Computer Network

* Computer Network:

1. Set of devices connected through links.
2. Node can be a computer, printer or any other device capable of sending and receiving data.
3. Links connecting the nodes are known as communication channels.
4. Task is divided among several computers.

* IP:

1. IP stands for Internet Protocol.
2. Protocol defined in the TCP/IP model used for sending packets from source to destination based on IP address available in packet headers.
3. IP address is a unique number provided to each and every device.
4. It consists of integers separated by dots. (eg: 192 . 168 . 10 . 26)

* IP Header:

1. Prefix to an IP packet that contains information about IP version, length of packet, source & destination IP address, etc.

Version (4 bits)	Header Length (4 bits)	Priority and Type of Service (8 bits)	Total Length(16 bits
Identification (16 bits)	Flags (3 bits)	fragmented offset (13 bits)	
Time to live (8 bits)	Protocols (8 bits)	Header checksum (16 bits)	
Source IP address (32 bits)			
Destination IP address (32 bits.			
Data (32 bits)			

→ Version - version of the IP protocol

→ Header length - Length of the header in 32 bit words.

→ Priority and type of Service - Specifies how the datagram should be handled.

→Total length - Length of the entire packet (Header + Data)

→Identification - To differentiate fragmented packets from different datagrams.

→Flags - Used to control or identify fragments.

→fragmented offset - Used for fragmentation / Reassembly.

→Time to live - limits a datagram's lifetime.

→Protocol - Defines the protocol used in data portion of IP datagram.

→Header checksum :- Used for error checking of the header.

→Source IP address :- IP address of Host that sent the packet.

→Destination IP address - IP address of Host that received the packet.

* IP fragmentation:

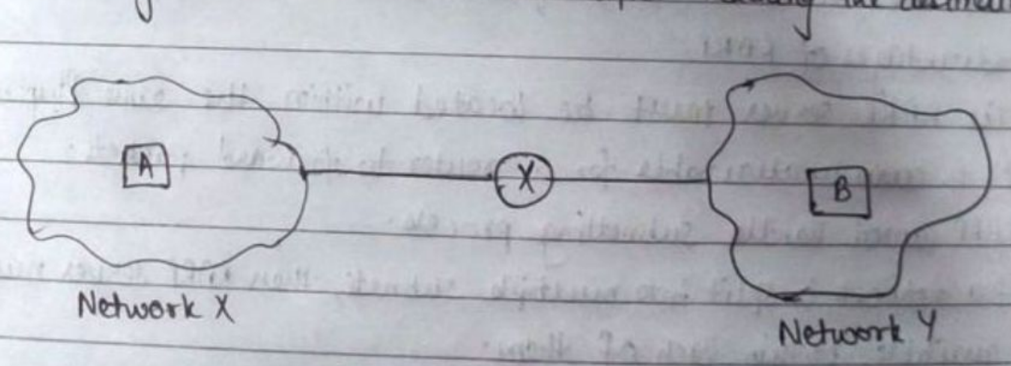
①. Every local Network supports a Maximum size of IP packet.

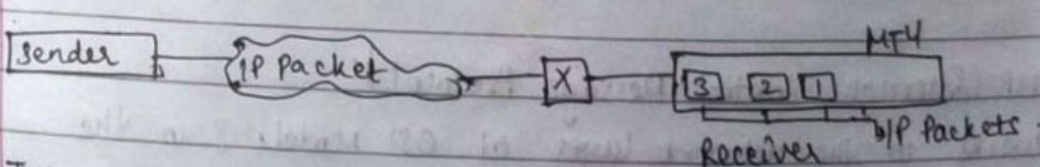
②. When a host transfers an IP packet, it shouldn't be larger than the Maximum Transmission Unit (MTU) size.

③. If the packet is larger than the MTU size, then it is divided into small packets and this process is called IP fragmentation.

④. The data would be broken into multiple pieces and carried in new fragments that are smaller than or equal to size of MTU.

⑤. This fragmented data reassembles after reaching the destination.





eg. This would happen when network X tries to send an IP packet whose MTU size is greater than that of the MTU size of the IP packet that Y can receive.

* ARP (Address Resolution Protocol):

①. Used to find MAC address of a device when IP address is known

②. The source knows the IP address of destination device but not the MAC address.

③. MAC address is required because you cannot communicate with a device in local area N/W without knowing its MAC Address.

$\boxed{\text{IP Address}} \xrightarrow{\text{ARP}} \boxed{\text{MAC Address}}$

④ It uses the value 1 for requests and 2 for responses.

→ Types of Mapping in ARP:

①. Static Mapping

②. Dynamic Mapping.

* RARP (Reverse Address Resolution Protocol):

①. Used to convert Ethernet Address into IP address.

②. Available for LAN technologies.

③. MAC address is known and IP address is requested.

④ Uses the value 3 for requests and 4 for responses.

Disadvantages of RARP:

①. The RARP server must be located within the same Physical N/W

②. It is unattainable for a router to forward packets.

③. RARP cannot handle subnetting process.

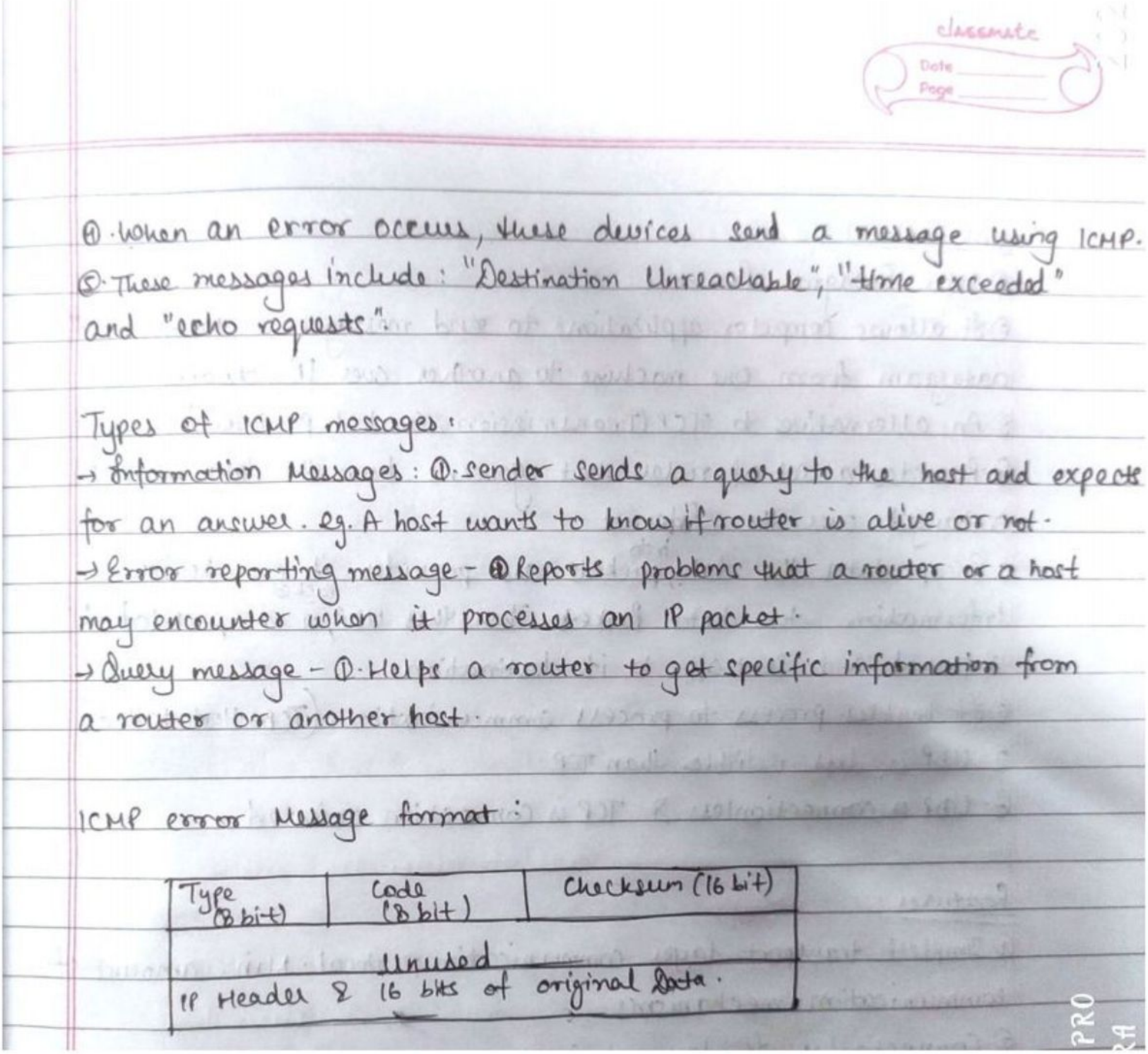
④. If a network is split into multiple subnets, then RARP server must be available within each of them.

* ICMP (Internet Control Message Protocol):

①. Works in the network layer of OSI Model & in the internet layer of the TCP/IP model.

②. Used to send control messages to network devices & hosts.

③. Routers and other N/W devices monitor the operation of the N/W



④. When an error occurs, these devices send a message using ICMP.
⑤. These messages include: "Destination Unreachable", "time exceeded" and "echo requests".

Types of ICMP messages:

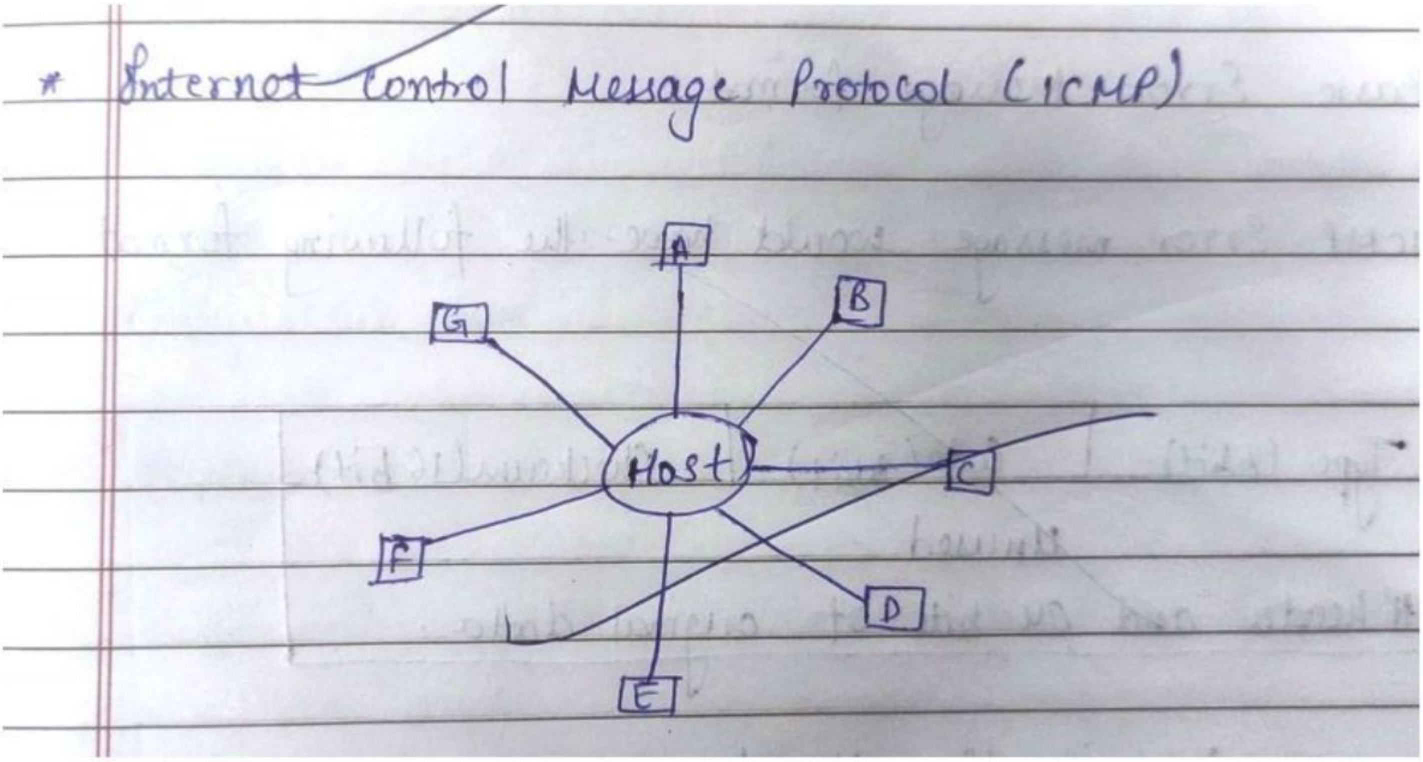
→ Information Messages: ①. sender sends a query to the host and expects for an answer. eg. A host wants to know if router is alive or not.

→ Error reporting message - ① Reports problems that a router or a host may encounter when it processes an IP packet.

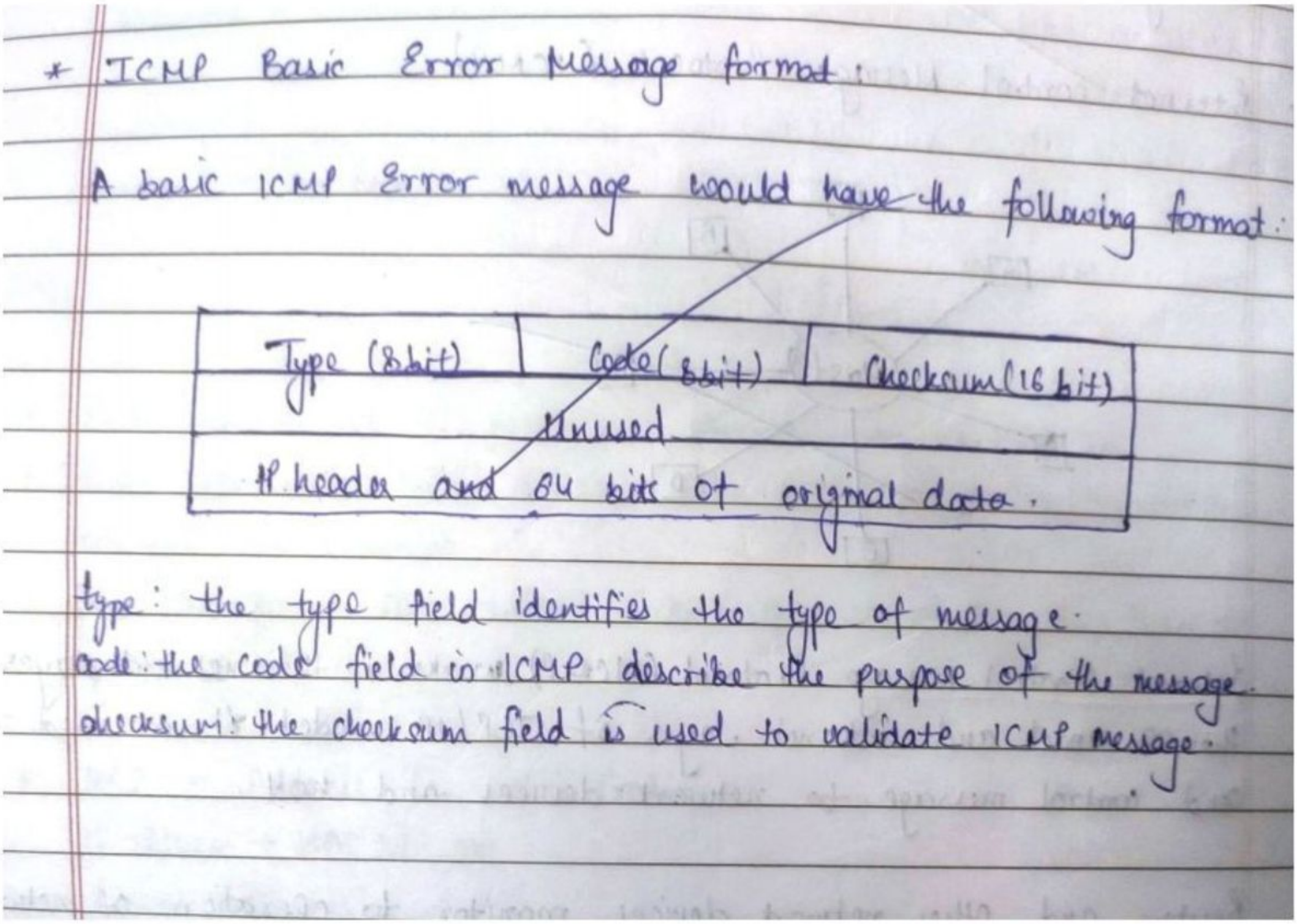
→ Query message - ①. Helps a router to get specific information from a router or another host.

ICMP error Message format:

Type (8 bit)	Code (8 bit)	Checksum (16 bit)
Unused		
IP Header & 16 bits of original Data.		



* Internet Control Message Protocol (ICMP)

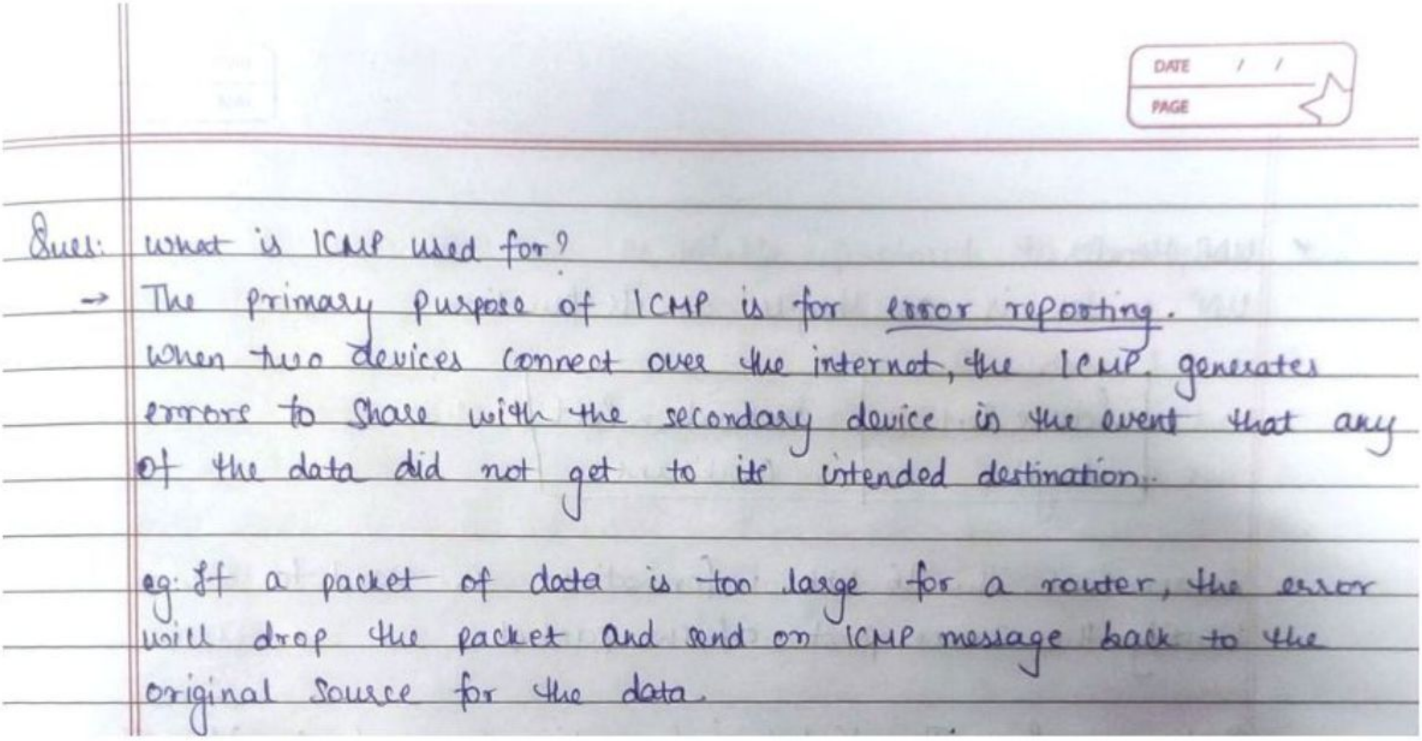


* ICMP Basic Error Message format :

A basic ICMP Error message would have the following format:

Type (8bit)	Code(8bit)	Checksum(16 bit)
Unused		
IP header and 64 bits of original data.		

type: the type field identifies the type of message.
code: the code field in ICMP describe the purpose of the message.
checksum: the checksum field is used to validate ICMP message.

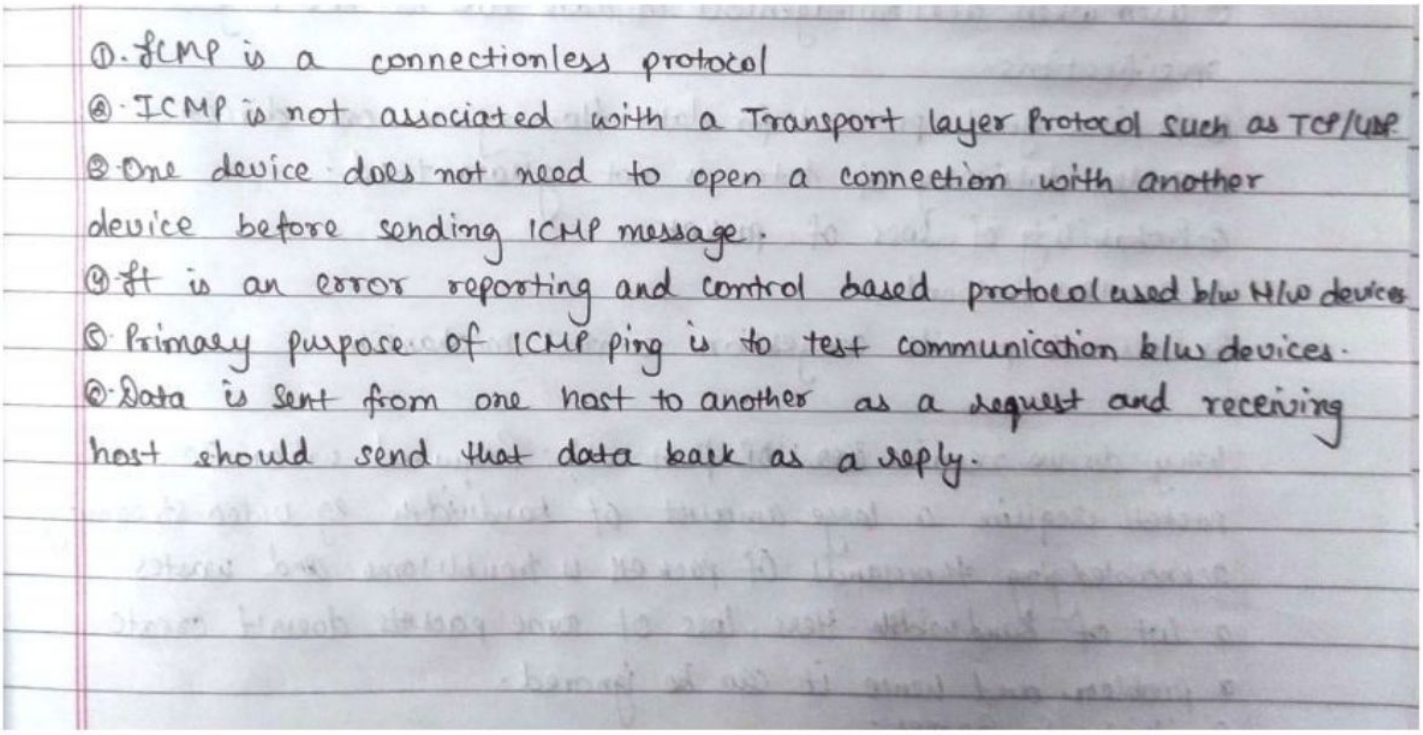


DATE / /
PAGE

Ques: what is ICMP used for?

→ The primary purpose of ICMP is for error reporting.
When two devices connect over the internet, the ICMP generates errors to share with the secondary device in the event that any of the data did not get to its intended destination.

eg: If a packet of data is too large for a router, the error will drop the packet and send on ICMP message back to the original source for the data.



①. ICMP is a connectionless protocol

②. ICMP is not associated with a Transport layer Protocol such as TCP/UDP

③. One device does not need to open a connection with another device before sending ICMP message.

④. It is an error reporting and control based protocol used b/w N/w devices

⑤. Primary purpose of ICMP ping is to test communication b/w devices.

⑥. Data is sent from one host to another as a request and receiving host should send that data back as a reply.

* UDP Protocol :

① User Datagram Protocol ·

② It allows computer applications to send messages in the form of datagram from one machine to another over IP network.

③ An alternative to TCP (Transmission control protocol).

④ Provides a set of rules that governs how the data must be exchanged over the Internet ·

⑤ Encapsulates the data into packets and provides its own header information to data packets. Then this UDP packet is encapsulated to IP packet and sent off to its destination.

⑥ It enables process to process communication. (TCP→ Host to Host)

⑦ UDP is less reliable than TCP.

⑧ UDP is connectionless & TCP is connection oriented.

Features :

① Simplest transport layer communication protocol. Min. amount of communication mechanisms.

② Connectionless: It does not create a virtual path to transfer data

③ Used when acknowledgement of data does not hold any specifications.

④ It is a good protocol for data flowing in one direction.

⑤ Order delivery of data is not guaranteed.

⑥ Probability of loss of packets.

⑦ Stateless Protocol.

⑧ Does not provide congestion control mechanism.

Why do we require UDP protocol?: Required when the packets require a large amount of bandwidth. eg. video streaming acknowledging thousands of packets is troublesome and wastes a lot of bandwidth. Here, loss of some packets doesn't create a problem and hence it can be ignored. Similarly in Games.

UDP Header

0 15/16 31

Source Port	Destination Port
Length	Checksum.

→ Source Port: 16 bit information which identifies which port is going to send the packet.

→ Destination Port: It Identifies which port is going to accept the info. (16-bit)

→ length → 16 bit field that specifies the length of entire packet (UDP) including Header.

→ Checksum: 16 bit field / Optional. Checks whether the info. is accurate or not as sometimes, the info gets corrupted while transmission. The checksum field is applied to the entire packet.

→ Characteristics & Importance of UDP:

① It offers low functionality with High Performance.

②. Optimal for Rate ~~pas~~ based small packet transfer.

③. Supports high throughput

④. Can send small, inefficient datagrams.

⑤. Supports multicast and Broadcast.

* TCP : (Transmission Control Protocol)

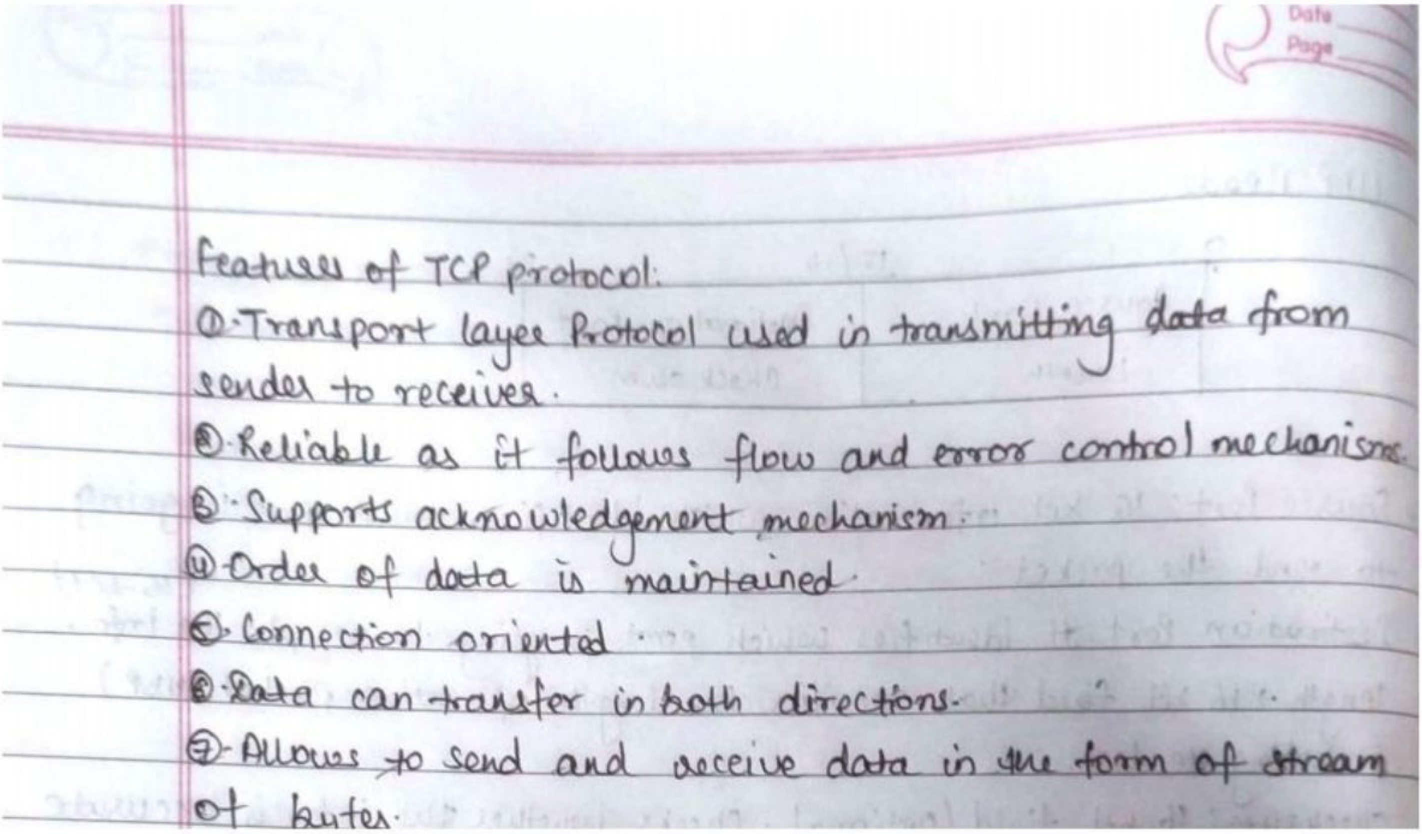
①. Transport layer protocol that facilitates transmission of packets from source to destination.

② Connection oriented. It establishes connection prior to communication.

③. Used with an IP protocol, they are referred to as TCP/IP.

④ Takes the data from application layer, divides the data into packets, provides numbering to these packets and then finally transmits them to application layer.

⑤. Connection will remain established until the communication is completed. b/w the sender and receiver.

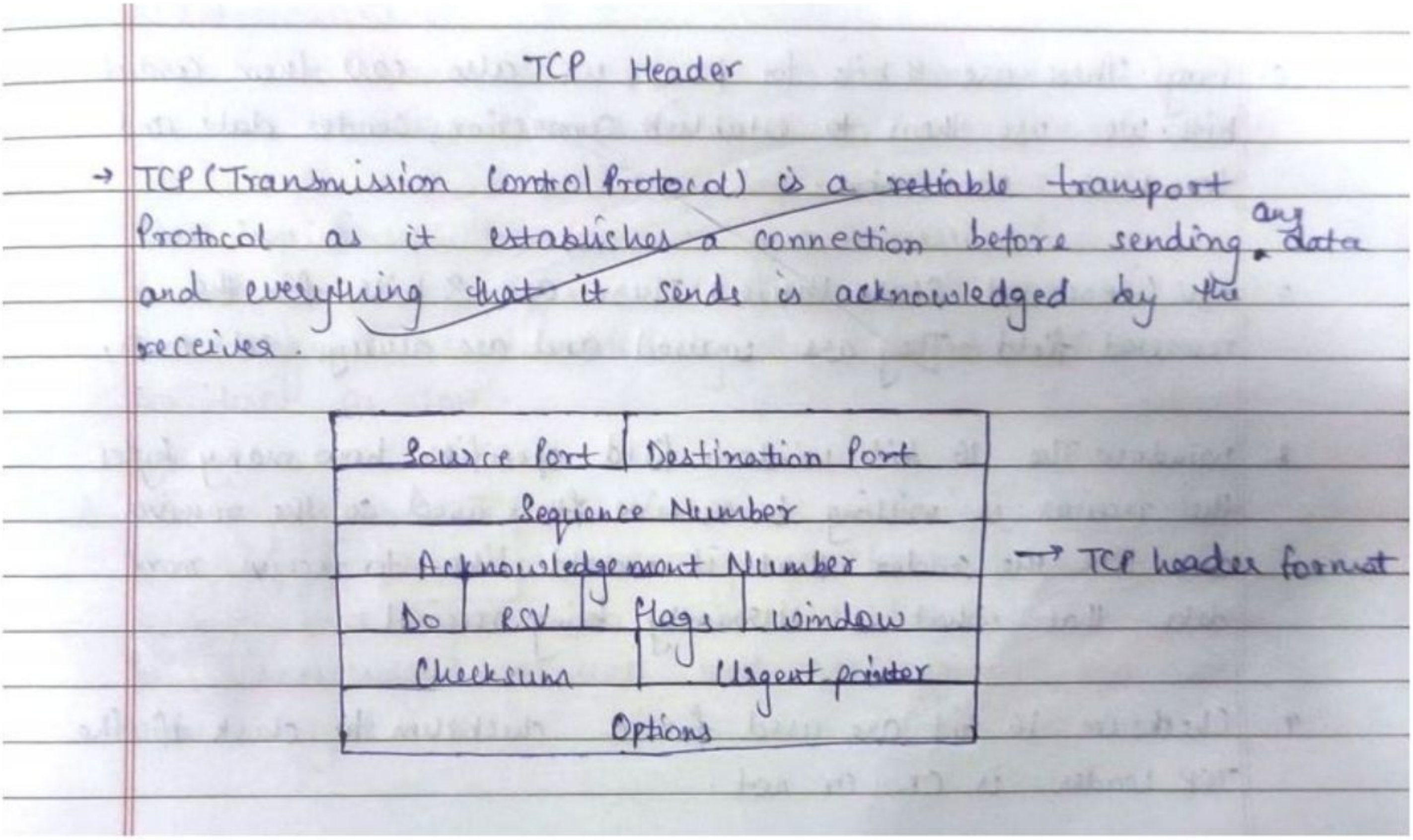


Date

Page

Features of TCP protocol:

1. Transport layer Protocol used in transmitting data from sender to receiver.
2. Reliable as it follows flow and error control mechanisms.
3. Supports acknowledgement mechanism.
4. Order of data is maintained.
5. Connection oriented.
6. Data can transfer in both directions.
7. Allows to send and receive data in the form of stream of bytes.



TCP Header

→ TCP (Transmission Control Protocol) is a reliable transport Protocol as it establishes a connection before sending any data and everything that it sends is acknowledged by the receiver.

Source Port		Destination Port	
Sequence Number			
Acknowledgement Number			
DO	RSV	Flags	Window
Checksum		Urgent pointer	
Options			

→ TCP header format

1. Souce Port: This is 16 bit field that specifies the port number of the sender.

2. Destination Port: This is a 16 bit field that specifies the port number of the receiver.

3. Sequence number: The sequence number is a 32-bit field that indicate how much data is sent during the TCP session.

4. Acknowledgement Number: This 32 bit field is used by the Receiver to request the next TCP. segment. This value will be the sequence number incremented by 1.

5. DO: This is the 4 bit data offset field, also known as the header length. It indicates the length of the TCP header so that we know where the actual data begins.

6. Flags: These are 9 bits for flags, we also call them control bits. We use them to establish connection, sender data and terminate connection.

7. RSV (Reserved State Value): These are 3 bits for the reserved field. They are unused and are always set to 0.

8. Window: The 16 bit window field specifies how many bytes the receiver is willing to receive. It is used so the receiver can tell the sender that it would like to receive more data that what is currently being received.

9. Checksum: 16 bit are used for a checksum to check if the TCP header is OK or not.

10 Urgent Pointer: These 16 bits are used when the URG bit has been set. The urgent Pointer is used to indicate where the urgent data ends.

11 Options: This field is optional and can be anywhere between 0 and 320 bits.

* Error control in TCP.

TCP Protocol has methods for finding out corrupted segments, missing segments, error segments and duplicated segments.

Error control in TCP is mainly done through use of three simple techniques:
(1). Checksum
(2) Acknowledgement
(3). Retransmission.
↳ Retransmission after RTO(Retransmission Time out)
↳ Retransmission after three duplicate ACK segments.

1. Check Sum: Every segment contains a checksum field which is used to find Corrupted Segment. If the segment is corrupted, then that segment is discarded by the destination TCP and is considered as lost.

2. Acknowledgement: TCP has another mechanism called acknowledgement to affirm that the data segments have been delivered. Control segments that contain no data but has sequence number will be acknowledged as well but ACK segments are not acknowledged.

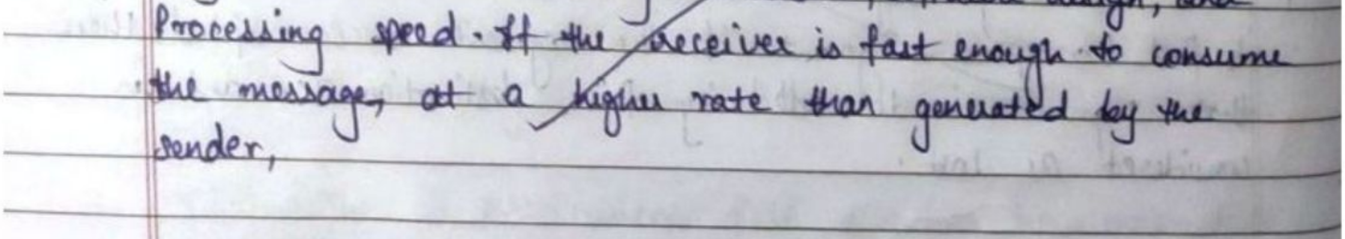
(3). Retransmission: when a segment is missing, delayed to delivered to receiver corrupted when it is checked by receiver than that segment is retransmitted again. Segments are retransmitted only during two events: when the sender receives three duplicate acknowledgement (ACK) or when a retransmission timer expires.

(a). Retransmission after RTO: When the timer matures, i.e., time out TCP resends the segment in front of the queue.

(b). Retransmission aftee three duplicates ACK segments: The previous rule about retransmission of a segment is sufficient if the value of RTO is not large. If three duplicate acknowledgements arrive for a segment, the next segment is retransmitted without waiting for the time-out.

TCP Flow Control:

When the network hosts start Communicating with each other, one sends packets, and the other receives them. Both may have different hosting hardware, software design, and Processing speed. If the receiver is fast enough to consume the message, at a higher rate than generated by the sender,

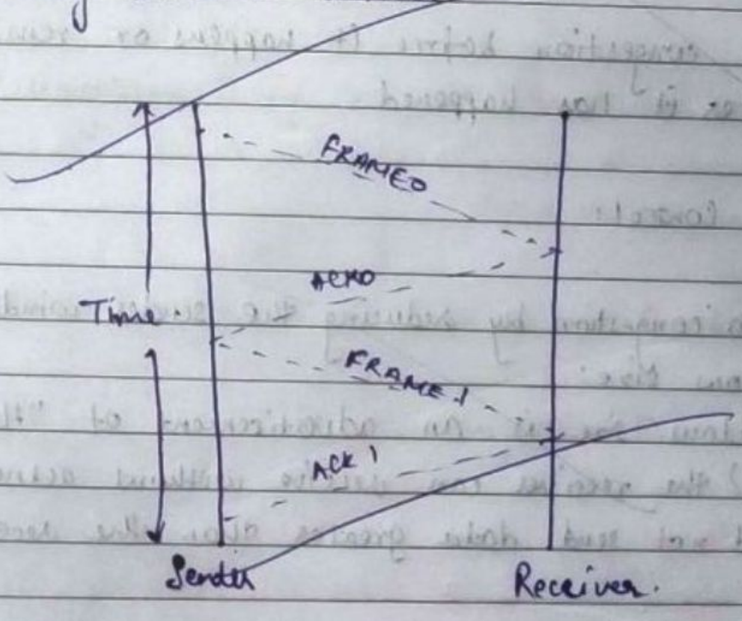

Sender
TCP/IP
sending at 10 messages/ second
TCP/IP
TCP Receiver
Receiving at 5 messages/ second
Sender
Receiver

* TCP flow Control is a protocol designed to manage the data flow between the user and the server.

* featurs:

→ Slow sender, speedy receiver - No flow control required.
→ fast sender, slow receiver - flow control is required.
→ The sender is sending message at the rate of 10 messages/second while the receiver is receiving at the rate of 5 messages/second.

There are two types:
(1) Stop and wait Protocol
(2) Sliding window Protocol.



* Congestion in Network :

→ Congestion refers to a network state where the message traffic becomes so heavy that it slows down the network response time.

→ Congestion is an important issue that can arise in PSN (Packet Switched Network)

→ Congestion leads to the loss of Packet in transit.

→ So, it is necessary to control the congestion in network.

→ It is not possible to completely avoid the congestion.

* Congestion Control:

→ Either prevent congestion before it happens or remove the congestion after it has happened.

* TCP Congestion Control:

→ TCP reacts to congestion by reducing the sender window size.

1. Receiver window size:

- Receiver window size is an advertisement of "How much data (in bytes) the receiver can receive without acknowledgment
- Sender should not send data greater than the 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.

II Congestion window:

→ Sender should not send data greater than congestion window size.

→ Otherwise it leads to dropping the TCP segments which causes TCP Retransmission.

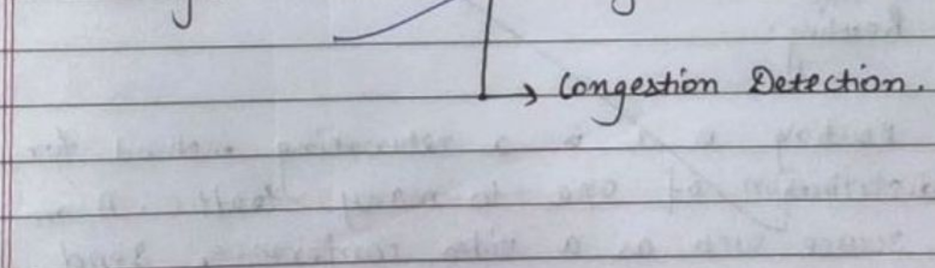
→ So, sender should always send data less than equal to congestion window ~~size~~.

→ 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.

→ formula ⇒ Sender window size = Minimum(Receiver window size, Congestion Window Size).

* TCP congestion Policy basically based on 3 parts.


TCP congestion Control

Slow start

Congestion Avoidance.

Congestion Detection.

* Multicasting:

Casting: Casting in computer Network means transmitting data (stream of packets) over a network.

These are three types of casting:

1. Unicast Transmission
2. Broadcast Transmission.
3. Multicast Transmission.

Multicasting: Multicast is a method of group Communication where the sender sends data to multiple receivers or nodes present in the network simultaneously.

→ Multicasting is a type of one-to-~~one~~ many and many-to-many communication as it allows sender or senders to send data Packets to multiple receivers at once across LANs or WANs.

* Multicast Routing:

Multicast Routing is ~~a~~ is a networking method for efficient distribution of one-to-many traffic. A ~~m~~ multicast source such as a video, conference, send traffic in one stream to a multicast group. The multicast group containing receivers such as computer devices, etc.

* Common use include these technologies:-

(1). Voice over IP (VOIP)
(2). Video On Demand (VOD)
(3). Video Conferencing.
(4). IP television (IPTV)

10/10

Good

Debugging tools: There are several tools used for debugging. Here we will learn two tools that use ICMP for debugging. The two tools are **ping** and **traceroute**.

1. **ping:** By ping tool we can send echo-request and echo-reply messages that check whether the host or a router is alive or running.

2. **traceroute:** Traceroute is a tool that tracks the route taken by a packet on an IP network from source to destination. It records the time taken by the packet on each hop during its route from source to destination. Traceroute uses ICMP messages and TTL values. The TTL value is calculated; if the TTL value reaches zero, the packet gets discarded. Traceroute uses small TTL values as they get quickly expired. If the TTL value is 1 then the message is produced by router 1; if the TTL value is 2 then the message is produced by router 2, and so on.

Example: Suppose A and B are two different hosts, and A wants to send the packet to the host B. Between A and B, 3 routers exist. To determine the location of the routers, we use the traceroute tool.

TTL value =1: First, host A sends the packet to router 1 with TTL value 1, and when the packet reaches to router 1 then router reduces the value of TTL by one and TTL values becomes 0. In this case, router 1 generates the time-exceeded message and host A gets to know that router 1 is the first router in a path.

TTL value=2: When host A sends the packet to router 1 with TTL value 2, and when the packet reaches to router 1 then the TTL value gets decremented by 1 and the TTL value becomes 1. Then router 1 sends the packet to router 2, and the TTL value becomes 0, so the router generates a time-exceeded message. The host A gets to know that router 2 is the second router on the path.

TTL value=3: When host A sends the packet to router 1 with TTL value 3, then the router decrements its value by one, and the TTL value becomes 2. Then, router 1 sends the packet to router 2, and the TTL value becomes 1. Then, router 2 sends the packet to router 3, and the TTL value becomes 0. As TTL value becomes 0, router 3 generates a time-exceeded message. In this way, host A is the third router on a path.