PART-A

When you are sending a message to your friend, you will probably use a browser based chat messenger(like whatsapp). When you hit send, these are the layers through which your message reaches your friends in Germany

1) **Application layer:** In this layer, the user directly interacts with the app. The app prepares the message and needs to send it to the chat server.It uses an API request (like an HTTP POST) or a socket message (over TCP or WebSocket).The application layer passes the data to the transport layer.Before sending the packet, your computer must know the IP address of the chat server, so it sends out a DNS query and recieves the IP address as response.

2) **Transport layer:** TCP establishes a connection using the three-way handshake:
   SYN (from you to server)- The client sends out a request to the server asking that it wants to send data
   SYN-ACK (server to you)- The server acknowledges this and tells that is ready to begin a connection, and it includes the server's own initial sequence number.
   ACK (you to server)-The client receives the SYN-ACK and sends a final packet with the ACK flag set.This final ACK packet acknowledges the server's SYN-ACK.
   UDP (used for low-latency apps) uses no handshake, no ACKs, and no retransmission. It is faster but less reliable than TCP.

3) **Network layer:** Your TCP segment is wrapped inside an IP packet with source and destination IP. Since your local IP is private, your router replaces it with your public IP before sending the packet to the Internet. The packet reaches from your PC to your Wi-Fi router and then travels through router to your ISP's edge router. ISP passes it through their core routers. ISP → connects to international Internet Exchange Points (IXPs). Then it travels across multiple Autonomous Systems (AS), each controlled by a large ISP. Finally, it reaches your friend's ISP in Germany.That ISP routes it internally to your friend's local network and then to their PC.

4) **Data link and physical layers:** When a packet arrives in a network, it is the responsibility of the DLL to transmit it to the host using its MAC address. Packet in the Data Link layer is referred to as Frame.The packet received from the Network layer is further divided into frames. DLL also encapsulates Sender and Receiver's MAC address in the header. The receiver's MAC address is obtained by placing an ARP request onto the wire asking, "Who has that IP address?" and the destination host will reply with its MAC address.Physical Layer is responsible for transmitting individual bits from one node to the next. When receiving data, this layer will get the signal received and convert it into 0s and 1s and send them to the Data Link layer, which will put the frame back together.

5) **Reverse path**: During the reverse journey, the destination and source addresses are reversed. The packet travels back (possibly through a different route) but through the same layers. On

reaching your router, NAT converts the public IP back to your local IP and delivers it to the correct device. TCP reassembles the message, ensuring all segments arrive and are acknowledged in order.

6) If a router’s queue fills up, it drops packets (congestion).TCP detects missing acknowledgments and triggers retransmission. It also performs congestion control, adjusting its sending rate to match network capacity. Whereas UDP just considers it as data loss but doesnt retransmit data.