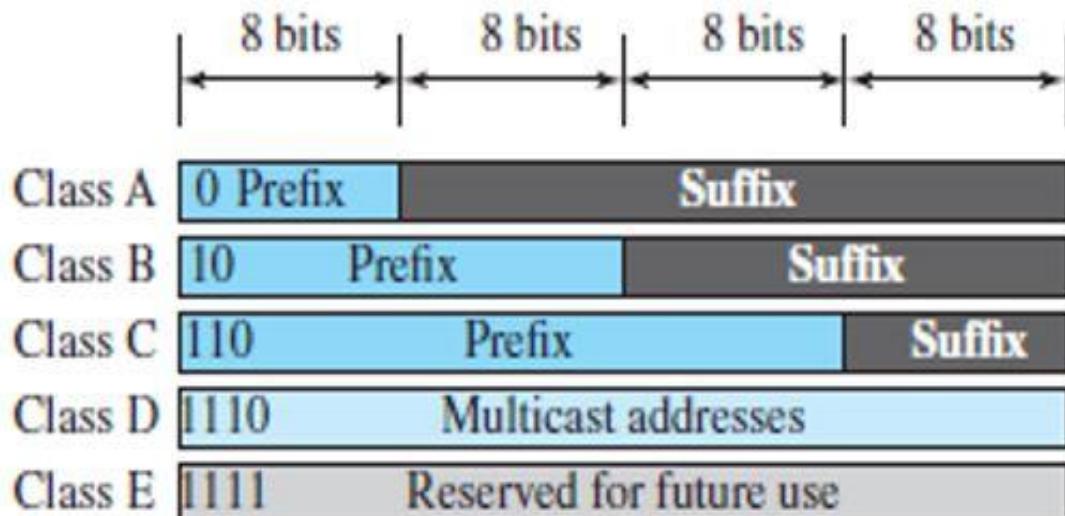


- IPv4 was first designed as a fixed-length prefix and is referred to as classful addressing
- In classful addressing, the address space is divided into five classes: A, B, C, D, and E. Each class occupies some part of the address space.
- To accommodate both small and large networks, three fixed-length prefixes were designed ( $n = 8$ ,  $n = 16$ , and  $n = 24$ ).



## Class A

- In Class A NetID = 8 bits and HostID = 24.
- How to identify class A address
  - First bit is reserved to 0 in binary notation
  - Range of 1<sup>st</sup> octet is [0, 127] in dotted decimal notation

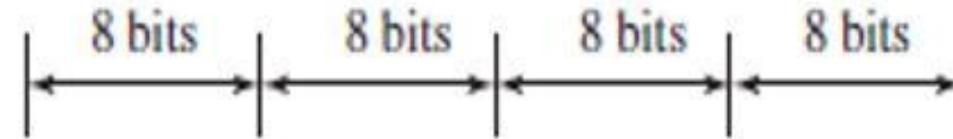


Class	Prefixes	First byte
A	$n = 8$ bits	0 to 127
B	$n = 16$ bits	128 to 191
C	$n = 24$ bits	192 to 223
D	Not applicable	224 to 239
E	Not applicable	240 to 255

- Total number of connections in class A is  $2^{31}$  (2,14,74,83,648)
- There are  $2^7 - 2 = 126$  networks in Class A network.
  - In Class A, total network available are 2 less, because:
  - IP Address 0.0.0.0 is reserved for broadcasting requirements
  - IP Address 127.0.0.1 is reserved for loopback address used for software testing.
  - The range of 1<sup>st</sup> octet is [0, 127] but since two addresses are reserved it is: [1, 126].
- There are  $2^{24} - 2$  (1,67,77,214) HostID in Class A.
  - In all the classes, total number of hosts that can be configured are 2 less because:
  - This is to account for the two reserved IP addresses in which all the bits for host ID are either zero or one.
  - When all Host ID bits are 0, it represents the Network ID for the network.
  - When all Host ID bits are 1, it represents the Broadcast Address.
- Class A is used by organizations requiring very large size networks like Indian Railways.

## Class B

- In Class B NetID = 16 bits and HostID = 16.
- How to identify class B address
  - First two bits are reserved to 10 in binary notation
  - Range of 1<sup>st</sup> octet is [128, 191] in dotted decimal notation
- Total number of connections in class B is  $2^{30}$  (1,07,37,41,824)
- Total number of networks available in class B is  $2^{14}$  (16,384)
- Total number of hosts that can be configured in every network in class B is  $2^{16} - 2$  (65,534)
- Class B is used by organizations requiring medium size networks

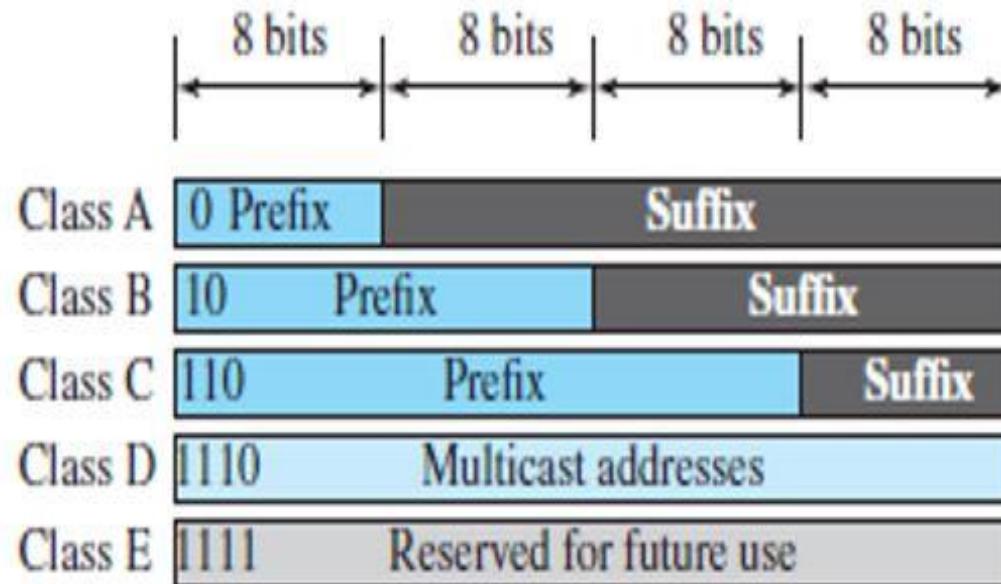


Class	Prefix	Suffix
Class A	0	
Class B	10	
Class C	110	
Class D	1110	Multicast addresses
Class E	1111	Reserved for future use

Class	Prefixes	First byte
A	$n = 8$ bits	0 to 127
B	$n = 16$ bits	128 to 191
C	$n = 24$ bits	192 to 223
D	Not applicable	224 to 239
E	Not applicable	240 to 255

# Class C

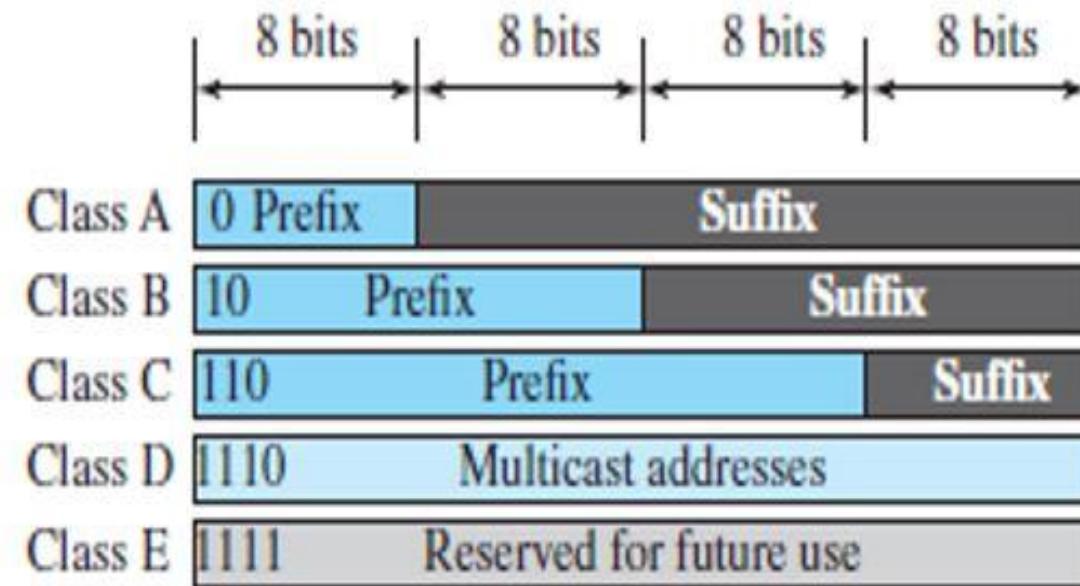
- In Class C NetID = 24 bits and HostID = 8.
- How to identify class C address
  - First three bits are reserved to 110 in binary notation
  - Range of 1<sup>st</sup> octet is [192, 223] in dotted decimal notation
- Total number of connections in class C is  $2^{29}$  (53,68,70,912)
- Total number of networks available in class C is  $2^{21}$  (20,97,152)
- Total number of hosts that can be configured in every network in class C is  $2^8 - 2$  (254)
- Class C is used by organizations requiring small to medium size networks.



Class	Prefixes	First byte
A	$n = 8$ bits	0 to 127
B	$n = 16$ bits	128 to 191
C	$n = 24$ bits	192 to 223
D	Not applicable	224 to 239
E	Not applicable	240 to 255

## Class D

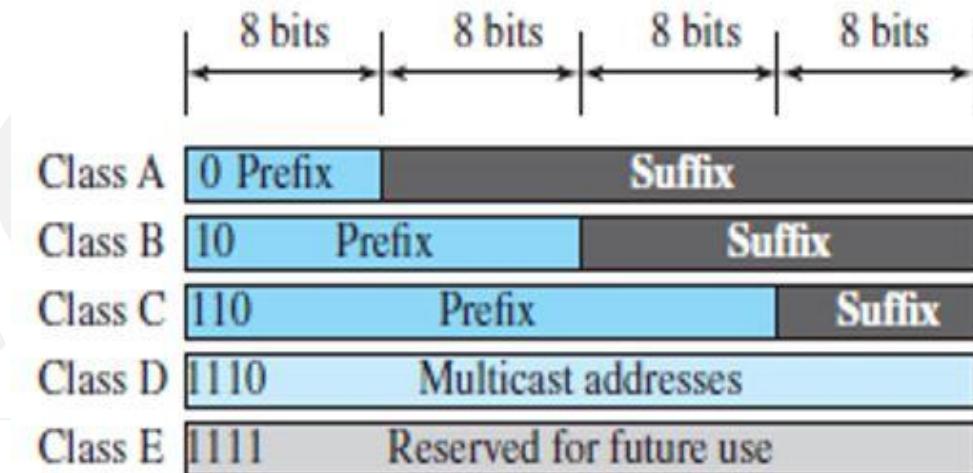
- Class D is not divided into Network ID and Host ID.
- How to identify class D address
  - First four bits are reserved to 1110 in binary notation
  - Range of 1<sup>st</sup> octet is [224, 239] in dotted decimal notation
- Total number of IP Addresses available in class D =  $2^{28}$  (26,84,35,456)
- Class D is reserved for multicasting, in multicasting, there is no need to extract host address from the IP Address, this is because data is not destined for a particular host.



Class	Prefixes	First byte
A	$n = 8$ bits	0 to 127
B	$n = 16$ bits	128 to 191
C	$n = 24$ bits	192 to 223
D	Not applicable	224 to 239
E	Not applicable	240 to 255

# Class E

- Class E is not divided into Network ID and Host ID.
- How to identify class E address
  - First four bits are reserved to 1111 in binary notation
  - Range of 1<sup>st</sup> octet is [240, 255] in dotted decimal notation
- Total number of IP Addresses available in class E =  $2^{28}$  (26,84,35,456)
- Class E is reserved for future or experimental purposes.



Class	Prefixes	First byte
A	$n = 8$ bits	0 to 127
B	$n = 16$ bits	128 to 191
C	$n = 24$ bits	192 to 223
D	Not applicable	224 to 239
E	Not applicable	240 to 255

## **Points to note**

- All the hosts in a single network always have the same network ID but different Host ID.
- Two hosts in two different networks can have the same host ID.
- Only those devices which have the network layer will have IP Address, switches, hubs and repeaters does not have any IP Address.

**Q** In the IPv4 addressing format, the number of networks allowed under Class C addresses is **(Gate-2012) (1 Marks)**

(A)  $2^{14}$

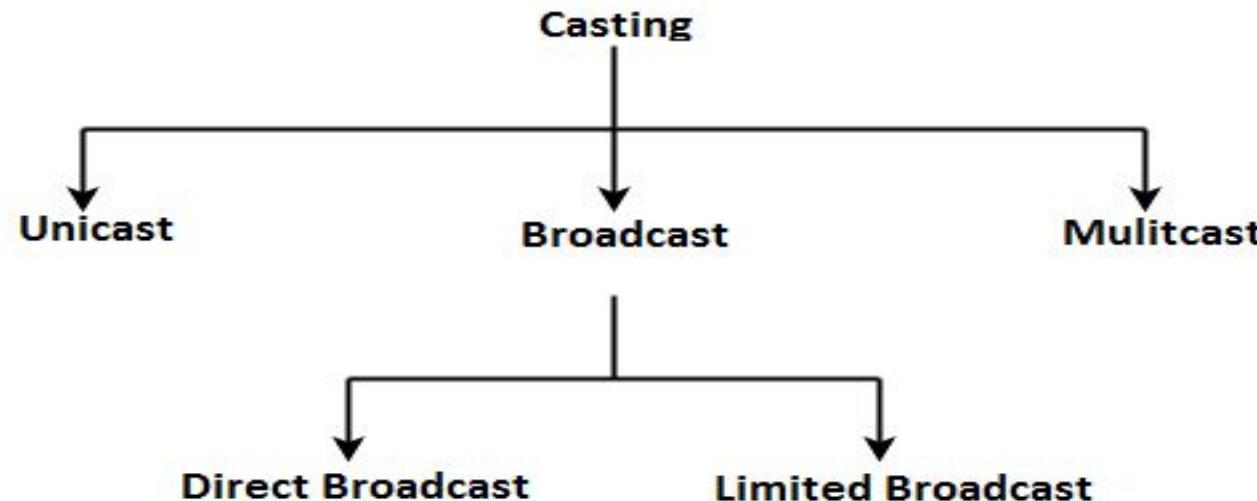
(B)  $2^7$

(C)  $2^{21}$

(D)  $2^{24}$

# Casting in Networks

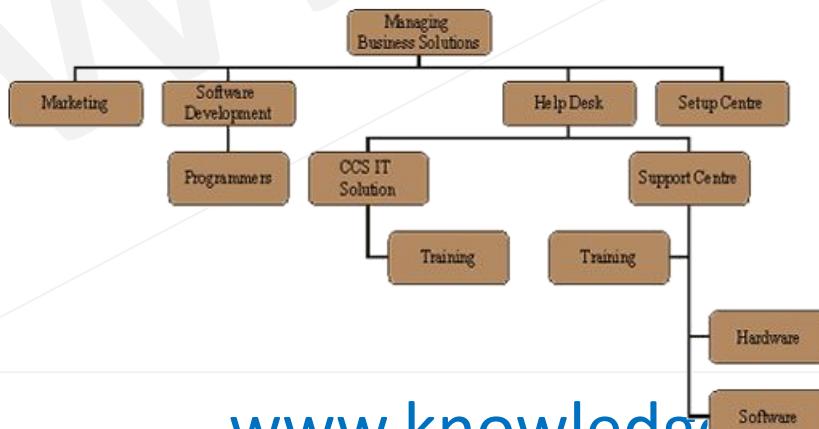
- Casting in networking refers to the way data is transmitted between devices. There are three primary types:
  - **Unicast:** One-to-one communication, where a single sender transmits data to a single receiver.
  - **Multicast:** One-to-many communication, where a single sender transmits data to multiple specific receivers.
  - **Broadcast:** One-to-all communication, where a single sender transmits data to all devices on the network.



- **Unicast:** Transmitting data from one source host to one destination host is called as **unicast**. It is a one to one transmission.
- **Broadcast:** Transmitting data from one source host to all other hosts residing in a network either same or other network is called as **broadcast**. It is a one to all transmission.
  - **Limited Broadcast:** Transmitting data from one source host to all other hosts residing in the same network is called as limited broadcast. Limited Broadcast Address for any network is All 32 bits set to 1 =  
 $11111111.11111111.11111111.11111111 = 255.255.255.255$
  - **Direct Broadcast:** Transmitting data from one source host to all other hosts residing in some other network is called as direct broadcast. Direct Broadcast Address for any network is the IP Address where, Network ID is the IP Address of the network where all the destination hosts are present and Host ID bits are all set to 1.
- **Multicast:** Transmitting data from one source host to a particular group of hosts having interest in receiving the data is called as multicast. It is a one to many transmissions.

# Reason For Subnetting

- Subnetting is implemented to improve network management, security, and efficiency in large networks. Here's why subnetting is essential:
- Challenges of Large Networks**
  - Difficult Maintenance:** Managing large networks like Class A or Class B is challenging for network administrators due to their size.
  - Security Concerns:** Placing all computers from different departments on the same network compromises security from a company's perspective.
- Purpose of Subnetting**
  - Dividing Address Blocks:** Organizations with large address blocks (Class A or B) can divide them into **smaller subnets** to simplify management and improve security.
  - Network Segmentation:** Each subnet can be assigned to different departments or purposes, creating isolated and secure network segments.

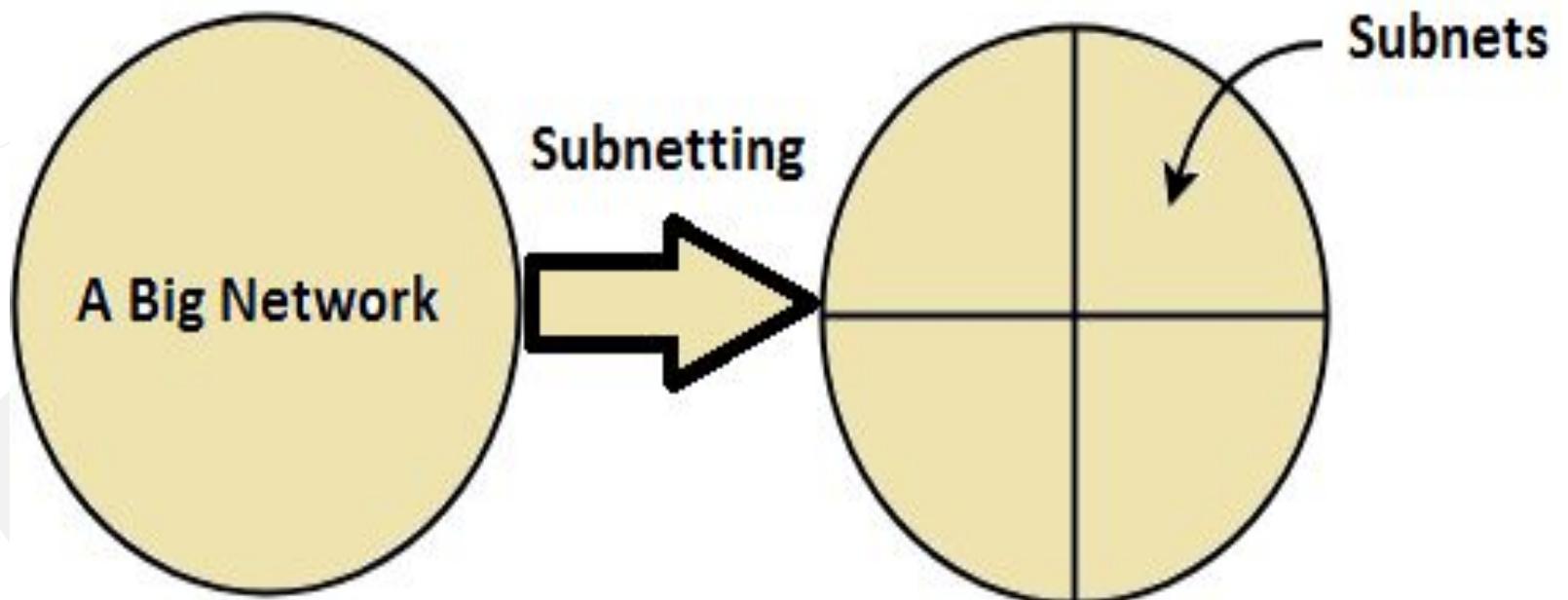


## Advantages

- It improves the security.
- The maintenance and administration of subnets is easy.

## Disadvantages

- Identification of a station is difficult
- Not possible to directed broadcast from outside network.



**Q Consider the network having IP Address 200.1.2.0. Divide this network into two subnets.**

**Q Consider the network having IP Address 200.1.2.0. Divide this network into two subnets.**

## **1st Subnet**

- IP Address of the subnet / Subnet id = 200.1.2.0
- Direct Broadcast Address = 200.1.2.01111111 = 200.1.2.127
- Total number of IP Addresses =  $2^7 = 128$
- Range of IP Addresses = [200.1.2.0, 200.1.2.127]
- Total number of hosts that can be configured =  $128 - 2 = 126$
- Range of Allocated IP Addresses = [200.1.2.1, 200.1.2.126]

## **2nd Subnet**

- IP Address of the subnet / Subnet id = 200.1.2.128
- Direct Broadcast Address = 200.1.2.11111111 = 200.1.2.255
- Total number of IP Addresses =  $2^7 = 128$
- Range of IP Addresses = [200.1.2.128, 200.1.2.255]
- Total number of hosts that can be configured =  $128 - 2 = 126$
- Range of Allocated IP Addresses = [200.1.2.129, 200.1.2.254]

**Q** Consider we have a big single network having IP Address 200.1.2.0. We want to do subnetting and divide this network into 4 subnets.

**Q Consider we have a big single network having IP Address 200.1.2.0. We want to do subnetting and divide this network into 4 subnets.**

#### **1st Subnet**

- IP Address of the subnet / Subnet id = 200.1.2.0
- Direct Broadcast Address = 200.1.2.0**00111111** = 200.1.2.63
- Total number of IP Addresses =  $2^6 = 64$
- Range of IP Addresses = [200.1.2.0, 200.1.2.63]
- Total number of hosts that can be configured =  $64 - 2 = 62$
- Range of Allocated IP Addresses = [200.1.2.1, 200.1.2.62]

#### **2nd Subnet**

- IP Address of the subnet / Subnet id = 200.1.2.64
- Direct Broadcast Address = 200.1.2.0**11111111** = 200.1.2.127
- Total number of IP Addresses =  $2^6 = 64$
- Range of IP Addresses = [200.1.2.64, 200.1.2.127]
- Total number of hosts that can be configured =  $64 - 2 = 62$
- Range of Allocated IP Addresses = [200.1.2.65, 200.1.2.126]

#### **3rd Subnet**

- IP Address of the subnet / Subnet id = 200.1.2.128
- Direct Broadcast Address = 200.1.2.**10111111** = 200.1.2.191
- Total number of IP Addresses =  $2^6 = 64$
- Range of IP Addresses = [200.1.2.128, 200.1.2.191]
- Total number of hosts that can be configured =  $64 - 2 = 62$
- Range of Allocated IP Addresses = [200.1.2.129, 200.1.2.190]

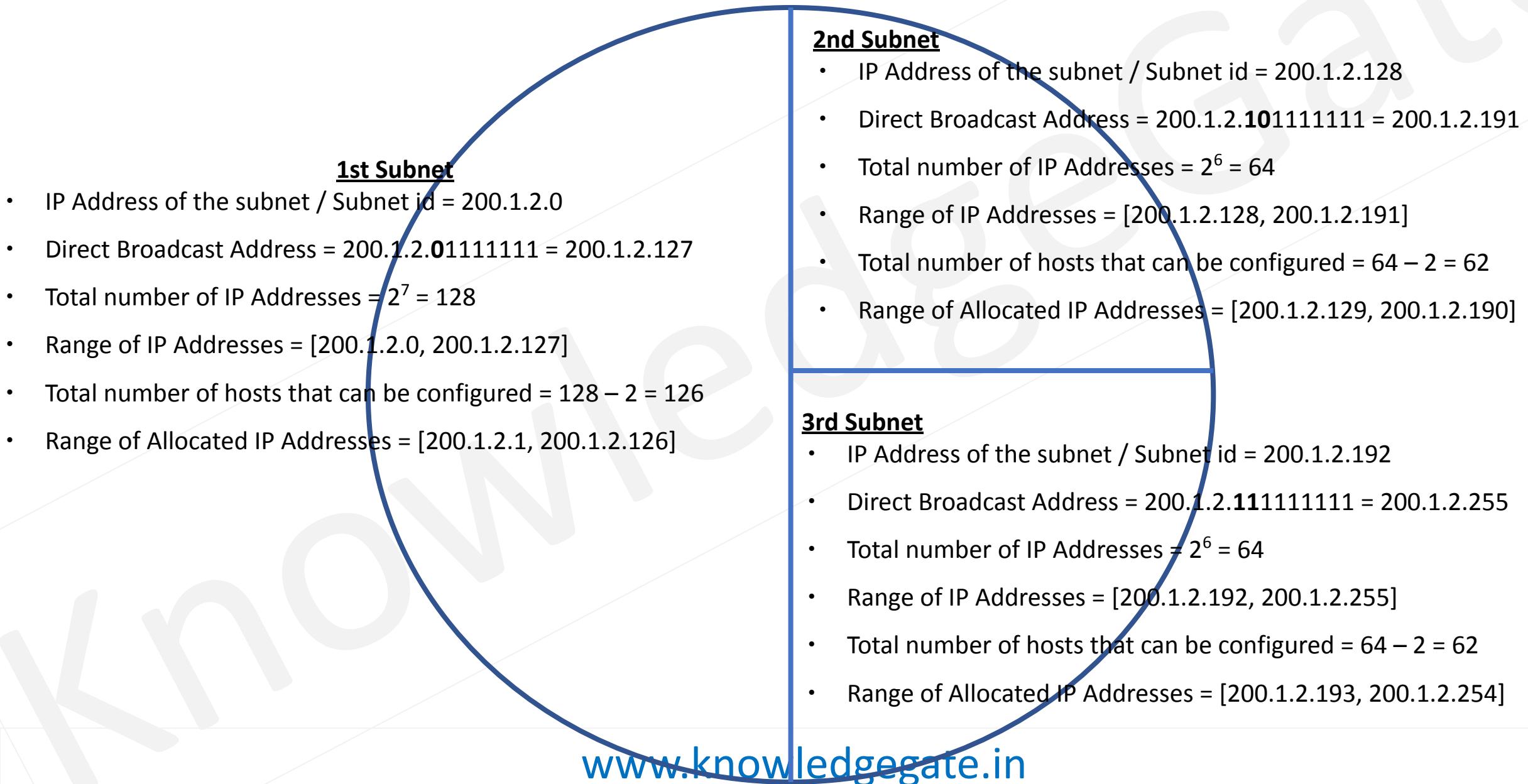
#### **4th Subnet**

- IP Address of the subnet / Subnet id = 200.1.2.192
- Direct Broadcast Address = 200.1.2.**11111111** = 200.1.2.255
- Total number of IP Addresses =  $2^6 = 64$
- Range of IP Addresses = [200.1.2.192, 200.1.2.255]
- Total number of hosts that can be configured =  $64 - 2 = 62$
- Range of Allocated IP Addresses = [200.1.2.193, 200.1.2.254]

- **Types of Subnetting**
  - **Fixed Length Subnetting** Fixed length subnetting (classful subnetting) divides the network into subnets such that:
    - All the subnets are of same size.
    - All the subnets have equal number of hosts.
    - All the subnets have same subnet mask.
  - **Variable Length Subnetting** Variable length subnetting (classless subnetting) divides the network into subnets such that:
    - All the subnets are not of same size.
    - All the subnets do not have equal number of hosts.
    - All the subnets do not have same subnet mask.

Q Consider we have a big single network having IP Address 200.1.2.0. We want to do subnetting and divide this network into 3 subnets, such that first contains 126 hosts, and other two contains 62 hosts each?

**Q** Consider we have a big single network having IP Address 200.1.2.0. We want to do subnetting and divide this network into 3 subnets, such that first contains 126 hosts, and other two contains 62 hosts each?



**Q** Consider we have a big single network having IP Address 200.1.2.0. We want to do subnetting and divide this network into 3 subnets, such that first contains 126 hosts, and other two contains 62 hosts each?

#### 1st Subnet

- IP Address of the subnet / Subnet id = 200.1.2.0
- Direct Broadcast Address = 200.1.2.**00111111** = 200.1.2.63
- Total number of IP Addresses =  $2^6 = 64$
- Range of IP Addresses = [200.1.2.0, 200.1.2.63]
- Total number of hosts that can be configured =  $64 - 2 = 62$
- Range of Allocated IP Addresses = [200.1.2.1, 200.1.2.62]

#### 3rd Subnet

- IP Address of the subnet / Subnet id = 200.1.2.128
- Direct Broadcast Address = 200.1.2.**11111111** = 200.1.2.255
- Total number of IP Addresses =  $2^7 = 128$
- Range of IP Addresses = [200.1.2.128, 200.1.2.255]
- Total number of hosts that can be configured =  $128 - 2 = 126$
- Range of Allocated IP Addresses = [200.1.2.129, 200.1.2.254]

#### 2nd Subnet

- IP Address of the subnet / Subnet id = 200.1.2.64
- Direct Broadcast Address = 200.1.2.**01111111** = 200.1.2.127
- Total number of IP Addresses =  $2^6 = 64$
- Range of IP Addresses = [200.1.2.64, 200.1.2.127]
- Total number of hosts that can be configured =  $64 - 2 = 62$
- Range of Allocated IP Addresses = [200.1.2.65, 200.1.2.126]

## Subnet Masks

- In case of subnetting the problem is how to identify to which subnet the incoming packet from outside the network must be delivered. To solve this problem, we use the idea of subnet mask.
- Subnet mask is a 32-bit number which is a sequence of 1's followed by a sequence of 0's where:
  - 1's represents the Network ID part along with the subnet ID.
  - 0's represents the host ID part.
- Default mask for different classes of IP Address are:
  - Default subnet mask of Class A = 255.0.0.0
  - Default subnet mask for Class B = 255.255.0.0
  - Default subnet mask for Class C = 255.255.255.0
- Networks of same size always have the same subnet mask.

**Q** Suppose computers A and B have IP addresses 10.105.1.113 and 10.105.1.91 respectively and they both use the same netmask N. Which of the values of N given below should not be used if A and B should belong to the same network? **(Gate-2010) (2 Marks)**

10.105.1.113 = 00001010 01101001 00000001 01110001

10.105.1.91 = 00001010 01101001 00000001 01011011

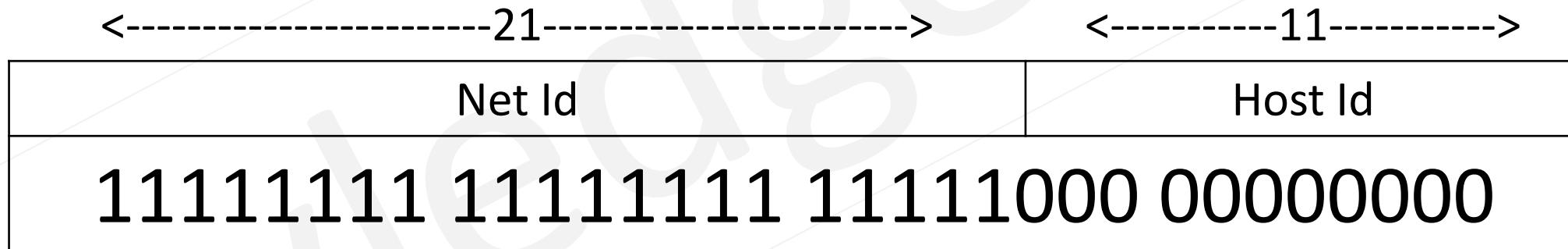
- a) 255.255.255.0 11111111 11111111 11111111 00000000
- b) 255.255.255.128 11111111 11111111 11111111 10000000
- c) 255.255.255.192 11111111 11111111 11111111 11000000
- d) 255.255.255.224 11111111 11111111 11111111 11100000

**Q** If a class B network on the Internet has a subnet mask of 255.255.248.0, what is the maximum number of hosts per subnet? **(Gate-2008) (2 Marks)**

**(A) 1022**



**(B) 1023**



**(C) 2046**

**(D) 2047**

**Q** Host X has IP address 192.168.1.97 and is connected through two routers  $R_1$  and  $R_2$  to another host Y with IP address 192.168.1.80. Router  $R_1$  has IP addresses 192.168.1.135 and 192.168.1.110.  $R_2$  has IP addresses 192.168.1.67 and 192.168.1.155. The netmask used in the network is 255.255.255.224. Which IP address should X configure its gateway as? **(Gate-2008) (2 Marks)**

**(A)** 192.168.1.67

**(B)** 192.168.1.110

**(C)** 192.168.1.135

**(D)** 192.168.1.155

**Q** The address of a class B host is to be split into subnets with a 6-bit subnet number. What is the maximum number of subnets and the maximum number of hosts in each subnet? **(Gate-2007) (2 Marks)**

**(A)** 62 subnets and 262142 hosts.

**(B)** 64 subnets and 262142 hosts.

**(C)** 62 subnets and 1022 hosts.

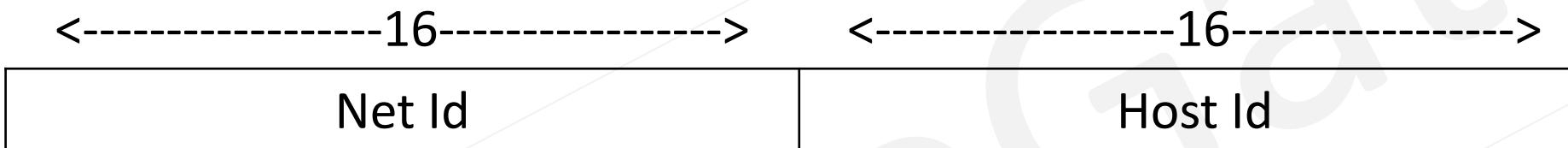
**(D)** 64 subnets and 1024 hosts.

**Q** A sub netted Class B network has the following broadcast address: 144.16.95.255. Its subnet mask **(Gate-2006) (2 Marks)**

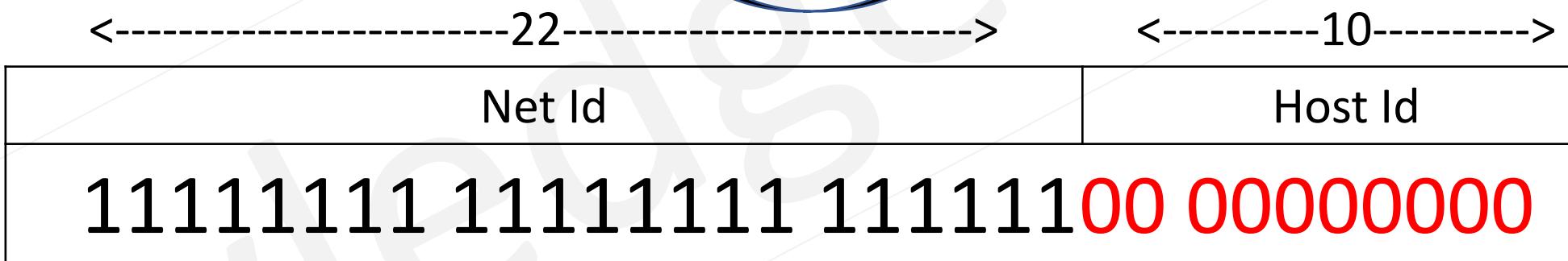
- (A) is necessarily 255.255.224.0
- (B) is necessarily 255.255.240.0
- (C) is necessarily 255.255.248.0
- (D) could be any one of 255.255.224.0, 255.255.240.0, 255.255.248.0

**Q** An organization has a class B network and wishes to form subnets for 64 departments. The subnet mask would be: **(Gate-2005) (1 Marks)**

**(a)** 255.255.0.0



**(b)** 255.255.64.0



**(c)** 255.255.128.0

11111111 11111111 11111111 00 00000000

**(d)** 255.255.252.0

**Q** A company has a class C network address of 204.204.204.0. It wishes to have three subnets, one with 100 hosts and two with 50 hosts each. Which one of the following options represents a feasible set of subnet address/subnet mask pairs? **(Gate-2005) (2 Marks)**

- |  |   |
|--|---|
| <b>(A)</b> 204.204.204.128/255.255.255.192<br>204.204.204.0/255.255.255.128<br>204.204.204.64/255.255.255.128    | <b>(B)</b> 204.204.204.0/255.255.255.192<br>204.204.204.192/255.255.255.128<br>204.204.204.64/255.255.255.128 |
| <b>(C)</b> 204.204.204.128/255.255.255.128<br>204.204.204.192/255.255.255.192<br>204.204.204.224/255.255.255.192 | <b>(D)</b> 204.204.204.128/255.255.255.128<br>204.204.204.64/255.255.255.192<br>204.204.204.0/255.255.255.192 |

<-----24----->      <-----8----->

Net Id	Host Id
1100110011001100 11001100	0 _____
11001100 11001100 11001100	01 _____
11001100 11001100 11001100	00 _____

**Q** A subnet has been assigned a subnet mask of 255.255.255.192. What is the maximum number of hosts that can belong to this subnet? **(Gate-2004) (1 Marks)**

**(A) 14**

	<-----26----->	<-----6----->
	Net Id	Host Id
<b>(B) 30</b>	11111111 11111111 11111111 11	000000

**(C) 62**

**(D) 126**

**Q** The subnet mask for a particular network is 255.255.31.0. Which of the following pairs of IP addresses could belong to this network? **(Gate-2003) (2 Marks)**

**(A)** 172.57.88.62 and 172.56.87.233

**(B)** 10.35.28.2 and 10.35.29.4

10.35.28.2	00001010 00100011 00011100 00000010
255.255.31.0	11111111 11111111 00011111 00000000
10.35.28.0	00001010 00100011 00011100 00000000
10.35.29.4	00001010 00100011 00011100 00000100
255.255.31.0	11111111 11111111 00011111 00000000
10.35.29.0	00001010 00100011 00011101 00000000

**(C)** 191.203.31.87 and 191.234.31.88

**(D)** 128.8.129.43 and 128.8.161.55

128.8.129.43	10000000 00001000 10000001 00101011
255.255.31.0	11111111 11111111 00011111 00000000
128.8.1.0	10000000 00001000 00000001 00000000
128.8.161.55	00001010 00100011 10100001 00110111
255.255.31.0	11111111 11111111 00011111 00000000
128.8.1.0	10000000 00001000 00000001 00000000

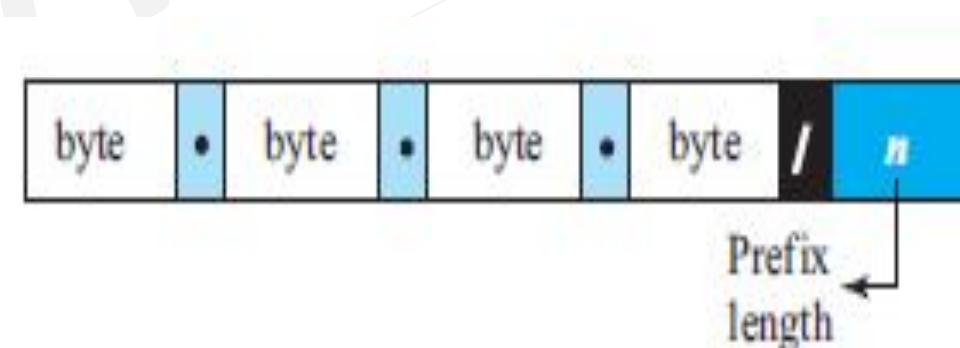
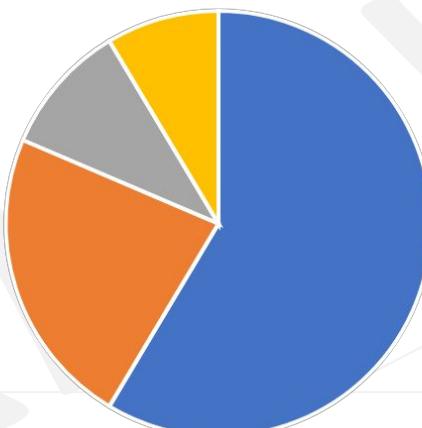
# Address Depletion

- **Problems in Classful Addressing**

- **Improper Distribution:** Classes A and B had large address spaces unsuitable for most organizations, while Class C was often too small.
- **Lack of Flexibility:** Organizations needing a small number of addresses (e.g., 50) had to acquire larger blocks (e.g., 256), leading to wasted addresses.
- **Unused Class E:** Class E addresses were reserved for experimental purposes and almost never used, resulting in a wasted class.

- **Transition to Classless Addressing**

- To address these issues, classless addressing (or **CIDR - Classless Interdomain Routing**) was introduced. Key changes include:
- **No Class Restrictions:** Address allocation is no longer limited to fixed classes. Variable-length blocks are assigned based on actual requirements, improving efficiency.
- **CIDR Notation:** Uses a **slash notation** (e.g., 192.168.1.0/24) to specify the **prefix (net\_id)**, where n indicates the number of bits in the network identifier.



Examples:

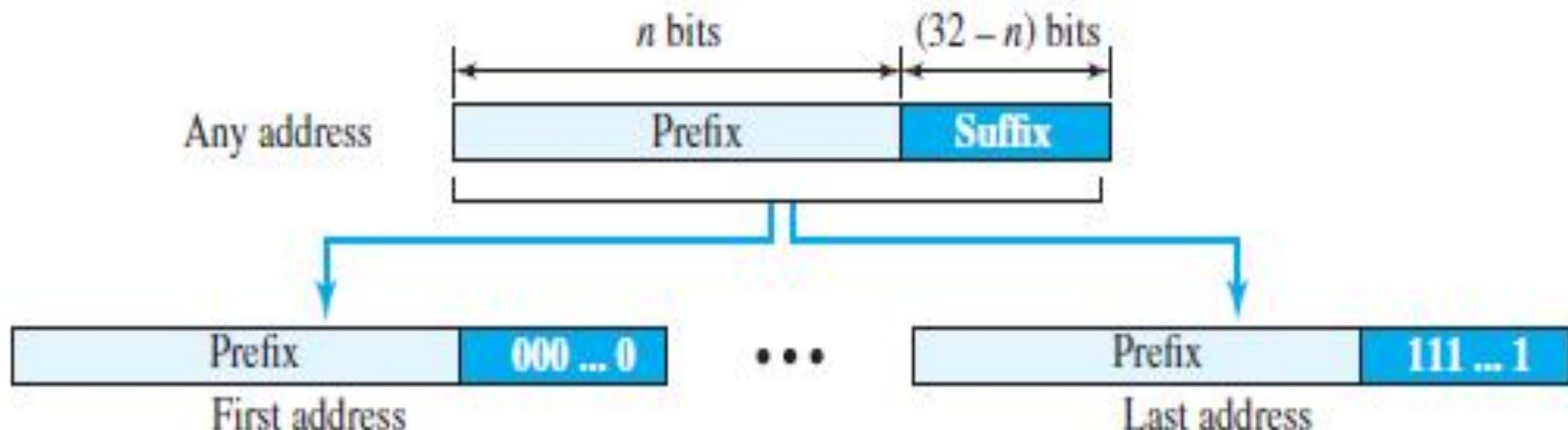
12.24.76.8/8

23.14.67.92/12

220.8.24.255/25

# Extracting Information from an Address

- The number of addresses in the block is found as  $N = 2^{32-n}$ .
- To find the first address, we keep the  $n$  leftmost bits and set the  $(32 - n)$  rightmost bits all to 0s.
- To find the last address, we keep the  $n$  leftmost bits and set the  $(32 - n)$  rightmost bits all to 1s.



**Q** Find the Number of addresses, first and last address of the CIDR block to which Following Address belongs to 167.199.170.82/27 (10100111 11000111 10101010 01010010)

167.199.170.82/27 (10100111 11000111 10101010 01010010)

10100111 11000111 10101010 010\_\_\_\_\_

**No of Address =  $2^5$**

The mask by setting 27 leftmost bits to 1 and 5 rightmost bits to 0 in dotted-decimal notation is:

255.255.255.224 (11111111 11111111 11111111 11100000)

- Number of addresses in the block:  $N = \text{NOT}(\text{mask}) + 1 =$
- $00000000\ 00000000\ 00000000\ 00011111 + 1 = 0.0.0.31 + 1 = 32 \text{ addresses}$

167.199.170.82/27 (10100111 11000111 10101010 01010010)

10100111 11000111 10101010 0100000<sup>64</sup>

10100111 11000111 10101010 0101111<sup>95</sup>

- First address: First = (address) **AND** (mask) =
  - 10100111 11000111 10101010 01010010
  - 11111111 11111111 11111111 11100000
  - 167.199.170.64
- Last address: Last = (address) **OR** (**NOT** mask) =
  - 10100111 11000111 10101010 01010010
  - 00000000 00000000 00000000 00011111
  - 167.199.170.95

## **Rules for Creating CIDR Block (Network)**

- All the IP Addresses in the CIDR block must be contiguous.
- The size of the block (total number of IP Addresses contained in the block) must be presentable as power of 2, size of any CIDR block will always be in the form  $2^1$ ,  $2^2$ ,  $2^3$ ,  $2^4$ ,  $2^5$  and so on. (calculation can be easy)
- First IP Address of the block must be divisible by the size of the block. (so that we get the host id from all 0 to all 1)

# Address Mask

- The address mask is a 32-bit number in which the  $n$  leftmost bits are set to 1s and the rest of the bits ( $32 - n$ ) are set to 0s.
- It is another way to find the first and last addresses in the block.
- Using the three bit-wise operations NOT, AND, and OR a computer can find:
  1. The number of addresses in the block  $N = \text{NOT}(\text{mask}) + 1$ .
  2. The first address in the block = (Any address in the block) **AND** (mask).
  3. The last address in the block = (Any address in the block) **OR** [**NOT** (mask)].

**Q** In the network 200.10.11.144/27, the fourth octet (in decimal) of the last IP address of the network which can be assigned to a host is \_\_\_\_\_ (Gate-2015) (2 Marks)

Net Id	Host Id
11001000 00001010 00001011 100	10000
11001000 00001010 00001011 100	00000
11001000 00001010 00001011 100	00001
11001000 00001010 00001011 100	11110(158)
11001000 00001010 00001011 100	11111

**Q** An Internet Service Provider (ISP) has the following chunk of CIDR-based IP addresses available with it:  
245.248.128.0/20. The ISP wants to give half of this chunk of addresses to Organization A, and a quarter to  
Organization B, while retaining the remaining with itself. Which of the following is a valid allocation of addresses to A  
and B? **(Gate-2012) (2 Marks)**

- (A) 245.248.136.0/21 and 245.248.128.0/22
- (B) 245.248.128.0/21 and 245.248.128.0/22
- (C) 245.248.132.0/22 and 245.248.132.0/21
- (D) 245.248.136.0/24 and 245.248.132.0/21

<----- 20 ----->

<----- 12 ----->

Net Id	Host Id
11110101 11111000 1000	0000 00000000
11110101 11111000 100000	00 0000000(B)
11110101 11111000 100001	00 0000000(ISP)
11110101 11111000 100010	00 0000000(A)

**Q. Which one of the following CIDR prefixes exactly represents the range of IP addresses 10.12.2.0 to 10.12.3.255? (Gate 2024 CS) (2 Marks) (MCQ)**

- (a) 10.12.2.0/23
- (b) 10.12.2.0/24
- (c) 10.12.0.0/22
- (d) 10.12.2.0/22

**Q Consider a block of IP Addresses ranging from 100.1.2.32 to 100.1.2.47.**

Is it a CIDR block?

If yes, give the CIDR representation?

#### **Rules for Creating CIDR Block (Network)**

- All the IP Addresses in the CIDR block must be contiguous.
- The size of the block (total number of IP Addresses contained in the block) must be presentable as power of 2, size of any CIDR block will always be in the form  $2^1, 2^2, 2^3, 2^4, 2^5$  and so on. (calculation can be easy)
- First IP Address of the block must be divisible by the size of the block. (so that we get the host id from all 0 to all 1)

**Q Consider a block of IP Addresses ranging from 20.10.30.32 to 20.10.30.63**

- Is it a CIDR block?
- If yes, give the CIDR representation?

#### **Rules for Creating CIDR Block (Network)**

- All the IP Addresses in the CIDR block must be contiguous.
- The size of the block (total number of IP Addresses contained in the block) must be presentable as power of 2, size of any CIDR block will always be in the form  $2^1, 2^2, 2^3, 2^4, 2^5$  and so on. (calculation can be easy)
- First IP Address of the block must be divisible by the size of the block. (so that we get the host id from all 0 to all 1)

**Q Consider a block of IP Addresses ranging from 150.10.20.64 to 150.10.20.127**

- Is it a CIDR block?
- If yes, give the CIDR representation?

#### **Rules for Creating CIDR Block (Network)**

- All the IP Addresses in the CIDR block must be contiguous.
- The size of the block (total number of IP Addresses contained in the block) must be presentable as power of 2, size of any CIDR block will always be in the form  $2^1, 2^2, 2^3, 2^4, 2^5$  and so on. (calculation can be easy)
- First IP Address of the block must be divisible by the size of the block. (so that we get the host id from all 0 to all 1)

## Subnetting in CIDR

Q Consider the network having IP Address 40.30.20.10/25 Divide this network into two subnets.

**Q Consider the network having IP Address 40.30.20.10/25 Divide this network into two subnets.**

## 1st Subnet

- 40.30.20.0001010
- Total number of IP Addresses =  $2^6 = 64$
- First Address of the Subnet = 40.30.20.0000000
- Last Address of the Subnet = 40.30.20.00111111
- Range of IP Addresses = [40.30.20.0, 40.30.20.63]
- Total number of hosts that can be configured =  $64 - 2 = 62$
- Range of Allocated IP Addresses = [40.30.20.1, 40.30.20.62]
- CIDR Representation 40.30.20. \_\_/26

40.30.20.0001010

## 2st Subnet

- 40.30.20.01001010
- Total number of IP Addresses =  $2^6 = 64$
- First Address of the Subnet = 40.30.20.01000000
- Last Address of the Subnet = 40.30.20.01111111
- Range of IP Addresses = [40.30.20.64, 40.30.20.127]
- Total number of hosts that can be configured =  $64 - 2 = 62$
- Range of Allocated IP Addresses = [40.30.20.65, 40.30.20.126]
- CIDR Representation 40.30.20. \_\_/26

0,1,2,3,4,5,-----,63

64,65,66,67,-----,127

**Q** Consider we have a big single network having IP Address 200.1.2.0/24 We want to do subnetting and divide this network into 4 subnets.

**Q** Consider we have a big single network having IP Address 200.1.2.0/24. We want to do subnetting and divide this network into 4 subnets.

#### 1st Subnet

- IP Address of the subnet / Subnet id = 200.1.2.0
- Direct Broadcast Address = 200.1.2.0**00111111** = 200.1.2.63
- Total number of IP Addresses =  $2^6 = 64$
- Range of IP Addresses = [200.1.2.0, 200.1.2.63]
- Total number of hosts that can be configured =  $64 - 2 = 62$
- Range of Allocated IP Addresses = [200.1.2.1, 200.1.2.62]

#### 2nd Subnet

- IP Address of the subnet / Subnet id = 200.1.2.64
- Direct Broadcast Address = 200.1.2.0**11111111** = 200.1.2.127
- Total number of IP Addresses =  $2^6 = 64$
- Range of IP Addresses = [200.1.2.64, 200.1.2.127]
- Total number of hosts that can be configured =  $64 - 2 = 62$
- Range of Allocated IP Addresses = [200.1.2.65, 200.1.2.126]

#### 3rd Subnet

- IP Address of the subnet / Subnet id = 200.1.2.128
- Direct Broadcast Address = 200.1.2.**10111111** = 200.1.2.191
- Total number of IP Addresses =  $2^6 = 64$
- Range of IP Addresses = [200.1.2.128, 200.1.2.191]
- Total number of hosts that can be configured =  $64 - 2 = 62$
- Range of Allocated IP Addresses = [200.1.2.129, 200.1.2.190]

#### 4th Subnet

- IP Address of the subnet / Subnet id = 200.1.2.192
- Direct Broadcast Address = 200.1.2.**11111111** = 200.1.2.255
- Total number of IP Addresses =  $2^6 = 64$
- Range of IP Addresses = [200.1.2.192, 200.1.2.255]
- Total number of hosts that can be configured =  $64 - 2 = 62$
- Range of Allocated IP Addresses = [200.1.2.193, 200.1.2.254]

**Q** Consider we have a big single network having IP Address 200.1.2.0/24. We want to do subnetting and divide this network into 3 subnets, such that first contains 126 hosts, and other two contains 62 hosts each?

**Q** Consider we have a big single network having IP Address 200.1.2.0/24. We want to do subnetting and divide this network into 3 subnets, such that first contains 126 hosts, and other two contains 62 hosts each?

### 1st Subnet

- IP Address of the subnet / Subnet id = 200.1.2.0
- Direct Broadcast Address = 200.1.2.01111111 = 200.1.2.127
- Total number of IP Addresses =  $2^7 = 128$
- Range of IP Addresses = [200.1.2.0, 200.1.2.127]
- Total number of hosts that can be configured =  $128 - 2 = 126$
- Range of Allocated IP Addresses = [200.1.2.1, 200.1.2.126]

### 2nd Subnet

- IP Address of the subnet / Subnet id = 200.1.2.128
- Direct Broadcast Address = 200.1.2.10111111 = 200.1.2.191
- Total number of IP Addresses =  $2^6 = 64$
- Range of IP Addresses = [200.1.2.128, 200.1.2.191]
- Total number of hosts that can be configured =  $64 - 2 = 62$
- Range of Allocated IP Addresses = [200.1.2.129, 200.1.2.190]

### 3rd Subnet

- IP Address of the subnet / Subnet id = 200.1.2.192
- Direct Broadcast Address = 200.1.2.11111111 = 200.1.2.255
- Total number of IP Addresses =  $2^6 = 64$
- Range of IP Addresses = [200.1.2.192, 200.1.2.255]
- Total number of hosts that can be configured =  $64 - 2 = 62$
- Range of Allocated IP Addresses = [200.1.2.193, 200.1.2.254]

**Q** Consider we have a big single network having IP Address 200.1.2.0/24. We want to do subnetting and divide this network into 3 subnets, such that first contains 126 hosts, and other two contains 62 hosts each?

#### **1st Subnet**

- IP Address of the subnet / Subnet id = 200.1.2.0
- Direct Broadcast Address = 200.1.2.**00111111** = 200.1.2.63
- Total number of IP Addresses =  $2^6 = 64$
- Range of IP Addresses = [200.1.2.0, 200.1.2.63]
- Total number of hosts that can be configured =  $64 - 2 = 62$
- Range of Allocated IP Addresses = [200.1.2.1, 200.1.2.62]

#### **3rd Subnet**

- IP Address of the subnet / Subnet id = 200.1.2.128
- Direct Broadcast Address = 200.1.2.**11111111** = 200.1.2.255
- Total number of IP Addresses =  $2^7 = 128$
- Range of IP Addresses = [200.1.2.128, 200.1.2.255]
- Total number of hosts that can be configured =  $128 - 2 = 126$
- Range of Allocated IP Addresses = [200.1.2.129, 200.1.2.254]

#### **2nd Subnet**

- IP Address of the subnet / Subnet id = 200.1.2.64
- Direct Broadcast Address = 200.1.2.**01111111** = 200.1.2.127
- Total number of IP Addresses =  $2^6 = 64$
- Range of IP Addresses = [200.1.2.64, 200.1.2.127]
- Total number of hosts that can be configured =  $64 - 2 = 62$
- Range of Allocated IP Addresses = [200.1.2.65, 200.1.2.126]

## Designing subnets for CIDR Notations

- Assume:
  - The total number of addresses granted to the organization is  $N$
  - The prefix length is  $n$
  - The assigned number of addresses to each sub-network is  $N_{\text{sub}}$
  - The prefix length for each sub-network is  $n_{\text{sub}}$ .
- Then,
- The number of addresses in each sub-network should be a power of 2.
- The prefix length for each sub-network should be found using the following formula:  $n_{\text{sub}} = 32 - \log_2 N_{\text{sub}}$
- The starting address in each sub-network should be divisible by the number of addresses in that sub-network. This can be achieved if we first assign addresses to larger sub-networks.

**Q** 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.

- **Disadvantages of Subnetting**
  - **Loss of IP Addresses:** In each subnet, two IP addresses are lost—one for the **network address** and one for the **broadcast address (DBA)**.
  - **Increased Complexity:** The communication process becomes more complex with subnetting.
- **Problems with Subnetting**
  - **Increased Routing Table Size:** Subnetting leads to a larger routing table, making management and processing more complex.
  - **Limitations of Class C Blocks:** A Class C block, with a maximum of 256 addresses, was often insufficient for mid-sized organizations.
- **Counter Idea: Supernetting**
  - **Supernetting as a Solution:** To address the issues of larger routing tables and address limitations, supernetting (aggregation of smaller blocks) was introduced as an alternative.

# Super Netting in Classful addressing

- **Supernetting** is a technique to overcome the limitations of classful addressing by combining multiple smaller blocks into a larger block.
  - **Purpose:** Supernetting allows organizations to combine multiple contiguous network blocks to create a larger range of addresses, referred to as a **super network** or **supernet**.
  - **Example:** An organization needing 1000 addresses can be allocated four contiguous **Class C blocks**. These blocks are combined to form a single, larger supernet.
- **Benefits of Supernetting**
  - **Efficient Allocation:** Reduces the number of network identifiers (IDs) in the mask, making it more scalable for organizations with higher address requirements.
- **Rules for Supernetting in CIDR**
  - **Contiguous Networks:** All networks to be combined must be contiguous.
  - **Divisibility:** The first network ID should be divisible by the size of the block being combined.

**Q Perform CIDR aggregation on the following IP Addresses-**

**128.56.24.0/24**

**128.56.25.0/24**

**128.56.26.0/24**

**128.56.27.0/24**

- Rules for Super netting in CIDR
  - All network should be contiguous
  - first net id should be divisible by size of the block

**Q** Consider the following networks and merger them to have a supernet

200.1.0.0/24

200.1.1.0/24

200.1.2.0/24

200.1.2.0/24

- Rules for Super netting in CIDR
  - All network should be contiguous
  - first net id should be divisible by size of the block

**Q** Consider the following networks and merger them to have a supernet

100.1.2.0/25

100.1.2.128/26

100.1.3.192/26

- Rules for Super netting in CIDR
  - All network should be contiguous
  - first net id should be divisible by size of the block

- ISP provides four things
  - IP address
  - DGW (router which is connecting us to internet)
  - SM (using subnet mask we understand weather the destination is in our network or some other network)
  - DNS

**Q** Two computers  $C_1$  and  $C_2$  are configured as follows.  $C_1$  has IP address 203.197.2.53 and netmask 255.255.128.0.  $C_2$  has IP address 203.197.75.201 and netmask 255.255.192.0. which one of the following statements is true?  
**(Gate-2006) (2 Marks)**

203.197.2.53	11001011 11000101 00000010 00110101
255.255.128.0	11111111 11111111 10000000 00000000
<b>203.197.0.0</b>	<b>11001011 11000101 00000000 00000000</b>
203.197.75.201	11001011 11000101 01001011 11001001
255.255.128.0	11111111 11111111 10000000 00000000
<b>203.197.0.0</b>	<b>11001011 11000101 00000000 00000000</b>

203.197.2.53	11001011 11000101 00000010 00110101
255.255.192.0	11111111 11111111 11000000 00000000
<b>203.197.0.0</b>	<b>11001011 11000101 00000000 00000000</b>
203.197.75.201	11001011 11000101 01001011 11001001
255.255.192.0	11111111 11111111 11000000 00000000
<b>203.197.64.0</b>	<b>11001011 11000101 00100000 00000000</b>

- (A)**  $C_1$  and  $C_2$  both assume they are on the same network
- (B)**  $C_2$  assumes  $C_1$  is on same network, but  $C_1$  assumes  $C_2$  is on a different network
- (C)**  $C_1$  assumes  $C_2$  is on same network, but  $C_2$  assumes  $C_1$  is on a different network
- (D)**  $C_1$  and  $C_2$  both assume they are on different networks.

## **Special address**

- Testing self-connectivity / Loop back address
  - To test internet connection
  - To check server and client relation

## **Private IP address**

- 10.0.0.0 10.255.255.255
- 172.16.0.0 172.31.255.255
- 192.168.0.0 192.168.255.255

Q. Node X has a TCP connection open to node Y. The packets from X to Y go through an intermediate IP router R. Ethernet switch S is the first switch on the network path between X and R. Consider a packet sent from X to Y over this connection. Which of the following statements is/are TRUE about the destination IP and MAC addresses on this packet at the time it leaves X? **(Gate 2024 CS) (1 Mark) (MSQ)**

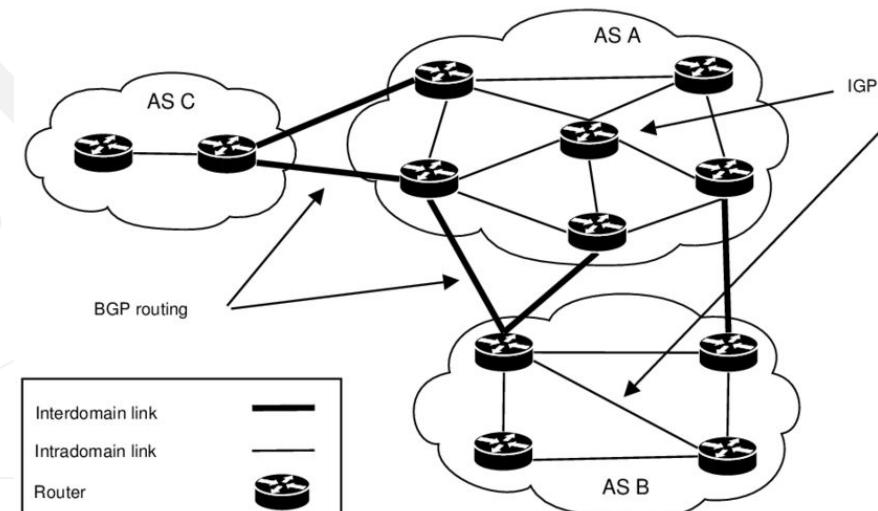
- (a) The destination IP address is the IP address of R
- (b) The destination IP address is the IP address of Y
- (c) The destination MAC address is the MAC address of S
- (d) The destination MAC address is the MAC address of Y

- **Problem and Routing Table**
  - When a router receives an IP packet, it must decide which interface to use to forward the packet towards its destination. This decision is made using a **routing table**.
  - The **routing process** is the task of creating or updating the routing table. Sending a packet through a specific path is referred to as **switching**.
- **Flooding: An Alternative Approach**
  - **Flooding** is a method used to deliver packets to the destination without the need for a routing table. The packet is sent through all possible paths, ensuring that at least one packet reaches the destination.
- **Advantages of Flooding**
  - **No Routing Algorithm Needed:** Flooding does not require a routing algorithm or table.
  - **Shortest Path Guaranteed:** The packet is guaranteed to take the shortest available path.
  - **High Reliability:** Increases the chances of successful packet delivery.
- **Disadvantages of Flooding**
  - **Duplicate Packets:** Multiple copies of the packet can reach the destination and intermediate routers.
  - **Increased Traffic:** Leads to higher network traffic due to the multiple copies.

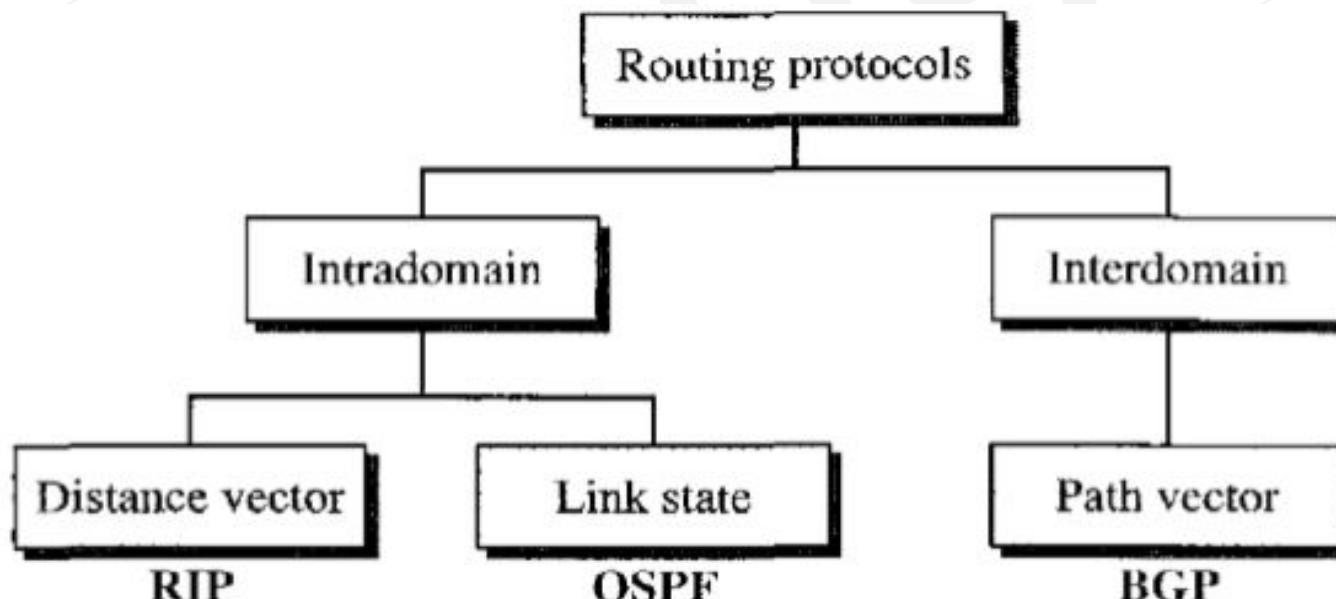
# ROUTING

- **Purpose of a Routing Table:** Contains information about the network and helps decide to which interface an incoming packet should be sent to reach its destination.
- **Types of Routing Tables:** Can be either static or dynamic.
- **Static Routing Table**
  - **Definition:** A static table contains manually configured entries based on known network information. Someone must have complete knowledge of all routers to manually compute and upload the routing information.
  - **Challenges:** Due to the dynamic nature of the internet with new routers being added and old ones going down, static routing is impractical for large-scale networks.
  - **Conclusion:** Static routing is not feasible for complex networks like the Internet.
- **Dynamic Routing Table**
  - **Definition:** A dynamic table automatically updates itself without human intervention based on changes in network topology or traffic patterns.
  - **Advantages:** It adapts to network changes, making it more suitable for the ever-changing nature of large networks like the Internet.

- **Unicast Routing Protocols**
  - **Purpose:** Designed to dynamically update routing tables.
  - **Functionality:** Routers share information to stay informed about network changes and failures. This way, a router can know about the status of a network even if it is far away geographically.
  - **Router Tables:** Protocols allow routers to maintain different routing tables based on the type of service needed.
- **Intra-domain and Interdomain Routing**
  - **Division of Internet:** Large networks are split into autonomous systems (AS) due to the complexity of handling them with a single protocol.
  - **Autonomous System (AS):** A group of networks and routers under one administrative domain.
    - **Intra-domain Routing:** Takes place within an autonomous system using specific protocols.
    - **Interdomain Routing:** Takes place between different autonomous systems using one standard interdomain routing protocol.



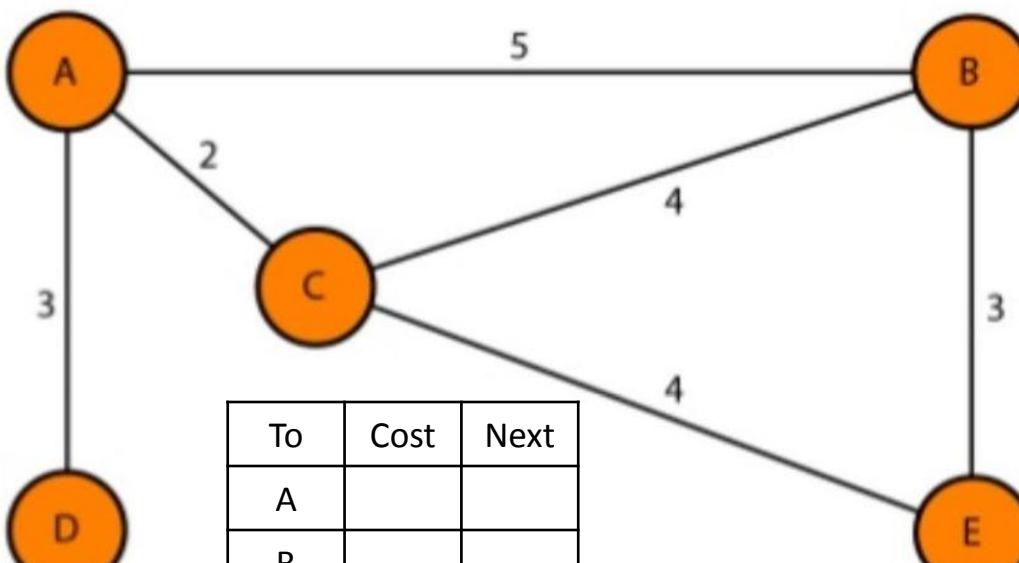
- **Optimum Pathway:** The most efficient or favorable path for data packets to travel from source to destination based on specific criteria (like cost or hop count).
- **Cost:** A metric that represents the value or weight of a path, which varies based on the routing protocol and network configuration.
- **Path Metrics in Routing Protocols**
  - **RIP (Routing Information Protocol):** Assigns the same cost to all networks, measured in hop counts. For example, if a packet crosses through 10 networks, the total cost would be 10 hops.
  - **OSPF (Open Shortest Path First):** Allows the network administrator to set a custom cost based on the type of service required. This means that different routes can have varying costs depending on their attributes and the services needed.



# Distance Vector Routing

- In Distance Vector Routing, each node keeps a routing table containing the minimum distance to other nodes.
- The table includes the next hop, indicating the next node to which a packet should be forwarded.

To	Cost	Next
A		
B		
C		
D		
E		



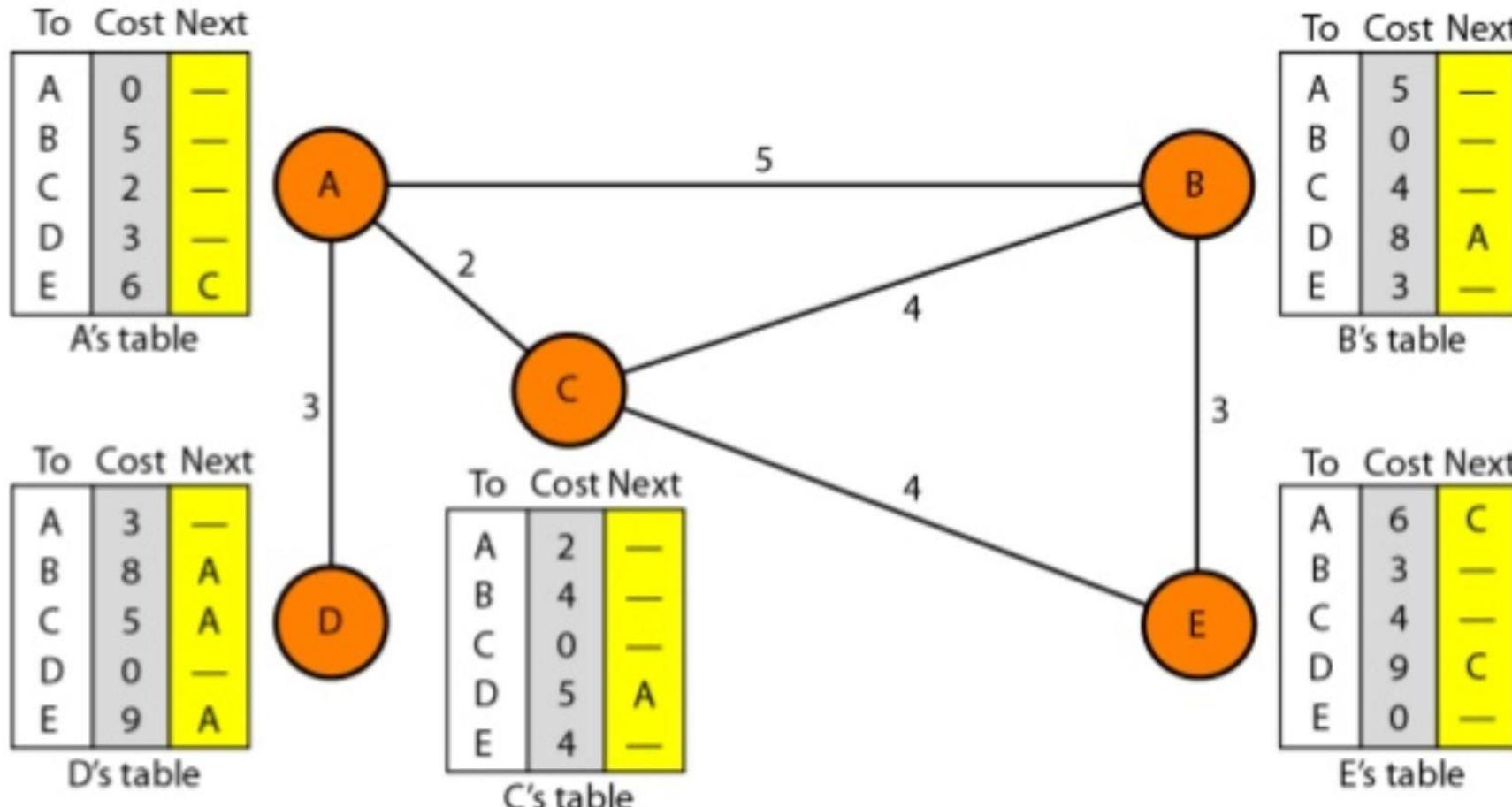
To	Cost	Next
A		
B		
C		
D		
E		

To	Cost	Next
A		
B		
C		
D		
E		

To	Cost	Next
A		
B		
C		
D		
E		

To	Cost	Next
A		
B		
C		
D		
E		

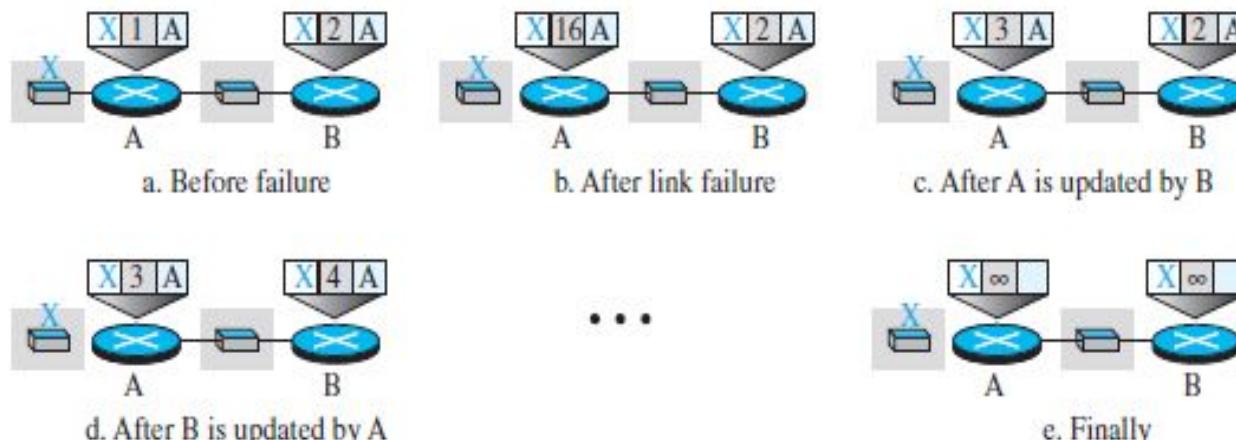
- Each node shares its routing table with its immediate neighbors.
- A node gains knowledge of distant nodes through its neighbor's routing table.
- When a node shares its table, it sends only the first two columns: destination node and distance. The next-hop column isn't useful for neighbors.



- **Update Mechanisms:**
  - **Periodic Updates:** Nodes periodically share their routing tables with neighbors (e.g., every 30 seconds).
  - **Triggered Updates:** These occur when a node detects a change in its routing table due to new information or link failure.
    - **Triggered Update Causes:**
      - Receiving a neighbor's table and detecting a necessary update.
      - Identifying link failures, causing the node to update its distances to infinity.

## Two-Node Loop Instability

- The concept of a **Two-Node Loop Instability** refers to a situation in distance-vector routing where, due to a link failure, nodes A and B mistakenly believe that each has a valid route to a destination (node X) through the other. Here's a simplified explanation:
  - Link Failure and Cost Change:** A link between node A and X breaks, causing node A to recognize a higher cost or an unreachable status to X. However, node B still assumes that A can reach X, based on old information.
  - Loop Problem:** Node A and B start exchanging routing updates, but due to the delay in propagating the correct information (infinite cost to X), each node updates its table based on incorrect assumptions from the other. Node A believes it can reach X via B, and vice versa, causing packets destined for X to bounce between nodes A and B.
  - Count to Infinity Issue:** This gradual increase in cost to infinity while each node adjusts its routes is called the *count to infinity* problem. It results in the system being unstable and packets unnecessarily looping between nodes A and B.

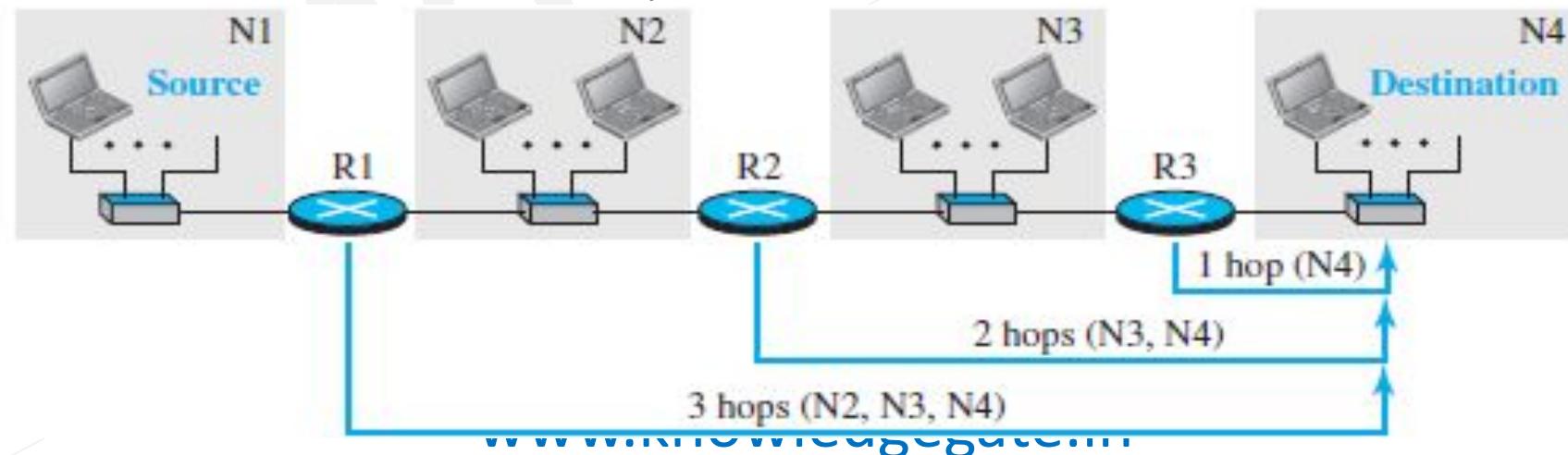


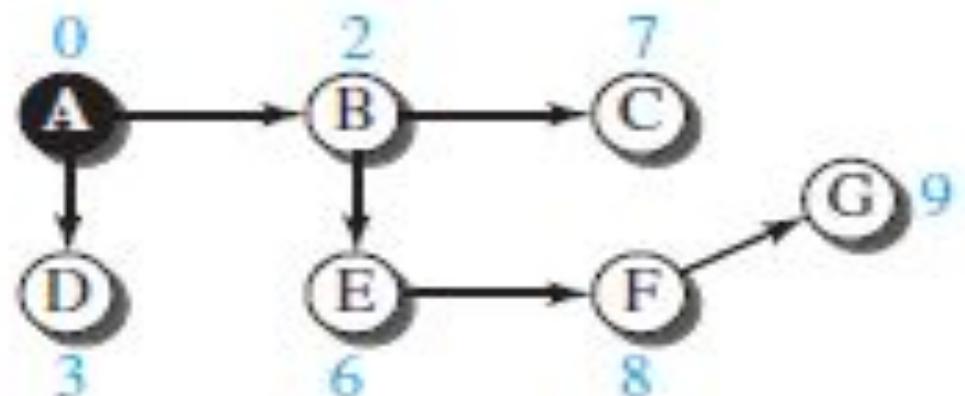
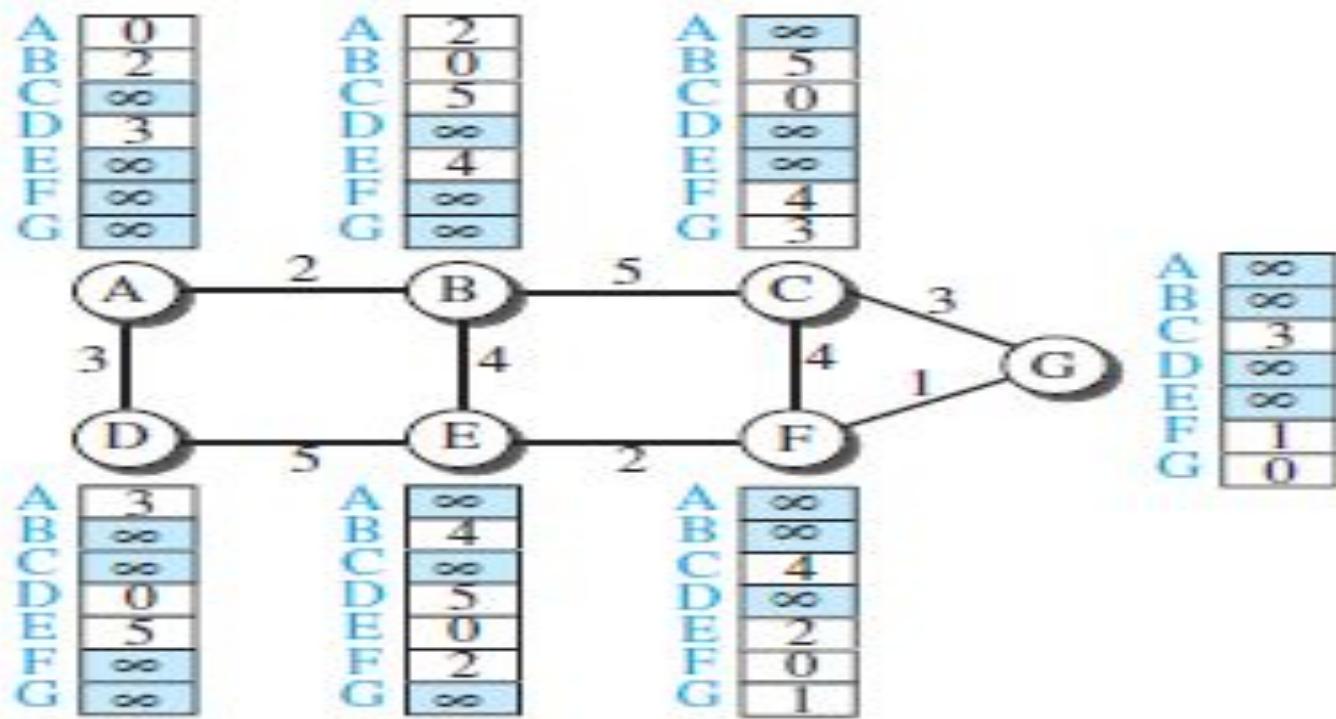
# Split Horizon

- **Split Horizon** is a technique in distance-vector routing to prevent routing loops and stabilize the network. It works by limiting the information shared between nodes:
  - **Selective Information Sharing:** Nodes do not share routing information with the same node from which they received that information. For example, if node B considers node A as the route to X, node B won't tell node A about its route to X, because node A is already aware of it.
  - **Eliminating Redundant Information:** This strategy prevents nodes from inadvertently creating confusion by sending back the information received from a neighbor.
  - **Effect:** This helps nodes like A and B keep the value of infinity for an unreachable destination (e.g., X) without mistakenly updating each other with incorrect routes.

# Routing Information Protocol (RIP)

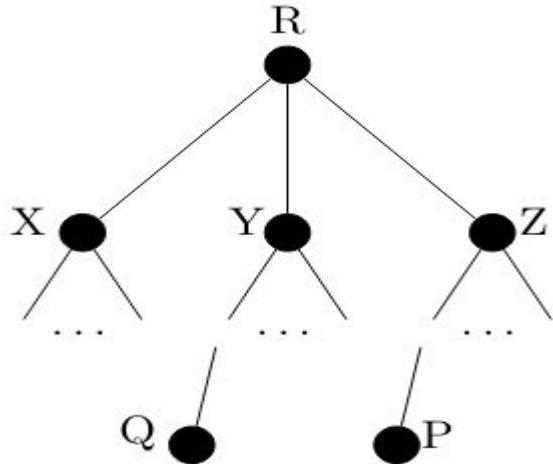
- **Summary of Routing Information Protocol (RIP):**
- **Metric (Hop Count):** RIP uses hop count as a metric, where each hop is a network or subnet a packet passes through. The maximum allowable hops are 15; 16 is considered unreachable (infinity).
- **RIP Implementation:** It works on the application layer using UDP port 520 to update forwarding tables.
- **RIP Algorithm:**
  - Each router shares its entire forwarding table, not just distance vectors.
  - Receivers increment the hop count and update their forwarding tables based on new route costs and next router details.
- **Timers in RIP:**
  - **Periodic Timer:** Prevents simultaneous messages to control traffic.
  - **Expiration Timer:** Set to 180 seconds; if no updates are received within this period, the route is marked expired.
  - **Garbage Collection Timer:** Set to 120 seconds after a route expires, after which it's removed.
- **Performance of RIP:**
  - **Less Traffic:** Minimal update messages reduce network load.
  - **Convergence:** Restricting the maximum hops helps in quick convergence.
  - **Robustness:** Issues in one router affect the entire system due to shared information.





A	0
B	2
C	7
D	3
E	6
F	8
G	9

**Q** Consider a computer network using the distance vector routing algorithm in its network layer. The partial topology of the network is shown below.



The objective is to find the shortest-cost path from the router R to routers P and Q. Assume that R does not initially know the shortest routes to P and Q. Assume that R has three neighbouring routers denoted as X, Y and Z. During one iteration, R measures its distance to its neighbours X, Y and Z as 3, 2 and 5, respectively. Router R gets routing vectors from its neighbours that indicate that the distance to router P from routers X, Y and Z are 7, 6 and 5, respectively. The routing vector also indicates that the distance to router Q from routers X, Y and Z are 4, 6 and 8 respectively. Which of the following statement(s) is/are correct with respect to the new routing table o R, after updation during this iteration? **(GATE 2021) (2 MARKS)**

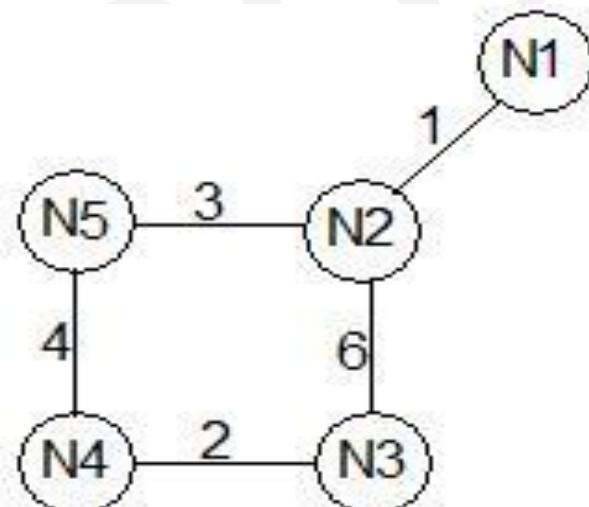
- (A) The distance from R to P will be stored as 10
- (B) The distance from R to Q will be stored as 7
- (C) The next hop router for a packet from R to P is Y
- (D) The next hop router for a packet from R to Q is Z

**Q** Consider a network with five nodes,  $N_1$  to  $N_5$ , as shown below. The network uses a Distance Vector Routing protocol. Once the routes have stabilized, the distance vectors at different nodes are as following.

Each distance vector is the distance of the best known path at that instance to nodes,  $N_1$  to  $N_5$ , where the distance to itself is 0. Also, all links are symmetric and the cost is identical in both directions. In each round, all nodes exchange their distance vectors with their respective neighbours. Then all nodes update their distance vectors. In between two rounds, any change in cost of a link will cause the two incident nodes to change only that entry in their distance vectors. The cost of link  $N_2-N_3$  reduces to 2 (in both directions). After the next round of update what will be the new distance vector at node,  $N_3$ ? **(GATE-2011) (2 Marks)**

$N_1 : (0, 1, 7, 8, 4)$   
 $N_2 : (1, 0, 6, 7, 3)$   
 $N_3 : (7, 6, 0, 2, 6)$   
 $N_4 : (8, 7, 2, 0, 4)$   
 $N_5 : (4, 3, 6, 4, 0)$

- (A)**  $(3, 2, 0, 2, 5)$
- (B)**  $(3, 2, 0, 2, 6)$
- (C)**  $(7, 2, 0, 2, 5)$
- (D)**  $(7, 2, 0, 2, 6)$



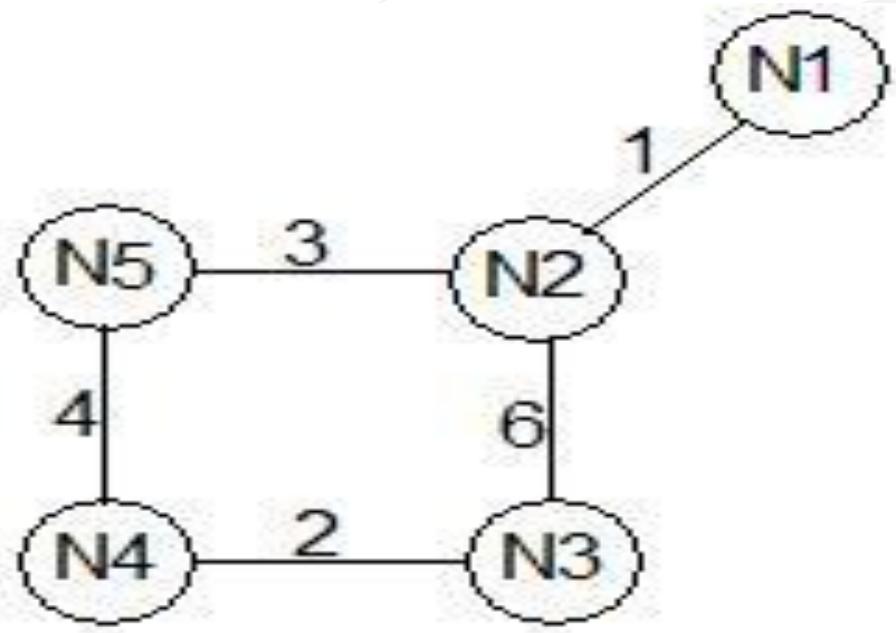
**Q** After the update in the previous question, the link  $N_1-N_2$  goes down.  $N_2$  will reflect this change immediately in its distance vector as cost,  $\infty$ . After the NEXT ROUND of update, what will be cost to  $N_1$  in the distance vector of  $N_3$ ? (GATE-2011) (2 Marks)

(A) 3

(B) 9

(C) 10

(D)  $\infty$



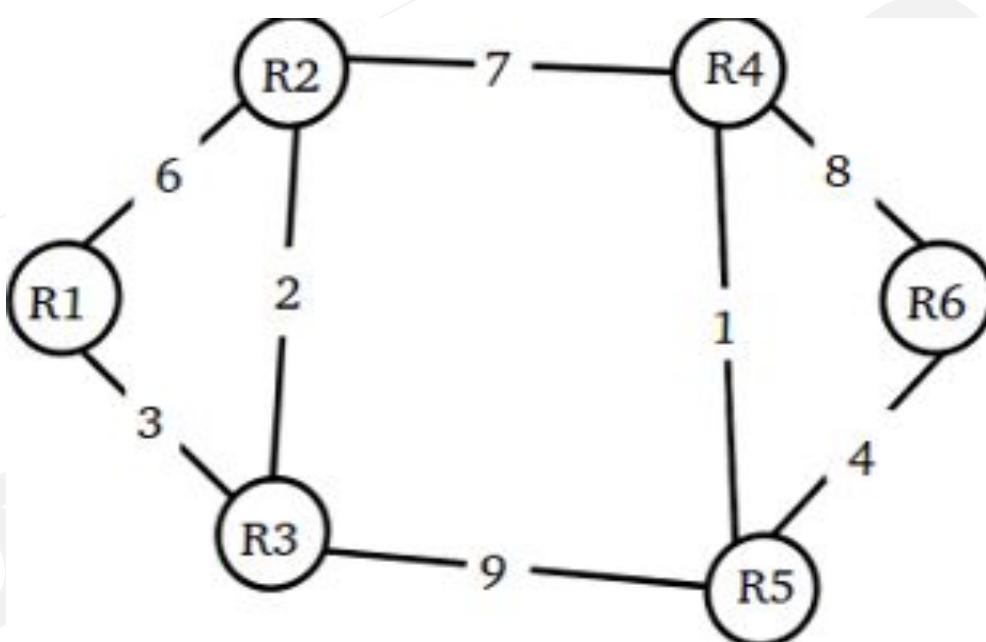
**Q** Consider a network with 6 routers  $R_1$  to  $R_6$  connected with links having weights as shown in the following diagram. All the routers use the distance vector-based routing algorithm to update their routing tables. Each router starts with its routing table initialized to contain an entry for each neighbour with the weight of the respective connecting link. After all the routing tables stabilize, how many links in the network will never be used for carrying any data? **(GATE-2010) (2 Marks)**

**(A) 4**

**(B) 3**

**(C) 2**

**(D) 1**



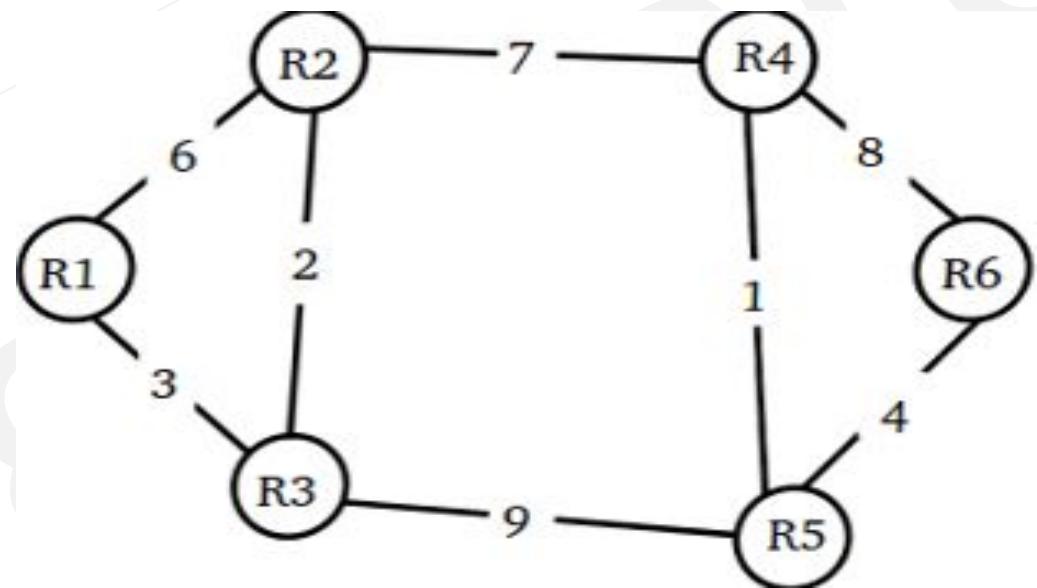
**Q** Suppose the weights of all unused links in the previous question are changed to 2 and the distance vector algorithm is used again until all routing tables stabilize. How many links will now remain unused? (GATE-2010) (2 Marks)

(A) 0

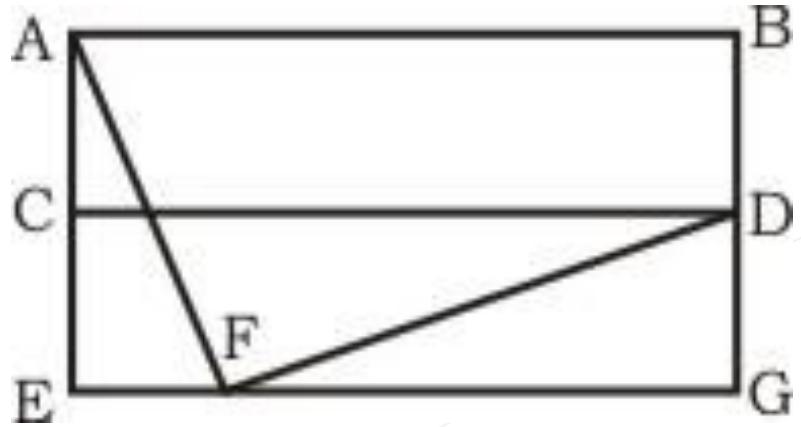
(B) 1

(C) 2

(D) 3



**Q** For the network given in the figure below, the routing tables of the four nodes A, E, D and G are shown. Suppose that F has estimated its delay to its neighbours, A, E, D and G as 8, 10, 12 and 6 msec respectively and updates its routing table using distance vector routing technique. (GATE-2007) (2 Marks)



A	8
B	20
C	17
D	12
E	10
F	0
G	6

a)

A	21
B	8
C	7
D	19
E	14
F	0
G	22

b)

A	8
B	20
C	17
D	12
E	10
F	16
G	6

c)

A	8
B	8
C	7
D	12
E	10
F	0
G	6

d)

Routing Table of A	
A	0
B	40
C	14
D	17
E	21
F	9
G	24

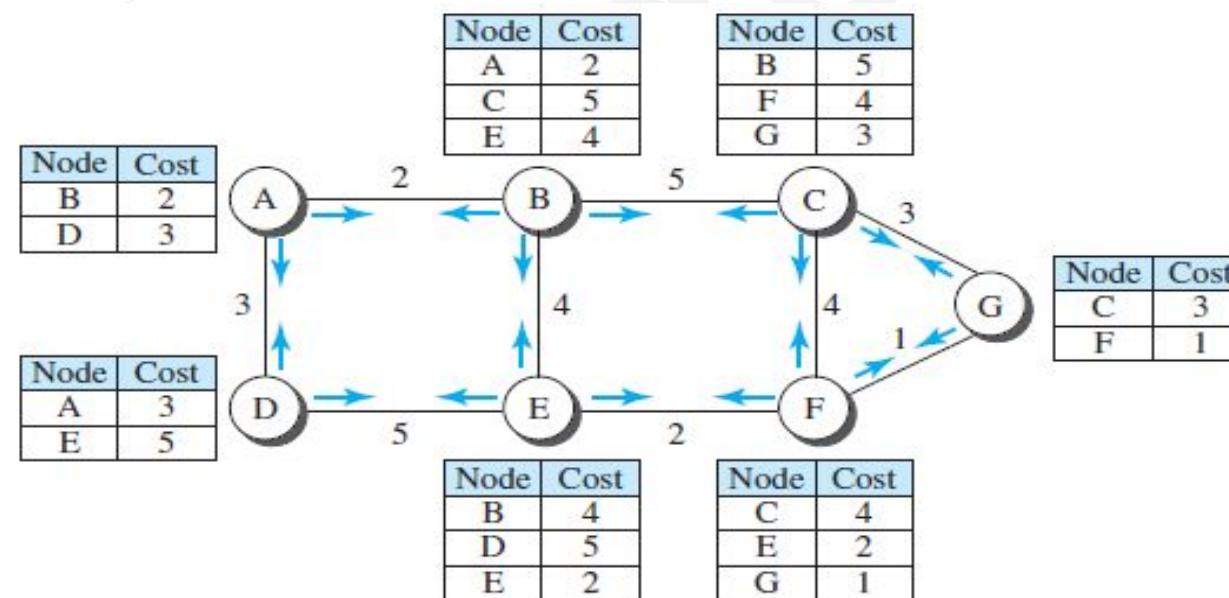
Routing Table of E	
A	24
B	27
C	7
D	20
E	0
F	11
G	22

Routing Table of D	
A	20
B	8
C	30
D	0
E	14
F	7
G	22

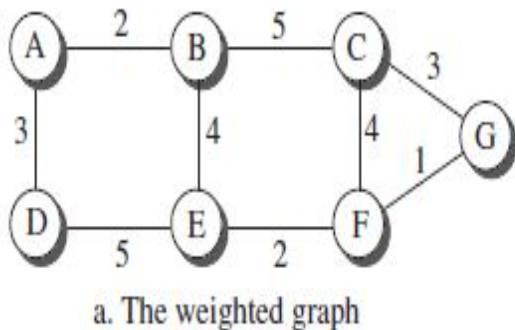
Routing Table of G	
A	21
B	24
C	22
D	19
E	22
F	10
G	0

# Link State Routing

- **Link State Routing** is a type of routing strategy that uses a different approach compared to distance vector routing. It operates based on the idea that each node within a network knows the entire network's topology, which includes:
- **List of nodes and links** along with their type, cost (metric), and condition (up or down).
- The routing process leverages **Dijkstra's Algorithm** to construct the shortest path tree from this network topology, thereby allowing each node to create its routing table. Even though every node has the same topology data, each node's routing table is unique due to the calculation being based on that node as the root of the tree.

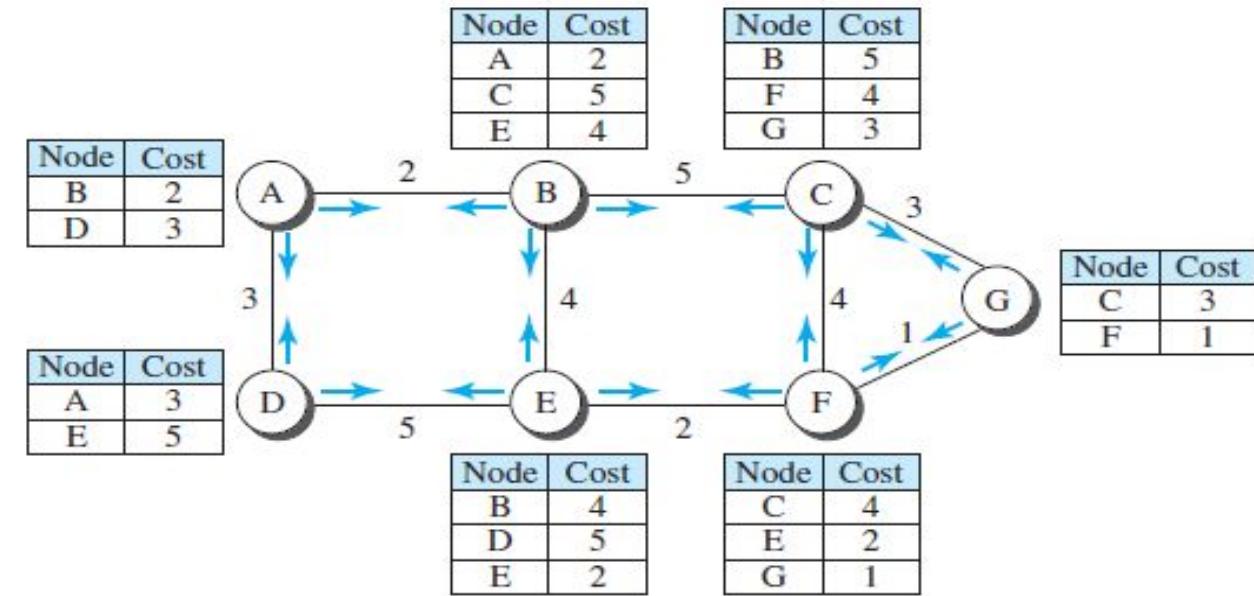


- **Link-State Database (LSDB):** A collection of the state of all links represented as a matrix, storing the cost of each link between nodes.
- **Dijkstra's Algorithm:** An iterative algorithm that starts with a single root node and builds a shortest path tree by adding the closest nodes and updating paths accordingly.

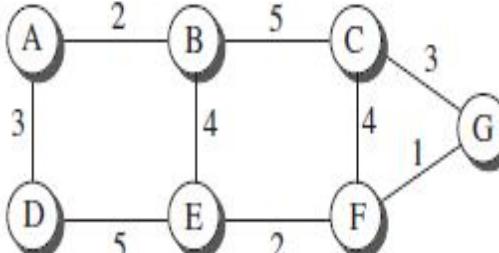


	A	B	C	D	E	F	G
A	0	2	$\infty$	3	$\infty$	$\infty$	$\infty$
B	2	0	5	$\infty$	4	$\infty$	$\infty$
C	$\infty$	5	0	$\infty$	$\infty$	4	3
D	3	$\infty$	$\infty$	0	5	$\infty$	$\infty$
E	$\infty$	4	$\infty$	5	0	2	$\infty$
F	$\infty$	$\infty$	4	$\infty$	2	0	1
G	$\infty$	$\infty$	3	$\infty$	$\infty$	1	0

b. Link state database



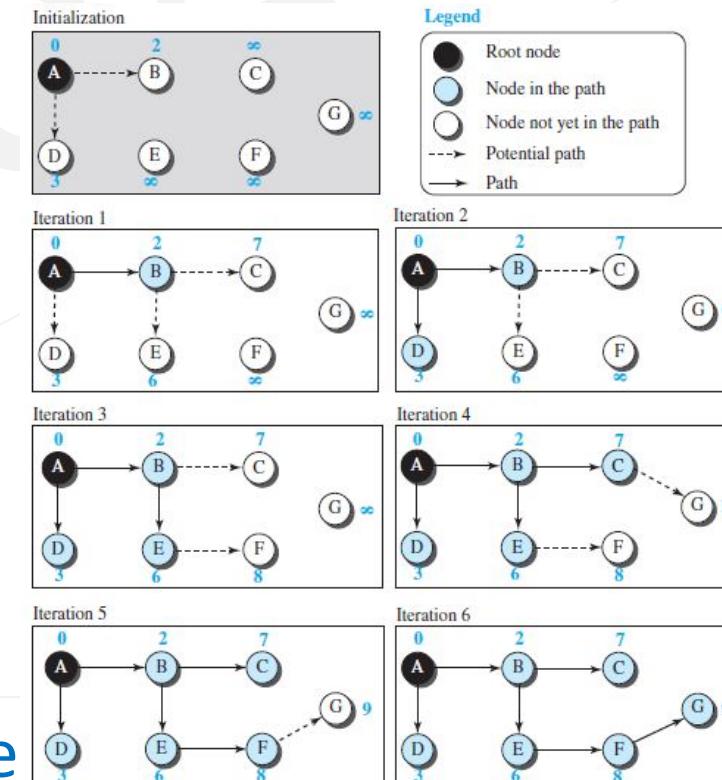
- **Key Steps Involved in Link State Routing**
- **Link State Packet (LSP) Creation:** Each node generates LSPs that describe the state of its direct links (type, condition, and cost).
- **Flooding of LSPs:** LSPs are sent reliably and efficiently to all nodes in the network through a process called flooding. Each node updates its link-state database (LSDB) by keeping the most recent LSPs.
- **Formation of Shortest Path Tree:** Each node uses the LSDB and Dijkstra's Algorithm to create a **shortest path tree** with itself as the root. The algorithm:
  - Divides nodes into tentative and permanent sets.
  - Adds nodes to the tree based on their cumulative cost from the root.
  - Updates the cost of other nodes as paths change.
- **Calculation of Routing Table:** From the shortest path tree, each node calculates its routing table, showing the least-cost path to every other node.



a. The weighted graph

	A	B	C	D	E	F	G
A	0	2	$\infty$	3	$\infty$	$\infty$	$\infty$
B	2	0	5	$\infty$	4	$\infty$	$\infty$
C	$\infty$	5	0	$\infty$	$\infty$	4	3
D	3	$\infty$	$\infty$	0	5	$\infty$	$\infty$
E	$\infty$	4	$\infty$	5	0	2	$\infty$
F	$\infty$	$\infty$	4	$\infty$	2	0	1
G	$\infty$	$\infty$	3	$\infty$	$\infty$	1	0

b. Link state database



# Open Shortest Path First (OSPF)

- **OSPF** is an intradomain routing protocol based on link-state routing, designed to handle routing within large Autonomous Systems (AS).
- **Key Features of OSPF:**
- **Metric Calculation:** The cost (or metric) of a route is based on various factors such as throughput, round-trip time, and reliability. Different service types can have varying weights as costs, providing flexibility.
- **Areas and Backbone Structure:**
  - Large ASs are divided into smaller **areas** to minimize flooding traffic.
  - Each area functions like an independent domain for managing and distributing link state packets (LSPs).
  - A special **backbone area** acts as the central area, connecting all smaller areas and facilitating the distribution of routing information.
- **OSPF Implementation:** OSPF operates at the network layer and leverages IP services for transmitting routing information.
- **Performance Considerations:**
  - **Update Messages:** OSPF uses more complex update messages, which may increase traffic and bandwidth usage in large areas.
  - **Convergence:** Once LSP flooding completes, routers quickly build their shortest path trees and forwarding tables, resulting in a fast convergence.
  - **Robustness:** OSPF is more robust than RIP, as corruption or failure in one router affects the overall network less severely.

**Q** Consider the following statements about the routing protocols, Routing Information Protocol (RIP) and Open Shortest Path First (OSPF) in an IPv4 network.

- I. RIP uses distance vector routing
- II. RIP packets are sent using UDP
- III. OSPF packets are sent using TCP
- IV. OSPF operation is based on link-state routing

Which of the following above are CORRECT? **(Gate-2017) (1 Marks)**

- (A) I and IV only
- (B) I, II and III only
- (C) I, II and IV only
- (D) II, III and IV only

**Q** Which one of the following is TRUE about interior Gateway routing protocols – Routing Information Protocol (RIP) and Open Shortest Path First (OSPF) **(GATE-2014) (1 Marks)**

- (A)** RIP uses distance vector routing and OSPF uses link state routing
- (B)** OSPF uses distance vector routing and RIP uses link state routing
- (C)** Both RIP and OSPF use link state routing
- (D)** Both RIP and OSPF use distance vector routing

**Q** Consider the following three statements about link state and distance vector routing protocols, for a large network with 500 network nodes and 4000 links.

- [S<sub>1</sub>] The computational overhead in link state protocols is higher than in distance vector protocols.
- [S<sub>2</sub>] A distance vector protocol (with split horizon) avoids persistent routing loops, but not a link state protocol.
- [S<sub>3</sub>] After a topology change, a link state protocol will converge faster than a distance vector protocol.

Which one of the following is correct about S<sub>1</sub>, S<sub>2</sub>, and S<sub>3</sub>? **(GATE-2014) (1 Marks)**

- (A) S<sub>1</sub>, S<sub>2</sub>, and S<sub>3</sub> are all true.
- (B) S<sub>1</sub>, S<sub>2</sub>, and S<sub>3</sub> are all false.
- (C) S<sub>1</sub> and S<sub>2</sub> are true, but S<sub>3</sub> is false
- (D) S<sub>1</sub> and S<sub>3</sub> are true, but S<sub>2</sub> is false

**Q** Two popular routing algorithms are Distance Vector (DV) and Link State (LS) routing. Which of the following are true? **(GATE-2008) (2 Marks)**

- (S<sub>1</sub>) Count to infinity is a problem only with DV and not LS routing
  - (S<sub>2</sub>) In LS, the shortest path algorithm is run only at one node
  - (S<sub>3</sub>) In DV, the shortest path algorithm is run only at one node
  - (S<sub>4</sub>) DV requires lesser number of network messages than LS
- 
- (A) S<sub>1</sub>, S<sub>2</sub> and S<sub>4</sub> only
  - (B) S<sub>1</sub>, S<sub>3</sub> and S<sub>4</sub> only
  - (C) S<sub>2</sub> and S<sub>3</sub> only
  - (D) S<sub>1</sub> and S<sub>4</sub> only

## Difference between DVR and LSR

<ul style="list-style-type: none"><li>• In the distance-vector routing algorithm, each router tells its neighbours what it knows about the whole internet.</li><li>• Was most popularly used around 1980's</li><li>• Based on the idea of Local Knowledge</li><li>• Bandwidth requirement is Less</li><li>• Roughly based on the idea of Bellman-Ford Algo</li><li>• Traffic is usually less</li><li>• Converge slowly</li><li>• Counts to Infinity</li><li>• RIP</li></ul>	<ul style="list-style-type: none"><li>• In the link-state routing algorithm, each router tells the whole internet what it knows about its neighbours.</li><li>• Was most popularly used around 1990's</li><li>• Based on the idea of Global Knowledge</li><li>• Bandwidth requirement is high</li><li>• Directly based on the idea of Dijkstra's</li><li>• Traffic is Usually high</li><li>• Converge faster</li><li>• No Counts to infinity</li><li>• OSPF</li></ul>
---	---

- **Routing Table**

- A **Routing Table** is maintained by an internal router to decide on which interface a data packet should be forwarded. It contains:
  - **IP Address** of the destination subnet.
  - **Subnet mask** of the subnet.
  - **Interface** for forwarding.

- **Routing in Subnets**

- When a data packet arrives, the router follows these steps:
  - **Bitwise AND Operation:** The router performs a bitwise AND between the destination IP address of the packet and each subnet mask in the routing table.
  - **Comparison:** The router compares each result with the corresponding IP address in the routing table.
  - **Three Possible Cases:**
    - **Single Match:** The router forwards the packet to the matching interface.
    - **Multiple Matches:** The router forwards the packet to the interface with the longest subnet mask.
    - **No Match:** The router forwards the packet using the default entry.

- **Fixed and Variable-Length Subnetting**

- **Fixed-Length Subnetting:** All subnets have the same subnet mask. Bitwise ANDing is performed once.
- **Variable-Length Subnetting:** Subnets have different subnet masks. The router performs ANDing with each mask and follows the three cases mentioned above.

**Q** Classless Inter-Domain Routing (CIDR) receives a packet with address 131.23.151.76. The router's routing table has the following entries: **(Gate-2014) (2 Marks)**

Prefix	Output Interface Identifier
131.16.0.0/12	3
131.28.0.0/14	5
131.19.0.0/16	2
131.22.0.0/15	1

131.23.151.76	10000011 00010111 10010111 01001100
255.240.0.0	11111111 11110000 00000000 00000000
131.16.0.0	10000011 00010000 00000000 00000000
131.23.151.76	10000011 00010111 10010111 01001100
255.252.0.0	11111111 11111100 00000000 00000000
131.20.0.0	10000011 00010100 00000000 00000000
131.23.151.76	10000011 00010111 10010111 01001100
255.255.0.0	11111111 11111111 00000000 00000000
131.23.0.0	10000011 00010111 00000000 00000000
131.23.151.76	10000011 00010111 10010111 01001100
255.254.0.0	11111111 11111110 00000000 00000000
131.22.0.0	10000011 00010110 00000000 00000000

The identifier of the output interface on which this packet will be forwarded is \_\_\_\_\_.

**Q** A router uses the following routing table:

Destination	Mask	Interface
144.16.0.0	255.255.0.0	eth0
144.16.64.0	255.255.224.0	eth1
144.16.68.0	255.255.255.0	eth2
144.16.68.64	255.255.255.224	eth3

A packet bearing a destination address 144.16.68.117 arrives at the router.  
On which interface will it be forwarded?  
**(Gate-2006) (2 Marks)**

- (A) eth0
- (B) eth1
- (C) eth2
- (D) eth3

144.16.68.117 255.255.0.0 144.16.0.0	10010000 00010000 01000100 01110101 11111111 11111111 00000000 00000000 10010000 00010000 00000000 00000000
144.16.68.117 255.255.224.0 144.16.64.0	10010000 00010000 01000100 01110101 11111111 11111111 11100000 00000000 10010000 00010000 01000000 00000000
144.16.68.117 255.255.255.0 144.16.68.0	10010000 00010000 01000100 01110101 11111111 11111111 11111111 00000000 10010000 00010000 01000100 00000000
144.16.68.117 255.254.255.224 144.16.68.96	10010000 00010000 01000100 01110101 11111111 11111111 11111111 11100000 10010000 00010000 10000100 01100000

**Q** The routing table of a router is shown below:

Destination	Sub net mask	Interface
128.75.43.0	255.255.255.0	Eth0
128.75.43.0	255.255.255.128	Eth1
192.12.17.5	255.255.255.255	Eth3
default		Eth2

On which interfaces will the router forward packets have addressed to destinations 128.75.43.16 and 192.12.17.10 respectively? **(Gate-2004) (2 Marks)**

- (A) Eth1 and Eth2
- (B) Eth0 and Eth2
- (C) Eth0 and Eth3
- (D) Eth1 and Eth3

128.75.43.16 255.255.255.0 <b>128.75.43.0</b>	10000000 01001011 00101011 00010000 11111111 11111111 11111111 00000000 <b>10000000 01001011 00101011 00000000</b>
128.75.43.16 255.255.255.128 <b>128.75.43.0</b>	10000000 01001011 00101011 00010000 11111111 11111111 11111111 10000000 <b>10000000 01001011 00101011 00000000</b>
128.75.43.16 255.255.255.255 <b>128.75.43.16</b>	<b>10000000 01001011 00101011 00010000</b> 11111111 11111111 11111111 11111111 <b>10000000 01001011 00101011 00010000</b>
192.12.17.10 255.255.255.0 <b>192.12.17.0</b>	11000000 00001100 00010001 00001010 11111111 11111111 11111111 00000000 <b>11000000 00001100 00010001 00000000</b>
192.12.17.10 255.255.255.128 <b>192.12.17.0</b>	11000000 00001100 00010001 00001010 11111111 11111111 11111111 10000000 <b>11000000 00001100 00010001 00000000</b>
192.12.17.10 255.255.255.255 <b>192.12.17.10</b>	11000000 00001100 00010001 00001010 11111111 11111111 11111111 11111111 <b>11000000 00001100 00010001 00001010</b>

**Q** A group of 15 routers is interconnected in a centralized complete binary tree with a router at each tree node. Router i communicates with router j by sending a message to the root of the tree. The root then sends the message back down to router j. The mean number of hops per message, assuming all possible router pairs are equally likely is  
**(GATE-2007) (2 Marks)**

- A) 3
- B) 4.26
- C) 4.53
- D) 5.26

**Q.** Consider the entries shown below in the forwarding table of an IP router. Each entry consists of an IP prefix and the corresponding next hop router for packets whose destination IP address matches the prefix. The notation “/N” in a prefix indicates a subnet mask with the most significant N bits set to 1. This router forwards 20 packets each to 5 hosts .The IP address of the hosts are 10.1.1.16, 10.1.1.72, 10.1.1.132, 10.1.1.191 and 10.1.1.205 . The number of packets forwarded via the next hop router R2 is \_\_\_\_\_ (Gate 2024,CS) (2 Marks) (NAT)

<b>Prefix</b>	<b>Next hop router</b>
10.1.1.0/24	R1
10.1.1.128/25	R2
10.1.1.64/26	R3
10.1.1.192/26	R4

**Q.** A packet with the destination IP address 145.36.109.70 arrives at a router whose routing table is shown.

Which interface will the packet be forwarded to? **(Gate 2025)**

- A) E3
- B) E1
- C) E2
- D) E5

Subnet Address	Subnet Mask (in CIDR notation)	Interface
145.36.0.0	/16	E1
145.36.128.0	/17	E2
145.36.64.0	/18	E3
145.36.255.0	/24	E4
Default	--	E5

**Q.** Consider the routing protocols given in List I and the names given in List II:

**List I**

- (i) Distance Vector routing      (a) Bellman-Ford
- (ii) Link state routing      (b) Dijkstra

**List II**

For matching of items in List I with those in List II, which ONE of the following options is CORRECT? **(GATE 2025)**

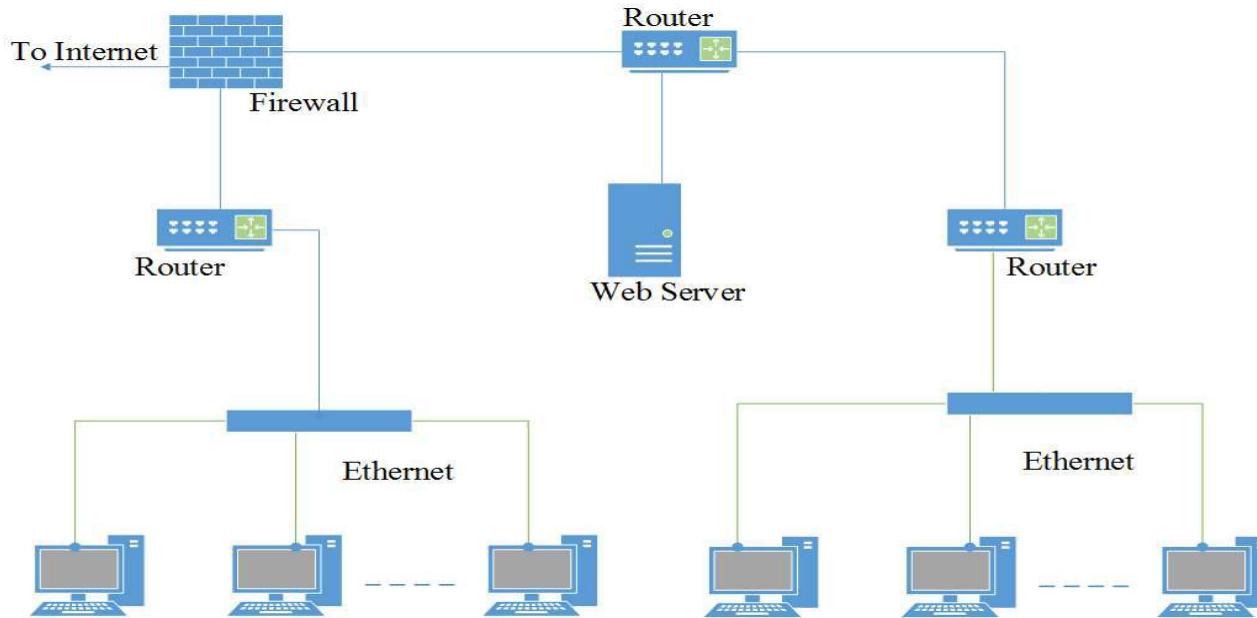
- A) (i) - (a) and (ii) - (b)
- B) (i) - (a) and (ii) - (a)
- C) (i) - (b) and (ii) - (a)
- D) (i) - (b) and (ii) - (b)

**Q.** A machine receives an IPv4 datagram. The protocol field of the IPv4 header has the protocol number of a protocol X.

Which ONE of the following is NOT a possible candidate for X?  
**(GATE 2025)**

- A)** Internet Control Message Protocol (ICMP)
- B)** Internet Group Management (IGMP)
- C)** Open Shortest Path First (OSPF)
- D)** Routing Information Protocol (RIP)

**Q** Consider an enterprise network with two Ethernet segments, a web server and a firewall, connected via three routers as shown below



What is the number of subnets inside the enterprise network? **(GATE 2022) (1 MARKS)**

- (A) 3
- (B) 12
- (C) 6
- (D) 8

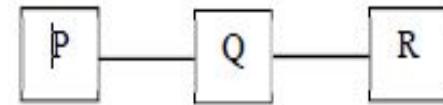
**Q** Consider routing table of an organization's router shown below:

Subnet Number	Subnet Mask	Next Hop
12.20.164.0	255.255.252.0	R1
12.20.170.0	255.255.254.0	R2
12.20.168.0	255.255.254.0	Interface 0
12.20.166.0	255.255.254.0	Interface 1
default		R3

Which of the following prefixes in CIDR notation can be collectively used to correctly aggregate all of the subnets in the routing table? **(GATE 2022) (2 MARKS)**

- (A) 12.20.164.0/20
- (B) 12.20.164.0/22
- (C) 12.20.164.0/21
- (D) 12.20.168.0/22

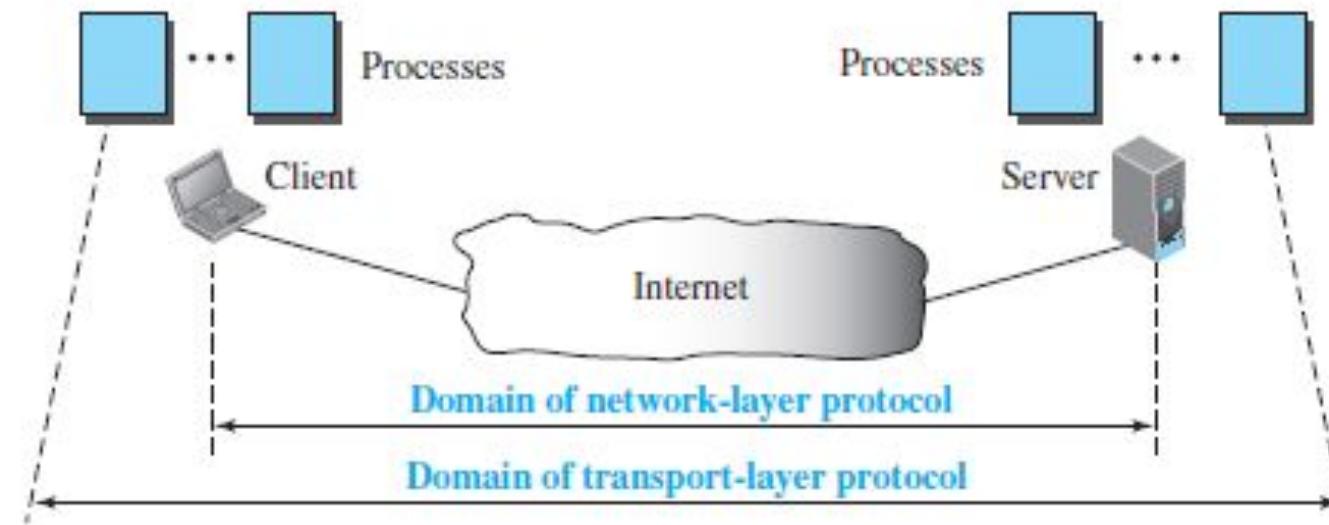
**Q** Consider a network with three routers P, Q, R shown in the figure below. All the links have cost of unity.



The routers exchange distance vector routing information and have converged on the routing tables, after which the link Q–R fails. Assume that P and Q send out routing updates at random times, each at the same average rate. The probability of a routing loop formation (rounded off to one decimal place) between P and Q, leading to count-to-infinity problem, is \_\_\_\_\_ . **(GATE 2022) (2 MARKS)**

# Transport-Layer Services

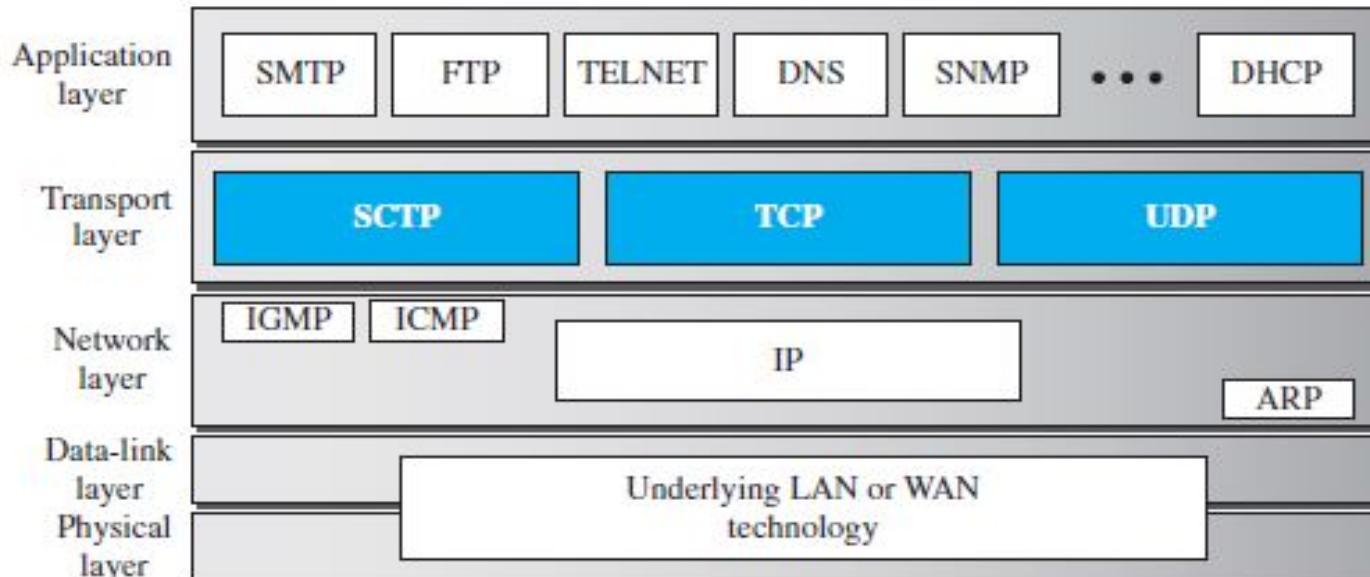
- The **transport layer (TL)** is responsible for enabling **process-to-process communication**, which involves delivering messages directly to specific applications or processes running on a computer.
  - The **network layer** manages host-to-host communication, ensuring messages reach the correct destination computer.
  - However, after a message reaches the computer, it needs to be forwarded to the appropriate process.
  - **Transport-layer protocols** provide **end-to-end communication**, ensuring that messages are delivered correctly from the source process to the destination process.



**Q** Which of the following functionalities must be implemented by a transport protocol over and above the network protocol? **(Gate-2003) (1 Marks)**

- (A)** Recovery from packet losses
- (B)** Detection of duplicate packets
- (C)** Packet delivery in the correct order
- (D)** End to end connectivity

- Transport layer protocols can be classified into **connectionless** and **connection-oriented**:
- **Connectionless Protocol:**
  - Treats each segment as an independent packet.
  - No setup or teardown of a connection is required.
  - Example: **UDP (User Datagram Protocol)** is connectionless and unreliable.
- **Connection-Oriented Protocol:**
  - Establishes a connection with the destination transport layer before data transfer.
  - After the transfer, the connection is terminated.
  - Examples: **TCP (Transmission Control Protocol)** and **SCTP (Stream Control Transmission Protocol)** are connection-oriented and reliable.

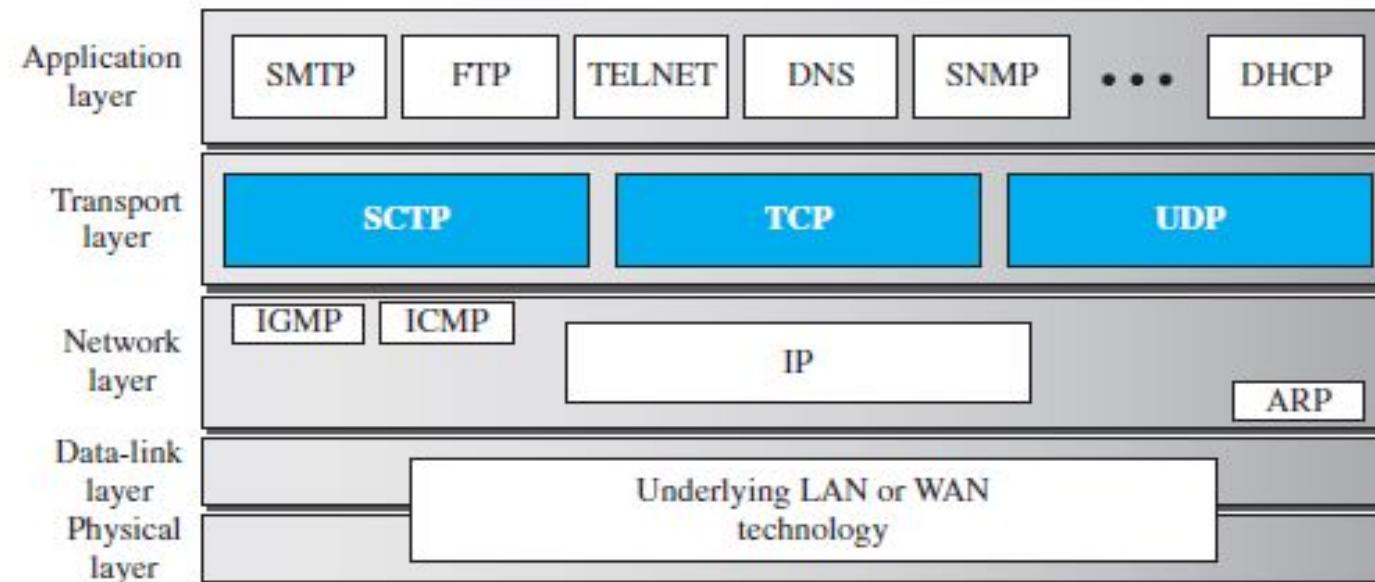


- **Key Responsibilities:**

- **Flow and Error Control:** Transport layer protocols can handle these end-to-end, rather than on a single link.
- **Reliability:** Reliable protocols like TCP and SCTP implement mechanisms for flow and error control, ensuring accurate delivery at the cost of complexity and speed. On the other hand, unreliable protocols like UDP prioritize speed and simplicity for use cases such as real-time applications.

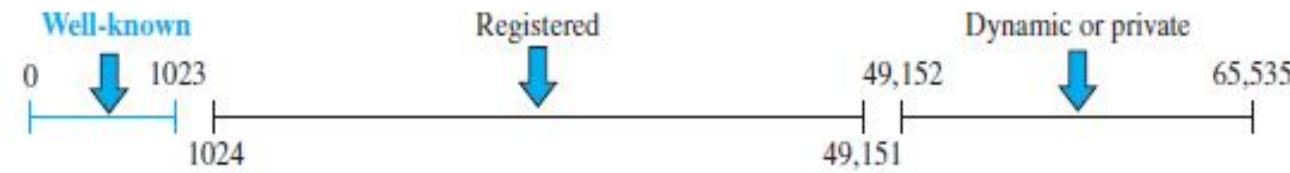
- **Summary of Protocols:**

- **UDP:** Connectionless, unreliable, fast.
- **TCP:** Connection-oriented, reliable, offers full-duplex communication (data can flow both ways simultaneously).
- **SCTP:** Also connection-oriented and reliable, combining features of TCP and UDP.



# Addressing: Port Numbers

- In communication over a network, identifying both hosts and processes is essential. Here's how port numbers work:
- **Hosts Identification:**
  - Local and remote hosts are identified using IP addresses.
- **Process Identification:**
  - Each process within a host is identified by a **port number**, a 16-bit integer ranging from 0 to 65535.

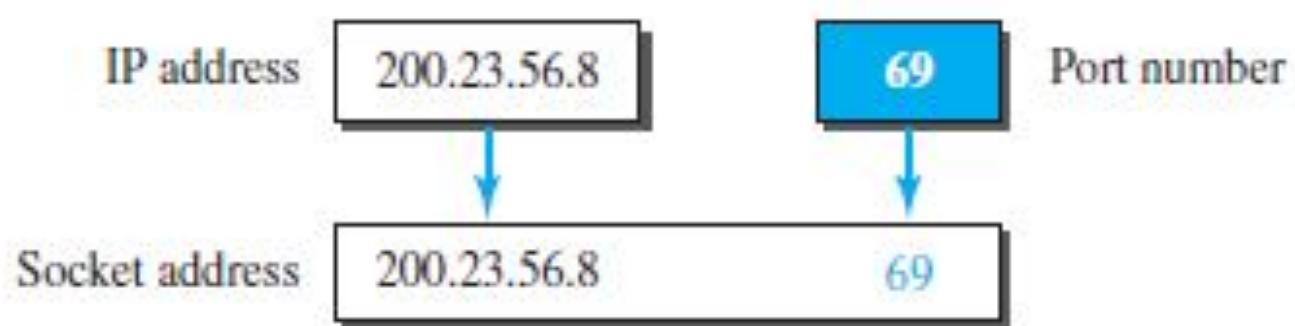


Port Number	Protocol	Application
20	TCP	FTP data
21	TCP	FTP control
22	TCP	SSH
23	TCP	Telnet
25	TCP	SMTP
53	UDP,TCP	DNS
67, 68	UDP	DHCP
69	UDP	TFTP
80	TCP	HTTP (WWW)
110	TCP	POP3
161	UDP	SNMP
443	TCP	HTTPS (SSL)
16384–32767	UDP	RTP-based voice (VoIP) and video

- **Types of Port Numbers**
- **Well-known Ports (0 to 1023):**
  - Assigned and controlled by IANA (Internet Assigned Numbers Authority).
  - Used by server processes to identify well-known services like HTTP (port 80) and FTP (port 21). These numbers are standardized to ensure client-server communication.
- **Registered Ports (1024 to 49151):**
  - Not assigned or controlled by IANA but can be registered to prevent duplication.
  - Used by applications that aren't as universally recognized as those with well-known ports.
- **Dynamic/Ephemeral Ports (49152 to 65535):**
  - Not controlled or registered by IANA.
  - Used temporarily by client processes and selected randomly by the transport layer.
  - Referred to as "ephemeral" due to their short-lived nature in client applications.

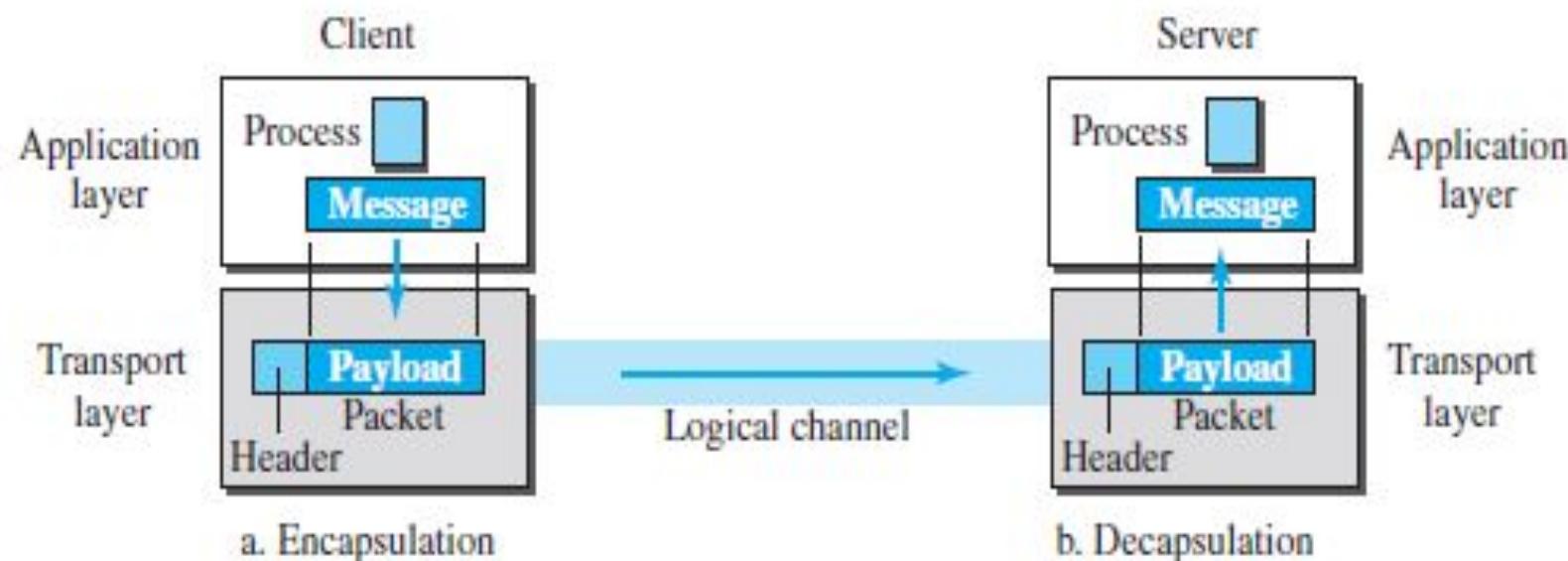
# Socket Addresses

- A transport-layer protocol in the TCP suite needs both the IP address and the port number, at each end, to make a connection. To use the services of the transport layer in the Internet, we need a pair of socket addresses: the client socket address and the server socket address.
- The combination of an IP address and a port number is called a **socket address**.



# Encapsulation and Decapsulation

- To transmit a message between processes, the transport layer uses two key operations: **encapsulation** and **decapsulation**.
- **Encapsulation (Sender Side):**
  - When a process sends a message, it passes it to the transport layer along with a pair of socket addresses (source and destination).
  - The transport layer then adds a **transport-layer header** to the message, creating a segment.
- **Decapsulation (Receiver Side):**
  - At the destination, the transport layer receives the segment and removes the transport-layer header.
  - The resulting message is then delivered to the appropriate process at the application layer.



**Q** What is the maximum size of data that the application layer can pass on to the TCP layer below? **(Gate-2008) (1 Marks)**

- (A)** Any size
- (B)**  $2^{16}$  bytes – size of TCP header
- (C)**  $2^{16}$  bytes
- (D)** 1500 bytes

**Q** A TCP message consisting of 2100 bytes is passed to IP for delivery across two networks. The first network can carry a maximum payload of 1200 bytes per frame and the second network can carry a maximum payload of 400 bytes per frame, excluding network overhead. Assume that IP overhead per packet is 20 bytes. What is the total IP overhead in the second network for this transmission? **(Gate-2004)**

**(2 Marks)**

**(A) 40 bytes**

**(B) 80 bytes**

**(C) 120 bytes**

**(D) 160 bytes**

**Q** In TCP, a unique sequence number is assigned to each **(Gate-2004) (1 Marks)**

**(A)** byte

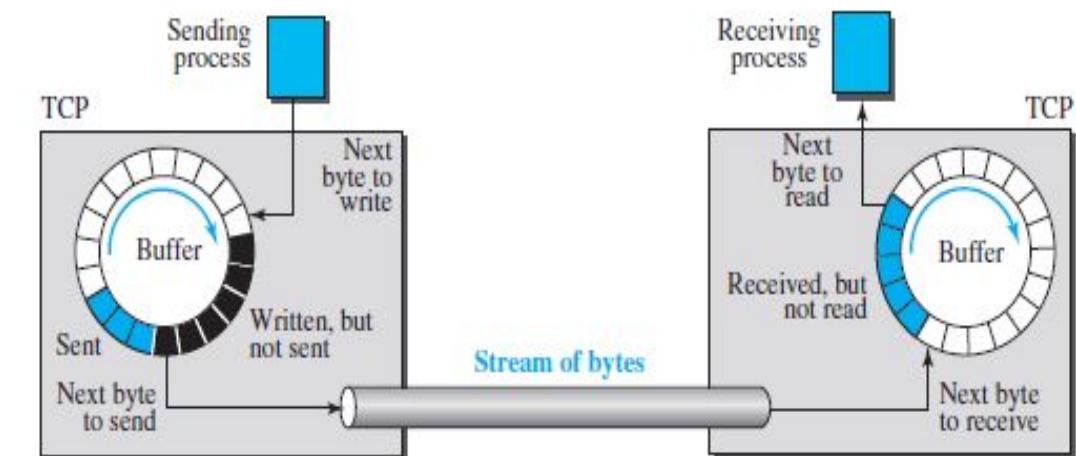
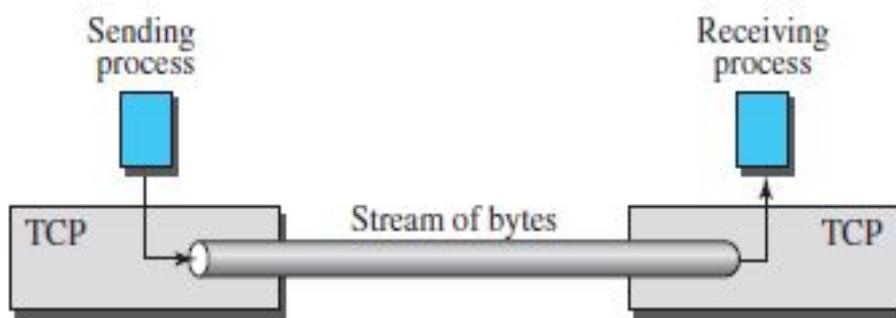
**(B)** word

**(C)** segment

**(D)** message

# TCP (Transmission Control Protocol)

- TCP is a reliable, connection-oriented transport layer protocol that is essential for applications requiring dependable data transfer. Key characteristics and functions of TCP include:
- **Connection-Oriented Communication:**
  - Establishes a **virtual connection** between sender and receiver, ensuring both ends are synchronized.
  - Reserves essential resources like bandwidth, CPU, and buffer space at the receiver's end.
- **Stream-Based Data Transfer:**
  - Allows the sender to transmit data as a **stream of bytes**, which the receiver processes in the same byte stream format, creating a continuous data flow.
- **Buffering:**
  - Uses **sending and receiving buffers** to handle different data rates between sender and receiver, ensuring smooth data flow.



**Q** In the TCP/IP protocol suite, which one of the following is NOT part of the IP header?

**(Gate-2004) (2 Marks)**

**(A)** Fragment Offset

**(B)** Source IP address

**(C)** Destination IP address

**(D)** Destination port number

**Q. Identify the ONE CORRECT matching between the OSI layers and their corresponding functionalities as shown.(Gate 2025)**

**OSI Layers**

- (a) Network layer
- (b) Transport layer handling
- (c) Datalink layer communication

**Functionalities**

- (I) Packet routing
- (II) Framing and error
- (III) Host to host

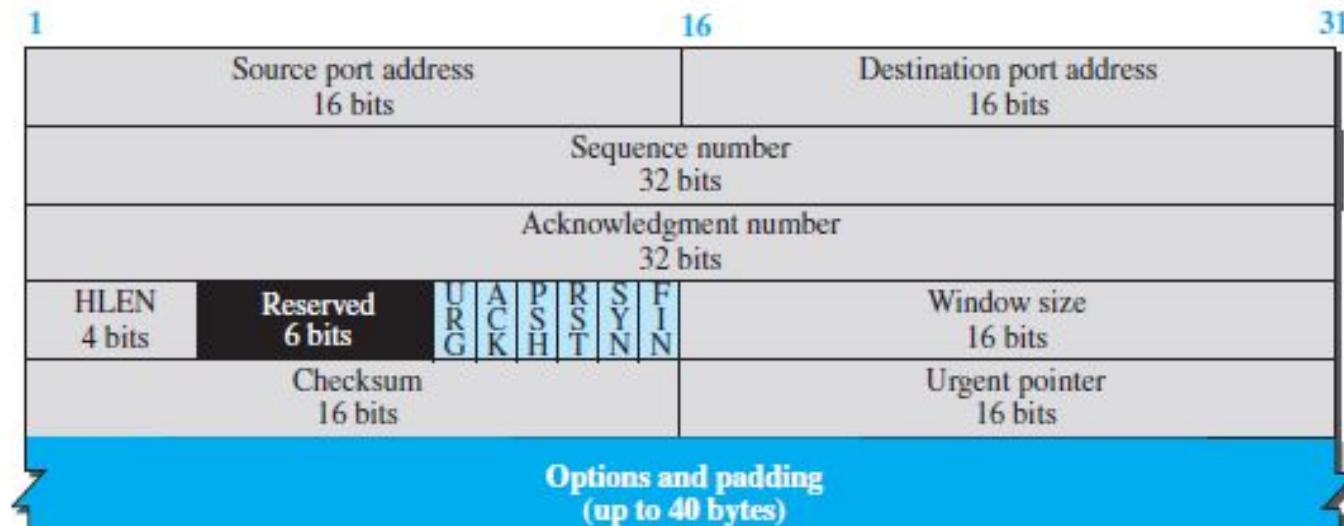
- A) (a)-(I), (b)-(II), (c)-(III)
- B) (a)-(I), (b)-(III), (c)-(II)
- C) (a)-(II), (b)-(I), (c)-(III)
- D) (a)-(III), (b)-(II), (c)-(I)

## TCP Header

- The segment consists of a 20- to 60-byte header, followed by data from the application program. The header is 20 bytes if there are no options and up to 60 bytes if it contains options.



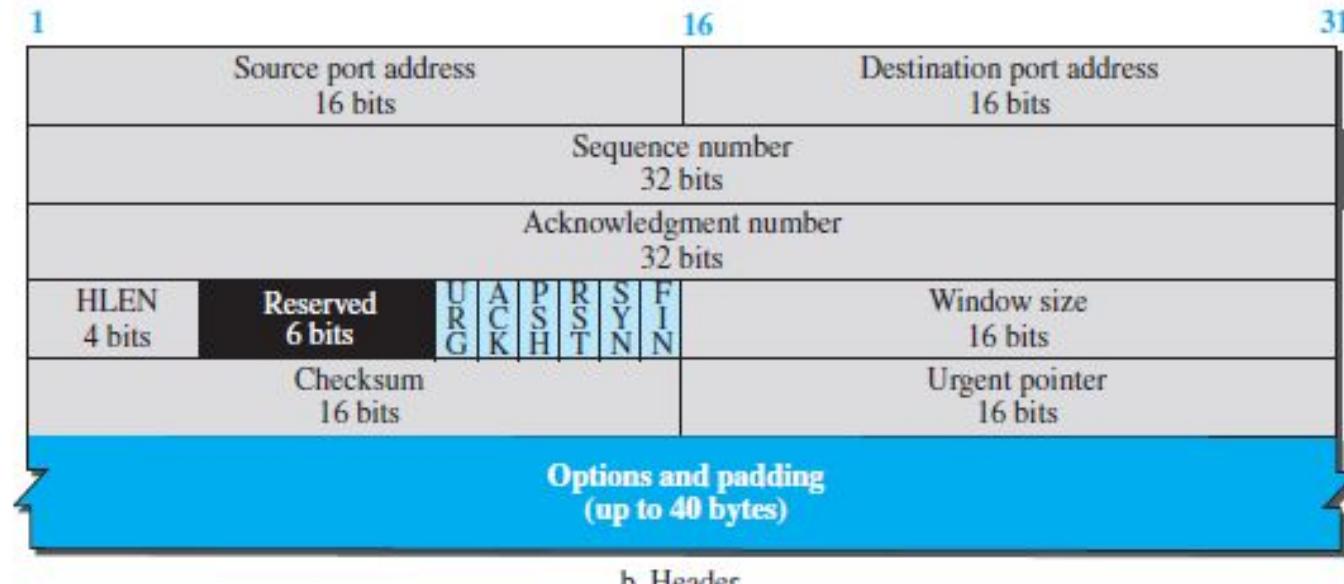
a. Segment



b. Header

## Source & Destination port address

- **Source port address.** This is a 16-bit field that defines the port number of the application program in the host that is sending the segment.
- **Destination port address.** This is a 16-bit field that defines the port number of the application program in the host that is receiving the segment.

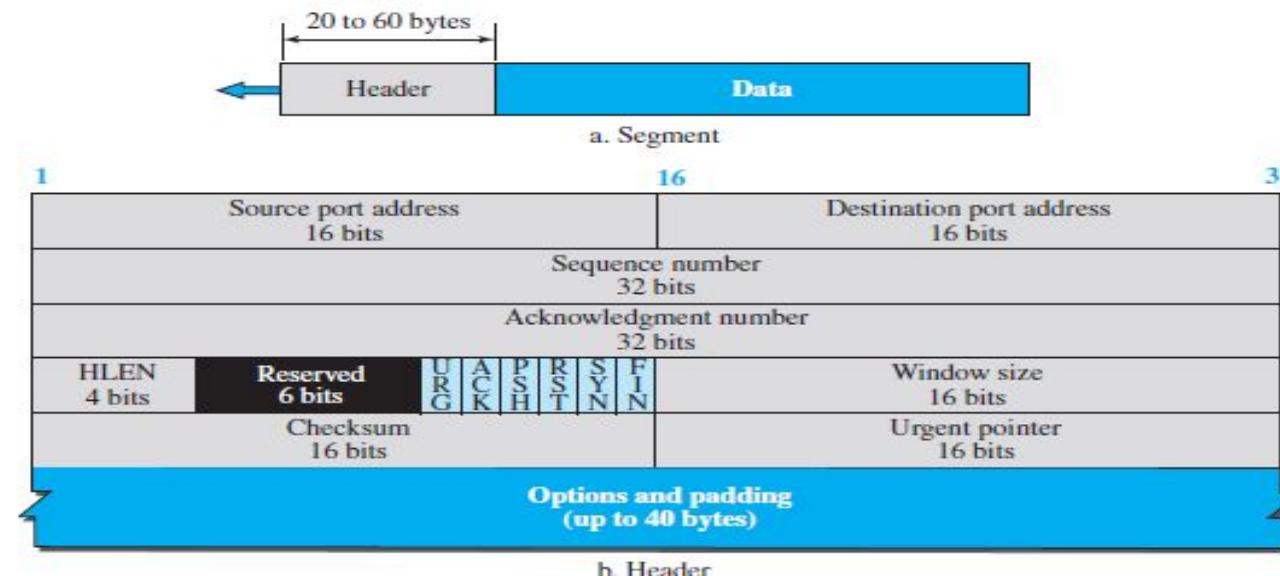


## Byte Number

- **Byte Numbering in TCP:**
  - **Numbering Mechanism:** TCP assigns a unique number to each data byte (octet) in a connection. This numbering is independent for each direction.
  - **Arbitrary Start Point:** TCP does not start numbering from 0. Instead, it chooses a random number between 0 and  $2^{32} - 1$  for the first byte.
  - **Purpose:** This method of numbering helps maintain order and ensure reliable delivery in TCP connections.
- **Sequence Number in TCP:**
  - **Definition:** The **sequence number** is a 32-bit field in the TCP header that specifies the number assigned to the first byte in a segment. Each segment's sequence number ensures continuity and reliability of data transmission.
  - **Range:** Sequence numbers range from 0 to  $2^{32} - 1$ .
  - **ISN (Initial Sequence Number):** During connection setup, each side randomly generates an ISN to prevent duplicate data and enhance security.
  - **Updating Sequence Numbers:** The sequence number of the next segment is calculated as the sum of the current segment's sequence number and the number of bytes in that segment.
- **Comparison to Other Layers:**
  - **IP Layer:** Counts packets.
  - **Data Link Layer (DLL):** Counts bits (as in HDLC protocol).

## Sequence number

- TCP is a stream transport protocol. To ensure connectivity, each byte to be transmitted is numbered. Sequence number is 32-bit field defines the number assigned to the first byte of data contained in this segment.
- So, maximum number of possible sequence numbers =  $2^{32}$ . These sequence numbers lie in the range  $[0, 2^{32} - 1]$ .
- In IP every packet is counted not Byte, in DLL every bit is counted with HDLC protocol.
- During connection establishment, each party uses a random number generator to create an initial sequence number (ISN), which is usually different in each direction. Sequence number should be started at random, to remove duplication problem.
- The sequence number of any other segment is the sequence number of the previous segment plus the number of bytes (real or imaginary) carried by the previous segment.



Example: Suppose a TCP connection is transferring a file of 5000 bytes. The first byte is numbered 10001. What are the sequence numbers for each segment if data are sent in five segments, each carrying 1000 bytes?

## Wrap Around in TCP

- **Wrap Around Concept:** Although TCP sequence numbers range from 0 to  $2^{32} - 1$ , this does not limit data transmission to 4 GB. TCP allows the sequence numbers to **wrap around** and restart from 0 once the maximum number is reached, enabling continuous data transmission.
- **Wrap Around Time (WAT):**
  - **Definition:** The time taken to exhaust all  $2^{32}$  sequence numbers and wrap around to 0 is called **Wrap Around Time**.
  - **Dependency on Bandwidth:** The wrap around time is inversely proportional to the network's bandwidth.  
Mathematically:
    - Time taken to use up all the  $2^{32}$  sequence numbers is called as **wrap around time**.
    - It depends on the bandwidth of the network i.e. the rate at which the bytes go out.
    - **Wrap Around Time  $\propto 1 / \text{Bandwidth}$** 
      - If bandwidth of the network =  $x$  bytes/sec, then
    - **Wrap Around Time =  $2^{32} / x \text{ sec.}$**

## **Life Time of a TCP Segment:**

- The **lifetime** of a TCP segment is defined as **180 seconds (3 minutes)**. This is the maximum time a TCP segment is expected to take to reach its destination.

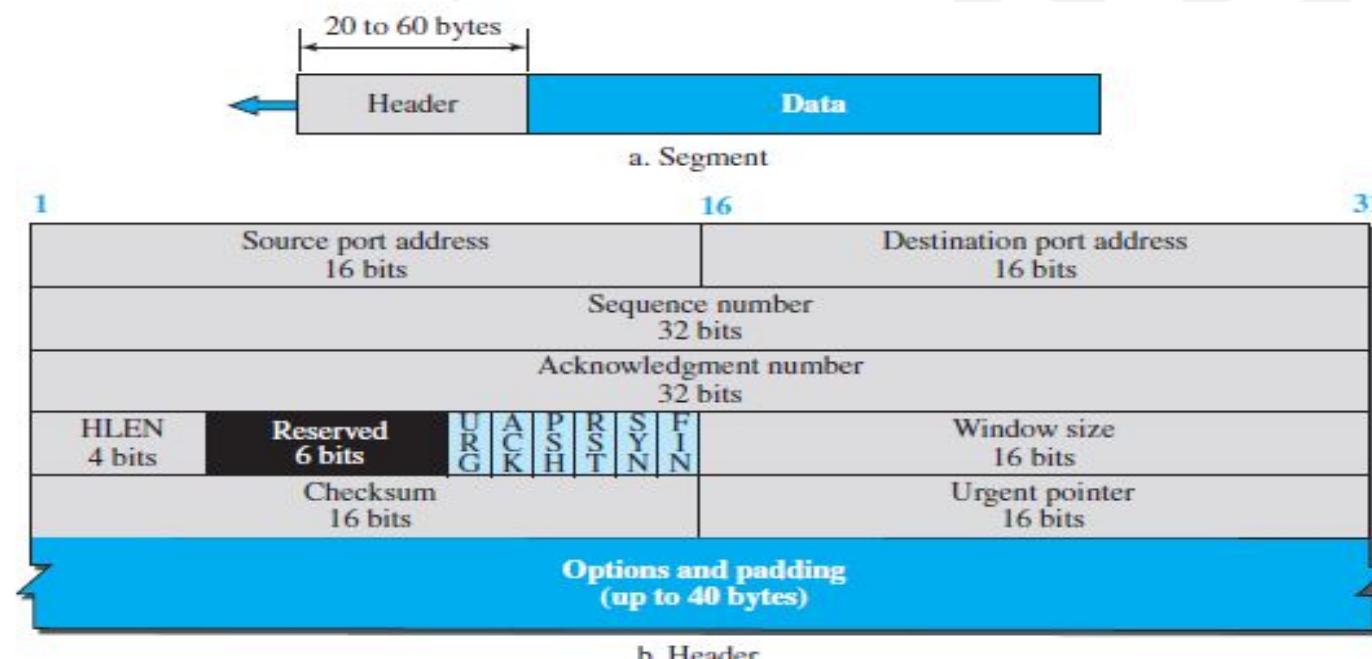
## **Considerations for Wrap Around:**

- **No Problem Scenario:** If the **wrap around time (WAT)** is greater than the **lifetime (LT)** of a TCP segment, there are no issues with reusing sequence numbers.
- **Conflict Scenario:** If WAT is less than LT, there is a risk of the receiver encountering duplicate sequence numbers, which can lead to confusion.

**Q** Consider a long-lived TCP session with an end-to-end bandwidth of  $1 \text{ Gbps} (= 10^9 \text{ bits / second})$ . The session starts with a sequence number of 1234. The minimum time (in seconds, rounded to the closest integer) before this sequence number can be used again is \_\_\_\_\_. **(GATE-2018)**  
**(1 Marks)**

## Acknowledgment Number

- This 32-bit field defines the byte number that the receiver of the segment is expecting to receive from the other party. If the receiver of the segment has successfully received byte number  $x$  from the other party, it defines  $x + 1$  as the acknowledgment number. Acknowledgment and data can be piggybacked together.
- The acknowledgment number is cumulative, which means that the party takes the number of the last byte that it has received, safe and sound, adds 1 to it, and announces this sum as the acknowledgment number.
- Acknowledgment number can be calculated by subtracting header length of IP and TCP to get the total byte count of the TCP segment and then can find the ack no

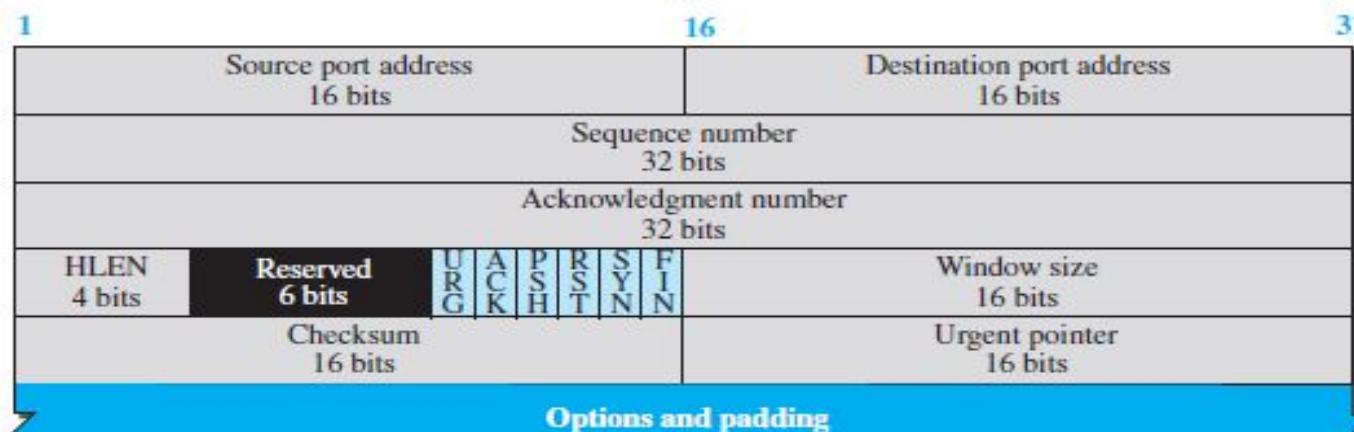


# Header length

- Header length: This 4-bit field indicates the number of 4-byte words in the TCP header. The length of the header can be between 20 and 60 bytes. Therefore, the value of this field can be between 5 ( $5 \times 4 = 20$ ) and 15 ( $15 \times 4 = 60$ ).
- Concept of Scaling Factor
  - $\text{Header length} = \text{Header length field value} \times 4 \text{ bytes}$
- Reserved. This is a 6-bit field reserved for future use.



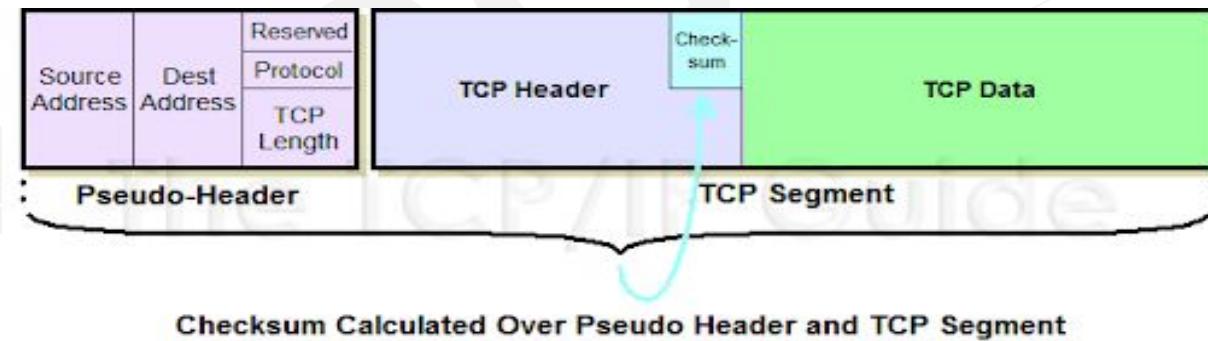
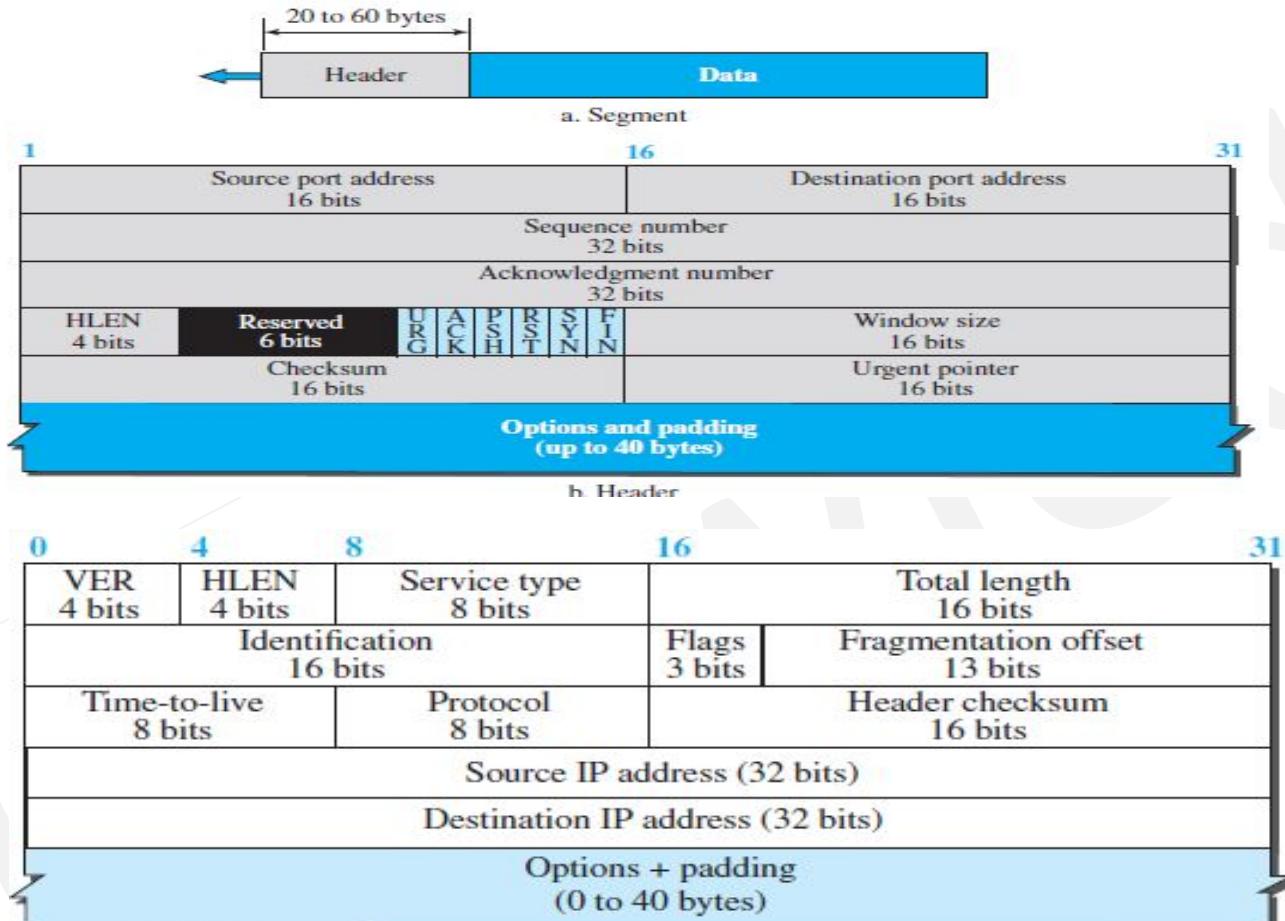
a. Segment



b. Header

# Checksum

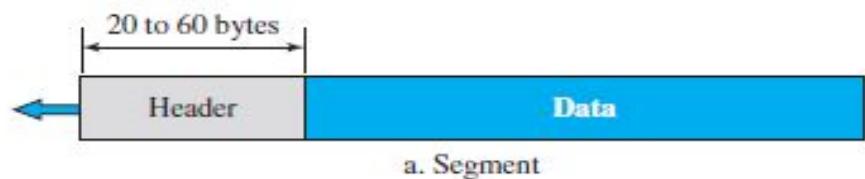
- **Checksum:** This 16-bit field contains the checksum.
- While calculation of the checksum for TCP, Entire TCP segment and pseudo header (IP) is considered.
- For the TCP pseudo header, the value for the protocol field is 6.



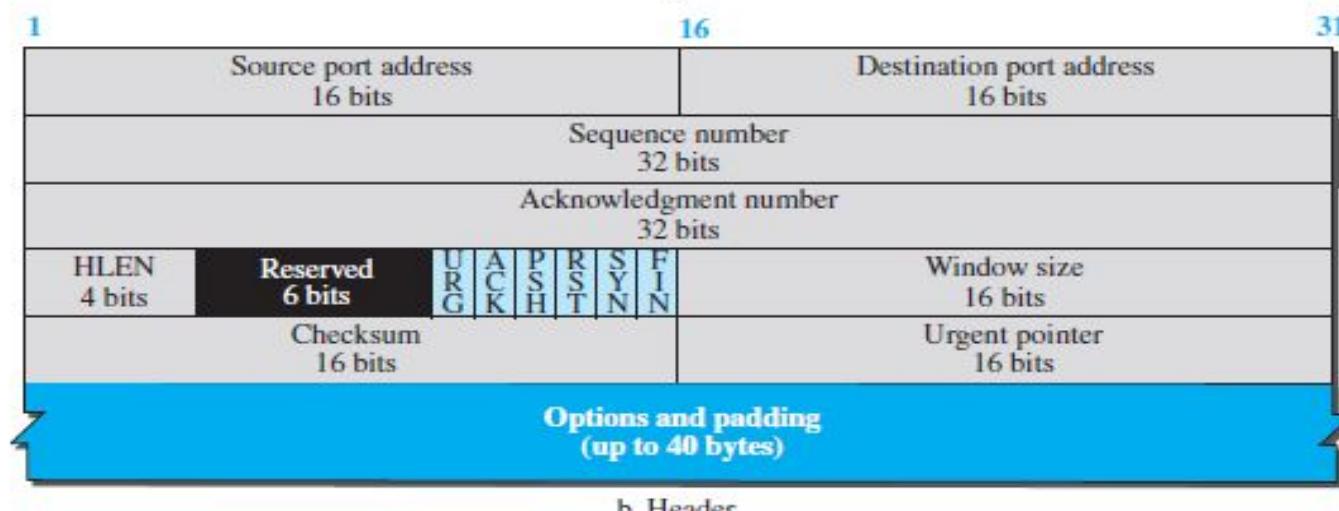
Checksum Calculated Over Pseudo Header and TCP Segment

## Window Size/Advertisement window(Flow control)

- Basics idea is a sender should never send what a receiver can not receive
- **Window size:** This field defines the size of the window, in bytes, that the receiver have reserved for the incoming data from sender.
- Note that the length of this field is 16 bits, which means that the maximum size of the window is 65,535 bytes. This value is normally referred to as the receiving window and is determined by the receiver. The sender must obey the dictation of the receiver in this case.



a. Segment



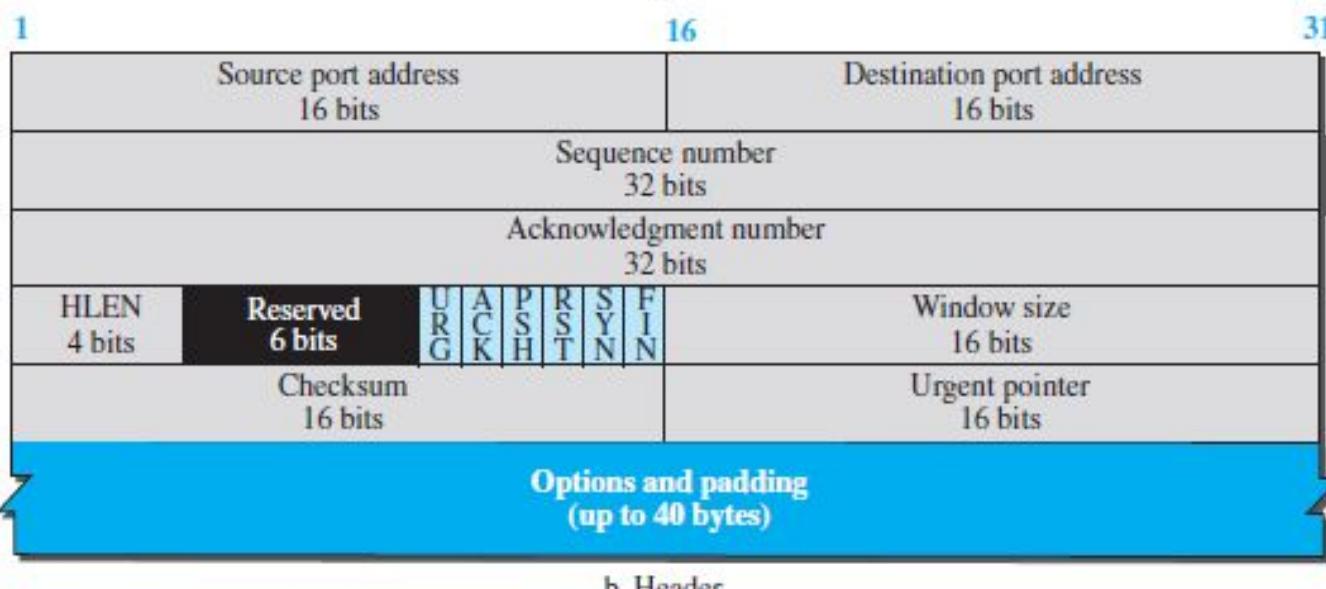
b. Header

## Window Size/Advertisement window(Flow control)

- **Persistent Timer** – To deal with a zero-window-size deadlock situation, TCP uses a persistence timer.
- When the sending TCP receives an acknowledgment with a window size of zero, it starts a persistence timer. When the persistence timer goes off, the sending TCP sends a special segment called a probe.
- This segment contains only 1 byte of new data. It has a sequence number, but its sequence number is never acknowledged. The probe causes the receiving TCP to resend the acknowledgment which was lost.



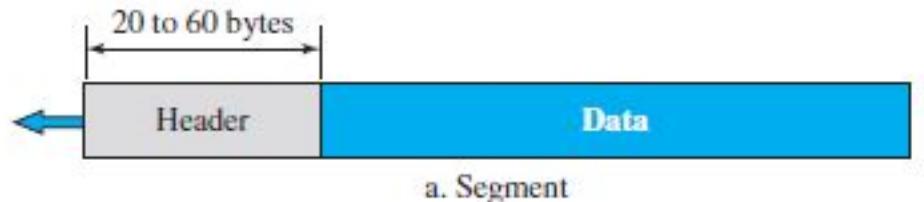
a. Segment



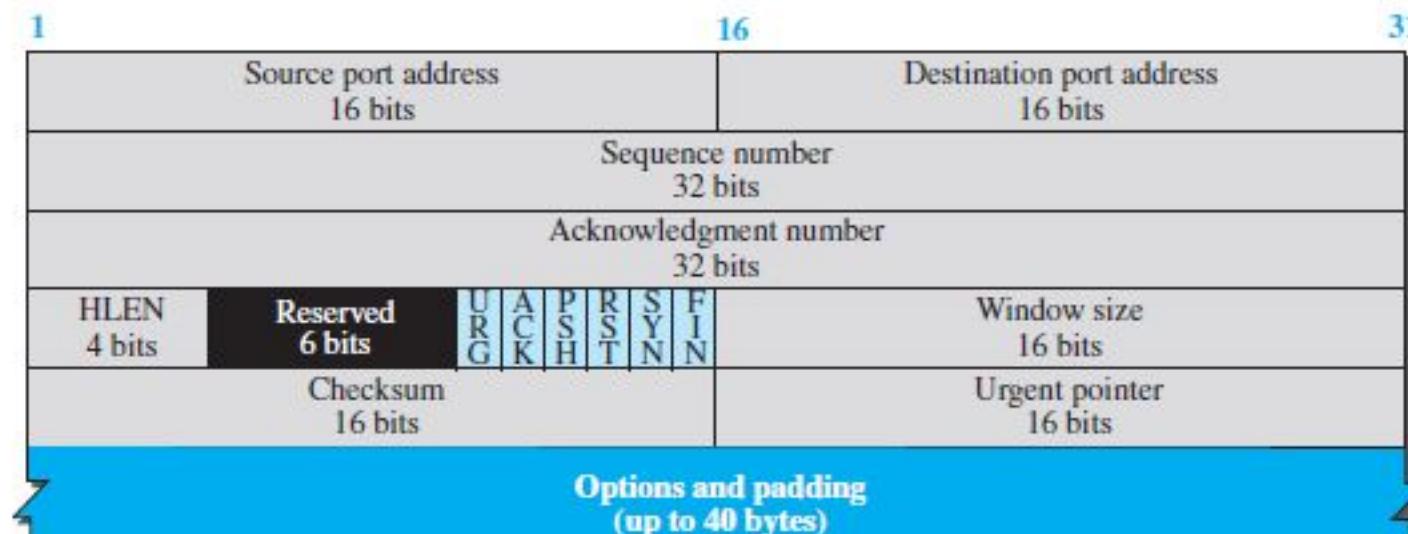
b. Header

## Window Size/Advertisement window(Flow control)

- One problem with this idea is window size is only 16 bits long, so very small amount of data can be advertise in todays world context.
- A solution is additional 14 bits can be taken from options so total size become 30 bits or 1GB.

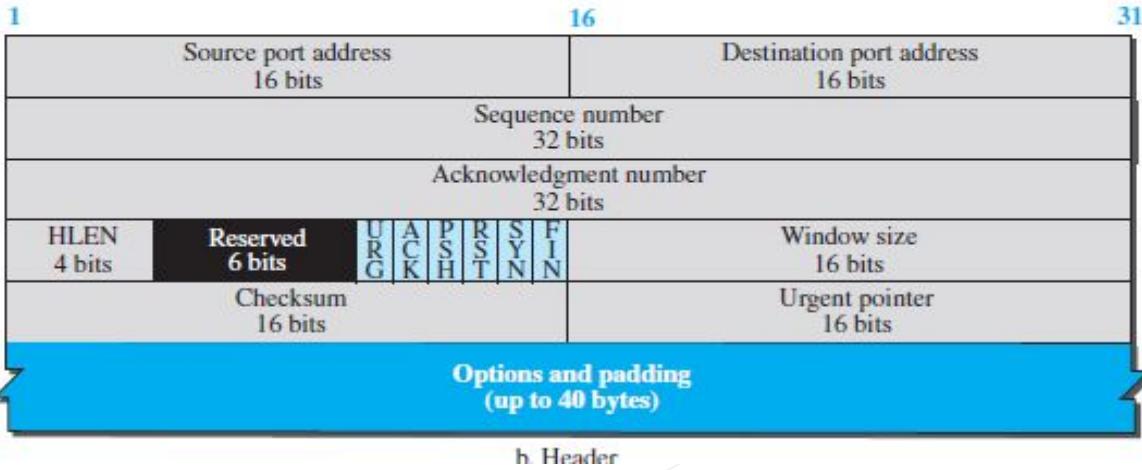


a. Segment



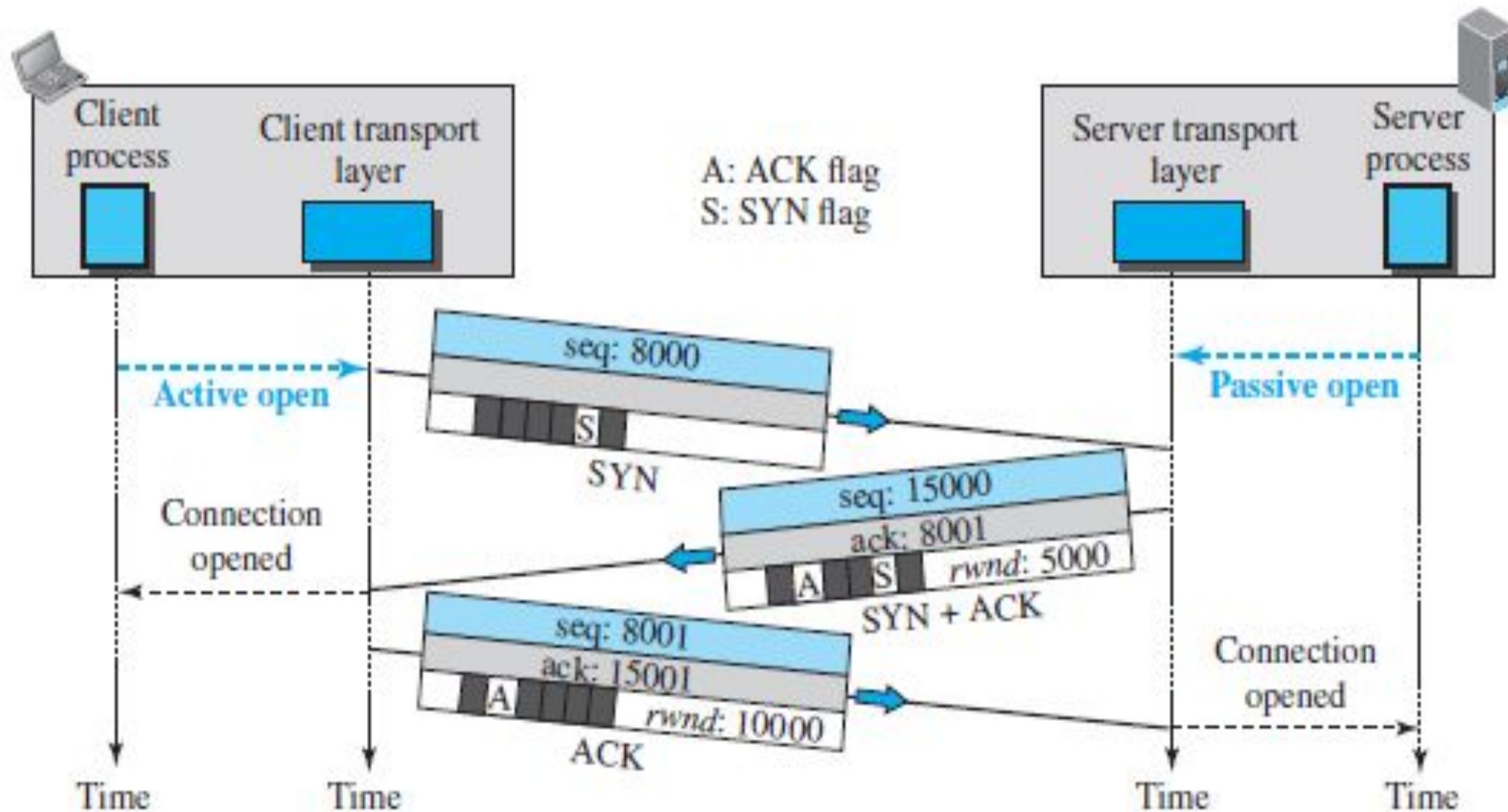
b. Header

- **Urgent Pointer:**
  - The **Urgent Pointer** is a 16-bit field that is only active if the **urgent flag** is set. It indicates the position of the last byte of urgent data by adding this value to the sequence number. This helps identify which part of the data is urgent and needs immediate attention.
- **Control Flags:**
  - TCP has **6 control flags** that play a crucial role in managing connections and data transfer. These flags help in establishing, terminating, or aborting connections, controlling data flow, and defining the transfer mode.
  - **PUSH Flag (PSH):**
    - The **Push Flag (PSH)** is used when immediate data delivery is required. By default, TCP waits to accumulate enough data to fill a segment before sending it to minimize network traffic.
    - For interactive applications like chat, where real-time communication is essential, the PSH flag is set to **1**. This prompts the transport layer to send the segment immediately and tells the receiver to forward it to the application layer without buffering.
  - **Reset Flag (RST):**
    - The **Reset Flag (RST)** is used to abort a TCP connection when something goes wrong or when the connection is deemed invalid. It can be sent by the receiver when a packet arrives unexpectedly or when there's an error in the TCP connection.



<i>Flag</i>	<i>Description</i>
URG	The value of the urgent pointer field is valid.
ACK	The value of the acknowledgment field is valid.
PSH	Push the data.
RST	Reset the connection.
SYN	Synchronize sequence numbers during connection.
FIN	Terminate the connection.

# A TCP Connection

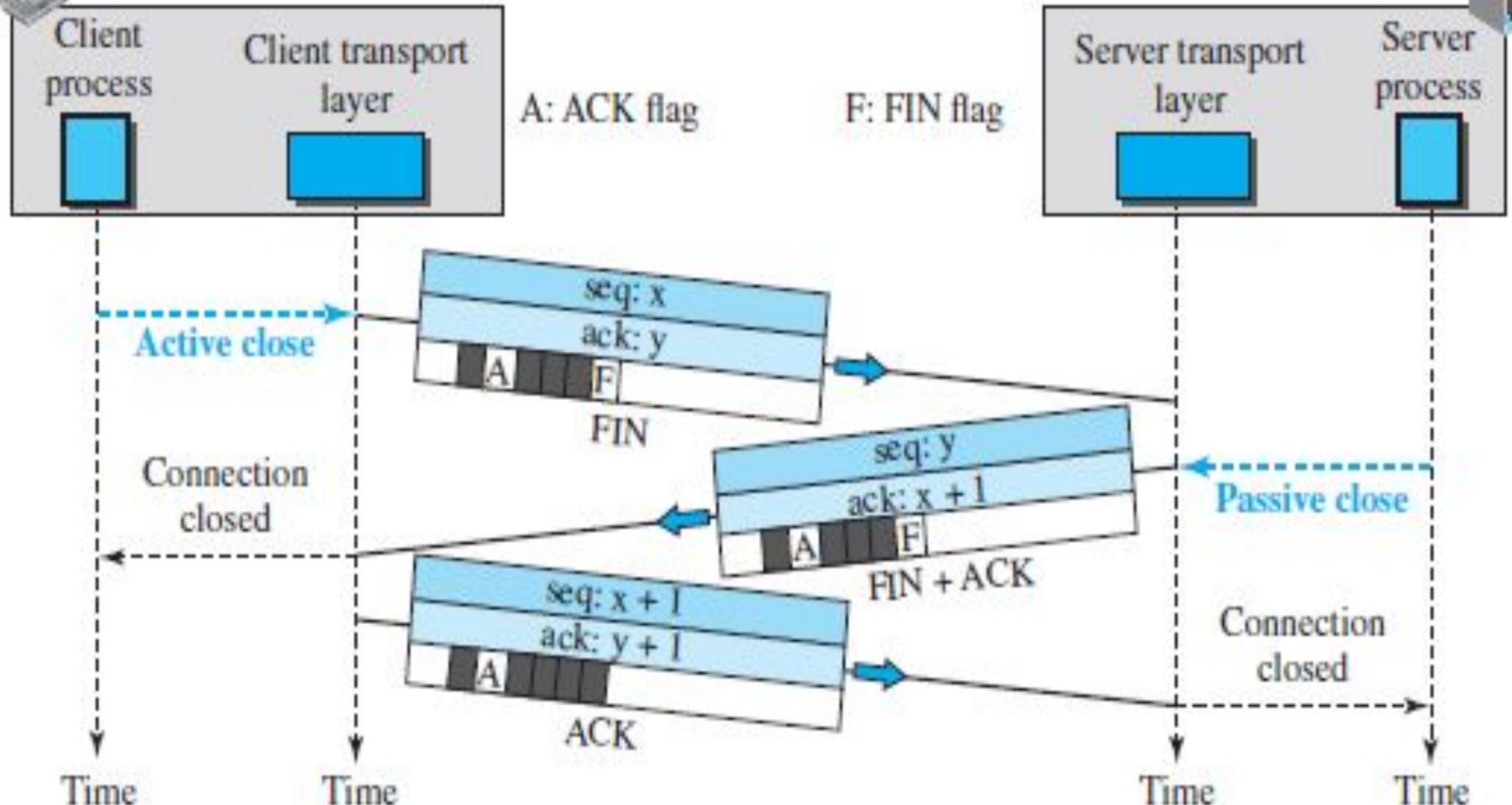


# Connection Establishment (Three-way handshaking)

- **Three-way handshaking** is the process of establishing a TCP connection between a client and a server. Here's how it works:
- **Passive Open by Server:**
  - The server signals its readiness to accept a connection. This state is referred to as **passive open**.
  - The server cannot initiate the connection; it only waits for a client to connect.
- **Active Open by Client:**
  - The client requests a connection by issuing an **active open** to a specific server.
- **Three-Way Handshake Process:**
  - **Step 1:** The client sends a **SYN segment** to the server to initiate the connection. This segment contains a sequence number chosen by the client.
  - **Step 2:** The server responds with a **SYN + ACK segment**:
    - **SYN:** Used to initiate a connection in the server-to-client direction.
    - **ACK:** Acknowledges the receipt of the client's SYN segment and displays the next expected sequence number from the client.
  - **Step 3:** The client sends an **ACK segment** to acknowledge the receipt of the server's SYN + ACK. This segment completes the handshake.
- **Key Points:**
- **Sequence Numbers:** The SYN and SYN + ACK segments consume a sequence number even though they carry no data.
- **Bidirectional Data Transfer:** Once the three-way handshake is complete, data can be transmitted in both directions.

## Connection Termination (Three-way handshaking)

- The termination of a TCP connection also involves a **three-way handshake** similar to its establishment. Here's how it works:
- **Initiating Closure:**
  - Either the client or server can initiate the closure. For example, the client decides to close the connection.
  - The client's TCP sends a **FIN segment** to the server, indicating the end of data transmission. This segment consumes one sequence number.
- **Server Acknowledgment and Closure:**
  - Upon receiving the FIN segment, the server acknowledges it by sending a **FIN + ACK segment**:
    - **FIN**: Indicates the server is ready to close its side of the connection.
    - **ACK**: Confirms the receipt of the client's FIN segment.
  - The server's FIN + ACK segment consumes one sequence number if it carries no data.
- **Final Acknowledgment from Client:**
  - The client responds with an **ACK segment**, acknowledging the server's FIN. This segment does not carry data and does not consume a sequence number.



**Q** Consider the three-way handshake mechanism followed during TCP connection establishment between hosts P and Q. Let X and Y be two random 32-bit starting sequence numbers chosen by P and Q respectively. Suppose P sends a TCP connection request message to Q with a TCP segment having SYN bit =1, SEQ number =X, and ACK bit =0. Suppose Q accepts the connection request. Which one of the following choices represents the information present in the TCP segment header that is sent by Q to P? **(GATE 2021) (1 MARKS)**

- a) SYN bit =1, SEQ number =X+1, ACK bit =0, ACK number =Y, FIN bit =0
- b) SYN bit =0, SEQ number =X+1, ACK bit =0, ACK number =Y, FIN bit =1
- c) SYN bit =1, SEQ number =Y, ACK bit =1, ACK number =X+1, FIN bit =0
- d) SYN bit =1, SEQ number =Y, ACK bit =1, ACK number =X, FIN bit =0

Q.TCP client P successfully establishes a connections to TCP server Q. Let  $N_p$  denote the sequence number in the SYN sent from P to Q. Let  $N_q$  denote the acknowledgement number in the SYN ACK from Q to P. Which of the following statements is/are CORRECT?

**(Gate 2024,CS) (1 Marks) (MSQ)**

- (a) The sequence number  $N_p$  is chosen randomly by P
- (b) The sequence number  $N_p$  is always 0 for a new connections
- (c) The acknowledgement number  $N_q$  is equal to  $N_p$
- (d) The acknowledgement number  $N_q$  is equal to  $N_p + 1$

**Q.** Consider the 3-way handshaking protocol for TCP connection establishment.

Let the three packets exchanged during the connection establishment be denoted as P1, P2, and P3, in order.

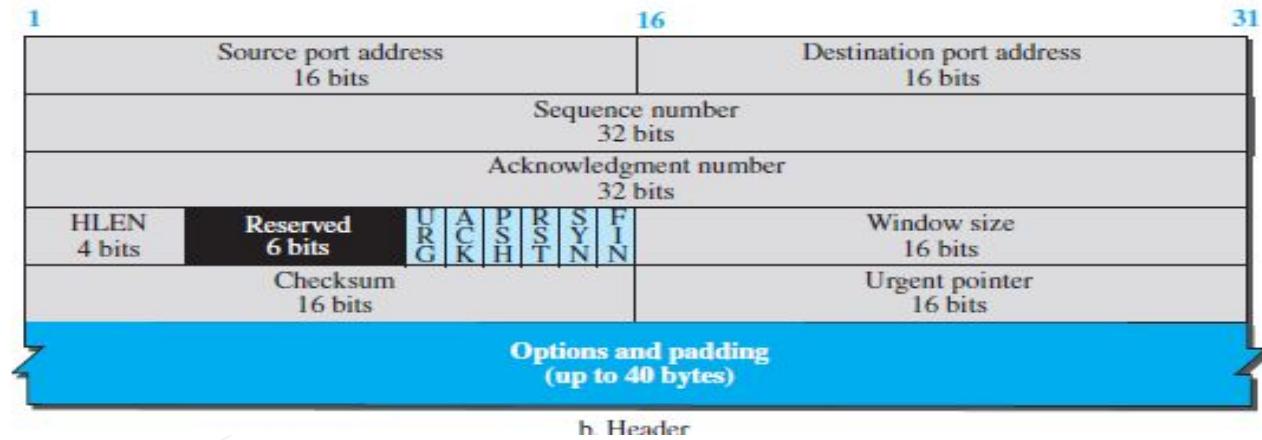
Which of the following option(s) is/are TRUE with respect to TCP header flags that are set in the packets?

**(Gate 2025)**

- A) P3 : SYN = 1, ACK = 1
- B) P2 : SYN = 1, ACK = 1
- C) P2 : SYN = 0, ACK = 1
- D) P1 : SYN = 1

## Options

- There can be up to 40 bytes of optional information in the TCP header.
  - Time Stamp
  - Window Size Extension
  - Padding



**Q** Suppose two hosts use a TCP connection to transfer a large file. Which of the following statements is/are **False** with respect to the TCP connection? **(Gate-2015) (1 Marks)**

1. If the sequence number of a segment is  $m$ , then the sequence number of the subsequent segment is always  $m+1$ .
  2. If the estimated round-trip time at any given point of time is  $t$  sec, the value of the retransmission timeout is always set to greater than or equal to  $t$  sec.
  3. The size of the advertised window never changes during the course of the TCP connection.
  4. The number of unacknowledged bytes at the sender is always less than or equal to the advertised window
- (A) 3 only**
- (B) 1 and 3 only**
- (C) 1 and 4 only**
- (D) 2 and 4 only**

**Q** Assume that the bandwidth for a TCP connection is 10,48,560 bits/sec. Let  $\alpha$  be the value of RTT in milliseconds (rounded off to the nearest integer) after which the TCP window scale option is needed. Let  $\beta$  be the maximum possible window size with window scale option. Then the values of  $\alpha$  and  $\beta$  are.  
**(Gate-2015) (2 Marks)**

- (A)** 63 milliseconds,  $65535 \times 2^{14}$
- (B)** 63 milliseconds,  $65535 \times 2^{16}$
- (C)** 500 milliseconds,  $65535 \times 2^{14}$
- (D)** 500 milliseconds,  $65535 \times 2^{16}$

**Q Consider the following statements. (Gate-2015) (1 Marks)**

- I. TCP connections are full duplex.
- II. TCP has no option for selective acknowledgment
- III. TCP connections are message streams.

- (A) Only I is correct
- (B) Only I and II are correct
- (C) Only II and III are correct
- (D) All of I, II and III are correct

**Q** Consider a TCP connection in a state where there are no outstanding ACKs. The sender sends two segments back to back. The sequence numbers of the first and second segments are 230 and 290 respectively. The first segment was lost, but the second segment was received correctly by the receiver. Let X be the amount of data carried in the first segment (in bytes), and Y be the ACK number sent by the receiver. The values of X and Y (in that order) are  
**(Gate-2007) (1 Marks)**

- (A)** 60 and 290
- (B)** 230 and 291
- (C)** 60 and 231
- (D)** 60 and 230

**Q** A TCP server application is programmed to listen on port number P on host S. A TCP client is connected to the TCP server over the network. Consider that while the TCP connection was active, the server machine S crashed and rebooted. Assume that the client does not use the TCP keepalive timer. Which of the following behaviors is/are possible? **(GATE 2021) (2 MARKS)**

- a)** If the client was waiting to receive a packet, it may wait indefinitely
- b)** The TCP server application on S can listen on P after reboot
- c)** If the client sends a packet after the server reboot, it will receive a RST segment
- d)** If the client sends a packet after the server reboot, it will receive a FIN segment

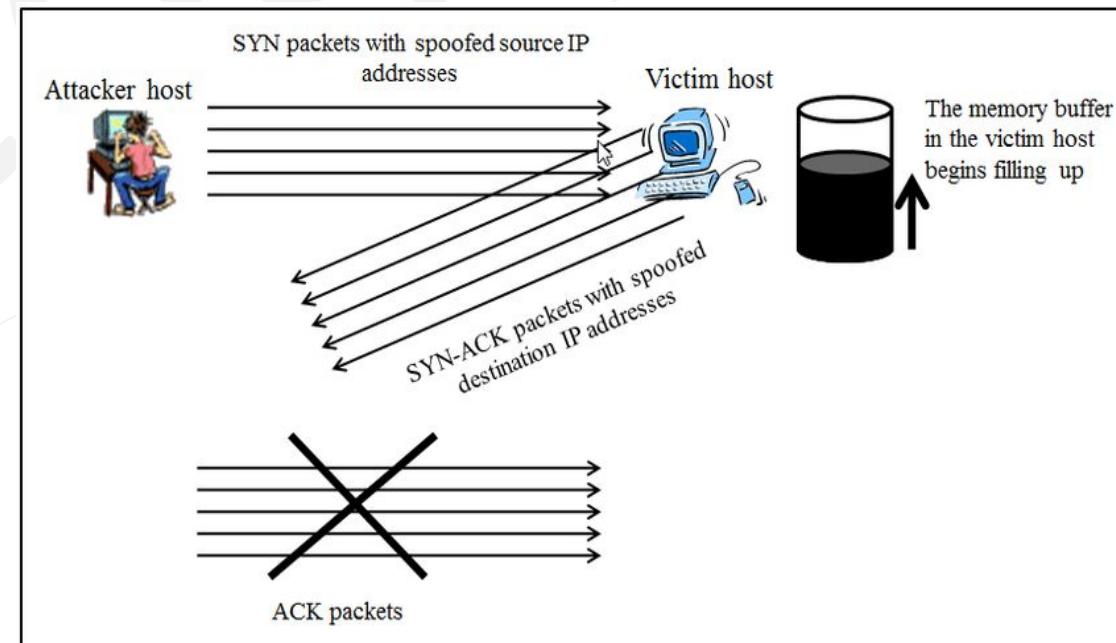
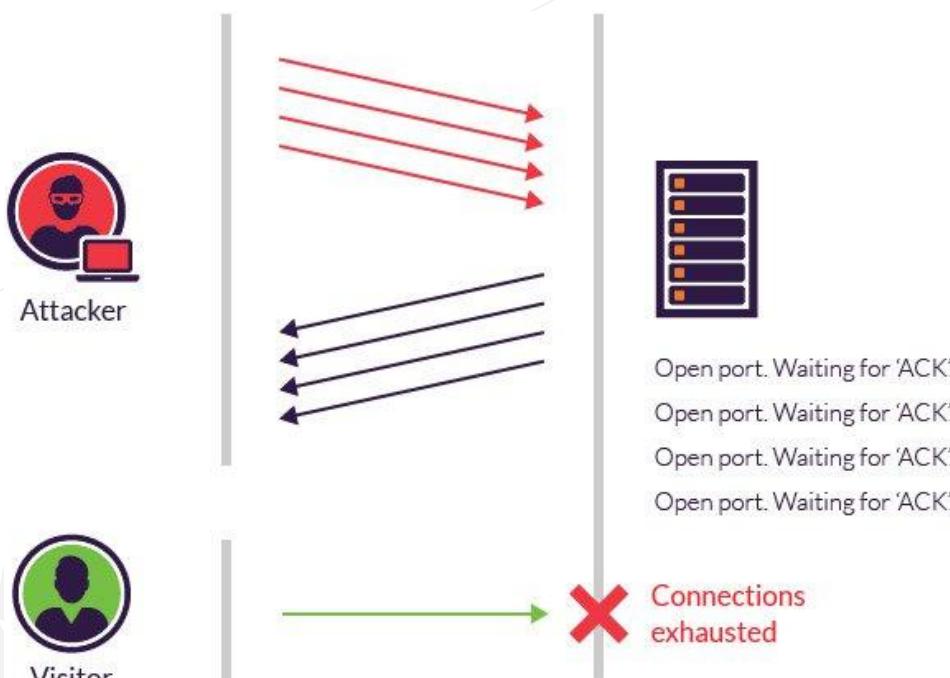
**Q** Consider the data transfer using TCP over a 1 Gbps link. Assuming that the maximum segment lifetime (MSL) is set to 60 seconds, the minimum number of bits required for the sequence number field of the TCP header, to prevent the sequence number space from wrapping around during the MSL is \_\_\_\_\_. **(GATE 2022) (2 MARKS)**

**Q** Consider two hosts P and Q connected through a router R. The maximum transfer unit (MTU) value of the link between P and R is 1500 bytes, and between R and Q is 820 bytes. A TCP segment of size 1400 bytes was transferred from P to Q through R, with IP identification value as 0x1234. Assume that the IP header size is 20 bytes. Further, the packet is allowed to be fragmented, i.e., Don't Fragment (DF) flag in the IP header is not set by P. Which of the following statements is/are correct? (GATE 2021) (2 MARKS)

- a) Two fragments are created at R and the IP datagram size carrying the second fragment is 620 bytes.
- b) If the second fragment is lost, R will resend the fragment with the IP identification value 0x1234.
- c) If the second fragment is lost, P is required to resend the whole TCP segment.
- d) TCP destination port can be determined by analysing only the second fragment.

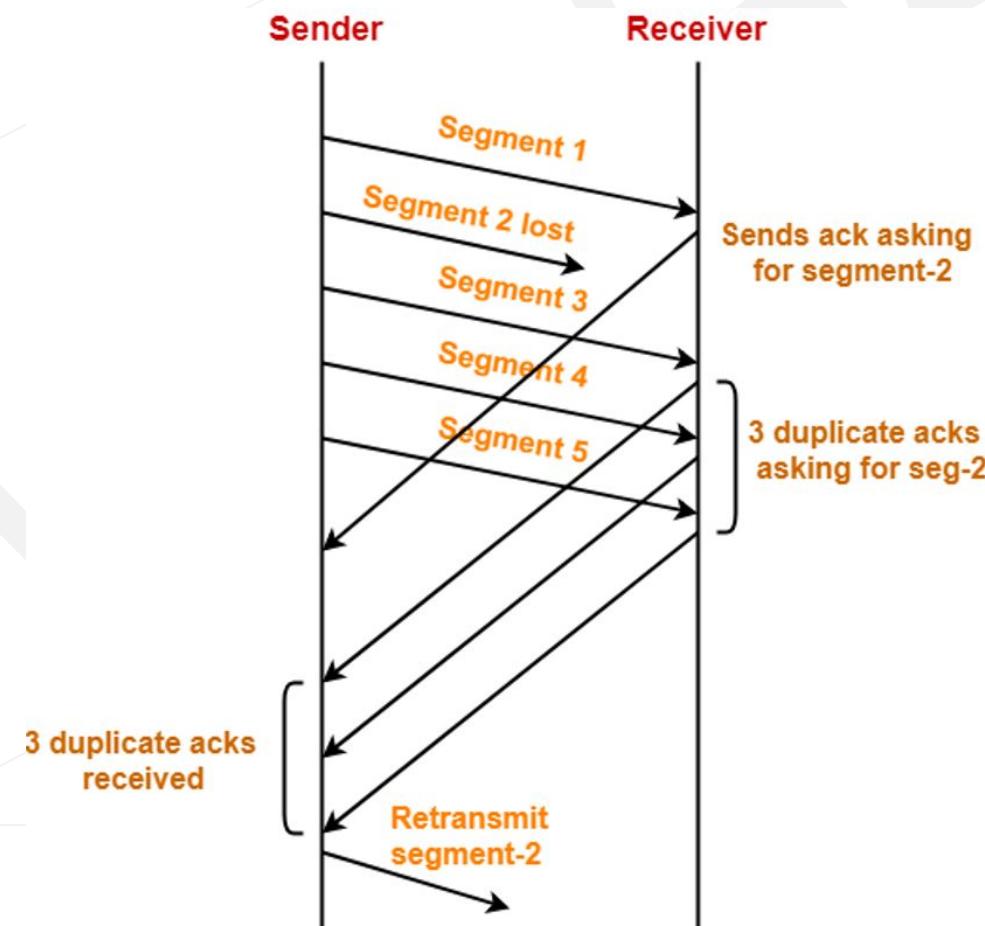
# SYN Flooding Attack(Denial of Service)

- **Attack Description:** In a SYN flooding attack, malicious attackers send numerous **SYN segments** to a server using **fake source IP addresses**. Each segment appears to come from a different client.
- **Server Resource Allocation:** The server allocates resources, such as creating **transfer control blocks (TCB)** and setting timers, in response to each SYN segment.
- **No Response to SYN + ACK:** The server sends **SYN + ACK segments** to the fake clients, which do not exist. As a result, these responses are lost, and the server never receives the final ACK to complete the handshake.
- **Resource Exhaustion:** The server waits for the ACK but, in the meantime, consumes resources without establishing actual connections. If many SYN segments are sent, the server can **run out of resources**, leading to a **denial of service** for legitimate clients.



# TCP Retransmission

- **Definition:** After a TCP connection is established, the sender transmits segments to the receiver. If a segment is lost in transit, the receiver sends **duplicate ACKs** with the same acknowledgment number to the sender.
- **Retransmission Process:** The sender detects a lost segment either through the expiration of a **timeout timer** or by receiving **three duplicate ACKs**.
- **Action:** Upon detection, the sender retransmits the lost segment to the receiver, ensuring reliable data delivery.

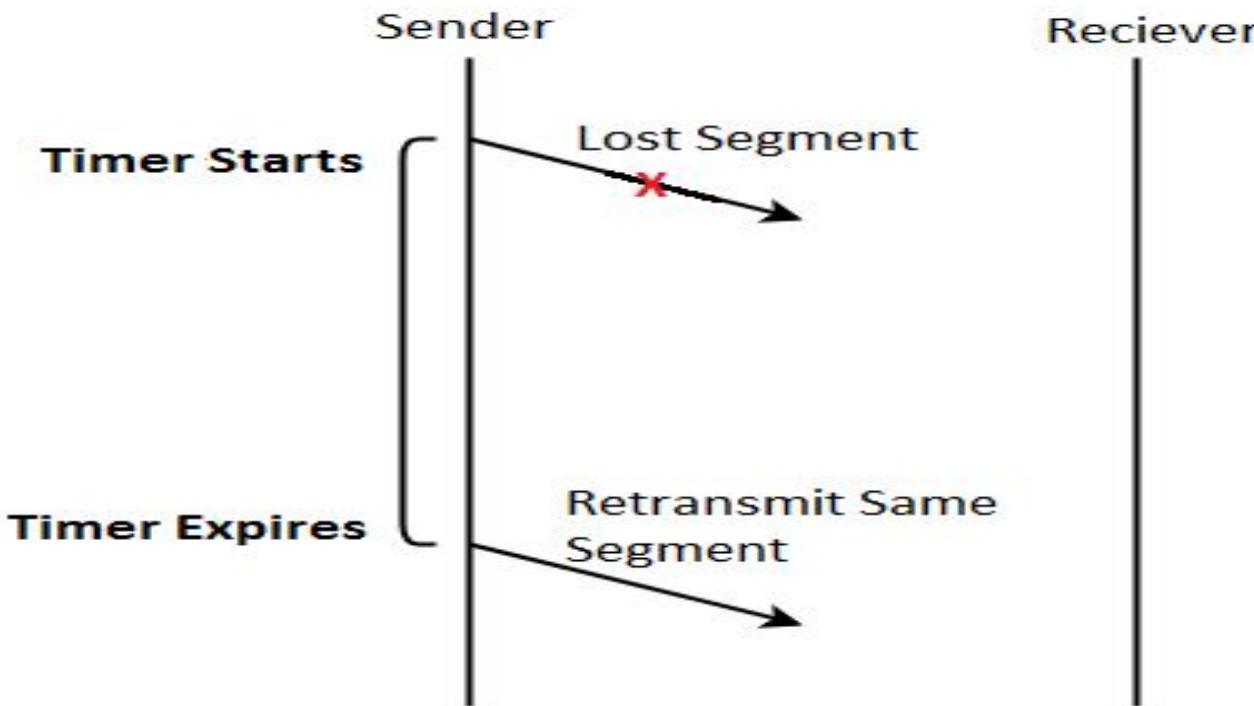


- **Retransmission after Timeout Expiry:**

- **Process:** When a sender transmits a TCP segment, it starts a **Timeout Timer**.
  - **Case 1:** If the sender receives an acknowledgment before the timer expires, it stops the timer.
  - **Case 2:** If no acknowledgment is received and the timer expires, the sender assumes the segment is lost. The sender **retransmits the segment** and resets the timer.

- **2. Early Retransmission (After 3 Duplicate ACKs):**

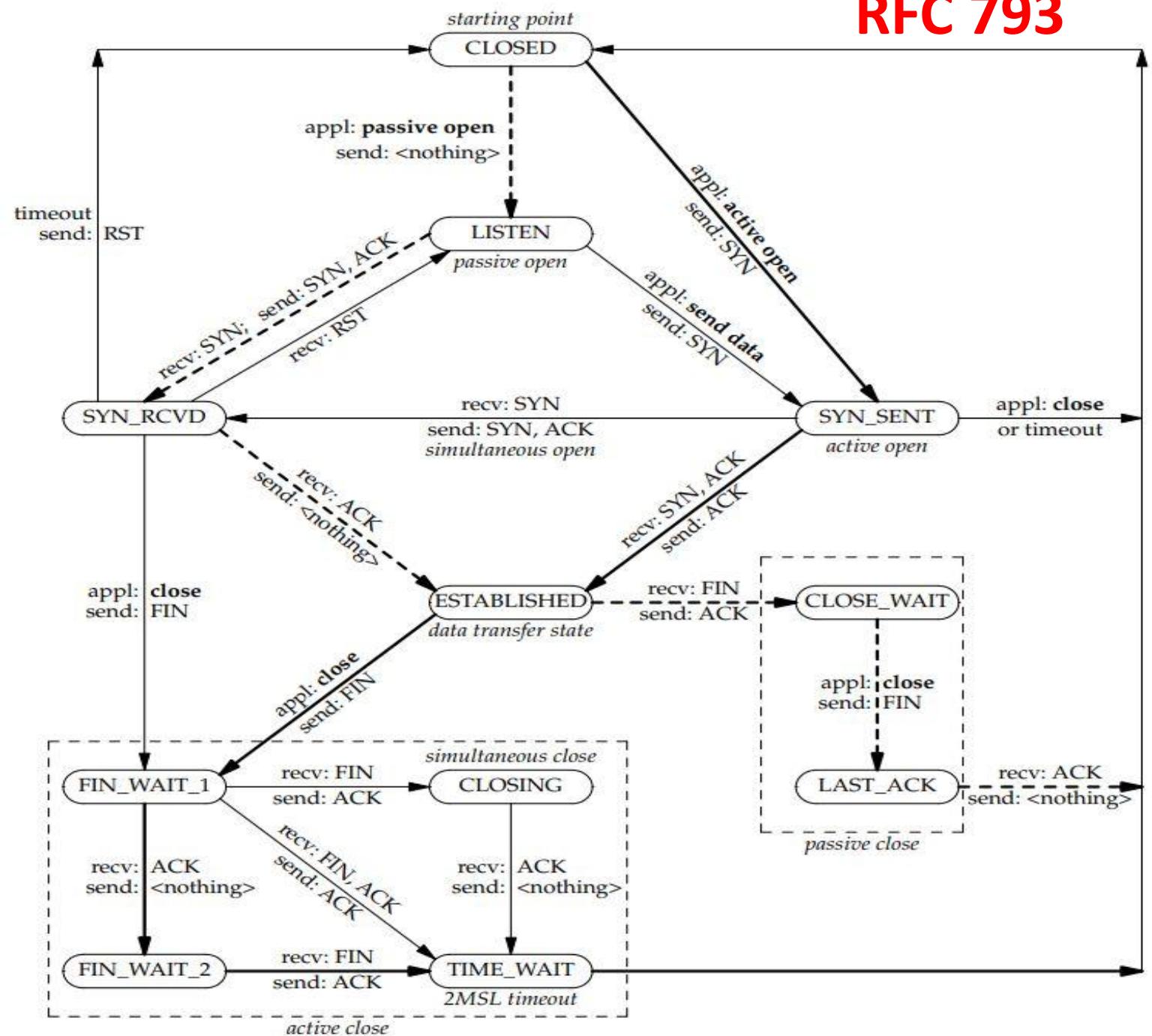
- **Process:** If the sender receives **three duplicate acknowledgments** for a segment, it presumes the segment is lost and retransmits it **without waiting for the timer** to expire. This is known as **early retransmission** or **fast retransmission**.
- **Reason for 3 Duplicate ACKs:** Experimental results have shown that using three duplicate acknowledgments effectively balances detection accuracy and performance.



## Points to Note

- **Timeout Expiry:** If a timeout occurs before an acknowledgment is received, it indicates a high likelihood of network congestion.
- **Performance Improvement:** Retransmission after three duplicate ACKs is faster and more efficient compared to waiting for the timeout, enhancing overall TCP performance.
- **4. TCP's Use of SR and GBN:**
  - TCP uses a combination of **Selective Repeat (SR)** and **Go-Back-N (GBN)** strategies:
    - **SR (Selective Repeat):** Handles out-of-order packets by accepting them.
    - **GBN (Go-Back-N):** Uses cumulative acknowledgments for simplicity and efficiency.

# RFC 793



→ normal transitions for client  
 → normal transitions for server  
 appl: state transitions taken when application issues operation  
 recv: state transitions taken when segment received  
 send: what is sent for this transition

State	Description
CLOSED	No connection exists
LISTEN	Passive open received; waiting for SYN
SYN-SENT	SYN sent; waiting for ACK
SYN-RCVD	SYN + ACK sent; waiting for ACK
ESTABLISHED	Connection established; data transfer in progress
FIN-WAIT-1	First FIN sent; waiting for ACK
FIN-WAIT-2	ACK to first FIN received; waiting for second FIN
CLOSE-WAIT	First FIN received, ACK sent; waiting for application to close
TIME-WAIT	Second FIN received, ACK sent; waiting for 2MSL time-out
LAST-ACK	Second FIN sent; waiting for ACK
CLOSING	Both sides decided to close simultaneously

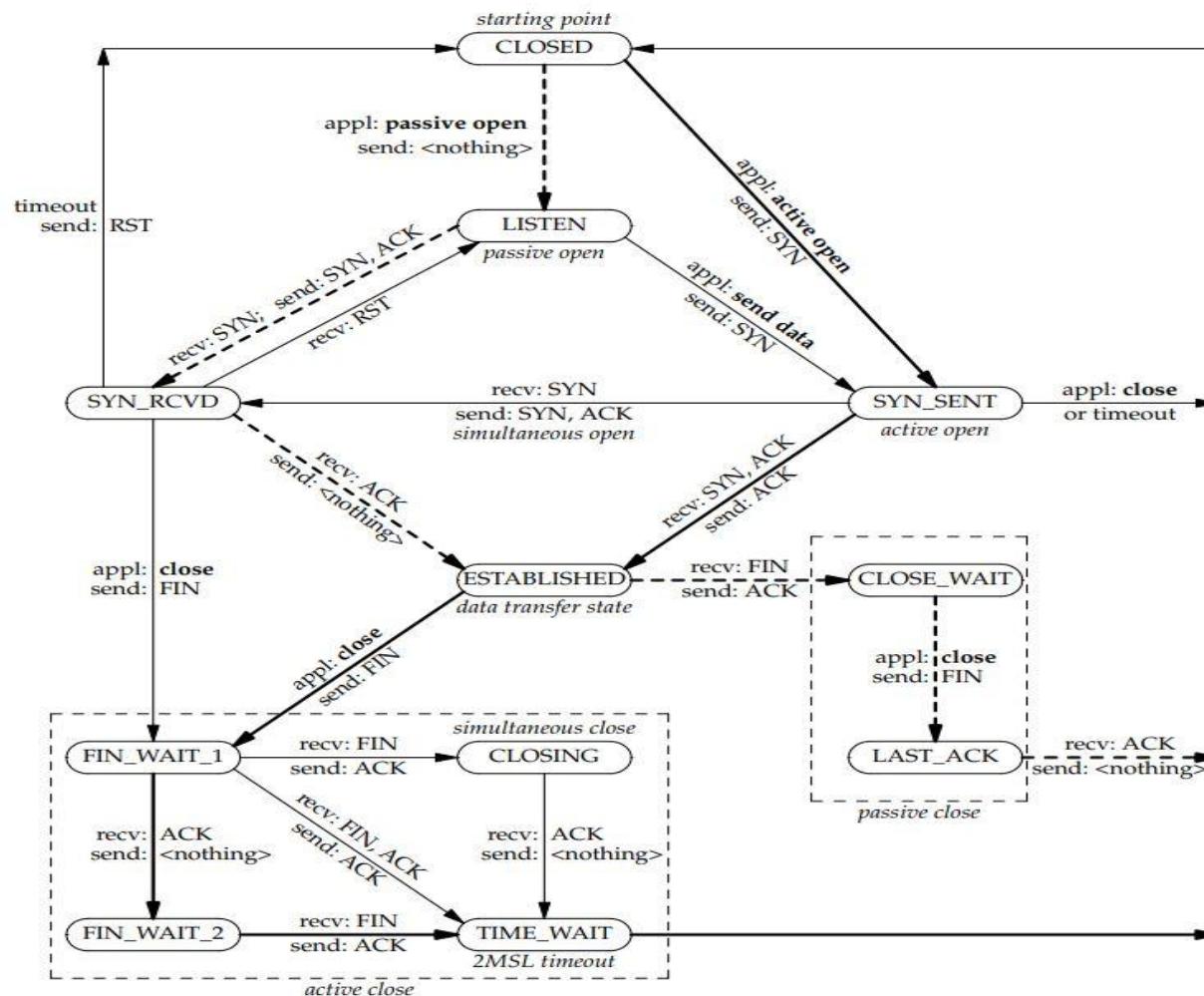
**Q** Consider a TCP client and a TCP server running on two different machines. After completing data transfer, the TCP client calls *close* to terminate the connection and a FIN segment is sent to the TCP server. Server-side TCP responds by sending an ACK, which is received by the client-side TCP. As per the TCP connection state diagram (RFC 793), in which state does the client-side TCP connection wait for the FIN from the server-side TCP? **(GATE-2017) (1 Marks)**

**(a) LAST-ACK**

**(b) TIME-WAIT**

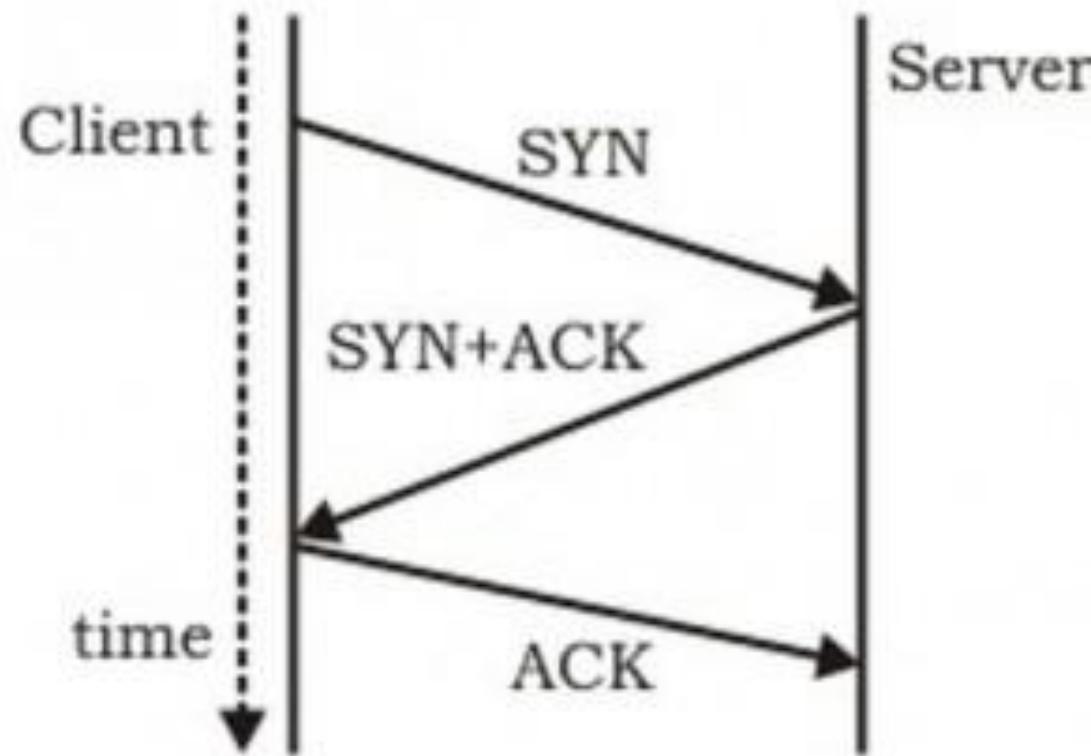
**(c) FIN-WAIT-1**

**(d) FIN-WAIT-2**



**Q** The three-way handshake for TCP connection establishment is shown below. Which of the following statements are TRUE?

- (S<sub>1</sub>) Loss of SYN + ACK from the server will not establish a connection
  - (S<sub>2</sub>) Loss of ACK from the client cannot establish the connection
  - (S<sub>3</sub>) The server moves LISTEN → SYN\_RCVD → SYN\_SENT → ESTABLISHED in the state machine on no packet loss
  - (S<sub>4</sub>) The server moves LISTEN → SYN\_RCVD → ESTABLISHED in the state machine on no packet loss. **(Gate-2008) (2 Marks)**
- (A) S<sub>2</sub> and S<sub>3</sub> only
  - (B) S<sub>1</sub> and S<sub>4</sub>
  - (C) S<sub>1</sub> and S<sub>3</sub>
  - (D) S<sub>2</sub> and S<sub>4</sub>



**Q.** Consider the following statements:

- (i)** Address Resolution Protocol (ARP) provides a mapping from an IP address to the corresponding hardware (link-layer) address.
- (ii)** A single TCP segment from a sender S to a receiver R cannot carry both data from S to R and acknowledgement for a segment from R to S.

Which one of the following is CORRECT? **(GATE 2025)**

- A)** Both (i) and (ii) are TRUE
- B)** (i) is TRUE and (ii) is FALSE
- C)** (i) is FALSE and (ii) is TRUE
- D)** Both (i) and (ii) are FALSE

# Congestion Control

- **Congestion:** Congestion refers to a network state where, the message traffic becomes so heavy that it slows down the network response time.
- Congestion control refers to techniques and mechanisms that can: Either prevent congestion before it happens or remove congestion after it has happened
  - TCP reacts to Congestion by reducing the sender window size.
  - TCP uses a combination of GBN and SR protocols to provide reliability.

## Windows in TCP

- TCP uses two windows (send window and receive window) for each direction of data transfer, i.e. four windows for a bidirectional communication.
- ***Send Window***
  - The size of the sender window is determined by the following two factors
  - **Receiver window size and Congestion window size.**
- ***Receive Window***
  - Sender should not send data greater than receiver window size. Otherwise, it leads to dropping the TCP segments which causes TCP Retransmission.
  - So, sender should always send data less than or equal to receiver window size. Receiver dictates its window size to the sender through TCP Header.

- **Congestion Window**

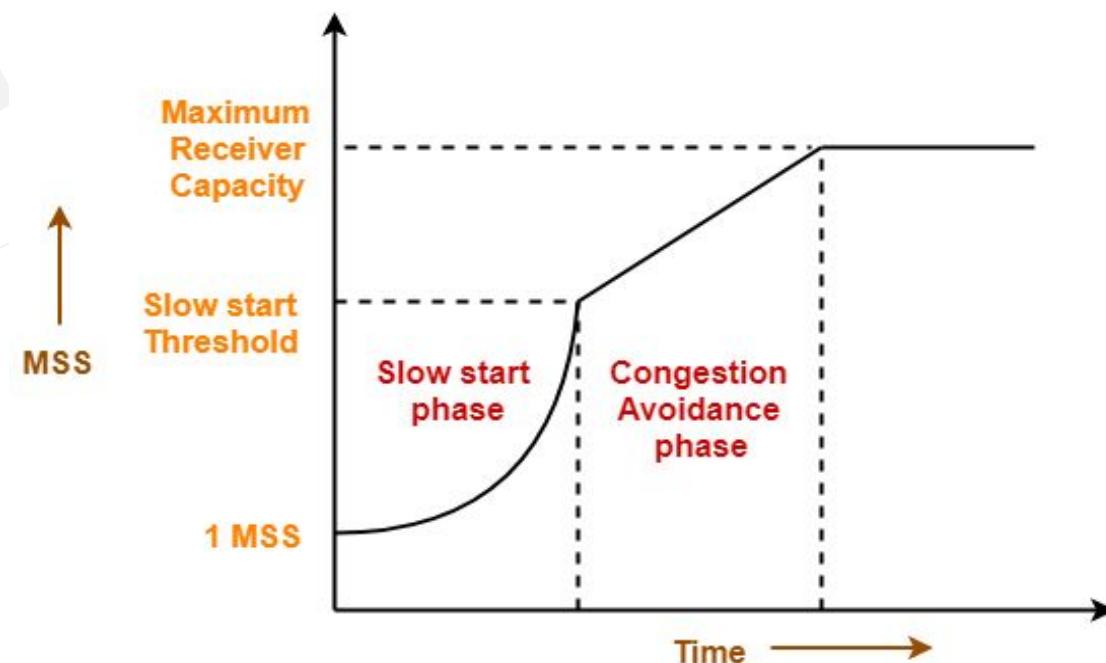
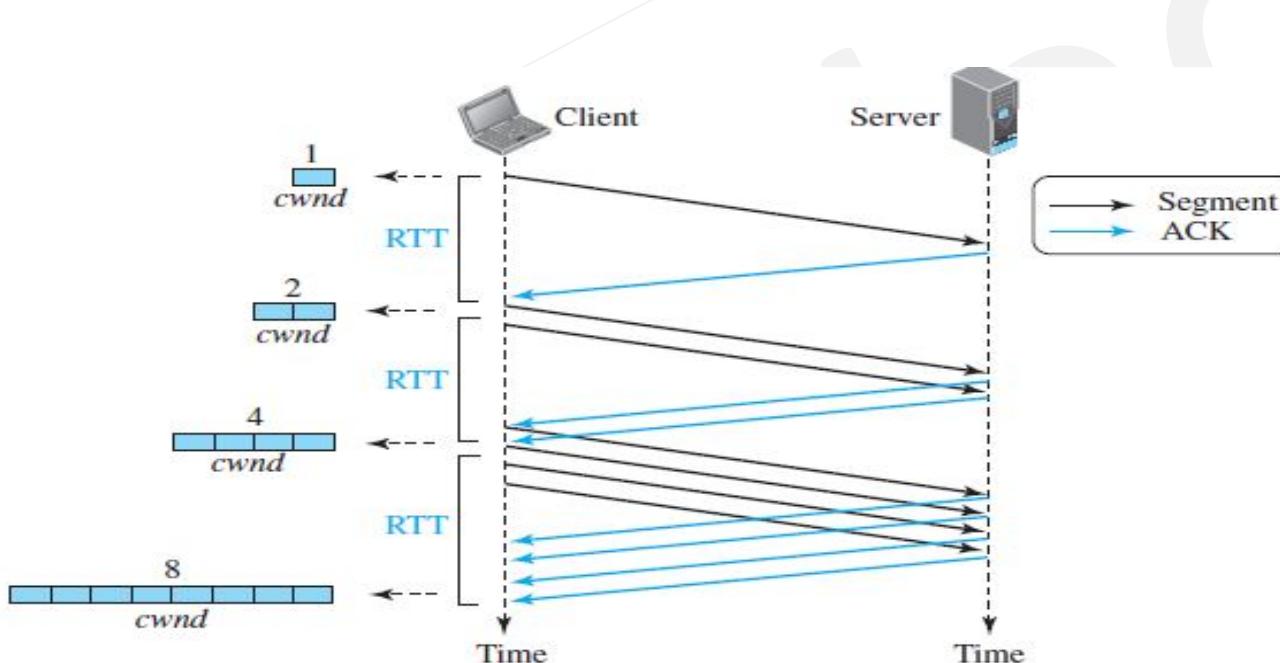
- Sender should not send data greater than congestion window size. Otherwise, it leads to dropping the TCP segments which causes TCP Retransmission.
  - So, sender should always send data less than or equal to congestion window size.
  - Different variants of TCP use different approaches to calculate the size of congestion window. Congestion window is known only to the sender and is not sent over the links.
- 
- **In general, Sender window size = Minimum (Receiver window size, Congestion window size)**

## TCP Congestion Policy

- TCP's general policy for handling congestion consists of following three phases
  - Slow Start (Exponential Increase)
  - Congestion Avoidance (Additive Increase)
  - Congestion Detection

## Slow Start Phase (Exponential Increase)

- Initially, sender sets congestion window size = Maximum Segment Size (1 MSS).
- After receiving each acknowledgment, the size of congestion window increases exponentially.
- After 1 round trip time, congestion window size =  $(2)^1$  = 2 MSS
- After 2 round trip time, congestion window size =  $(2)^2$  = 4 MSS
- After 3 round trip time, congestion window size =  $(2)^3$  = 8 MSS and so on.
- This phase continues until the congestion window size reaches the slow start threshold.
- Threshold = Maximum number of TCP segments that receiver window can accommodate / 2 = (Receiver window size / Maximum Segment Size) / 2**



**Q** Consider the following statements regarding the slow start phase of the TCP congestion control algorithm. Note that  $cwnd$  stands for the TCP congestion window and MSS denotes the Maximum Segment Size.

- (i) The  $cwnd$  increase by 2 MSS on every successful acknowledgement.
- (ii) The  $cwnd$  approximately doubles on every successful acknowledgement.
- (iii) The  $cwnd$  increase by 1 MSS every round-trip time.
- (iv) The  $cwnd$  approximately doubles every round-trip time.

Which one of the following is correct? **(GATE-2018) (1 Marks)**

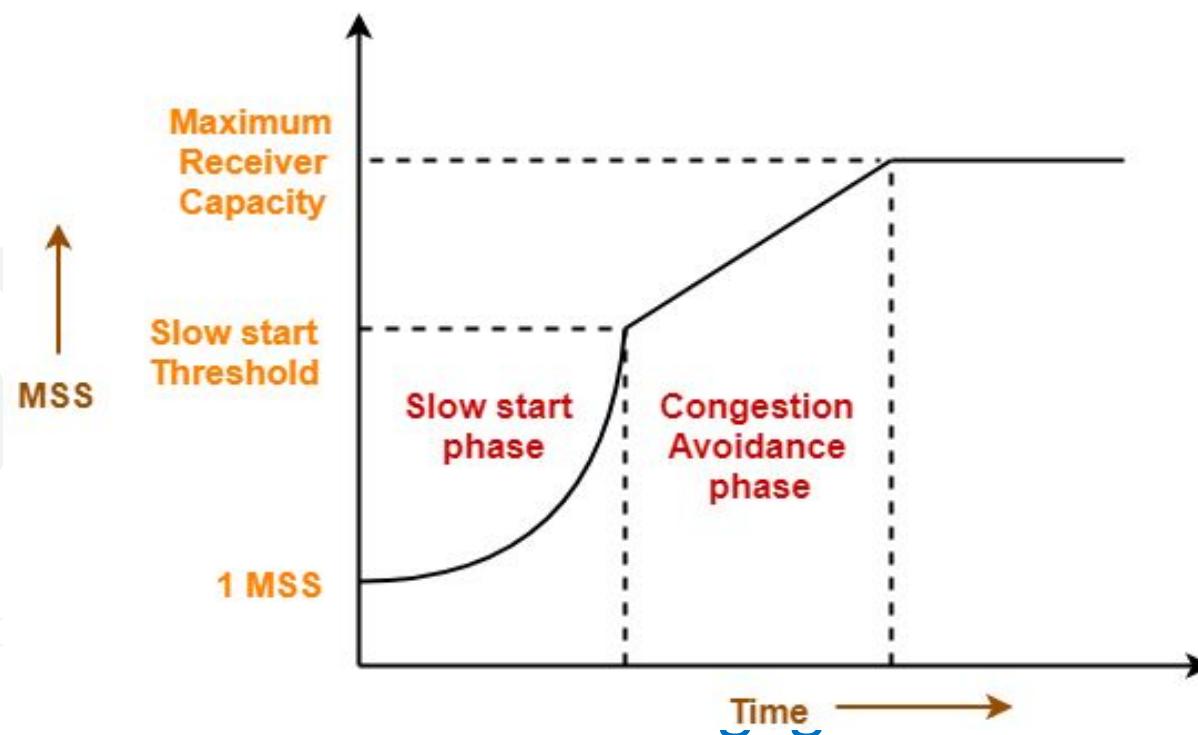
- (a) Only (ii) and (iii) are true
- (b) Only (i) and (iii) are true
- (c) Only (iv) is true
- (d) Only(i) and (iv) is true

**Q** In the slow start phase of the TCP congestion algorithm, the size of the congestion window:  
**(Gate-2008) (2 Marks)**

- a) does not increase
- b) increase linearly
- c) increases quadratically
- d) increases exponentially

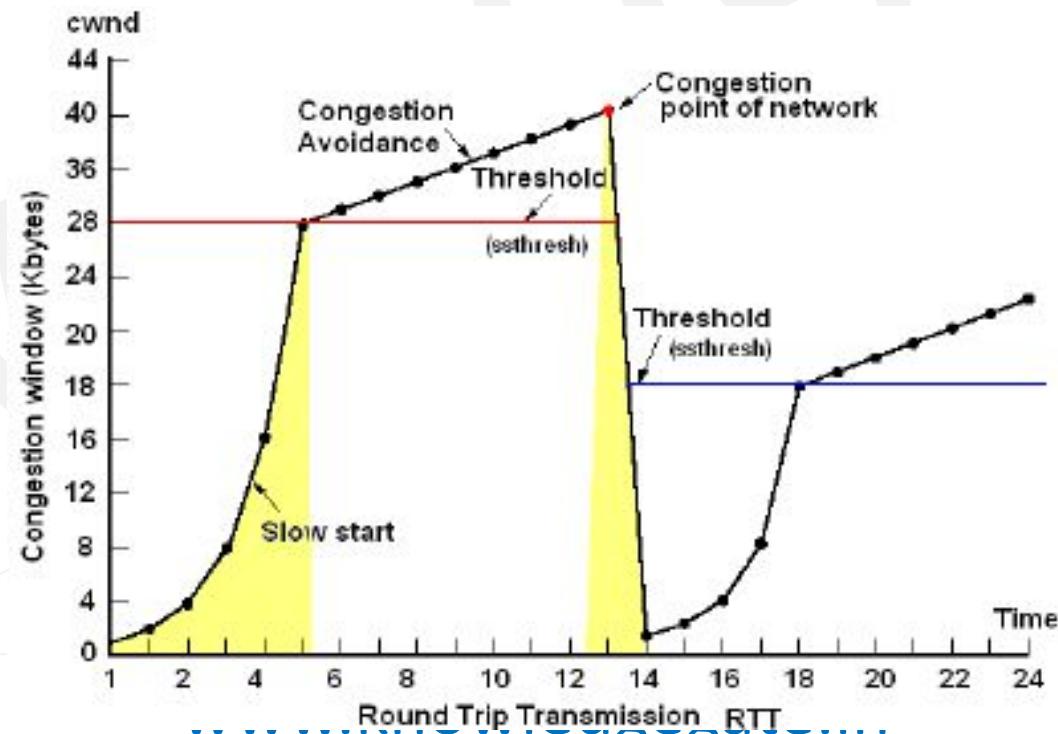
## Congestion Avoidance Phase

- After reaching the threshold, Sender increases the congestion window size linearly to avoid the congestion.
- On receiving each acknowledgement, sender increments the congestion window size by 1.
- **Congestion window size = Congestion window size + 1**, This phase continues until the congestion window size becomes equal to the receiver window size.



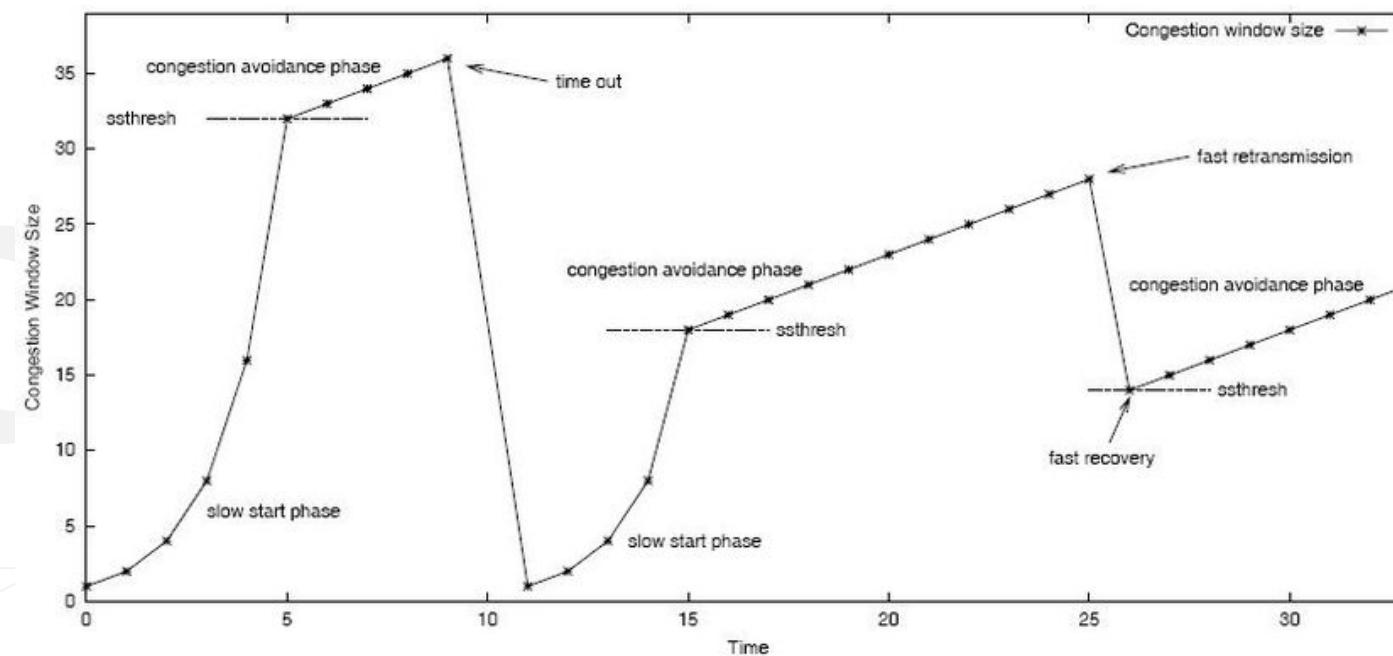
## Congestion Detection Phase

- When sender detects the loss of segments, it reacts in different ways depending on how the loss is detected
- Detection On Time Out
  - Time Out Timer expires before receiving the acknowledgement for a segment. It suggests the strong possibility of congestion in the network. There are chances that a segment has been dropped in the network.
  - Reaction: In this case, sender reacts by
    - Setting the slow start threshold to half of the current congestion window size.
    - Decreasing the congestion window size to 1 MSS.
    - Resuming the slow start phase.



## • Detection On Receiving 3 Duplicate Acknowledgements

- Sender receives 3 duplicate acknowledgements for a segment. This case suggests the weaker possibility of congestion in the network. There are chances that a segment has been dropped but few segments sent later may have reached.
- Reaction
  - In this case, sender reacts by setting the slow start threshold to half of the current congestion window size.
  - Decreasing the congestion window size to slow start threshold.
  - Resuming the congestion avoidance phase.



**Q** Let the size of congestion window of a TCP connection be 32 KB when a timeout occurs. The round-trip time of the connection is 100 msec and the maximum segment size used is 2 KB. The time taken (in msec) by the TCP connection to get back to 32 KB congestion window is \_\_\_\_\_. **(Gate-2014) (2 Marks)**

**Q** Consider an instance of TCP's Additive Increase Multiplicative Decrease (AIMD) algorithm where the window size at the start of the slow start phase is 2 MSS and the threshold at the start of the first transmission is 8 MSS. Assume that a timeout occurs during the fifth transmission. Find the congestion window size at the end of the tenth transmission. **(Gate-2012) (2 Marks)**

- a) 8 MSS
- b) 14 MSS
- c) 7 MSS
- d) 12 MSS

**Q** On a TCP connection, current congestion window size is Congestion Window = 4 KB. The window size advertised by the receiver is Advertise Window = 6 KB. The last byte sent by the sender is LastByteSent = 10240 and the last byte acknowledged by the receiver is LastByteAcked = 8192. The current window size at the sender is **(Gate-2005) (2 Marks)**

**(A)** 2048 bytes

**(B)** 4096 bytes

**(C)** 6144 bytes

**(D)** 8192 bytes

**Q** Suppose that the maximum transmit window size for a TCP connection is 12000 bytes. Each packet consists of 2000 bytes. At some point of time, the connection is in slow-start phase with a current transmit window of 4000 bytes. Subsequently, the transmitter receives two acknowledgements. Assume that no packets are lost and there are no time-outs. What is the maximum possible value of the current transmit window? **(Gate-2004) (2 Marks)**

**(A)** 4000 bytes

**(B)** 8000 bytes

**(C)** 10000 bytes

**(D)** 12000 bytes

**Q Which one of the following statements is FALSE? (Gate-2004) (1 Marks)**

- (A) TCP guarantees a minimum communication rate
- (B) TCP ensures in-order delivery
- (C) TCP reacts to congestion by reducing sender window size
- (D) TCP employs retransmission to compensate for packet loss

**Q. Consider a TCP connection operating at a point of time with the congestion window of size 12 MSS (Maximum Segment Size), when a timeout occurs due to packet loss. Assuming that all the segments transmitted in the next two RTTs (Round Trip Time) are acknowledged correctly, the congestion window size (in MSS) during the third RTT will be \_\_\_\_\_ (Gate 2024 CS) (2 Marks)(NAT)**

# Timers

- TCP uses different types of timers to ensure reliable data transmission and proper connection management:
- **Time-Wait Timer:**
  - **Purpose:** Prevents issues with late packets after a connection is closed.
  - **Action:** Keeps the connection in a wait state for  $2 \times \text{Lifetime (LT)}$  before completely closing it. This prevents the port number from being immediately reassigned and avoids conflicts with late-arriving packets.
- **Keep-Alive Timer:**
  - **Purpose:** Detects and closes **idle connections**.
  - **Action:** The server periodically checks if the connection is still active. If there is no response within the **keep-alive time**, the server sends up to 10 probe messages at intervals of 75 seconds. If no reply is received, the connection is closed.
- **Persistent Timer:**
  - **Purpose:** Handles the case of a **zero window size advertisement** by the receiver.
  - **Action:** Ensures that the sender occasionally checks if the window size has increased, preventing deadlock situations.
- **Acknowledgment Timer:**
  - **Purpose:** Used to send **cumulative acknowledgments** efficiently, usually in **piggybacking** mode.
  - **Action:** When a segment arrives, the station starts the acknowledgment timer. All segments received within this time are acknowledged with a single cumulative ACK.

- **Timer Setting:**
  - **Data Link Layer (DLL):** In the DLL, setting a timeout timer is simple since it only deals with adjacent nodes.
  - **Transport Layer:** In contrast, the transport layer handles complex scenarios involving multiple layers, networks, and varying paths. Accurately setting the timeout timer (TOT) is challenging, as incorrect settings can lead to **congestion** or inefficient resource use.
- **Network Traffic Impact on Timeout Timer**
  - **High Traffic:** When the network has high traffic, ACKs take longer to reach the sender. The **timeout timer** should be set to a larger value to account for the delay. If the timer is set too short, the sender may wrongly assume segment loss and retransmit unnecessarily, worsening congestion.
  - **Low Traffic:** When there is low traffic, ACKs reach the sender quickly. The timeout timer should be set to a smaller value. If the timer is too long, the sender waits excessively for an ACK that may already be lost, causing delays.

## Algorithms for Computing Timeout Timer

- TCP uses various algorithms to dynamically compute the timeout timer:
  - **Basic Algorithm:** A straightforward method based on round-trip time (RTT) measurements.
  - **Jacobson's Algorithm:** An improved algorithm that considers variability in RTT to better estimate the timeout.
  - **Karn's Modification:** A technique to handle retransmitted segments accurately by ignoring RTT measurements for retransmitted packets.

## General Rules for Algorithms (Basic algorithm)

- **Rule 1:** Increase the **timeout timer** if the actual round-trip time (RTT) increases, indicating high traffic.
- **Rule 2:** Decrease the **timeout timer** if the actual RTT decreases, indicating low traffic.
- **Steps in Basic Algorithm:**
- **Sending the 1st Segment:**
  - The sender starts with an initial RTT, denoted as **IRTT<sub>1</sub>** (an assumed random value).
  - Sets the timeout timer for the 1st segment as **TOT<sub>1</sub> = 2 × IRTT<sub>1</sub>**.
  - After sending the 1st segment, the actual RTT (ARTT<sub>1</sub>) is calculated based on the acknowledgment arrival time.
- **Sending the 2nd Segment:**
  - Sender computes the value of initial RTT for the 2nd segment using the relation
    - $IRTT_{n+1} = \alpha IRTT_n + (1 - \alpha) ARTT_n$
  - Here,  $\alpha$  is called smoothing factor where  $0 \leq \alpha \leq 1$  (Its value will be given in questions)
  - Substituting  $n=1$ , sender gets  $IRTT_2 = \alpha IRTT_1 + (1 - \alpha) ARTT_1$ .
  - So, after sending the 2nd segment, sender expects its ACK to arrive in time  $IRTT_2$ .
  - Sender sets time out timer value (TOT) for the 2nd segment to be- **TOT<sub>2</sub> = 2 × IRTT<sub>2</sub>**
  - Suppose ACK for the 2nd segment arrives in time  $ARTT_2$ .
  - Here,  $ARTT_2$  = Actual Round-Trip Time for the 2nd segment.
- **Iterative Process:** The algorithm continues using this method to calculate the initial RTT and set the timeout timer for each subsequent segment.
- **Advantages:**
  - The **timeout timer** value adapts dynamically to changing round-trip times.
  - Takes into account previously sent segments to refine the current RTT estimation.
- **Disadvantage:**
  - The algorithm **always uses a multiplier of 2** for the timeout timer value, which lacks a logical basis and may not be optimal for all scenarios.

## Van Jacobson

- An American computer scientist born in 1950.
- Known for his significant contributions to **TCP/IP network performance** and scalability.
- One of the key figures in the development of the **TCP/IP protocol stack**, which forms the backbone of today's Internet.
- Since 2013, he has been an adjunct professor at the University of California, Los Angeles (UCLA) working on **Named Data Networking**.
- **Jacobson's Algorithm:**
  - A refined and modified version of the **basic algorithm** for calculating **timeout timers** in TCP.
  - Offers **improved performance** over the basic algorithm by addressing the limitations and incorporating better RTT measurement techniques.



- **Step-01: Sending 1st Segment-**

- Sender assumes any random value of initial RTT say  $\text{IRTT}_1$ .
- So, after sending the 1st segment, sender expects its ACK to arrive in time  $\text{IRTT}_1$ .
- Sender assumes any random value of initial deviation say  $\text{ID}_1$ .
- So, after sending the 1st segment, sender expects there will be a deviation of  $\text{ID}_1$  time from  $\text{IRTT}_1$ .
- Sender sets time out timer value (TOT) for the 1st segment to be-
  - $\text{TOT}_1 = 4 \times \text{ID}_1 + \text{IRTT}_1$
- Suppose ACK for the 1st segment arrives in time  $\text{ARTT}_1$ . Here,  $\text{ARTT}_1$  = Actual Round-Trip Time for the 1st segment.
- Then, Actual deviation from  $\text{IRTT}_1$  is given by-
- $\text{AD}_1 = | \text{IRTT}_1 - \text{ARTT}_1 |$

- **Step-02: Sending 2nd Segment-**

- Sender computes the value of initial RTT for the 2nd segment using the relation-
  - $\text{IRTT}_{n+1} = \alpha \text{IRTT}_n + (1 - \alpha) \text{ARTT}_n$
  - Here,  $\alpha$  is called smoothing factor where  $0 \leq \alpha \leq 1$  (Its value will be given in questions)
- Sender computes the value of initial deviation for the 2nd segment using the relation
  - $\text{ID}_{n+1} = \alpha \text{ID}_n + (1 - \alpha) \text{AD}_n$
  - Here,  $\alpha$  is called smoothing factor where  $0 \leq \alpha \leq 1$  (Its value will be given in questions)
- Substituting  $n=1$ , sender gets
  - $\text{IRTT}_2 = \alpha \text{IRTT}_1 + (1 - \alpha) \text{ARTT}_1$
  - $\text{ID}_2 = \alpha \text{ID}_1 + (1 - \alpha) \text{AD}_1$
- So after sending the 2nd segment, sender expects its ACK to arrive in time  $\text{IRTT}_2$  with deviation of  $\text{ID}_2$  time.
- Sender sets time out timer value (TOT) for the 2nd segment to be-
  - $\text{TOT}_2 = 4 \times \text{ID}_2 + \text{IRTT}_2$
- Suppose ACK for the 2nd segment arrives in time  $\text{ARTT}_2$ . Here,  $\text{ARTT}_2$  = Actual Round Trip Time for the 2nd segment.
- Then, Actual deviation from  $\text{IRTT}_2$  is given by-
  - $\text{AD}_2 = | \text{IRTT}_2 - \text{ARTT}_2 |$
- In the similar manner, algorithm computes the time out timer value for all the further segments.

## Problems with Basic Algorithm and Jacobson's Algorithm

- Karn's Modification aims to improve the accuracy of timeout estimations in TCP transmissions. The key challenge addressed is the situation where a packet is retransmitted due to a timeout before the acknowledgment (ACK) arrives. Since the actual round-trip time (RTT) cannot be determined accurately in such cases, Karn's algorithm recommends doubling the timeout timer (TOT) instead of using the measured RTT. Here's a concise summary:
- **Problem in Basic and Jacobson's Algorithms:** Both algorithms rely on the actual RTT of previous segments to calculate the initial RTT for the current segment. However, if the ACK arrives late and causes a timeout, the RTT cannot be accurately measured, leading to retransmission and incorrect calculations.
- **Karn's Solution:** Karn's algorithm suggests ignoring the RTT measurements for retransmitted segments. Instead, it prescribes doubling the timeout timer each time the timer expires, ensuring that the next retransmission adapts to increased delays.



**Q** Consider the following statements about the timeout value used in TCP.

i. The timeout value is set to the RTT (Round Trip Time) measured during TCP connection establishment for the entire duration of the connection.

ii. Appropriate RTT estimation algorithm is used to set the timeout value of a TCP connection.

iii. Timeout value is set to twice the propagation delay from the sender to the receiver.

Which of the following choices hold? **(Gate-2007) (1 Marks)**

(A) (i) is false, but (ii) and (iii) are true

(B) (i) and (iii) are false, but (ii) is true

(C) (i) and (ii) are false, but (iii) is true

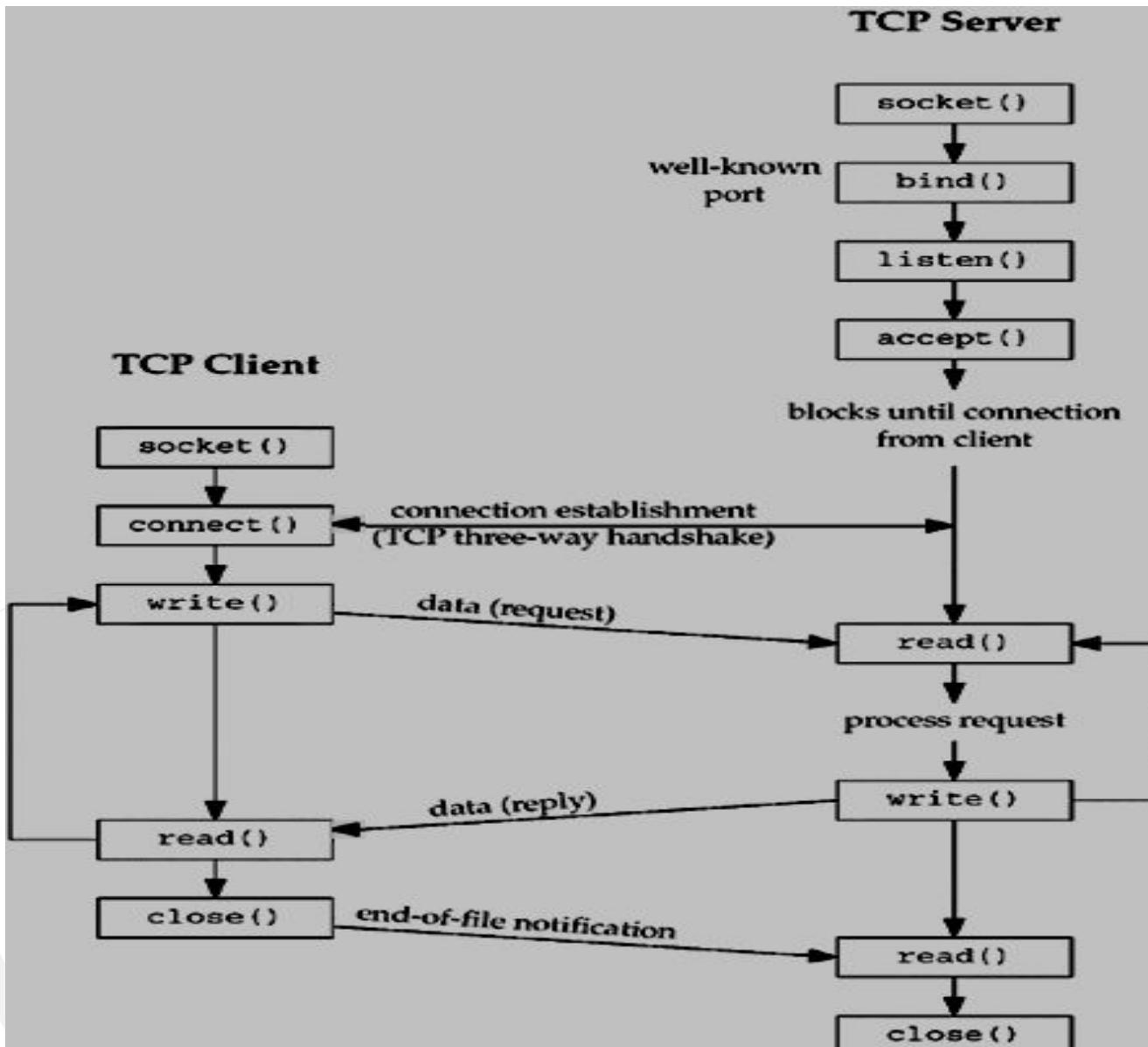
(D) (i), (ii) and (iii) are false

## Silly Window Syndrome

- Silly Window Syndrome (SWS) is a problem in TCP due to poor handling of data transfer, causing inefficiency in communication. It occurs when small amounts of data are repeatedly sent or received, which leads to smaller and smaller window sizes. This issue can happen in two scenarios:
- **Sender sends small segments repeatedly** - inefficient data transmission occurs due to the sender frequently sending tiny packets.
- **Receiver accepts small amounts of data repeatedly** - leading to a reduced and 'silly' window size, which becomes inefficient for communication.

- **Solutions for SWS:**
- **Nagle's Algorithm** - It focuses on preventing the sender from sending small segments:
  - The sender transmits the first small packet and buffers all subsequent data until the previously sent packet is acknowledged.
  - After receiving the acknowledgment, the buffered data is sent as a single segment, reducing overhead and network congestion.
- **Clark's Solution** - It addresses the receiver's acceptance of small packets:
  - The receiver should avoid sending window updates for tiny amounts of available space.
  - Instead, the receiver waits until there's a substantial window size or half the buffer is free before sending an update.
- **Important Notes:**
  - **Nagle's Algorithm** is not suitable for applications requiring instant data transmission, as it introduces delays by waiting for acknowledgment.
  - Both **Nagle's Algorithm and Clark's Solution** work together to ensure that senders don't transmit small packets, and receivers don't request them unnecessarily.

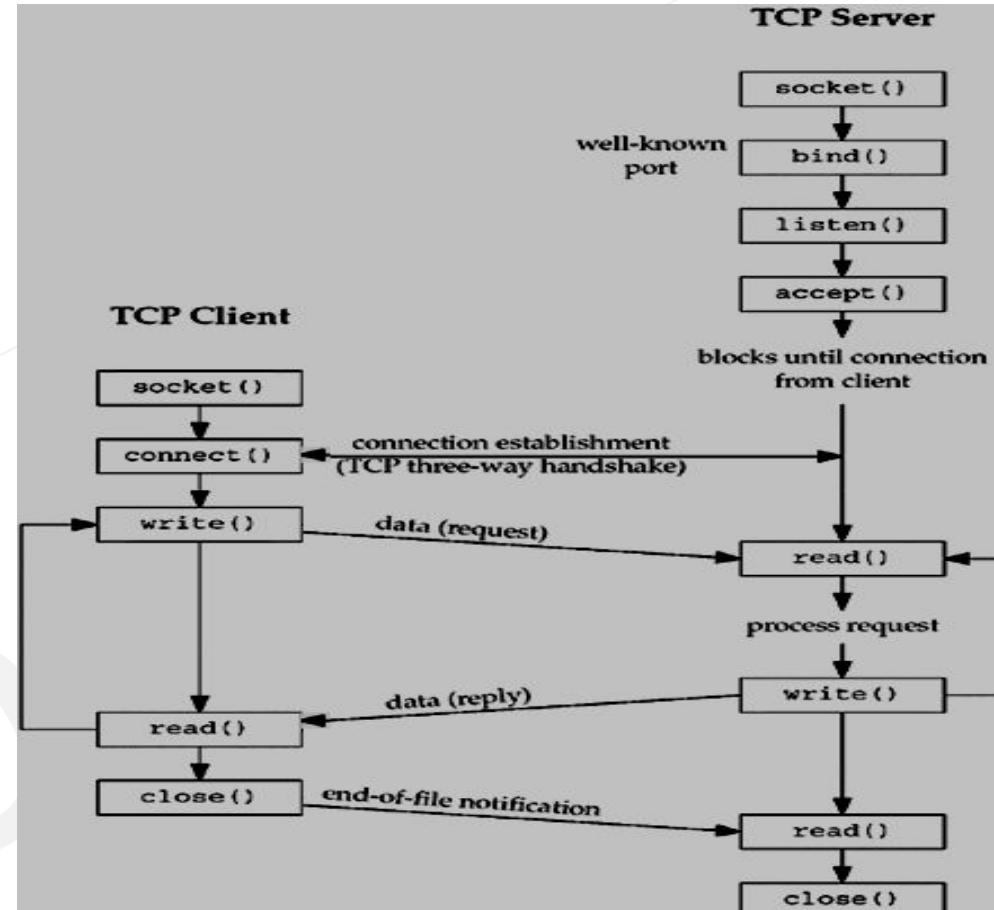
# UNIX socket API



**Q** Identify the correct order in which a server process must invoke the function calls accept, bind, listen, and recv according to UNIX socket API. **(Gate-2015) (1 Marks)**

**(A)** listen, accept, bind recv  
**(C)** bind, accept, listen, recv

**(B)** bind, listen, accept, recv  
**(D)** accept, listen, bind, recv



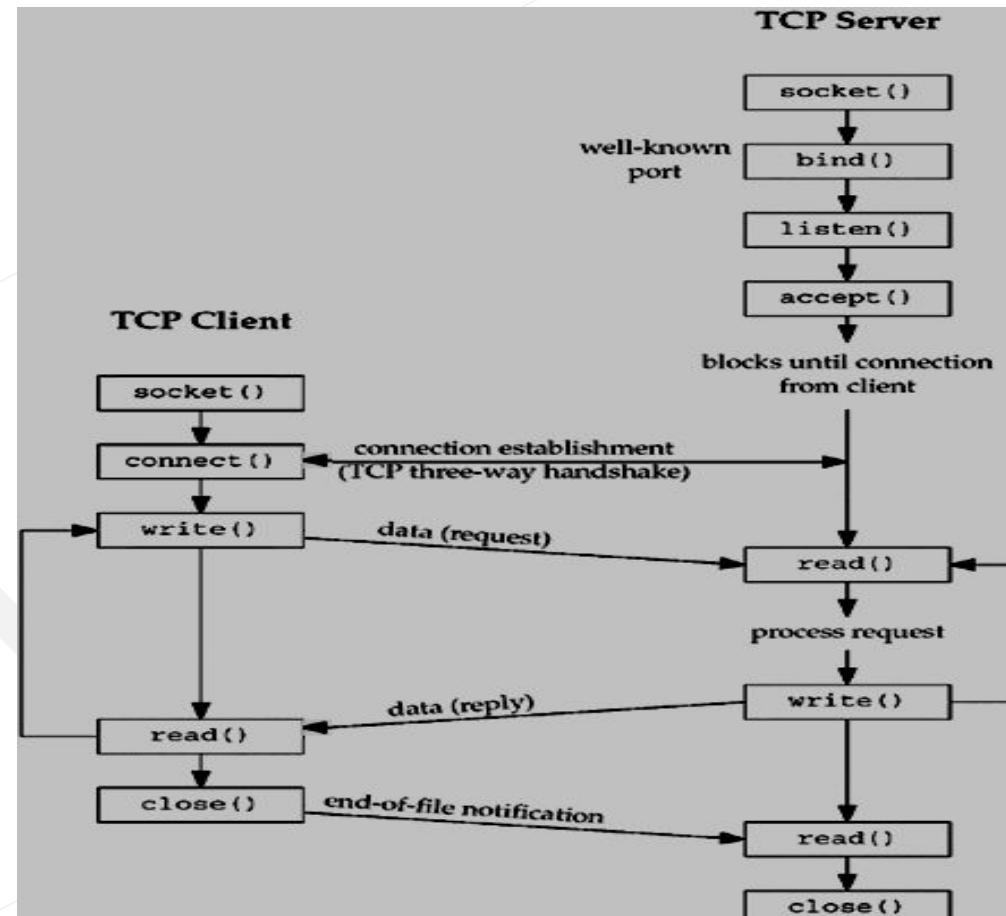
**Q** Which one of the following socket API functions converts an unconnected active TCP socket into a passive socket? **(Gate-2014) (1 Marks)**

**(a) CONNECT**

**(b) BIND**

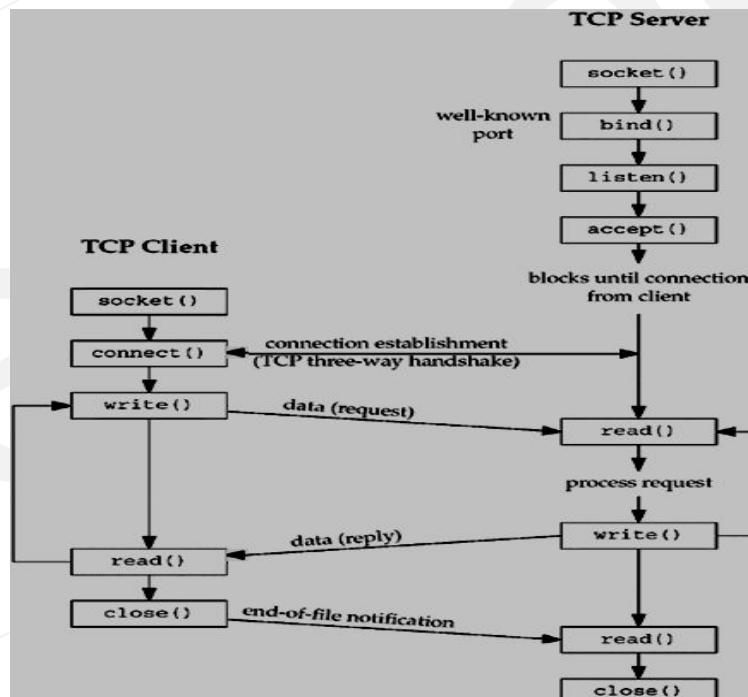
**(c) LISTEN**

**(d) ACCEPT**



**Q** A client process P needs to make a TCP connection to a server process S. Consider the following situation: the server process S executes a socket (), a bind () and a listen () system call in that order, following which it is pre-empted. Subsequently, the client process P executes a socket () system call followed by connect () system call to connect to the server process S. The server process has not executed any accept () system call. Which one of the following events could take place? **(Gate-2008) (2 Marks)**

- (A)** connect () system call returns successfully
- (B)** connect () system call blocks
- (C)** connect () system call returns an error
- (D)** connect () system call results in a core dump



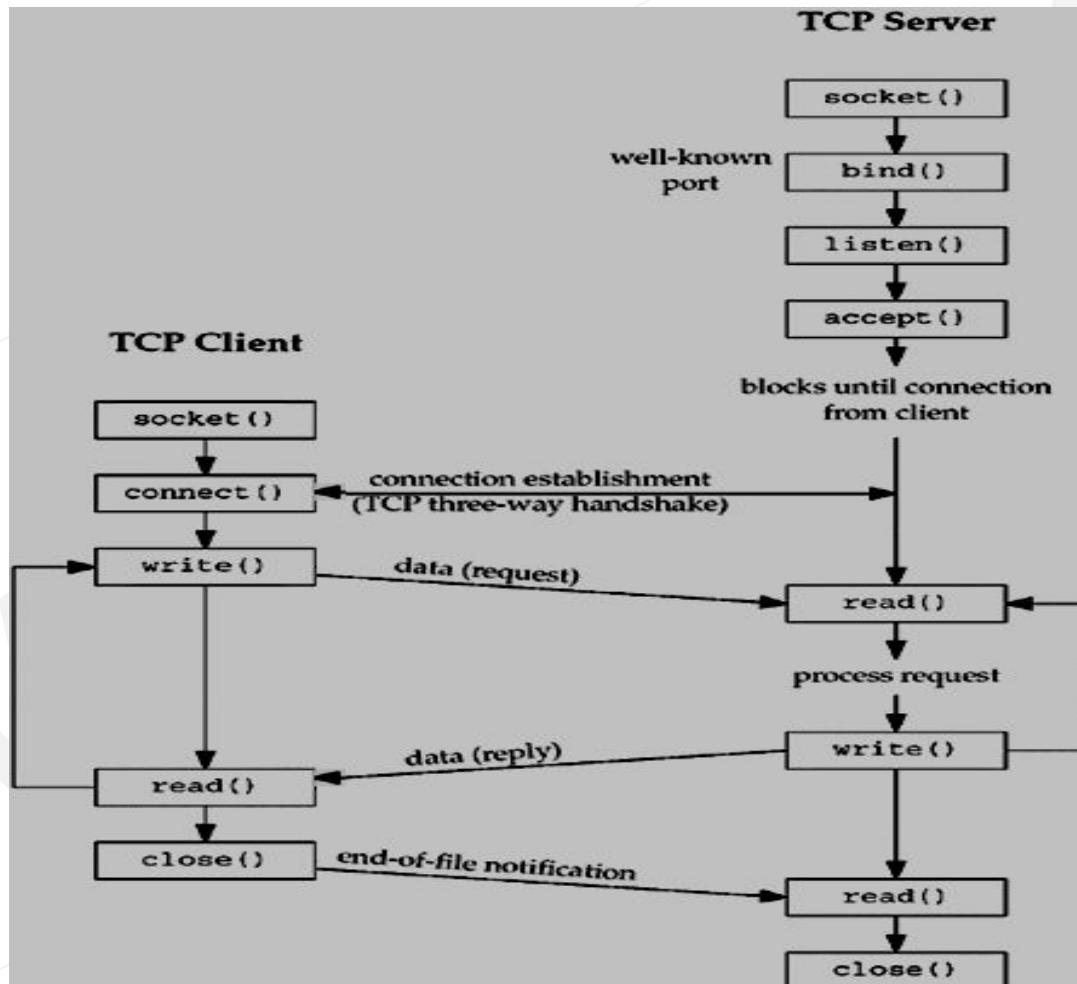
**Q Which of the following system calls results in the sending of SYN packets? (Gate-2008) (1 Marks)**

**(A) socket**

**(B) bind**

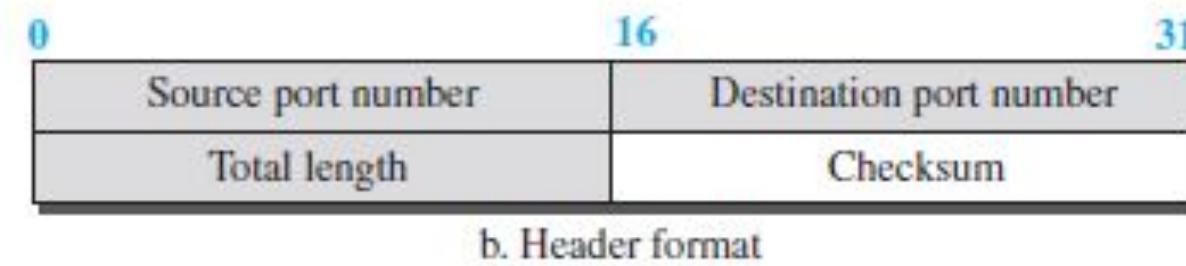
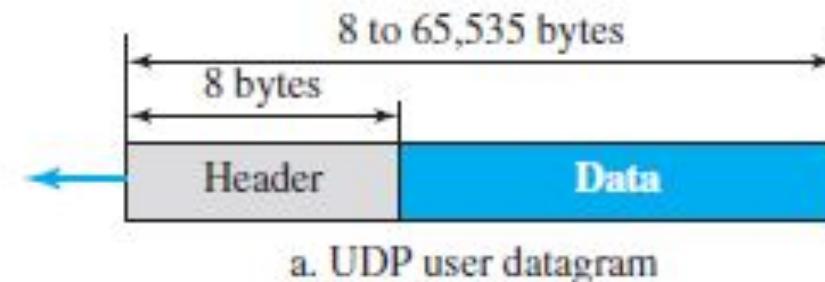
**(C) listen**

**(D) connect**



## USER DATAGRAM PROTOCOL (UDP)

- **Overview:**
- UDP is a connectionless and unreliable transport protocol.
- Unlike TCP, UDP does not guarantee reliability but focuses on simplicity and low overhead.
- It provides process-to-process communication using socket addresses (IP + port numbers).



- **Features of UDP:**

- **Lightweight and Simple:**

- Minimum overhead, used for quick communication.
- Ideal for applications that do not require heavy interaction between sender and receiver.

- **User Datagram Structure:**

- Consists of a fixed 8-byte header.
- Includes fields for source and destination port numbers, total length, and optional checksum.

- **Key Characteristics:**

- **Connectionless:** Each datagram is independent.
- **No numbering or sequencing of datagrams.**
- **No connection establishment or termination.**

- **Limitations:**

- No built-in flow control, error control, or congestion control mechanisms.
- Relies on simple checksum for basic error checking.

- **Applications of UDP:**

- Suitable for scenarios with short, request-response communication.
- Commonly used in applications where the process includes its own error or flow control (e.g., TFTP).
- Preferred for real-time applications (e.g., DNS queries, multimedia streaming) due to its tolerance for minor data loss over delays.
- Utilized in multicasting, SNMP management, and route updating protocols like RIP.

**Q Match the following: (GATE-2018) (1 Marks)**

<u>Field</u>	<u>Length in bits</u>
P. UDP Header's Port Number	I. 48
Q. Ethernet MAC Address	II. 8
R. IPv6 Next Header	III.32
S. TCP Header's Sequence Number	IV. 16

- (a) P-III, Q-IV, R-II, S-I  
(c) P-IV, Q-I, R-II, S-III

- (b) P-II, Q-I, R-IV, S-III  
(d) P-IV, Q-I, R-III, S-II

**Q Which of the following statements are TRUE? (Gate-2008) (2 Marks)**

- (S<sub>1</sub>) TCP handles both congestion and flow control
  - (S<sub>2</sub>) UDP handles congestion but not flow control
  - (S<sub>3</sub>) Fast retransmit deals with congestion but not flow control
  - (S<sub>4</sub>) Slow start mechanism deals with both congestion and flow control
- (A) S<sub>1</sub>, S<sub>2</sub> and S<sub>3</sub> only**
- (B) S<sub>1</sub> and S<sub>3</sub> only**
- (C) S<sub>3</sub> and S<sub>4</sub> only**
- (D) S<sub>1</sub>, S<sub>3</sub> and S<sub>4</sub> only**

**Q** A program on machine X attempts to open a UDP connection to port 5376 on a machine Y, and a TCP connection to port 8632 on machine Z. However, there are no applications listening at the corresponding ports on Y and Z. An ICMP Port Unreachable error will be generated by

**(Gate-2006) (2 Marks)**

(A) Y but not Z

(B) Z but not Y

(C) Neither Y nor Z

(D) Both Y and Z

**Q** Packets of the same session may be routed through different paths in: **(Gate-2005) (1 Marks)**

**(a)** TCP, but not UDP

**(b)** TCP and UDP

**(c)** UDP, but not TCP

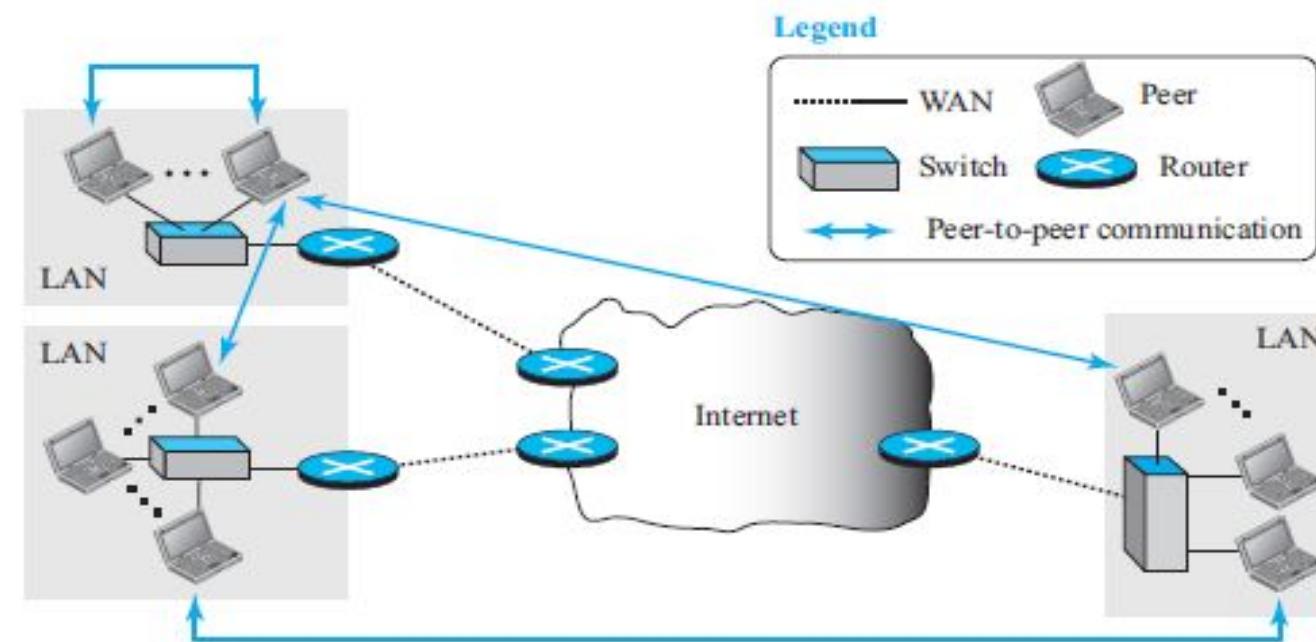
**(d)** Neither TCP nor UDP

**Q** A firewall is to be configured to allow hosts in a private network to freely open TCP connections and send packets on open connections. However, it will only allow external hosts to send packets on existing open TCP connections or connections that are being opened (by internal hosts) but not allow them to open TCP connections to hosts in the private network. To achieve this the minimum capability of the firewall should be that of **(GATE-2007) (1 Marks)**

- (A)** A combinational circuit
- (B)** A finite automaton
- (C)** A pushdown automaton with one stack
- (D)** A pushdown automaton with two stacks

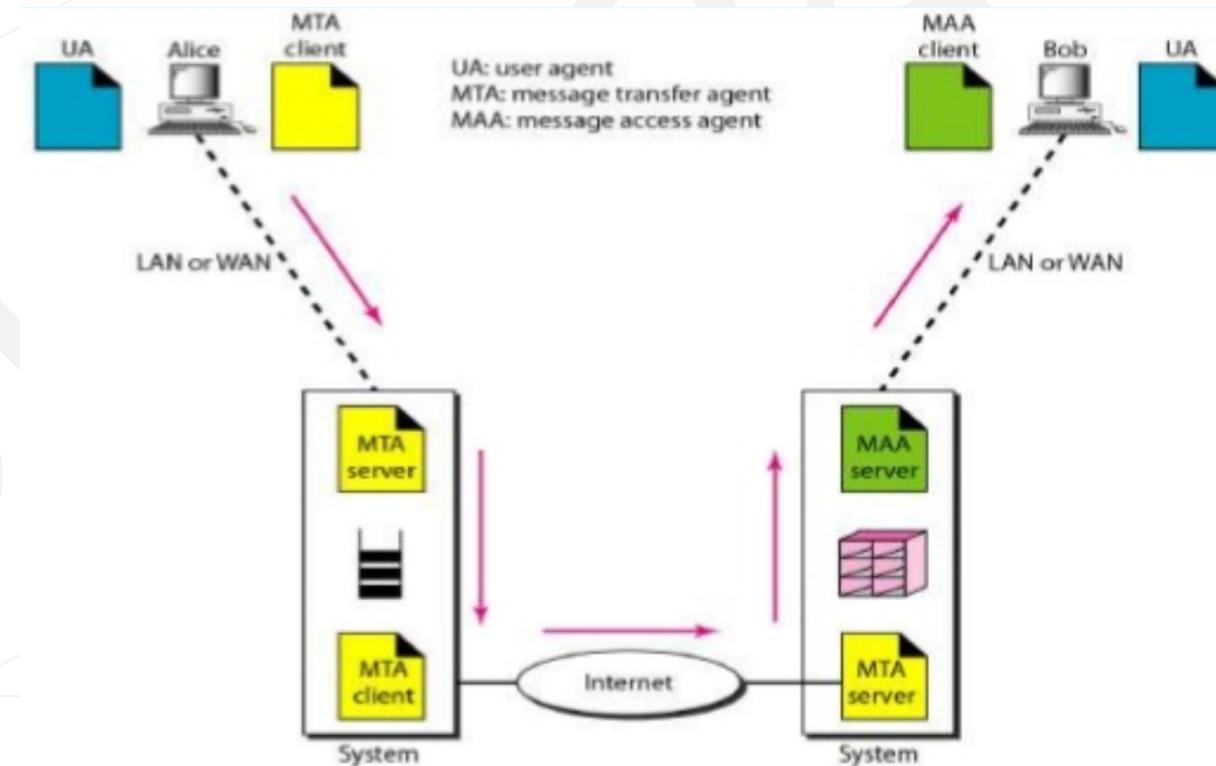
# Application Layer

- The **application layer** provides users with the interface to access network services such as email, file transfer, web browsing, and more. It facilitates communication between applications over a network through logical connections.
- **Two Key Paradigms:**
- **Client-Server Paradigm:**
  - **Structure:** There is a clear distinction between server (service provider) and client (service requester).
  - **Operation:** The server runs continuously, waiting for connections from clients to provide services. It is a centralized model.
- **Peer-to-Peer (P2P) Paradigm:**
  - **Structure:** Responsibilities are distributed among peers (computers). Each peer can act as both a client and a server.
  - **Operation:** Peers communicate directly without relying on a continuously running central server. This model is more flexible and scalable for sharing resources and services.



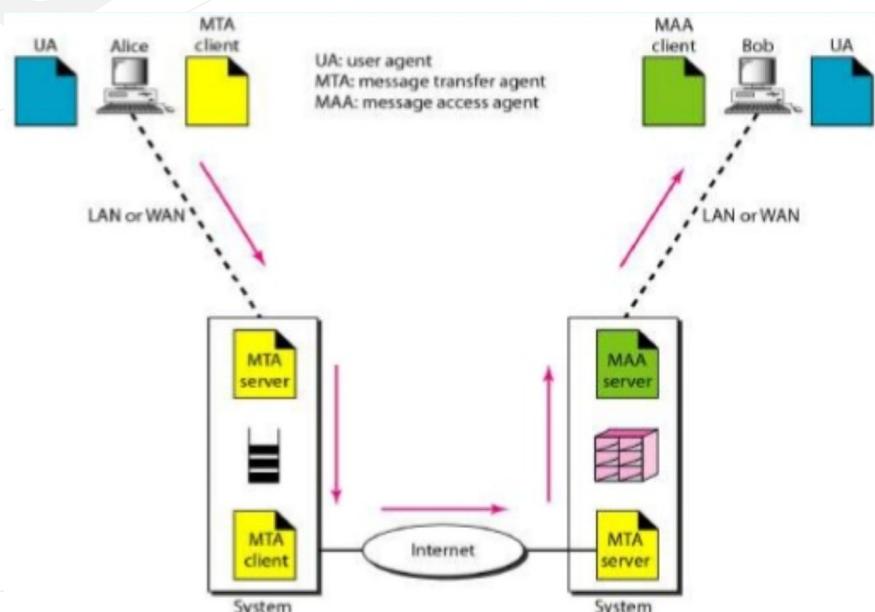
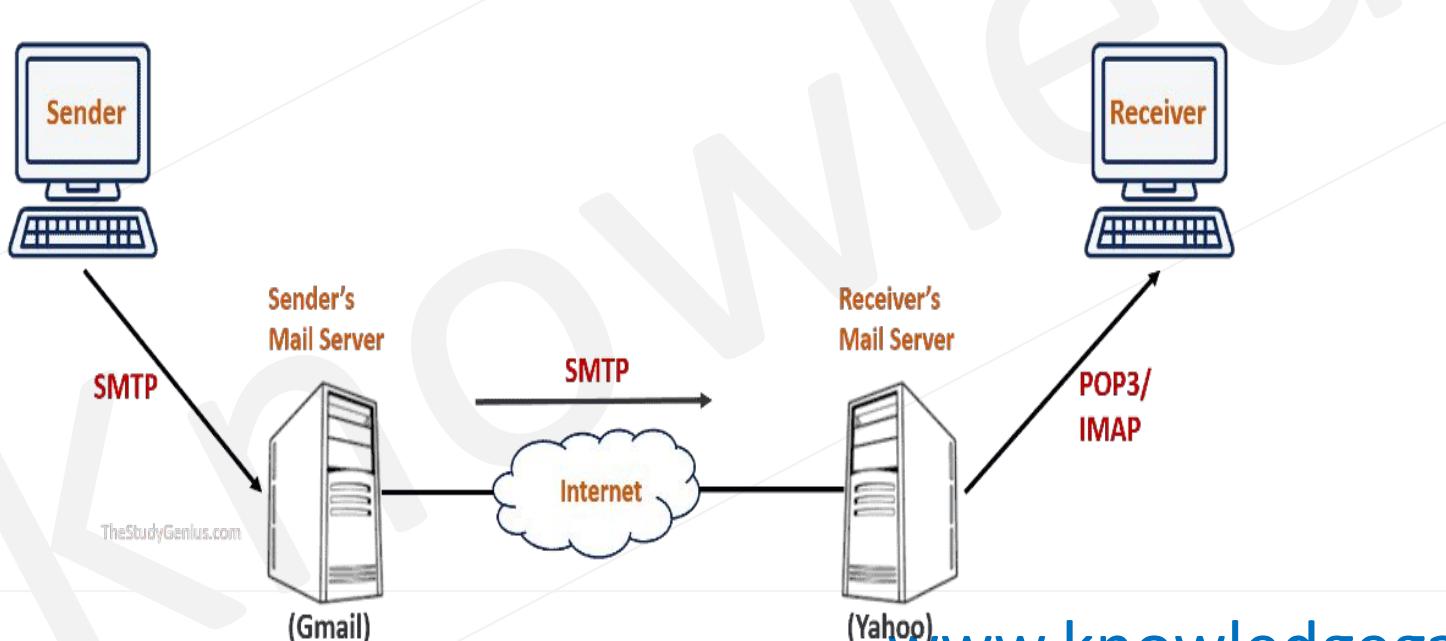
## ELECTRONIC MAIL

- Email is one of the most popular services on the Internet. Initially, emails consisted of only text messages meant for quick exchanges, but over time, it has evolved to support text, audio, and video, allowing messages to be sent to multiple recipients simultaneously.



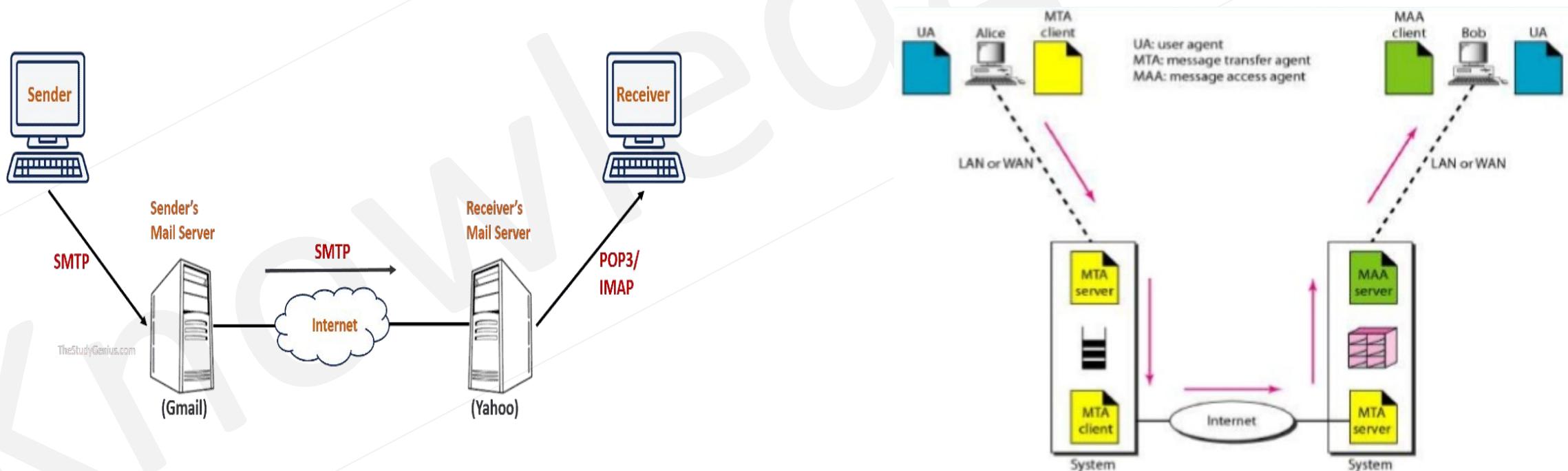
# Message Transfer Agent: SMTP

- **Message Transfer Agent (MTA) and SMTP**
- **Role of MTA:** Manages the actual transfer of emails. Sending and receiving systems must have MTA clients and servers, respectively.
- **SMTP Protocol:** Simple Mail Transfer Protocol (SMTP) is used twice:
  - Between the sender and their mail server.
  - Between the sender's and recipient's mail servers.
- **SMTP Functionality:** Defines the format for sending commands and responses to transfer messages.
- **Mail Transfer Phases:**
  - **Connection Establishment:** Establishes a link between sender and receiver mail servers.
  - **Mail Transfer:** Transfers the actual content of the email.
  - **Connection Termination:** Ends the established link.



## Message Access Agent: POP and IMAP

- **Third Stage Requirement:** SMTP, a push protocol, is not involved in the final stage of mail delivery. Instead, a pull protocol is needed to fetch emails from the server to the client.**Protocols Used:** **POP3 (Post Office Protocol, version 3):** Basic and widely used protocol for accessing emails.
- **IMAP4 (Internet Mail Access Protocol, version 4):** More sophisticated, offering better features for managing and accessing emails.



# POP3

- Post Office Protocol, version 3 (POP3) is simple and limited in functionality. The client POP3 software is installed on the recipient computer; the server POP3 software is installed on the mail server.
- Mail access starts with the client when the user needs to download e-mail from the mailbox on the mail server. The client opens a connection to the server on TCP port 110. It then sends its user name and password to access the mailbox.
- The user can then list and retrieve the mail messages, one by one. POP3 has two modes: the delete mode and the keep mode. In the delete mode, the mail is deleted from the mailbox after each retrieval. In the keep mode, the mail remains in the mailbox after retrieval.
- The delete mode is normally used when the user is working at her permanent computer and can save and organize the received mail after reading or replying.
- The keep mode is normally used when the user accesses her mail away from her primary computer (e.g., a laptop). The mail is read but kept in the system for later retrieval and organizing.

## IMAP4

- Another mail access protocol is Internet Mail Access Protocol, version 4 (IMAP4). IMAP4 is similar to POP3, but it has more features; IMAP4 is more powerful and more complex.
- POP3 is deficient in several ways. It does not allow the user to organize her mail on the server; the user cannot have different folders on the server. (Of course, the user can create folders on her own computer.)
- In addition, POP3 does not allow the user to partially check the contents of the mail before downloading. IMAP4 provides the following extra functions:
  - A user can check the e-mail header prior to downloading.
  - A user can search the contents of the e-mail for a specific string of characters prior to downloading.
  - A user can partially download e-mail. This is especially useful if bandwidth is limited and the e-mail contains multimedia with high bandwidth requirements.
  - A user can create, delete, or rename mailboxes on the mail server.
  - A user can create a hierarchy of mail boxes in a folder for e-mail storage.

## **MIME**

- Electronic mail has a simple structure. Its simplicity, however, comes at a price. It can send messages only in NVT 7-bit ASCII format. In other words, it has some limitations. For example, it cannot be used for languages that are not supported by 7-bit ASCII characters (such as Hindi, French, German, Hebrew, Russian, Chinese, and Japanese).
- Also, it cannot be used to send binary files or video or audio data.
- Multipurpose Internet Mail Extensions (MIME) is a supplementary protocol that allows non-ASCII data to be sent through e-mail.
- MIME transforms non-ASCII data at the sender site to NVT ASCII data and delivers them to the client MTA to be sent through the Internet. The message at the receiving side is transformed back to the original data.

## **Web-Based Mail**

- E-mail is such a common application that some websites today provide this service to anyone who accesses the site. Two common sites are Hotmail and Yahoo.
- The idea is very simple. Mail transfer from Alice's browser to her mail server is done through HTTP.
- The transfer of the message from the sending mail server to the receiving mail server is still through SMTP.
- Finally, the message from the receiving server (the Web server) to Bob's browser is done through HTTP.
- The last phase is very interesting. Instead of POP3 or IMAP4, HTTP is normally used. When Bob needs to retrieve his e-mails, he sends a message to the website (Hotmail, for example).
- The website sends a form to be filled in by Bob, which includes the log-in name and the password. If the log-in name and password match, the e-mail is transferred from the Web server to Bob's browser in HTML format.

**Q** Which of the following protocol pairs can be used to send and retrieve e-mails (in that order)? **(Gate-2019) (1 Marks)**

**(a)** SMTP, MIME

**(b)** SMTP, POP3

**(c)** IMAP, POP3

**(d)** IMAP, SMTP

**Q** Which of the following transport layer protocols is used to support electronic mail? **(Gate-2012) (1 Marks)**

**(A)** SMTP

**(B)** IP

**(C)** TCP

**(D)** UDP

**Q** Consider different activities related to email:

$m_1$ : Send an email from a mail client to a mail server

$m_2$ : Download an email from mailbox server to a mail client

$m_3$ : Checking email in a web browser

Which is the application level protocol used in each activity? **(GATE-2011) (1 Marks)**

(A)  $m_1$ : HTTP  $m_2$ : SMTP  $m_3$ : POP

(B)  $m_1$ : SMTP  $m_2$ : FTP  $m_3$ : HTTP

(C)  $m_1$ : SMTP  $m_2$ : POP  $m_3$ : HTTP

(D)  $m_1$ : POP  $m_2$ : SMTP  $m_3$ : IMAP

# WWW

- The **World Wide Web (WWW)**, introduced by Tim Berners-Lee in 1989 at CERN, is a global information system linked through web pages. Its user-friendly features, flexibility, and portability make it unique among internet services. Initially developed for managing scientific resources, it has evolved into a distributed client-server system where users access information through web browsers.
- **Architecture of WWW:**
  - **Client-Server Model:** Users access websites using a client (browser) that communicates with servers storing web pages.
  - **Web Pages and Links:** Web pages stored on servers may contain links to other pages within or outside the site, allowing navigation across documents.



- **Client (Browser) Components:**
  - **Controller:** Receives user input (keyboard/mouse) and manages access to web documents.
  - **Client Protocol:** Facilitates the communication with the server.
  - **Interpreters:** Render and display the web page on the screen.
- **Server Role:**
  - Servers store web pages and respond to client requests. Efficient servers use caching and multi-threading to handle multiple requests simultaneously. They rely on **Uniform Resource Locators (URLs)** for addressing pages.
- **Uniform Resource Locator (URL):**
  - A URL uniquely identifies a web page using four elements:
  - **Protocol:** Defines the client-server application (e.g., HTTP, FTP).
  - **Host:** Identifies the server (IP address or domain name).
  - **Port:** Defines the server's application port.
  - **Path:** Indicates the specific document on the server.



**Q** which of the following is not a client -server application? **(GATE-2010) (1 Marks)**

**a)** Internet chat

**b)** Web browsing

**c)** E-mail

**d)** Ping

# HTTP

**Hypertext Transfer Protocol (HTTP)** is a foundational protocol used for accessing information on the **World Wide Web (WWW)**. It defines how clients (browsers) and servers should interact to retrieve web pages and other resources. HTTP is a connection-oriented and reliable protocol, operating over **TCP on port 80**.

## **HTTP Characteristics:**

### **1. Connection Management:**

1. HTTP establishes a single TCP connection between a client and a server, unlike FTP which uses separate control and data connections.
2. It is similar to **SMTP** in terms of message structure but serves immediate delivery of web resources instead of storing and forwarding messages.

### **2. Proxy Server:**

1. HTTP supports **proxy servers**, which cache responses to reduce load on original servers, decrease network traffic, and improve latency. Proxy servers serve repeated client requests efficiently.

### **3. Connections:**

1. **Nonpersistent Connections:** For every request-response cycle, a separate TCP connection is created. For instance, if a webpage has multiple elements (like images), a new connection must be opened for each element, which introduces overhead.
2. **Persistent Connections:** HTTP 1.1 introduced persistent connections by default, where the server keeps the connection open for further requests, reducing the time and resources needed for connection establishment.

## HTTP Security:

- HTTP is inherently **insecure**, lacking built-in mechanisms for confidentiality, integrity, or authentication. To enhance security, **HTTP over SSL (Secure Socket Layer)** is used, commonly known as **HTTPS**. HTTPS ensures:
  - **Confidentiality**: Encrypting data to protect it from eavesdropping.
  - **Authentication**: Verifying the identities of the server and client.
  - **Data Integrity**: Protecting data from being altered during transmission.
- **Advantages of Persistent Connections**:
- Saves time and resources by avoiding repeated connection setups.
- Reduces the need for multiple sets of buffers and variables at each site.
- Lowers round-trip delays for connection management.

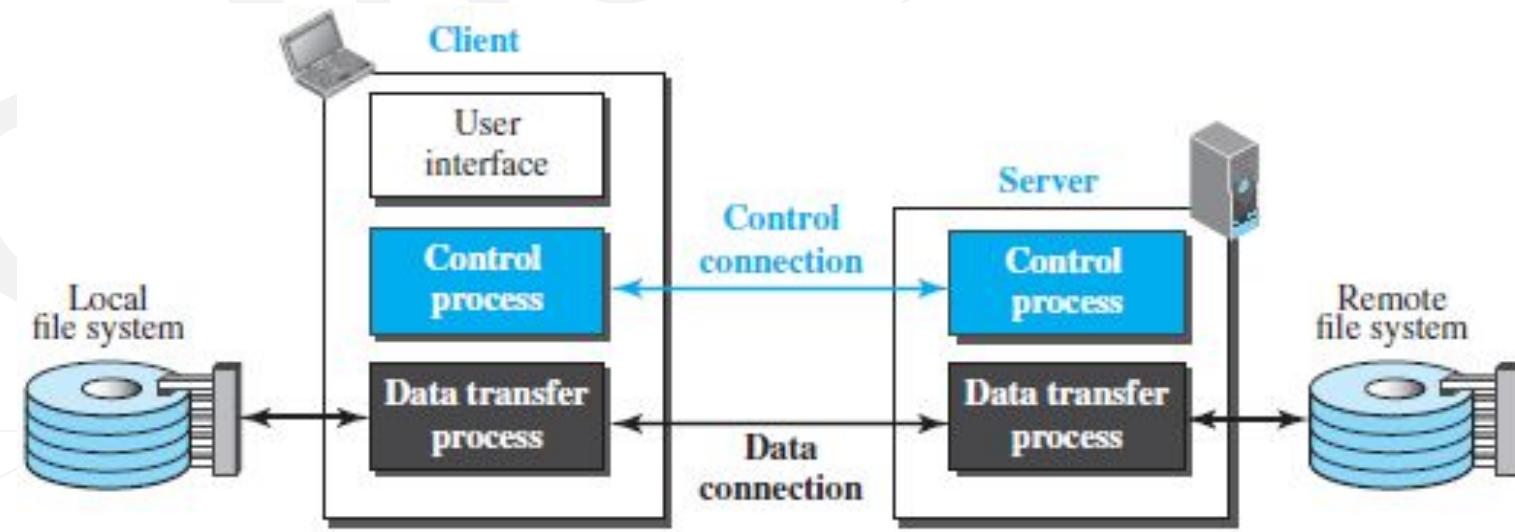


**Q** Identify the correct sequence in which the following packets are transmitted on the network by a host when a browser requests a webpage from a remote server, assuming that the host has just been restarted. **(GATE-2016) (2 Marks)**

- (A)** HTTP GET request, DNS query, TCP SYN
- (B)** DNS query, HTTP GET request, TCP SYN
- (C)** DNS query, TCP SYN, HTTP GET request
- (D)** TCP SYN, DNS query, HTTP GET request

## FILE TRANSFER

- File transfer is a fundamental task in networking and the internet. One of the primary protocols for file exchange is the **File Transfer Protocol (FTP)**. It enables users to upload and download files efficiently over the internet.
- **Key Features of FTP:**
- **Standard Protocol in TCP/IP:** FTP is the standard mechanism provided by TCP/IP for transferring files between hosts.
- **Two Connections:** FTP differs from other protocols by establishing two separate connections:
  - **Control Connection (Port 21):** Used to transmit commands and responses, following simple communication rules.
  - **Data Connection (Port 20):** Handles the actual file transfer and uses more complex rules due to the various types of data transferred.
- **Handling Different Systems:** FTP overcomes issues arising from:
  - Different file naming conventions.
  - Different representations of text and data.
  - Different directory structures.
- **Efficiency in File Transfer:** By separating control and data connections, FTP achieves greater efficiency in handling commands and data transfer.



- **Security Concerns with FTP:**

- **Plaintext Transmission:** FTP was designed without modern security in mind. Both passwords and data are transmitted in plaintext, making them vulnerable to interception.
- **Solution:** Adding a Secure Socket Layer (SSL) between the FTP and TCP layers creates **SSL-FTP**, enhancing security by encrypting both commands and data.

**Q** In one of the pairs of protocols given below, both the protocols can use multiple TCP connections between the same client and the server. Which one is that?

**(GATE-2015) (1 Marks)**

**(A) HTTP, FTP**

**(B) HTTP, TELNET**

**(C) FTP, SMTP**

**(D) HTTP, SMTP**

**Q Match the following: (GATE-2007) (2 Marks)**

(P) SMTP	(1) Application layer
(Q) BGP	(2) Transport layer
(R) TCP	(3) Data link layer
(S) PPP	(4) Network layer
	(5) Physical layer

**(A) P – 2 Q – 1 R – 3 S – 5**

**(B) P – 1 Q – 4 R – 2 S – 3**

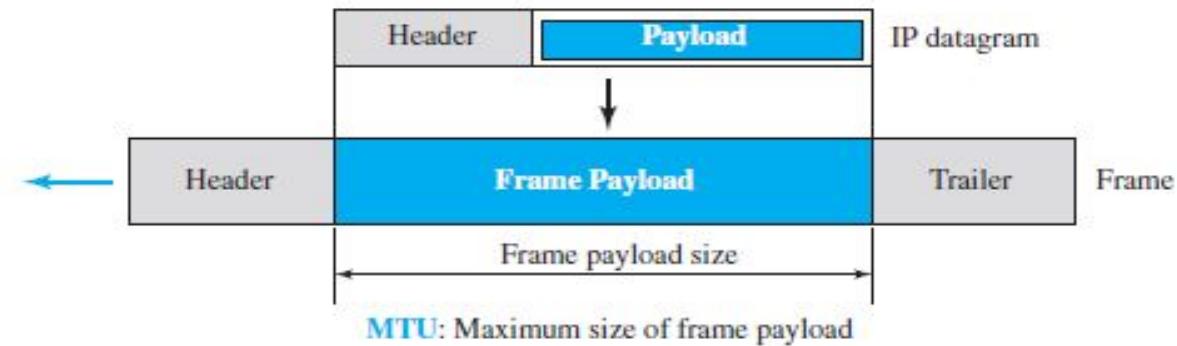
**(C) P – 1 Q – 4 R – 2 S – 5**

**(D) P – 2 Q – 4 R – 1 S – 3**

## Maximum Transfer Unit (MTU)

- Each link-layer protocol has its own frame format.
- One of the features of each format is the maximum size of the payload that can be encapsulated.
- In other words, when a datagram is encapsulated in a frame, the total size of the datagram must be less than this maximum size.

Protocol	MTU
Hyperchannel	65,535
Token Ring (16 Mbps)	17,914
Token Ring (4 Mbps)	4,464
FDDI	4,352
Ethernet	1,500
X.25	576
PPP	296



**Q** The protocol data unit (PDU) for the application layer in the Internet stack is  
**(GATE-2012) (1 Marks)**

**(A)** Segment

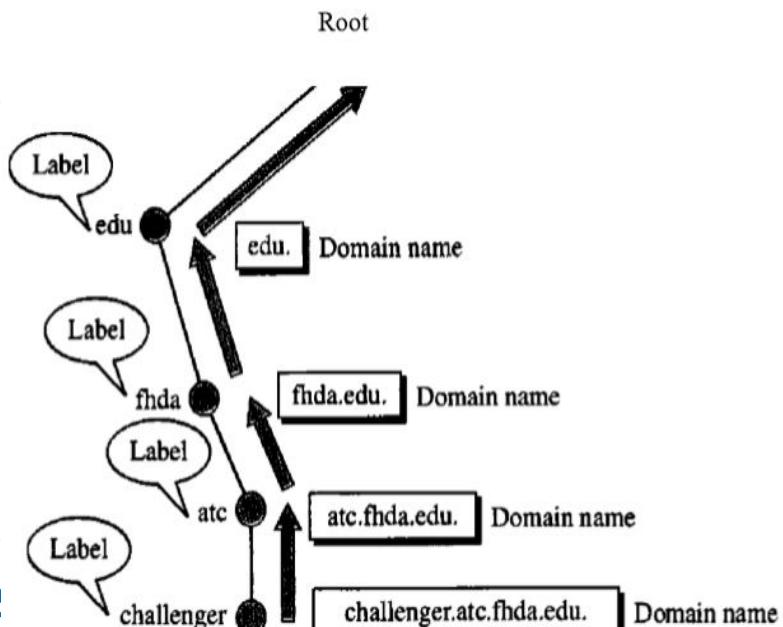
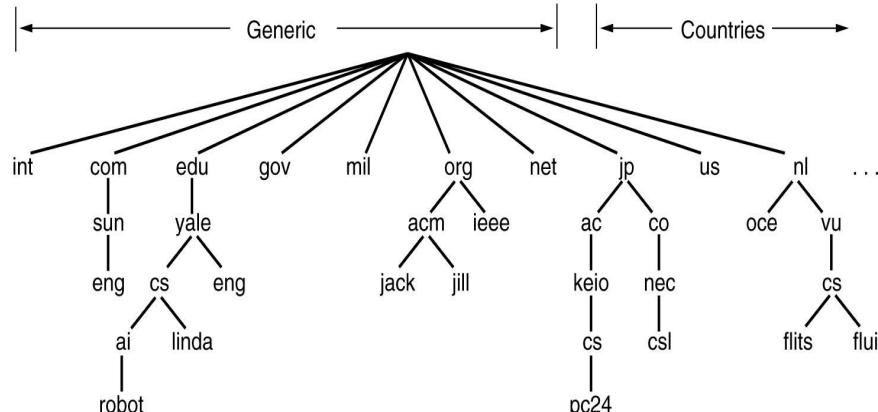
**(B)** Datagram

**(C)** Message

**(D)** Frame

# NAME SPACE

- A **name space** is used to uniquely map addresses to names. It can be organized in two ways: **flat** or **hierarchical**.
- **Flat Name Space:**
  - A flat name space consists of a sequence of characters without any structure.
  - It is not suitable for large systems like the Internet because it requires central control to avoid duplication and ambiguity.
- **Hierarchical Name Space:**
  - A hierarchical name space divides each name into several parts (labels). For example, one part can indicate the type of organization, another the name of the organization, and subsequent parts can indicate departments or specific resources within that organization.
  - The authority to manage name spaces is decentralized. A central authority only controls the top part of the name (e.g., .com, .edu), while individual organizations control the rest.
  - domain name ends with a dot, indicating the root.

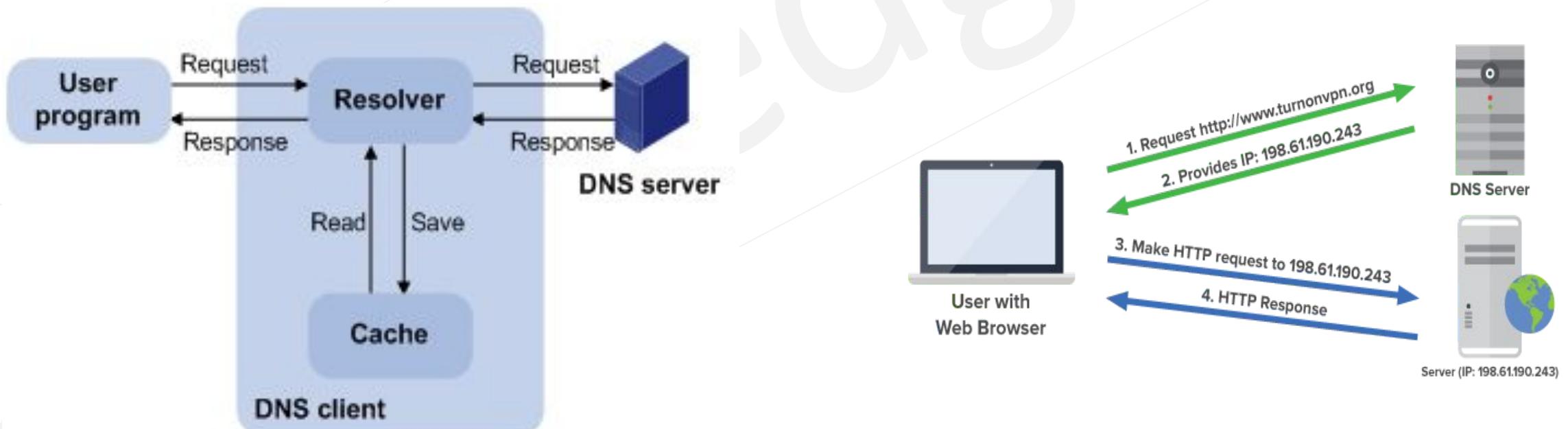


- **Generic Domains:** Include .com (commercial), .edu (educational), .mil (military), .org (non-profit), and .net.
- **Country Domains:** Include country-specific domains like .in (India), .uk (United Kingdom), .us (United States).
- **Inverse Domain:** Used for mapping an IP address to a domain name, which helps in reverse lookups.
- **Domain Name Space Structure:**
  - The domain name space is arranged as an **inverted tree** structure, with the root at the top. The tree is limited to 128 levels, from level 0 (root) to level 127.
  - Each node in the tree has a **label** (string with a maximum of 63 characters). The root label is represented by a null string (empty).
- **Domain Names:**
  - A **domain name** is formed by a sequence of labels separated by dots (.). Domain names are read from the specific node (leaf) up to the root. Each domain name ends with a dot, indicating the root.

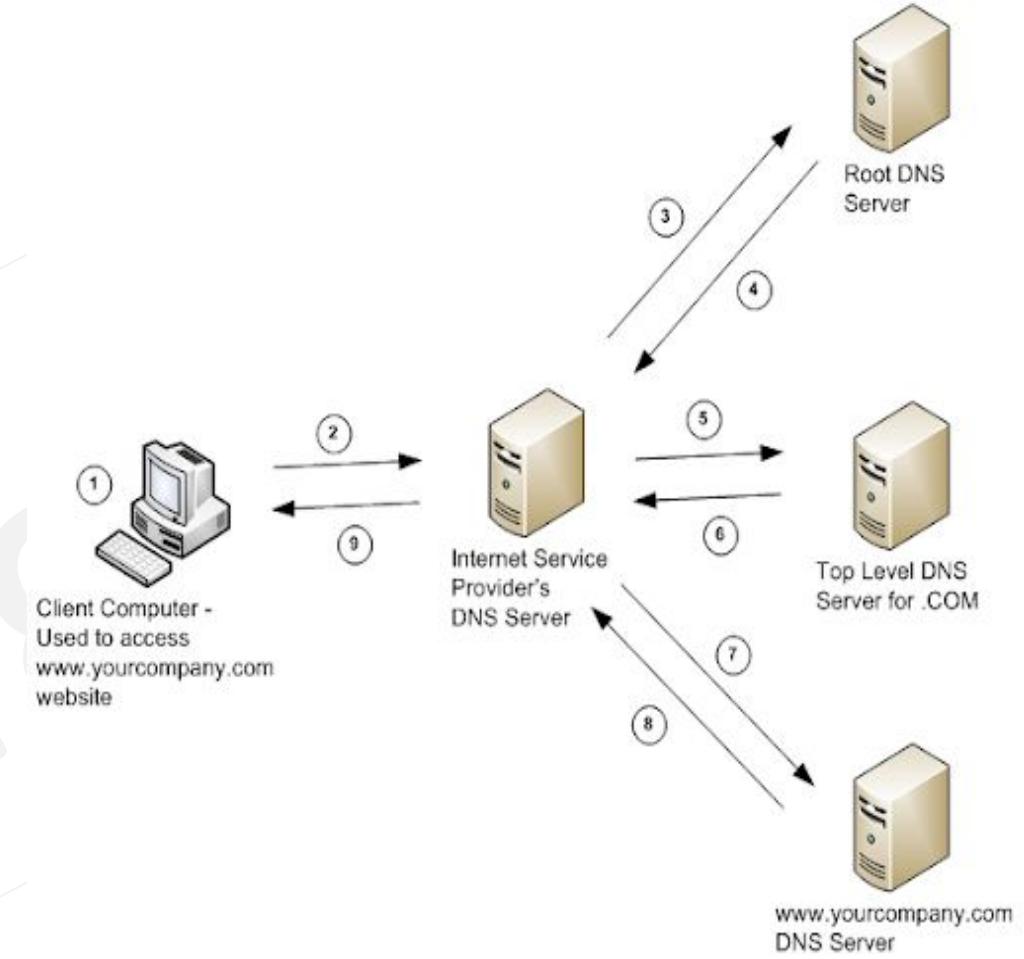
<i>Label</i>	<i>Description</i>
aero	Airlines and aerospace companies
biz	Businesses or firms (similar to "com")
com	Commercial organizations
coop	Cooperative business organizations
edu	Educational institutions
gov	Government institutions
info	Information service providers
int	International organizations
mil	Military groups
museum	Museums and other nonprofit organizations
name	Personal names (individuals)
net	Network support centers
org	Nonprofit organizations
pro	Professional individual organizations

# DNS

- **Domain Name System (DNS)** is a crucial mechanism that translates human-readable domain names into machine-readable IP addresses, simplifying navigation on the internet. Since remembering numerical IP addresses is difficult and they frequently change, DNS serves as a reliable method to map names to IPs.
- **How DNS Works:**
- **Name to IP Mapping:** A user inputs a domain name (like example.com) into a browser. The DNS client sends a query to a DNS server to resolve this name into an IP address.
- **Distributed Database:** DNS information is distributed and divided among multiple servers to manage the vast amount of domain data effectively and to provide fast and reliable IP lookups.



- **Hierarchy of Name Servers:**
- **Root Name Servers:** The starting point for DNS queries. If a local DNS server cannot resolve a query, it contacts a root server, which directs the query to the appropriate top-level server.
- **Top-Level Domain (TLD) Servers:** These handle TLDs such as .com, .org, and country-specific domains like .uk or .in. They store information about authoritative name servers responsible for second-level domains.
- **Authoritative Name Servers:** These servers store the final mapping of domain names to IP addresses. For example, if a user queries cse.dtu.in, the authoritative server holds the IP address for that specific domain.

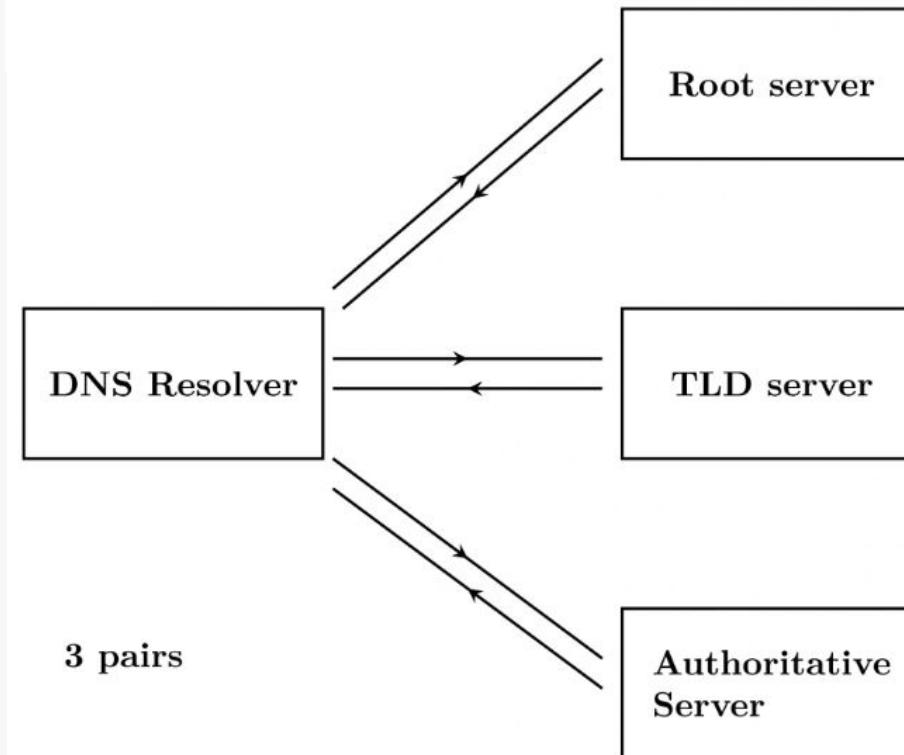


**Q** Consider the resolution of the domain name www.gate.org.in by a DNS resolver. Assume that no resource records are cached anywhere across the DNS servers and that iterative query mechanism is used in the resolution. The number of DNS query-response pairs involved in completely resolving the domain name is \_\_\_\_\_. **(GATE 2022) (1 MARKS)**

**Q** Consider the resolution of the domain name `www.gate.org.in` by a DNS resolver. Assume that no resource records are cached anywhere across the DNS servers and that iterative query mechanism is used in the resolution. The number of DNS query-response pairs involved in completely resolving the domain name is \_\_\_\_\_.

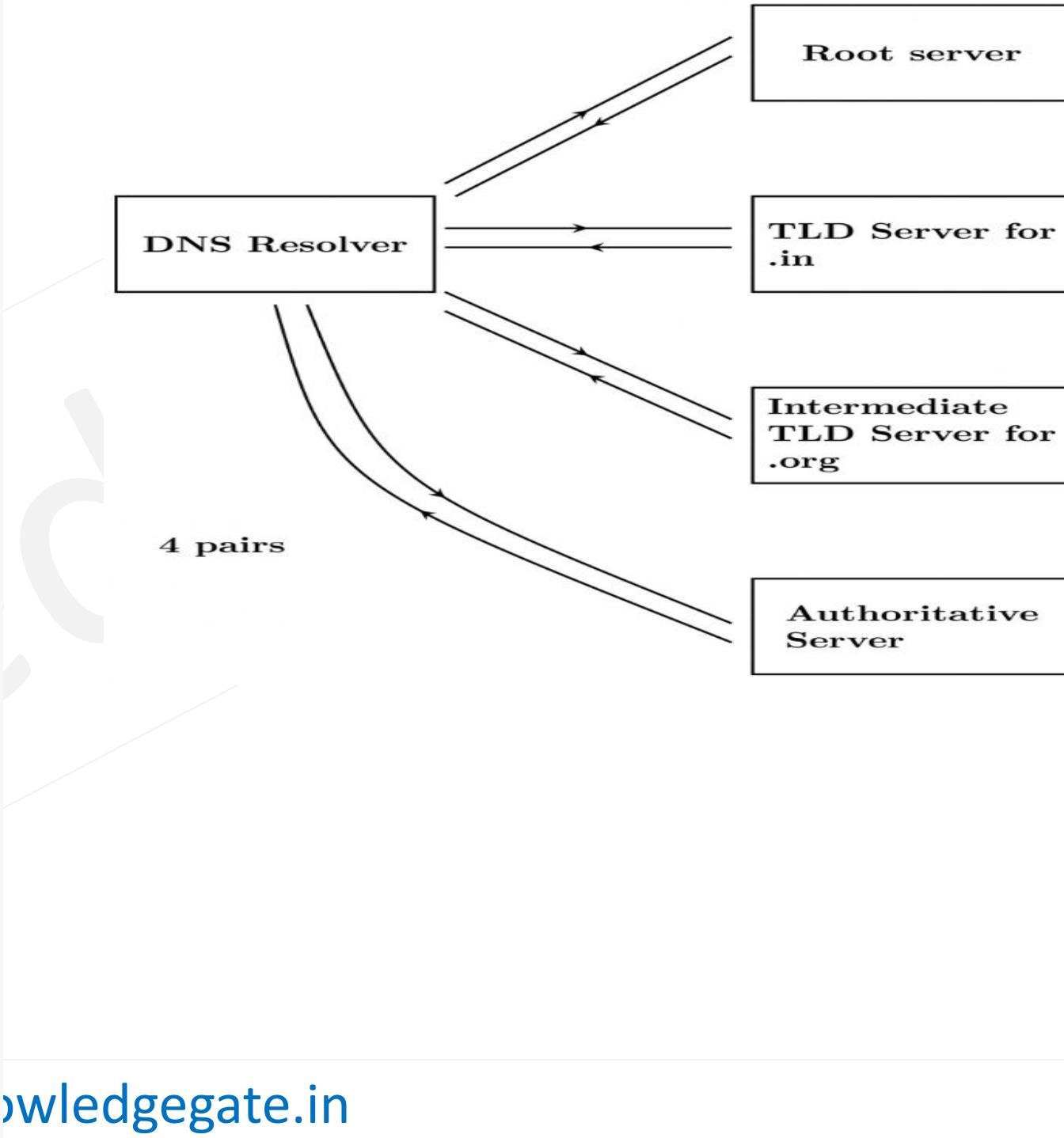
**(GATE 2022) (1 MARKS)**

- Root servers act as the entry point for DNS queries and guide them to the appropriate TLD servers. They are at the highest level of the DNS hierarchy but don't store domain-specific information.
- TLD servers handle the second step in DNS queries and are responsible for specific top-level domains like **.COM**, **.org**, etc. They direct queries to the authoritative servers where the actual IP mapping is stored.
- Authoritative servers are the final destination for DNS queries and hold the actual domain-to-IP mapping. They maintain zone files and are the definitive source for a specific domain or set of domains. They provide the final IP address to the client.



- Root servers act as the entry point for DNS queries and guide them to the appropriate TLD servers. They are at the highest level of the DNS hierarchy but don't store domain-specific information.

- TLD servers handle the second step in DNS queries and are responsible for specific top-level domains like **.com**, **.org**, etc. They direct queries to the authoritative servers where the actual IP mapping is stored.
- Authoritative servers are the final destination for DNS queries and hold the actual domain-to-IP mapping. They maintain zone files and are the definitive source for a specific domain or set of domains. They provide the final IP address to the client.



**Q Which one of the following uses UDP as the transport protocol? (GATE-2007) (1 Marks)**

(A) HTTP

(B) Telnet

(C) DNS

(D) SMTP

**Q** The transport layer protocols used for real time multimedia, file transfer, DNS and email, respectively are: **(GATE-2013) (1 Marks)**

(A) TCP, UDP, UDP and TCP

(B) UDP, TCP, TCP and UDP

(C) UDP, TCP, UDP and TCP

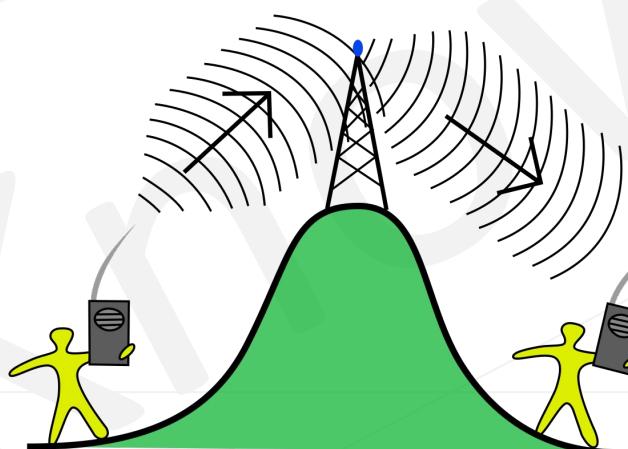
(D) TCP, UDP, TCP and UDP

**Q.** A user starts browsing a web page hosted at a remote server. The browser opens a single TCP connection to fetch the entire webpage from the server. The webpage consists of a top-level index page with multiple embedded image objects. Assume that all caches (e.g., DNS cache, browser cache) are all initially empty. The following packets leave the user's computer in some order. **(Gate 2024,CS) (1 Marks) (MCQ)**

- (i) HTTP GET request for the index page
  - (ii) DNS request to resolve the web server's name to its IP address
  - (iii) HTTP GET request for an image object
  - (iv) TCP SYN to open a connection to the web server
- 
- (a) (iv),(ii),(iii),(i)
  - (b) (ii),(iv),(iii),(i)
  - (c) (ii),(iv),(i),(iii)
  - (d) (iv),(ii),(i),(iii)

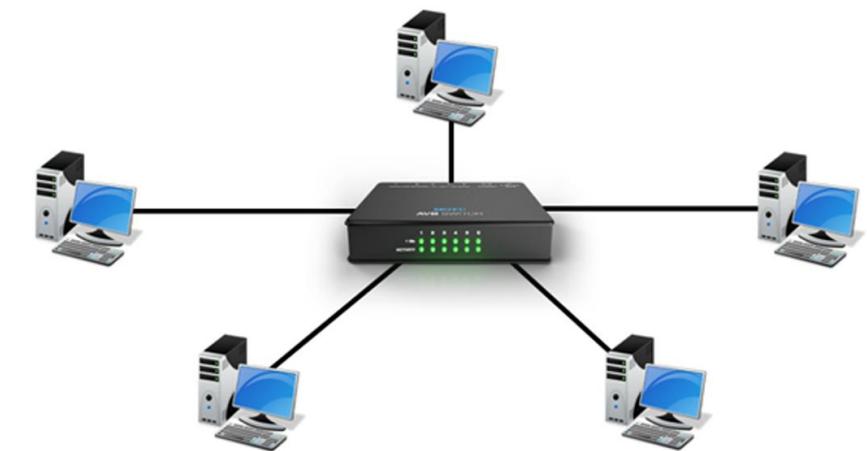
# Repeater

- When an information-bearing signal passes through a communication channel, it is progressively degraded due to loss of power. For example, when a telephone call passes through a wire telephone line, some of the power in the electric current which represents the audio signal is dissipated as heat in the resistance of the copper wire.
- The longer the wire is, the more power is lost, and the smaller the amplitude of the signal at the far end. So with a long enough wire the call will not be audible at the other end. Similarly, the farther from a radio station a receiver is, the weaker the radio signal, and the poorer the reception. A repeater is an electronic device in a communication channel that increases the power of a signal and retransmits it, allowing it to travel further. Since it amplifies the signal, it requires a source of electric power.
- Repeaters are used to extend transmissions so that the signal can cover longer distances or be received on the other side of an obstruction.
- In computer networking, because repeaters work with the actual physical signal, and do not attempt to interpret the data being transmitted, they operate on the physical layer, the first layer of the OSI model.
- Conclusion
  - Work in physical Layer
  - Collisions are possible
  - Range of Lan is increased



# Hub

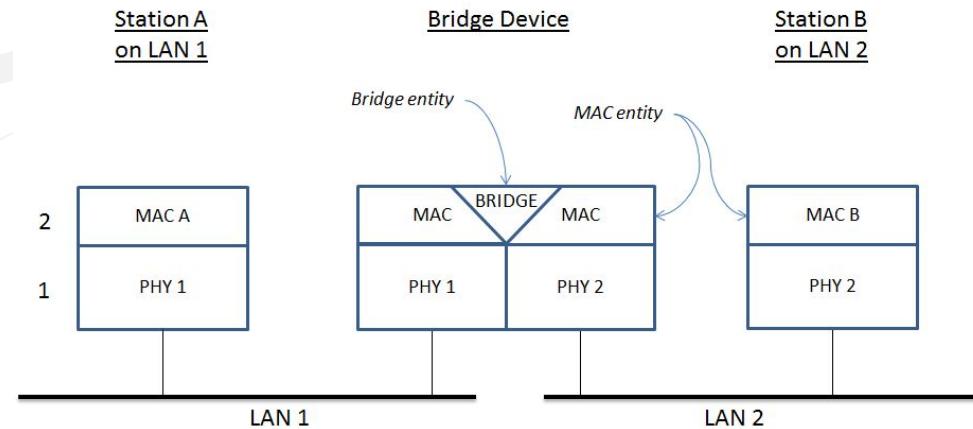
- Hubs are multiport repeater. An **Ethernet hub**, **active hub**, **network hub**, **repeater hub**, **multiport repeater**, or simply **hub** is a network hardware device for connecting multiple Ethernet devices together and making them act as a single network segment.
- It has multiple input/output (I/O) ports, in which a signal introduced at the input of any port appears at the output of every port except the original incoming. A hub works at the physical layer (layer 1) of the OSI model.
- Hubs are now largely obsolete, having been replaced by network switches except in very old installations.
- Conclusion
  - Work in physical Layer
  - Collisions are possible
  - Traffic is very high
  - Cost is less



# Bridge

- Bridge is used to connect two different lan
- A **network bridge** is a computer networking device that creates a single, aggregate network from multiple communication networks or network segments. Bridging connects two separate networks as if they were a single network.
- In the OSI model, bridging is performed in the data link layer (layer 2).
- Filtering, Forwarding, Flooding, and Store, no collision inside a bridge
- Conclusion
  - Work in physical Layer, Data Link Layer

A bridge connecting two LAN segments



# Switch

- Switch is Connecting device which generally have many port, in compare to bridge which generally have two interface
- A **network switch** (also called **switching hub**, **bridging hub**, and, by the IEEE, **MAC bridge**) is networking hardware that connects devices on a computer network by using packet switching to receive and forward data to the destination device.
- A network switch is a multiport network bridge that uses MAC addresses to forward data at the data link layer (layer 2) of the OSI model.
- Switches for Ethernet are the most common form of network switch. The first Ethernet switch was introduced by Kalpana in 1990.
- Unlike repeater hubs, which broadcast the same data out of each port and let the devices pick out the data addressed to them, a network switch learns the identities of connected devices and then only forwards data to the port connected to the device to which it is addressed.
- Conclusion
  - Work in physical Layer, Data Link Layer
  - Collisions are not possible
  - Traffic is very less
  - Cost is high



# Router

- A **router** is a networking device that forwards data packets between computer networks. Routers perform the traffic directing functions on the Internet. Data sent through the internet, such as a web page or email, is in the form of data packets. A packet is typically forwarded from one router to another router through the networks that constitute an internetwork (e.g. the Internet) until it reaches its destination node.
- A router is connected to two or more data lines from different IP networks. When a data packet comes in on one of the lines, the router reads the network address information in the packet header to determine the ultimate destination. Then, using information in its routing table or routing policy, it directs the packet to the next network on its journey.
- The most familiar type of IP routers are home and small office routers that simply forward IP packets between the home computers and the Internet. More sophisticated routers, such as enterprise routers, connect large business or ISP networks up to the powerful core routers that forward data at high speed along the optical fiber lines of the Internet backbone.
- Conclusion
  - Work in physical Layer, Data Link Layer, Network Layer
  - Filter, Forward, Flooding, Routing, no collision

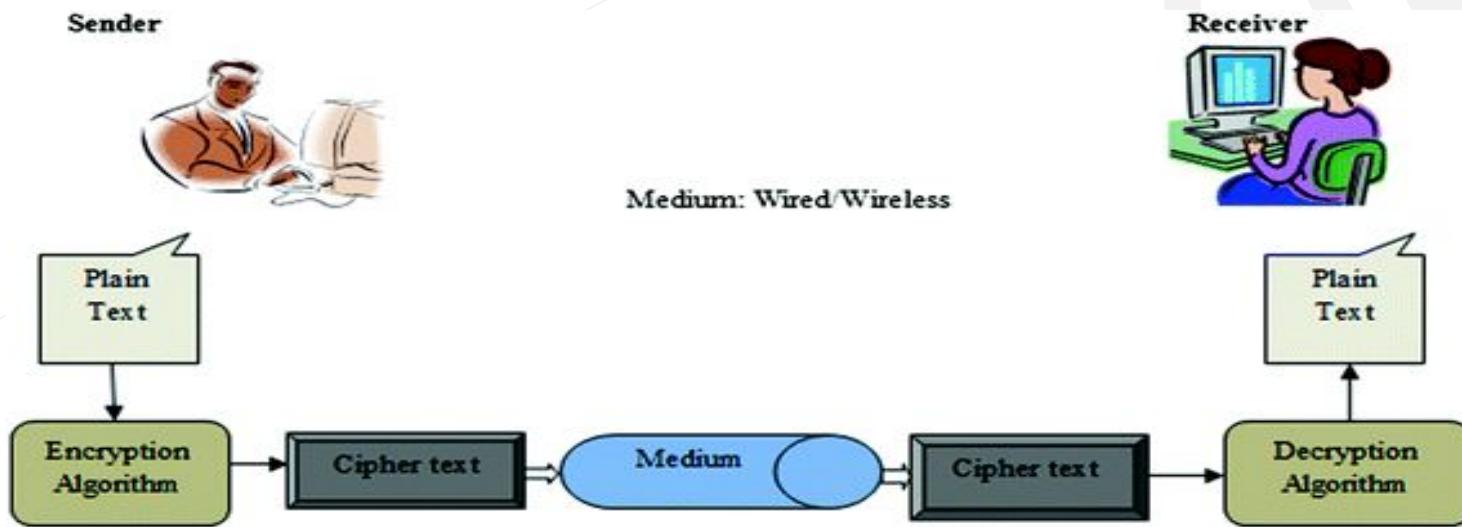


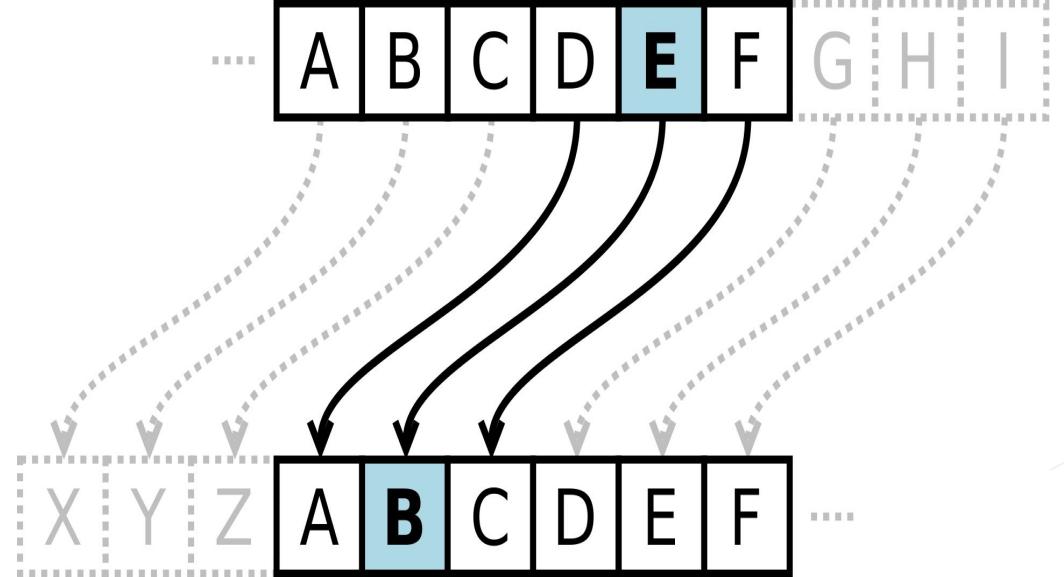
# Gateway

- A **gateway** is a piece of networking hardware or software used in telecommunications networks that allows data to flow from one discrete network to another.
- Gateways are distinct from routers or switches in that they communicate using more than one protocol to connect multiple networks and can operate at any of the seven layers of the open systems interconnection model (OSI).
- The term gateway can also loosely refer to a computer or computer program configured to perform the tasks of a gateway, such as a default gateway or router.
- Conclusion
  - Protocol Converter
  - Proxy
  - Network Address Translation
  - Firewall
  - Deep Packet Inspection

# Cryptography

- **Cryptography**, (from Ancient Greek: *kryptós* "hidden, secret"; and *graphein*, "to write"), is the practice and study of techniques for secure communication in the presence of third parties called adversaries.
- More generally, cryptography is about constructing and analysing protocols that prevent third parties or the public from reading private messages; various aspects in information security such as data confidentiality, data integrity, authentication, and non-repudiation are central to modern cryptography.
- Applications of cryptography include electronic commerce, chip-based payment cards, digital currencies, computer passwords, and military communications.



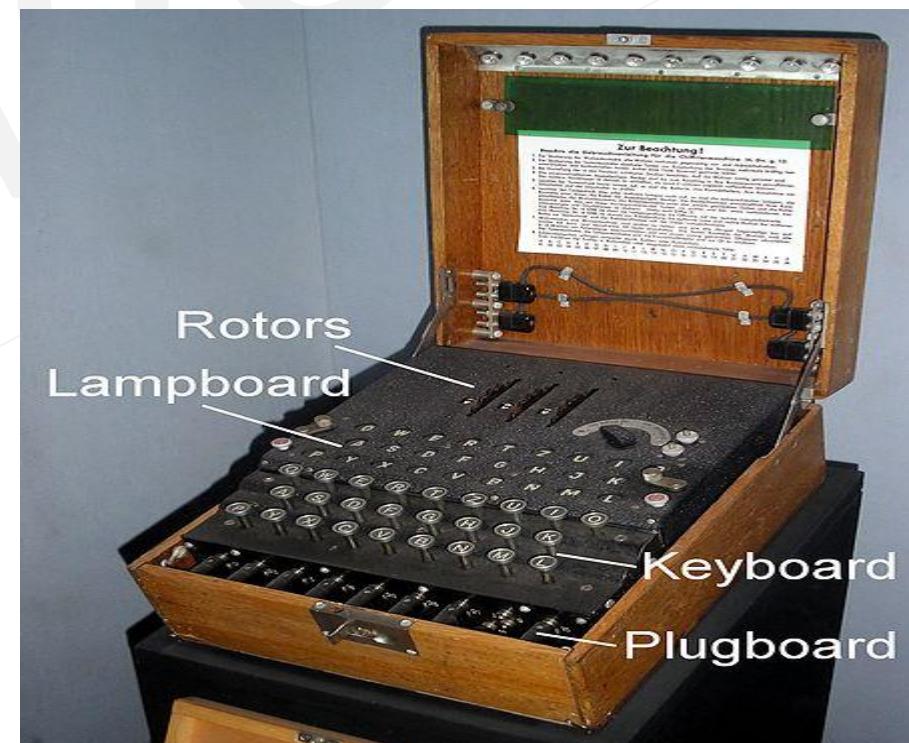
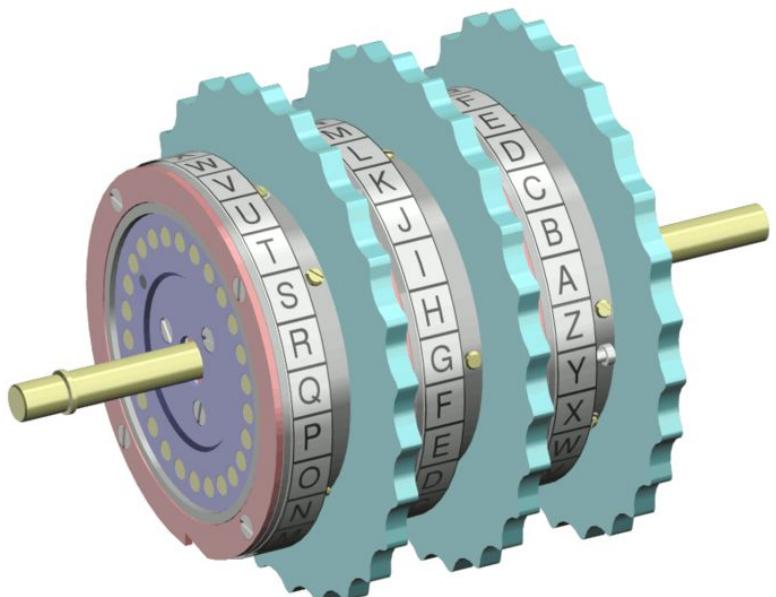


Alphabet shift ciphers are believed to have been used by Julius Caesar over 2,000 years ago.



16th-century book shaped French cipher machine, with arms of Henri II of France

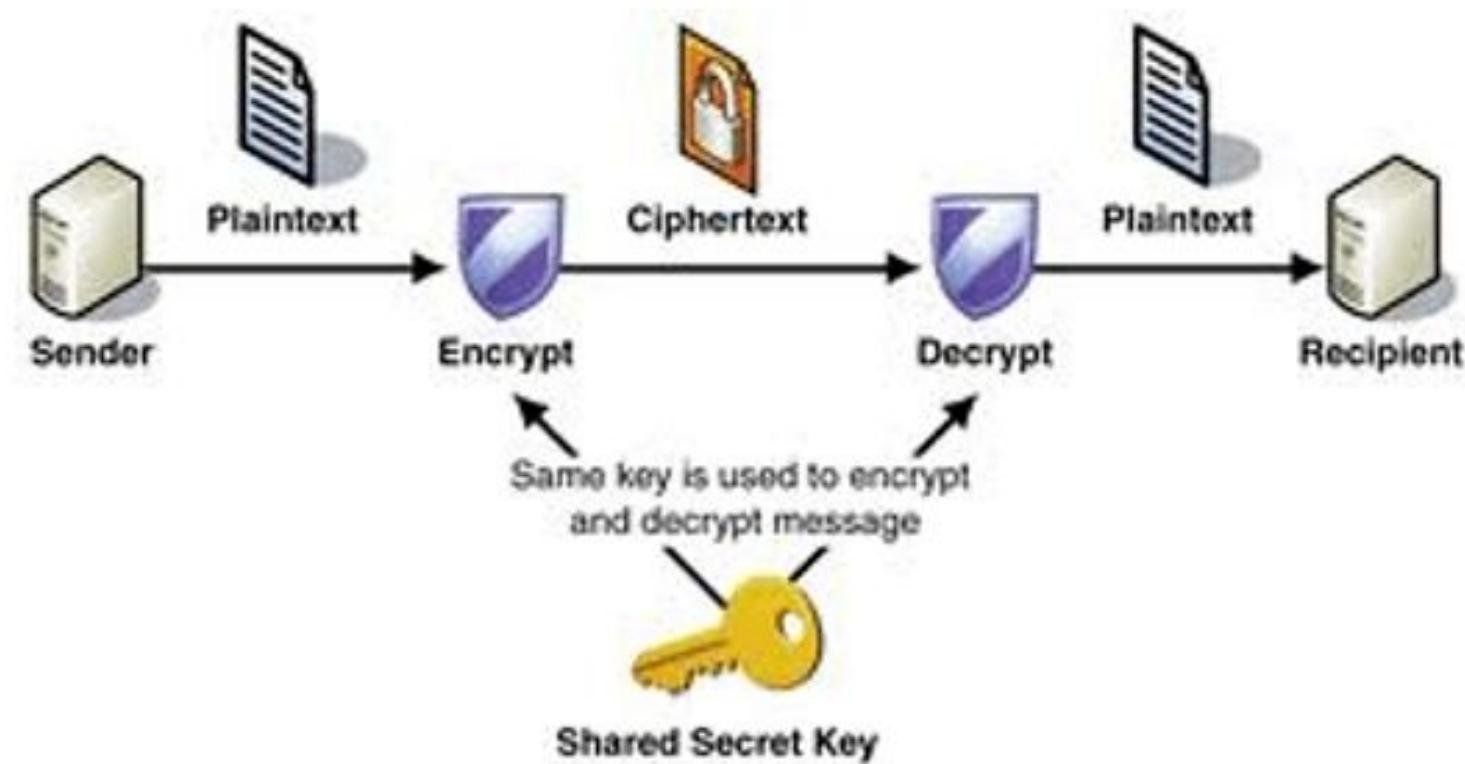
- Cryptography prior to the modern age was effectively synonymous with *encryption*, the conversion of information from a readable state to apparent nonsense. The originator of an encrypted message shares the decoding technique only with intended recipients to preclude access from adversaries.
- The cryptography literature often uses the names Alice ("A") for the sender, Bob ("B") for the intended recipient, and Eve ("eavesdropper") for the adversary.
- Since the development of rotor cipher machines in World War I and the advent of computers in World War II, the methods used to carry out cryptology have become increasingly complex and its application more widespread.



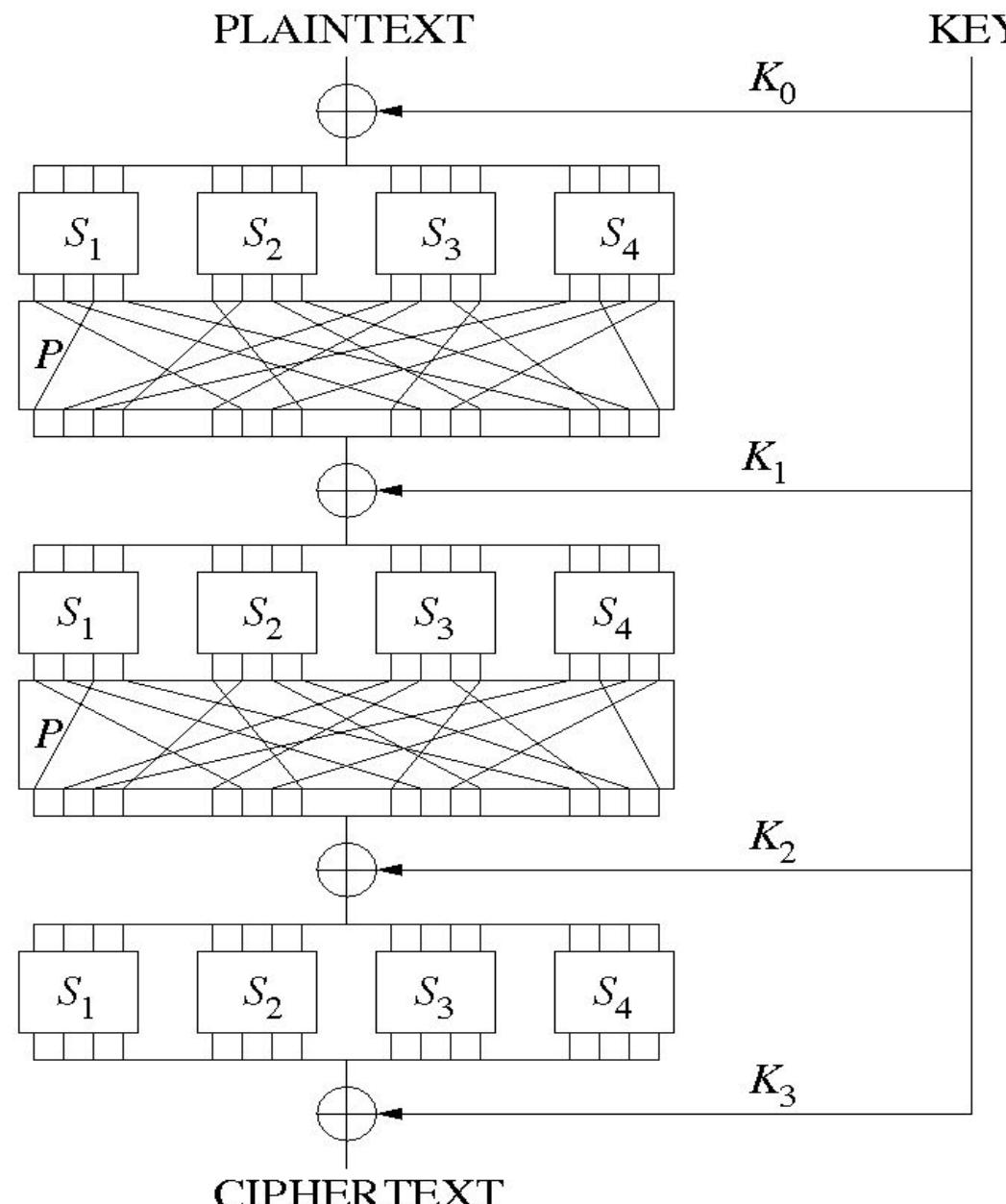
- Modern cryptography is heavily based on mathematical theory and computer science practice; cryptographic algorithms are designed around computational hardness assumptions, making such algorithms hard to break in practice by any adversary.
- It is theoretically possible to break such a system, but it is infeasible to do so by any known practical means. The growth of cryptographic technology has raised a number of legal issues in the information age.
- In some jurisdictions where the use of cryptography is legal, laws permit investigators to compel the disclosure of encryption keys for documents relevant to an investigation. Cryptography also plays a major role in digital rights management and copyright infringement of digital media.

## Symmetric key

- Symmetric-key cryptography refers to encryption methods in which both the sender and receiver share the same key (or, less commonly, in which their keys are different, but related in an easily computable way). This was the only kind of encryption publicly known until June 1976.



- The Data Encryption Standard (DES) and the Advanced Encryption Standard (AES) are block cipher designs that have been designated cryptography standards by the US government (though DES's designation was finally withdrawn after the AES was adopted).
- Despite its deprecation as an official standard, DES (especially its still-approved and much more secure triple-DES variant) remains quite popular; it is used across a wide range of applications, from ATM encryption to e-mail privacy and secure remote access.



**Q Consider the following two statements: (GATE-2007) (1 Marks)**

- i. A hash function (these are often used for computing digital signatures) is an injective function.
- ii. encryption technique such as DES performs a permutation on the elements of its input alphabet.

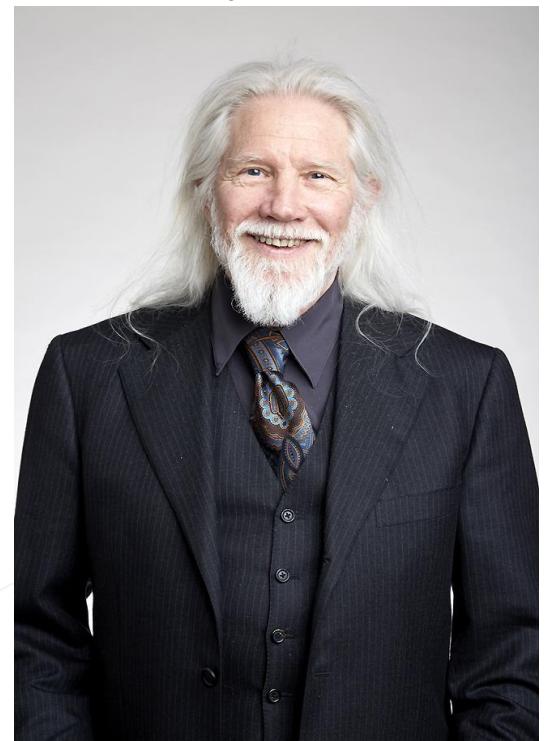
Which one of the following options is valid for the above two statements?

- (A) Both are false
- (B) Statement (i) is true and the other is false
- (C) Statement (ii) is true and the other is false
- (D) Both are true

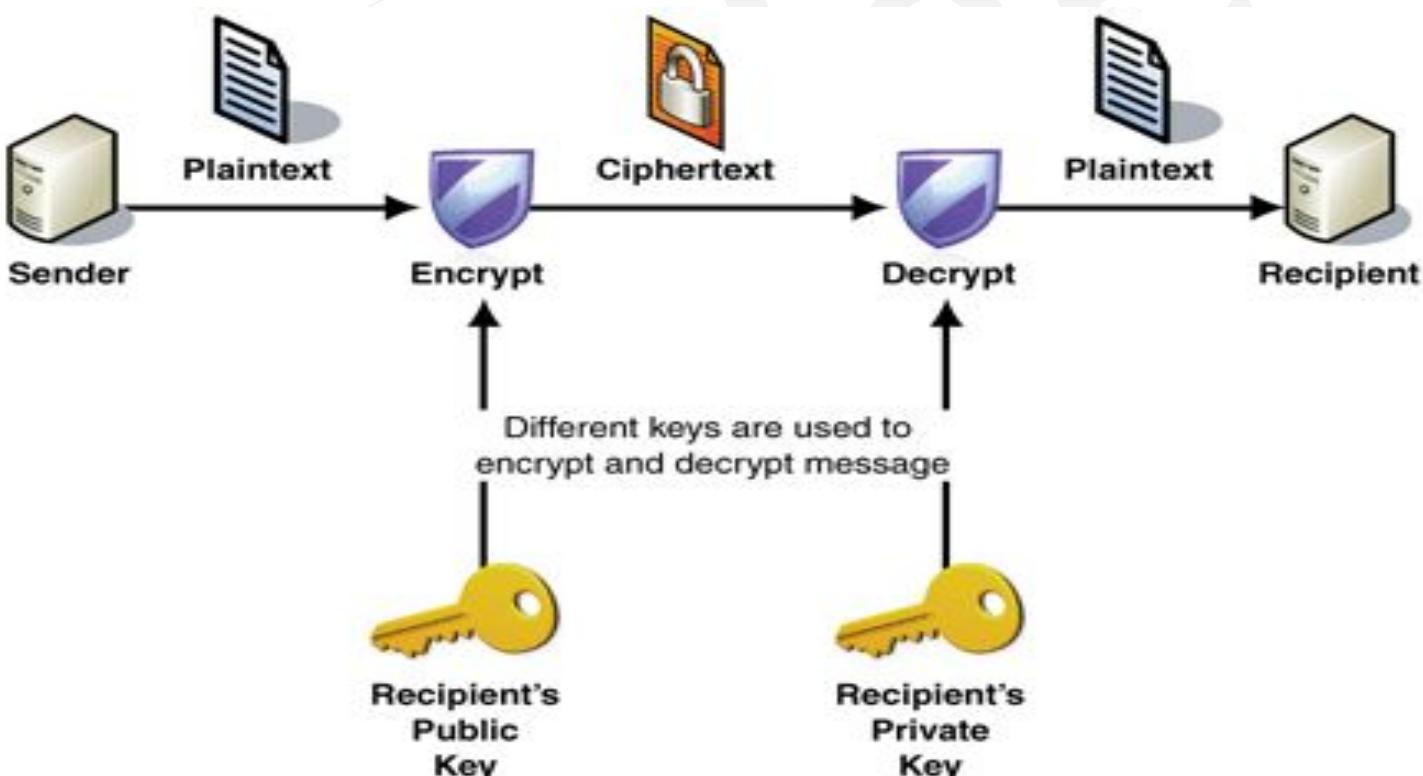
## **Asymmetric key (Public-key cryptography)**

- Symmetric-key cryptosystems use the same key for encryption and decryption of a message, although a message or group of messages can have a different key than others.
- A significant disadvantage of symmetric ciphers is the key management necessary to use them securely. Each distinct pair of communicating parties must, ideally, share a different key, and perhaps for each ciphertext exchanged as well.
- The number of keys required increases as the square of the number of network members, which very quickly requires complex key management schemes to keep them all consistent and secret.

- In a ground breaking 1976 paper, Whitfield Diffie and Martin Hellman proposed the notion of *public-key* (also, more generally, called *asymmetric key*) cryptography in which two different but mathematically related keys are used—a *public key* and a *private key*.
- A public key system is so constructed that calculation of one key (the 'private key') is computationally infeasible from the other (the 'public key'), even though they are necessarily related. Instead, both keys are generated secretly, as an interrelated pair.

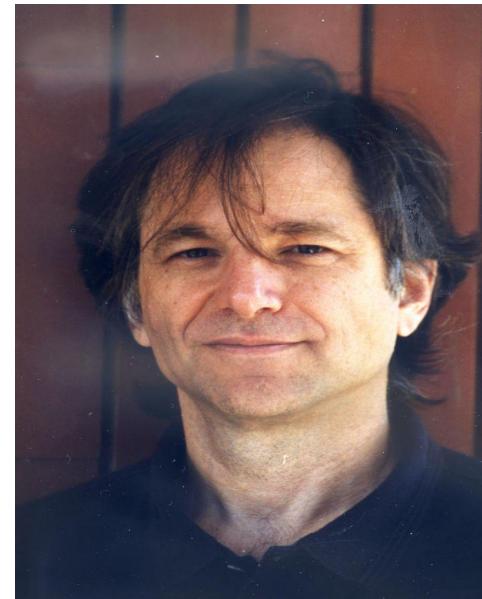


Whitfield Diffie



Martin Hellman

- In public-key cryptosystems, the public key may be freely distributed, while its paired private key must remain secret. In a public-key encryption system, the *public key* is used for encryption, while the *private* or *secret key* is used for decryption. While Diffie and Hellman could not find such a system, they showed that public-key cryptography was indeed possible by presenting the Diffie–Hellman key exchange protocol, a solution that is now widely used in secure communications to allow two parties to secretly agree on a shared encryption key.
- Diffie and Hellman's publication sparked widespread academic efforts in finding a practical public-key encryption system. This race was finally won in 1978 by Ronald Rivest, Adi Shamir, and Len Adleman, whose solution has since become known as the RSA algorithm.



- Reportedly, around 1970, James H. Ellis had conceived the principles of asymmetric key cryptography.



- In 1973, Clifford Cocks invented a solution that very similar in design rationale to RSA.

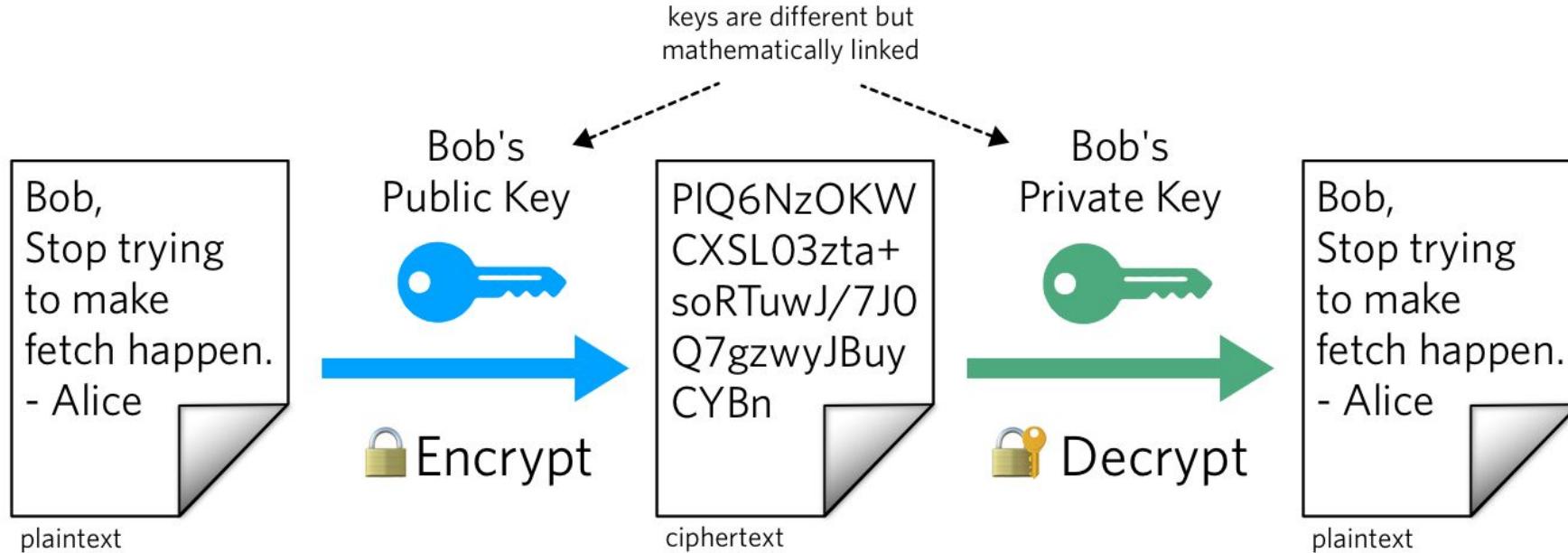


- And in 1974, Malcolm J. Williamson is claimed to have developed the Diffie–Hellman key exchange.



# Confidentiality

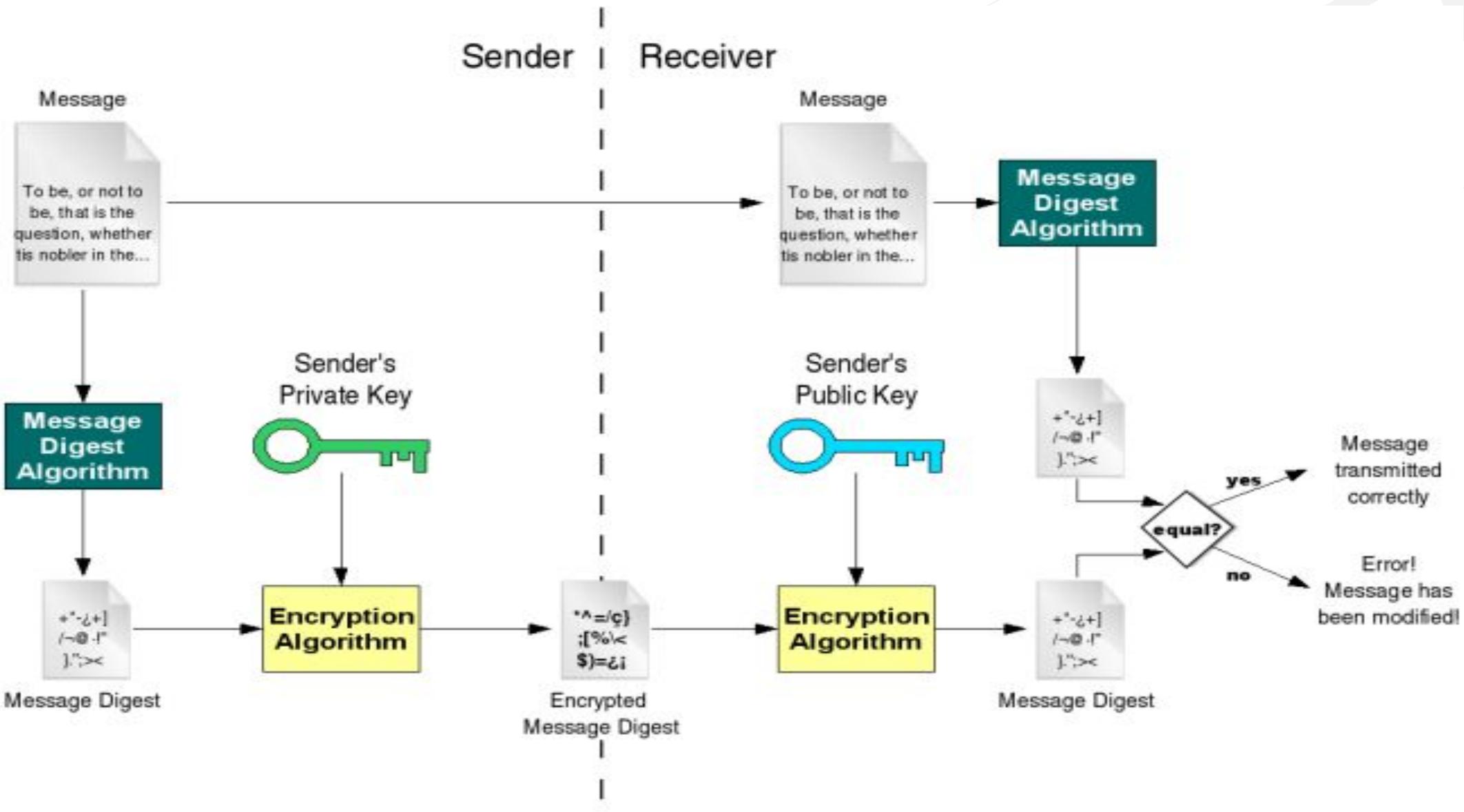
## Public Key Cryptography



**Q** Suppose that everyone in a group of  $N$  people wants to communicate secretly with the  $N-1$  other using symmetric key cryptographic system. The communication between any two persons should not be decodable by the others in the group. The number of keys required in the system as a whole to satisfy the confidentiality requirement is **(GATE-2015) (1 Marks)**

- (A)  $2N$
- (B)  $N(N - 1)$
- (C)  $N(N - 1)/2$
- (D)  $(N - 1)^2$

# Authentication



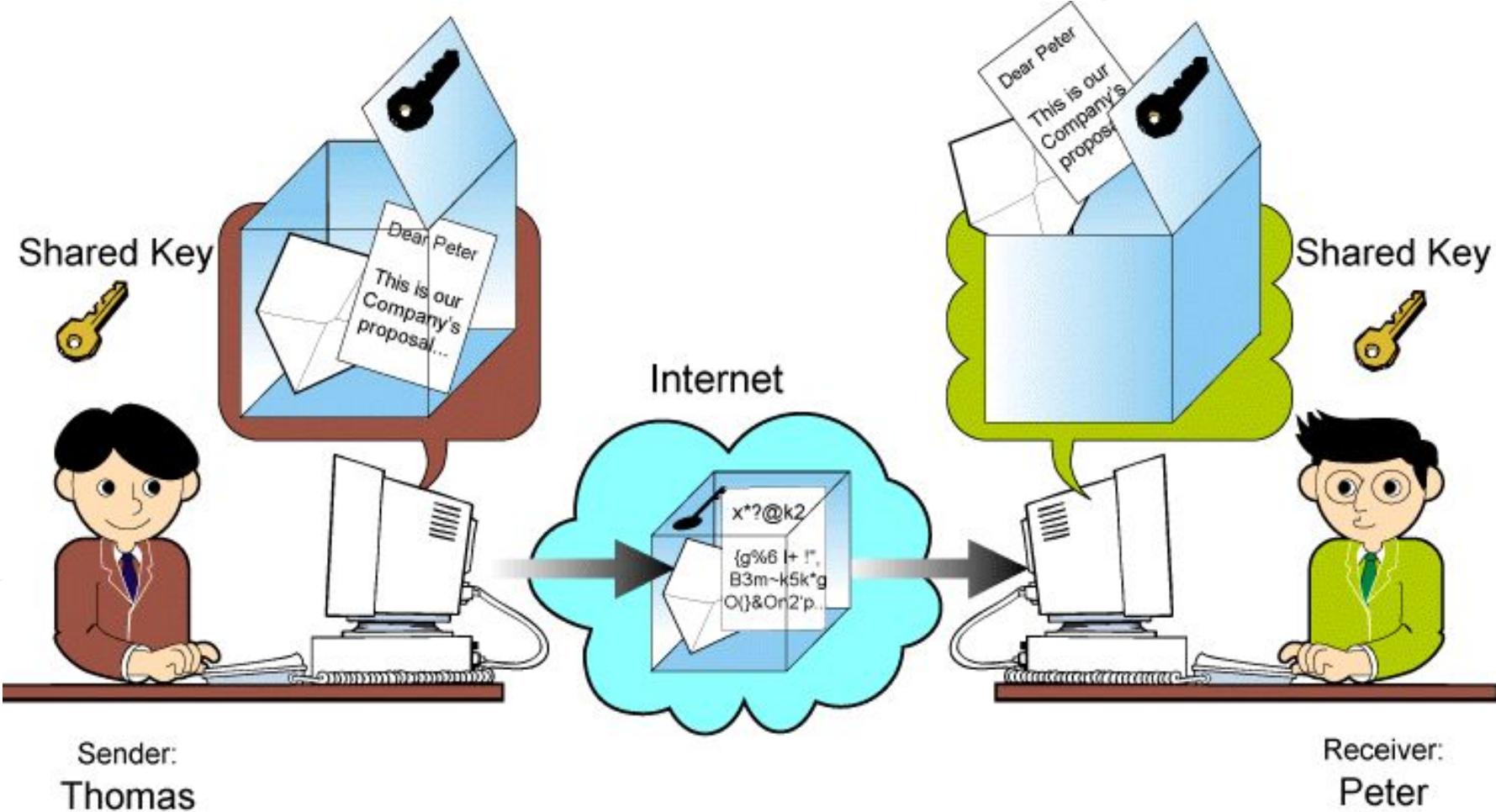
**Q** Anarkali digitally signs a message and sends it to Salim. Verification of the signature by Salim requires **(GATE-2016) (1 Marks)**

- (A) Anarkali's public key.
- (B) Salim's public key.
- (C) Salim's private key.
- (D) Anarkali's private key.

**Q** Consider that B wants to send a message  $m$  that is digitally signed to A. Let the pair of private and public keys for A and B be denoted  $K_x^-$  and  $K_x^+$  for  $x = A, B$ , respectively. Let  $K_x(m)$  represent the operation of encrypting  $m$  with a key  $K_x$  and  $H(m)$  represent the message digest. Which one of the following indicates the CORRECT way of sending the message  $m$  along with the digital signature to A? (GATE-2016) (2 Marks)

- (A)  $\{m, K_B^+(H(m))\}$
- (B)  $\{m, K_B^-(H(m))\}$
- (C)  $\{m, K_A^-(H(m))\}$
- (D)  $\{m, K_A^+(m)\}$

# Authentication Confidentiality



**Q** Using public key cryptography, X adds a digital signature  $\sigma$  to message M, encrypts  $\langle M, \sigma \rangle$ , and sends it to Y, where it is decrypted. Which one of the following sequences of keys is used for the operations? **(GATE-2013) (1 Marks)**

- (A)** Encryption: X's private key followed by Y's private key;  
Decryption: X's public key followed by Y's public key
- (B)** Encryption: X's private key followed by Y's public key;  
Decryption: X's public key followed by Y's private key
- (C)** Encryption: X's public key followed by Y's private key;  
Decryption: Y's public key followed by X's private key
- (D)** Encryption: X's private key followed by Y's public key;  
Decryption: Y's private key followed by X's public key

knowledgeGate

[www.knowledgegate.in](http://www.knowledgegate.in)

**Q** The total number of keys required for a set of  $n$  individuals to be able to communicate with each other using secret key and public key crypto-systems, respectively are: **(GATE-2008) (2 Marks)**

- (A)  $n(n-1)$  and  $2n$
- (B)  $2n$  and  $((n(n - 1))/2)$
- (C)  $((n(n - 1))/2)$  and  $2n$
- (D)  $((n(n - 1))/2)$  and  $n$

## RSA Algorithm

Bob chooses two large numbers,  $p$  and  $q$ , and calculates  $n = p \times q$  and  $\phi = (p - 1) \times (q - 1)$ . Bob then selects  $e$  and  $d$  such that  $(e \times d) \bmod \phi = 1$ . Bob advertises  $e$  and  $n$  to the community as the public key; Bob keeps  $d$  as the private key. Anyone, including Alice, can encrypt a message and send the ciphertext to Bob, using  $C = (P^e) \bmod n$ ; only Bob can decrypt the message, using  $P = (C^d) \bmod n$ . An intruder such as Eve cannot decrypt the message if  $p$  and  $q$  are very large numbers (she does not know  $d$ ).

For the sake of demonstration, let Bob choose 7 and 11 as  $p$  and  $q$  and calculate  $n = 7 \times 11 = 77$ . The value of  $\phi(n) = (7 - 1)(11 - 1)$ , or 60. If he chooses  $e$  to be 13, then  $d$  is 37. Note that  $e \times d \bmod 60 = 1$ . Now imagine that Alice wants to send the plaintext 5 to Bob. She uses the public exponent 13 to encrypt 5. This system is not safe because  $p$  and  $q$  are small.

Plaintext: 5

$$C = 5^{13} = 26 \text{ mod } 77$$

Ciphertext: 26

Ciphertext: 26

$$P = 26^{37} = 5 \text{ mod } 77$$

Plaintext: 5

Here is a more realistic example calculated using a computer program in Java. We choose a 512-bit  $p$  and  $q$ , calculate  $n$  and  $\phi(n)$ . We then choose  $e$  and calculate  $d$ . Finally, we show the results of encryption and decryption. The integer  $p$  is a 159-digit number.

$$p = \begin{array}{l} 9613034531358350457419158128061542790930984559499621582258315087964 \\ 7940455056470638491257160180347503120986666064924201918087806674210 \\ 96063354219926661209 \end{array}$$

The integer  $q$  is a 160-digit number.

$$q = \begin{array}{l} 1206019195723144691827679420445089600155592505463703393606179832173 \\ 1482148483764659215389453209175225273226830107120695604602513887145 \\ 524969000359660045617 \end{array}$$

The modulus  $n = p \times q$ . It has 309 digits.

$$n = \begin{array}{l} 1159350417396761496889250986461588752377145737545414477548552613761 \\ 4788540832635081727687881596832516846884930062548576411125016241455 \\ 2339182927162507656772727460097082714127730434960500556347274566628 \\ 0600999240371029914244722922157727985317270338393813346926841373276 \\ 22000966676671831831088373420823444370953 \end{array}$$

$\phi(n) = (p - 1)(q - 1)$  has 309 digits.

Bob chooses  $e = 35535$  (the ideal is 65537). He then finds  $d$ .

$\phi(n) =$	1159350417396761496889250986461588752377145737545414477548552613761 4788540832635081727687881596832516846884930062548576411125016241455 2339182927162507656751054233608492916752034482627988117554787657013 9234444057169895817281960982263610754672118646121713591073586406140 08885170265377277264467341066243857664128
-------------	---

$e =$	35535
-------	-------

$d =$	5800830286003776393609366128967791759466906208965096218042286611138 0593852822358731706286910030021710859044338402170729869087600611530 6202524959884448047568240966247081485817130463240644077704833134010 8509473852956450719367740611973265574242372176176746207763716420760 033708533328853214470885955136670294831
-------	---

Alice wants to send the message “THIS IS A TEST”, which can be changed to a numeric value using the 00–26 encoding scheme (26 is the *space* character).

The ciphertext calculated by Alice is  $C = P^e$ , which is shown below.

$P =$	1907081826081826002619041819
-------	------------------------------

$C =$	4753091236462268272063655506105451809423717960704917165232392430544 5296061319932856661784341835911415119741125200568297979457173603610 1278218847892741566090480023507190715277185914975188465888632101148 3541033616578984679683867637337657774656250792805211481418440481418 4430812773059004692874248559166462108656
-------	--

Bob can recover the plaintext from the ciphertext using  $P = C^d$ , which is shown below.

$P =$	1907081826081826002619041819
-------	------------------------------

The recovered plaintext is “THIS IS A TEST” after decoding.

**Q** In a RSA cryptosystem a particular A uses two prime numbers  $p = 13$  and  $q = 17$  to generate her public and private keys. If the public key of A is 35. Then the private key of A is \_\_\_\_\_.  
**(GATE-2017) (2 Marks)**

**Q** In the RSA public key cryptosystem, the private and public keys are  $(e, n)$  and  $(d, n)$  respectively, where  $n = p \cdot q$  and  $p$  and  $q$  are large primes. Besides,  $n$  is public and  $p$  and  $q$  are private. Let  $M$  be an integer such that  $0 < M < n$  and  $f(n) = (p-1)(q-1)$ . Now consider the following equations.

- I.  $M' = M^e \text{ mod } n, \quad M = (M')^d \text{ mod } n$
- II.  $ed \equiv 1 \pmod{n}$
- III.  $ed \equiv 1 \pmod{f(n)}$
- IV.  $M' = M^e \text{ mod } f(n), \quad M = (M')^d \text{ mod } f(n)$

Which of the above equations correctly represent RSA cryptosystem? **(GATE-2009) (2 Marks)**

- (A) I and II
- (B) I and III
- (C) II and IV
- (D) III and IV

**Q** A sender is employing public key cryptography to send a secret message to a receiver. Which one of the following statements is TRUE? **(GATE-2004) (1 Marks)**

- (A) Sender encrypts using receiver's public key
- (B) Sender encrypts using his own public key
- (C) Receiver decrypts using sender's public key
- (D) Receiver decrypts using his own public key

# Fermat little theorem

knowledgeGate

**Q** The minimum positive integer p such that  $3^p$  modulo 17 = 1 is (GATE-2007) (1 Marks)

- (A) 5      (B) 8      (C) 12      (D) 16

## Diffie Hellman

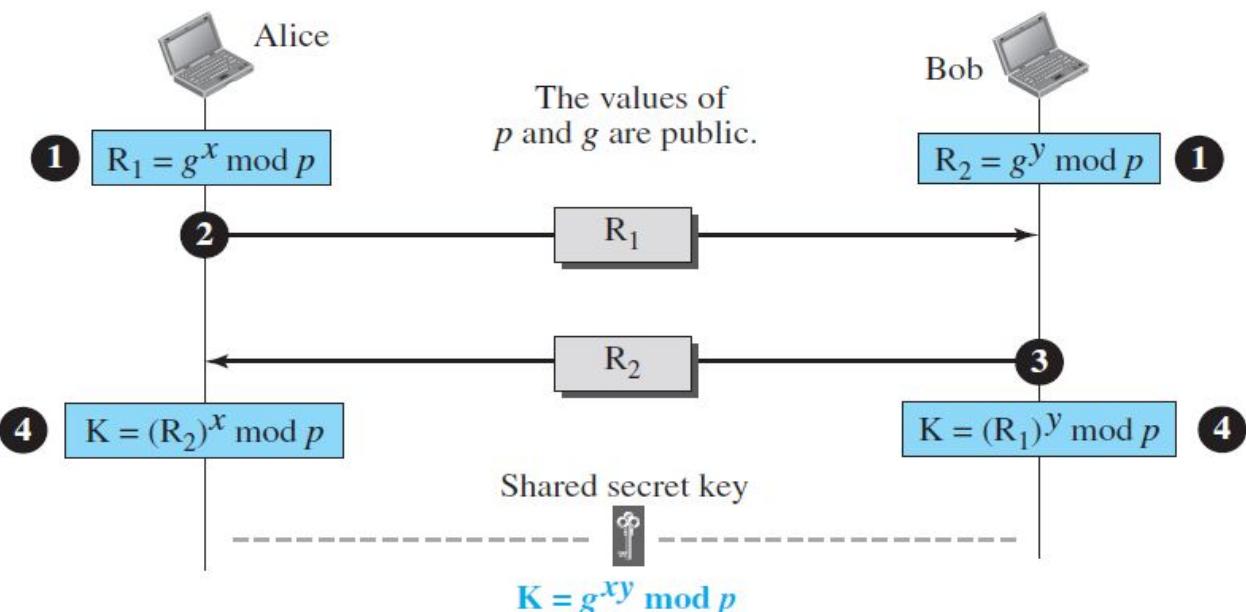
In the **Diffie-Hellman protocol** two parties create a symmetric session key without the need of a KDC. Before establishing a symmetric key, the two parties need to choose two numbers  $p$  and  $g$ . These two numbers have some properties discussed in number theory, but that discussion is beyond the scope of this book. These two numbers do not need to be confidential. They can be sent through the Internet; they can be public.

The steps are as follows:

1. Alice chooses a large random number  $x$  such that  $0 \leq x \leq p - 1$  and calculates  $R_1 = g^x \text{ mod } p$ . Bob chooses another large random number  $y$  such that  $0 \leq y \leq p - 1$  and calculates  $R_2 = g^y \text{ mod } p$ .
2. Alice sends  $R_1$  to Bob. Note that Alice does not send the value of  $x$ ; she sends only  $R_1$ .
3. Bob sends  $R_2$  to Alice. Again, note that Bob does not send the value of  $y$ , he sends only  $R_2$ .
4. Alice calculates  $K = (R_2)^x \text{ mod } p$ . Bob also calculates  $K = (R_1)^y \text{ mod } p$ .

$K$  is the symmetric key for the session.

$$K = (g^x \text{ mod } p)^y \text{ mod } p = (g^y \text{ mod } p)^x \text{ mod } p = g^{xy} \text{ mod } p$$



Bob has calculated  $K = (R_1)^y \bmod p = (g^x \bmod p)^y \bmod p = g^{xy} \bmod p$ . Alice has calculated  $K = (R_2)^x \bmod p = (g^y \bmod p)^x \bmod p = g^{xy} \bmod p$ . Both have reached the same value without Bob knowing the value of  $x$  and without Alice knowing the value of  $y$ .

**The symmetric (shared) key in the Diffie-Hellman method is  $K = g^{xy} \bmod p$ .**

Let us give a trivial example to make the procedure clear. Our example uses small numbers, but note that in a real situation, the numbers are very large. Assume that  $g = 7$  and  $p = 23$ . The steps are as follows:

1. Alice chooses  $x = 3$  and calculates  $R_1 = 7^3 \bmod 23 = 21$ . Bob chooses  $y = 6$  and calculates  $R_2 = 7^6 \bmod 23 = 4$ .
2. Alice sends the number 21 to Bob.
3. Bob sends the number 4 to Alice.
4. Alice calculates the symmetric key  $K = 4^3 \bmod 23 = 18$ . Bob calculates the symmetric key  $K = 21^6 \bmod 23 = 18$ .

The value of  $K$  is the same for both Alice and Bob;  $g^{xy} \bmod p = 7^{18} \bmod 23 = 18$ .

**Q** Suppose that two parties A and B wish to setup a common secret key (D-H key) between themselves using the Diffie-Hellman key exchange technique. They agree on 7 as the modulus and 3 as the primitive root. Party A chooses 2 and party B chooses 5 as their respective secrets. Their D-H key is **(GATE-2005) (2 Marks)**

- (A) 3                    (B) 4                    (C) 5                    (D) 6

**Q Which of the following are used to generate a message digest by the network security protocols? (GATE-2014) (1 Marks)**

- (P) RSA                    (Q) SHA-1                    (R) DES                    (S) MD5  
(A) P and R only        (B) Q and R only        (C) Q and S only        (D) R and S only

**Q A layer-4 firewall ( a device that can look at all protocol headers up to the transport layer) CANNOT (Gate-2011) (1 Marks)**

- (A) block HTTP traffic during 9:00PM and 5:00AM**
- (B) block all ICMP traffic**
- (C) stop incoming traffic from a specific IP address but allow outgoing traffic to same IP**
- (D) block TCP traffic from a specific user on a specific IP address on multi-user system during 9:00PM and 5:00AM**

**Q** An IP machine Q has a path to another IP machine H via three IP routers  $R_1$ ,  $R_2$ , and  $R_3$ .

Q— $R_1$ — $R_2$ — $R_3$ —H

H acts as an HTTP server, and Q connects to H via HTTP and downloads a file. Session layer encryption is used, with DES as the shared key encryption protocol. Consider the following four pieces of information:

[I<sub>1</sub>] The URL of the file downloaded by Q

[I<sub>2</sub>] The TCP port numbers at Q and H

[I<sub>3</sub>] The IP addresses of Q and H

[I<sub>4</sub>] The link layer addresses of Q and H

Which of I<sub>1</sub>, I<sub>2</sub>, I<sub>3</sub>, and I<sub>4</sub> can an intruder learn through sniffing at R<sub>2</sub> alone? **(GATE-2014) (2 Marks)**

- (A) Only I<sub>1</sub> and I<sub>2</sub>      (B) Only I<sub>1</sub>      (C) Only I<sub>2</sub> and I<sub>3</sub>    (D) Only I<sub>3</sub> and I<sub>4</sub>