10]

A) Write down the features of UDP.

⇒ UDP (User Datagram Protocol) is a transport layer protocol in the internet protocol suite. Features of UDP:

i. Connectionless: UDP doesn't establish a connection before transmitting data, making it suitable for quick and independent data transfers.
ii. Low overhead: UDP has minimal protocol overhead compared to TCP, resulting in faster transmission and lower network congestion.
iii. Datagram-oriented: UDP treats data as individual packets, allowing them to be transmitted independently without the need for sequencing.
iv. Fast & efficient: UDP's simplicity and lack of extensive error-checking mechanisms make it faster and more efficient than TCP, making it suitable for real-time applications.
v. Broadcast and multicast support: UDP supports broadcasting data to all devices on a network or multicasting data to specific groups of devices simultaneously making it useful for efficient content distribution.
vi. Suitable for one-way communication: UDP is commonly used for one-way data transmission, where the sender doesn't require a response from the receiver. This makes it

B] Explain UDP Socket Options

⇒ ~~Comm~~ UDP socket options are parameters that can be set on a UDP socket to control various aspects of its behavior. Some of the UDP socket options:

i) SO_BROADCAST: Enables sending broadcast messages from the UDP socket.
ii) SO_REUSEADDR: Allows multiple UDP sockets to bind to the same local address & port
iii) SO_TIMEOUT: Sets the timeout value for blocking operations on the UDP socket.

iv) IP_TOS : Sets the Type of Service field in the IP header for outgoing UDP packets.
v) IP_MULTICAST_TTL : Set the time-to-live value for outgoing multicast UDP packets
vi) SO_RCVBUF & SO_SNDBUF: Adjusts the receive and send buffer sizes for the UDP socket.
vii) IP_MULTICAST_LOOP : Controls whether multicast UDP packets sent by the sockets are received by the same packet.
viii) IP_MULTICAST_IF : Specifies the network interface to use for outgoing multicast UDP packets.

C] Explain about DatagramPacket and DatagramSocket classes.

⇒ The DatagramPacket and DatagramSocket classes are part of the Java networking API and are used for communication using the UDP.:

- DatagramPacket:
  The datagram packet class represents a UDP packet, which is a self-contained unit of data that can be sent over a network. It encapsulates the data to be sent or received and includes information about the source and destination address. DatagramPacket objects are used for sending and receiving data using UDP.

Key features and usage of DatagramPacket class:

- Data encapsulation: It encapsulates the data to be sent or received.
- Source and destination information: It holds information about the source and destination addresses of the packet.
- Data access: It provides methods to get and set the data buffer, data length, source and destination addresses.

- DatagramSocket:
  The DatagramSocket class represents a UDP socket, which is the end point for

sending and receiving UDP packets. It provides methods for establishing a connection, sending and receiving packets over the network.

key features and usage of DatagramSocket class:

- Socket creation: It creates a UDP socket that can be used for sending and receiving UDP packets.
- Binding to a port: It allows the socket to bind to a specific port on the local machine.
- Sending packets: It provides methods to send DatagramPacket objects to a specified destination.
- Receiving packets: It provides methods to receive DatagramPacket objects containing incoming data.
- Closing the socket: It provides a method to close the socket and release associated resources.

11]

A) What is Multicast? Explain with examples.

⇒ Multicast in a network communication method where a single sender can transmit data to multiple recipents simultaneously. It allows efficient distribution of data to a group of recipients who are interested in receiving the data.

In multicast communication, the sender sends a single copy of the data, and it gets replicated and delivered to multiple receivers in the network. This is in contrast to unicast communication, where the sender sends a separate copy of the data to each individual recipient.

Examples:

i) Multimedia Streaming: Efficiently streams audio and video content to multiple receivers by transmitting a single multicast stream.

ii) Software Updates: Distributed software updates or patches simultaneously to neimerous devices in a network using multicast.

iii) Real-time Collaboration: Enables synchronized communication and interaction in applications like video conferencing and online goming by multicast transmission of audio, video, or game data.

iv) Network Routing Protocols: Facilitates efficient packet delivery across networks by using multicast for distributing routing information among routers.

B) Describe the working methodology of Multicast.

⇒ The working methodology of multicast:

i) Group Formation: A multicast group is formed by specifying a multicast group address.

ii) Sender Transmission: The sender sends data to the multicast group address.

iii) Multicast Routing: Routers use multicast routing protocols to determine how

to forward multicast packets.

iv) Receiver joining: Receivers express their interest in receiving data from the multicast group.

v) Data Replication and Delivery: Routers replicate and forward multicast packets to reach interested receivers.

vi) Receiver Processing: Receivers receive and process the multicast data.

vii) Leave and Network Updates: Receivers can leave the multicast group, and the network updates its routing accordingly.

C) Explain the features of IP Multicast.

⇒ The features of IP Multicast:

i. Efficient Data Delivery: IP multicast sends a single copy of data, replicated and forwarded by routers, reducing network bandwidth usage and improving performance.

ii. Scalability: IP multicast supports a large number of receivers by delivering packets only to necessary network segments, minimizing network congestion.

iii. One-to-many communication: IP multicast enables one sender to communicate simultaneously with multiple receivers, distributing data efficiently.

iv. Group Membership Management: IP Multicast allows dynamic membership changes, letting receivers join or leave multicast groups as needed.

v. Application Agnostic: IP Multicast can be used by various applications and services for efficient data distribution.

vi. Reliable Delivery: IP Multicast operates over UDP, but application-level protocols can ensure reliable delivery if required.

vii. IGMP: Internet Group Management Protocol enables hosts to inform routers about their interest in receiving multicast traffic for specific groups, managing multicast group membership.

viii) Multicast Routing Protocols: IP Multicast utilizes routing protocols like PIM to

establish and maintain optimal paths for multicast traffic, ensuring efficient delivery.

D) Write about 'Multicast Address and Multicast Group.

• Multicast Address:

A multicast address is a special type of IP address used for multicast communication. It is used to identify a group of network devices that are interested in receiving multicast traffic. Multicast addresses are in the range of 224.0.0.0 to 239.255.255.255. When a sender wants to send data to multiple receivers, it uses the multicast address as the destination address to reach all the members of the multicast group.

• Multicast Group:

A multicast group is a logical group of network devices that have expressed their interest in receiving multicast traffic for a specific multicast addresses. Devices become members of a multicast group by joining the group using protocols like IGMP. Once a device joins a multicast group, it can receive multicast packets sent to the corresponding multicast address. Multicast group allows efficient one-to-many communication where a single sender can reach multiple receivers who are part of the group.

12]

1. Explain the RMI Architecture.

⇒ In an RMI application, we write two programs, a server program (resides on the server) and a client program (resides on the client).

- Inside the server program, a remote object is created and reference of that object is made.
- The client program requests the remote objects on the server and tries to invoke its methods.

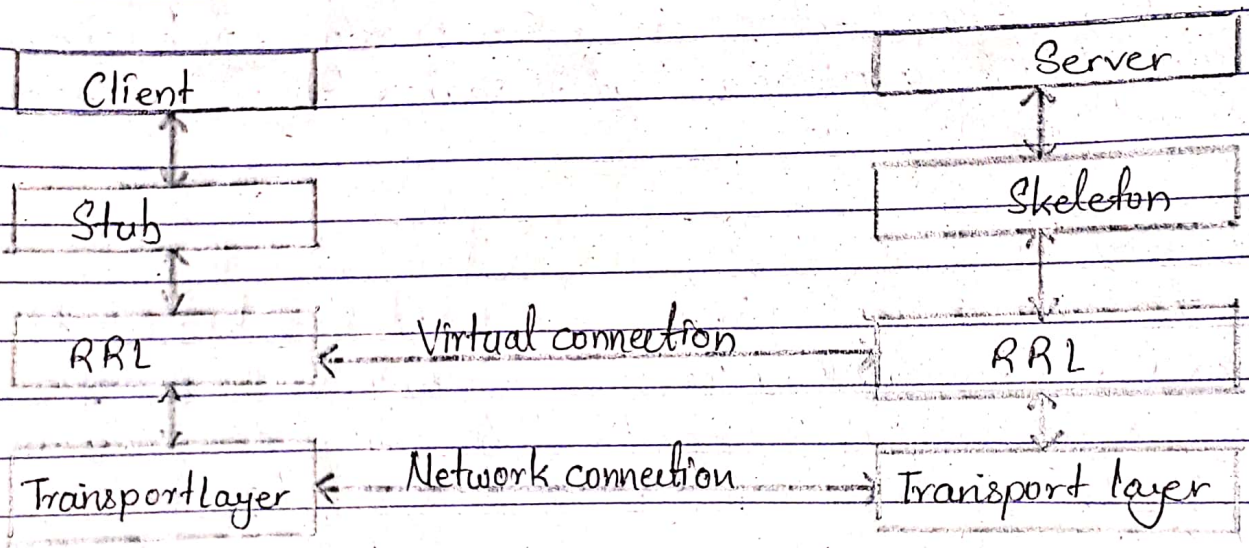

figure: Architecture of an RMI application

- Transport Layer: This layer connects the client and the server. It manages the existing connection and also sets up new connections.
- Stub - A stub is a representation (proxy) of the remote object at client. It resides in the client system; it acts as a gateway for the client program.
- Skeleton - This is the object which resides on the server side. stub communicates with this skeleton to pass request to the remote object.
- RRL (Remote Reference Layer) - It is the layer which manages the references made by the client to the remote object.

2) Explain the Stub and Skeleton role in RMI.

⇒ Role of Stub in RMI:

- Acts as a surrogate for the remote object on the client-side.
- Provides a local representation of the remote object
- Handles network communication, serialization, and deserialization of method calls and results.
- Hides the complexities of remote communication from the client.
- Allows method invocations on the remote object as if it were a local object.

Role of Skeleton in RMI:

- Resides on the server-side and acts as an intermediary between the network and the remote object.
- Receives method invocations from the Stub and forwards them to the remote object for execution.
- Marshals return values or exceptions from the remove object and transmits them back to the Stub.
- Handles the low-level details of network communication and object marshaling.
- Provides a bridge between the network and the remote object, ensuring proper method invocation and result transmission.

3) What are the different types of classes that are used in RMI.

The classes are:

- Remote class
  - It is the class that consists of the instances that can be accessed from the server and also consists of different properties that are required to be accessed by the client and server during request and response time.

- Remote class object:
  - Wherever this object is defined the instances of the class can be accessed using other object
  - On other computers that are not in the network it can accessed through the useof object handles.

- Serializable class:
  - It is the class that consists of instances that can be marshaled or turned in a linear sequence to represent the bits.
- Serializable class objects:
  - It allows the request to be transmitted from one computer to another.
  - It allows the easy implementation of the serializable class.

4) What is the method that is used by the RMI client to connect to remote RMI server?

⇒ The steps involved when an RMI client connects to a remote RMI server:

i. Locate the remote object: Obtain a reference to the remote object using a registry service or other means.
ii. Look up the remote object: Retrieve a stub object from the registry or other lookup mechanism.
iii. Establish a connection: Create a network connection with the remote server using TCP/IP.
iv. Marshaling and parameter passing: Serialize method arguments into a format suitable for network transmission.
v. Remote method invocation: Send the method call, including its name and parameters, to the server over the network connection.
vi. Unmarshaling and execution: Deserialize the method call on the server, invoke the method on the remote object, and perform the necessary operations.
vii) Return value and response: Serialize the return value (if any) into a response and send it back to the client.
viii) Client-side processing: Deserialize the response on the client side and retrieve the return value.