

## Distributed Systems - General Concepts

- 1. What is a distributed system?**  
A distributed system is a network of independent computers that work together to appear as a single cohesive system. Components communicate through message passing to achieve a shared goal.
  - 2. What are the advantages of distributed systems?**  
They provide scalability, fault tolerance, and resource sharing. These systems divide tasks across nodes for improved efficiency and reliability.
  - 3. What are the challenges in distributed systems?**  
Major challenges include synchronization, fault tolerance, network latency, and ensuring consistency across distributed components.
  - 4. What is consistency in distributed systems?**  
Consistency ensures that all nodes have the same data state at any given time. Strong consistency guarantees immediate synchronization, while eventual consistency allows temporary mismatches.
  - 5. What is the difference between client-server and peer-to-peer models?**
    - **Client-Server:** Clients request resources, and a centralized server provides them.
    - **Peer-to-Peer:** All nodes are equal and share resources directly without central coordination.
  - 6. Explain mutual exclusion in distributed systems.**  
Mutual exclusion ensures that only one process accesses a shared resource at a time, preventing conflicts and inconsistencies in distributed environments.
- 

## Practical-Oriented Questions

- 1. What is the purpose of socket programming?**  
Socket programming facilitates communication between two devices over a network. It is used to create servers and clients that exchange data streams.
- 2. What is the Producer-Consumer problem?**  
It's a synchronization problem where a producer creates data, and a consumer processes it. The challenge is to manage a shared buffer so that it neither overflows nor underflows.
- 3. How does the token ring algorithm ensure mutual exclusion?**  
A unique token circulates among processes, granting access to the process holding it. This eliminates conflicts and avoids the need for multiple message exchanges.
- 4. What is Remote Procedure Call (RPC)?**  
RPC allows a program to execute a procedure on a remote machine as if it were local. It uses stubs to send requests and responses between client and server.
- 5. What is Remote Method Invocation (RMI)?**  
RMI enables a Java program to invoke methods on remote objects. It uses interfaces and RMI registries to locate and interact with remote objects.

---

## Cloud Computing Questions

1. **What are the types of cloud services? Explain.**
  - **SaaS (Software as a Service):** Users access applications over the internet (e.g., Gmail, Google Docs).
  - **PaaS (Platform as a Service):** Developers use cloud platforms for app development (e.g., Google App Engine).
  - **IaaS (Infrastructure as a Service):** Provides virtualized computing resources like servers and storage (e.g., AWS EC2).
2. **What are the types of cloud deployment models?**
  - **Public Cloud:** Shared infrastructure managed by third parties, open to multiple organizations.
  - **Private Cloud:** Dedicated infrastructure for a single organization, offering greater control and security.
  - **Hybrid Cloud:** Combines public and private clouds for flexibility and optimized cost-performance balance.
3. **What is Storage as a Service?**

It is a cloud-based service where companies rent storage from providers. Data is stored remotely, ensuring accessibility, scalability, and reduced infrastructure costs.
4. **What is Identity and Access Management (IAM)?**

IAM manages user identities and access permissions to ensure only authorized users can access specific resources. It enhances security in cloud environments.
5. **What are the main benefits of cloud computing?**

Cloud computing offers scalability, cost-efficiency, and accessibility. Users can deploy resources on demand without maintaining physical hardware.

---

## Additional Questions

1. **What is clock synchronization in distributed systems?**

It ensures that all nodes in a distributed system maintain a consistent time. Techniques include logical clocks (e.g., Lamport clocks) and physical clock synchronization.
2. **What is a distributed shared memory (DSM)?**

DSM allows multiple systems to share memory, making it appear as a single memory space. It simplifies programming by abstracting data distribution.
3. **What are election algorithms?**

Election algorithms, like Bully and Ring algorithms, elect a leader node to coordinate tasks in a distributed system. They ensure system stability during failures.

4. **What is fault tolerance in distributed systems?**

Fault tolerance allows a system to continue functioning even if some components fail. Techniques include replication and checkpointing.

5. **What are load balancing strategies in distributed systems?**

Load balancing ensures even distribution of tasks among nodes. Static methods predefine load assignments, while dynamic methods adjust loads during runtime.

---

## **Distributed Systems - Advanced Concepts**

1. **What is the CAP theorem?**

The CAP theorem states that a distributed system can guarantee only two of the following three: Consistency, Availability, and Partition Tolerance. It highlights the trade-offs in system design.

2. **What is the difference between synchronous and asynchronous communication?**

- **Synchronous:** The sender waits for a response from the receiver before proceeding.
- **Asynchronous:** The sender continues processing without waiting for a response, using mechanisms like callbacks.

3. **Explain transparency in distributed systems.**

Transparency hides the complexity of distributed systems from users. Types include access transparency (hide location details), replication transparency (multiple copies appear as one), and fault transparency (system recovers from failures).

4. **What are the differences between a process and a thread?**

A process is an independent program with its own memory space, while a thread is a lightweight sub-process that shares memory with other threads in the same process.

5. **What are distributed file systems (DFS)?**

DFS manages files stored on multiple networked systems, making them accessible as a single file system. Examples include Hadoop HDFS and NFS.

6. **What is process migration?**

Process migration involves transferring a process from one machine to another for load balancing, fault tolerance, or improved performance.

7. **What are the different types of election algorithms?**

- **Bully Algorithm:** The highest priority process becomes the leader.
  - **Ring Algorithm:** Processes are organized in a logical ring, and the election message is passed around until a leader is chosen.
- 

## **Cloud Computing - Advanced Topics**

1. **What are the key differences between IaaS, PaaS, and SaaS?**

- **IaaS:** Provides virtualized computing resources (e.g., AWS EC2).

- **PaaS:** Offers a platform for app development (e.g., Heroku).
  - **SaaS:** Provides software applications over the internet (e.g., Dropbox).
2. **What is virtualization in cloud computing?**  
Virtualization creates virtual versions of physical resources (like servers and storage) to maximize hardware utilization and scalability.
  3. **What are the challenges in cloud computing?**  
Challenges include security, vendor lock-in, latency, compliance with regulations, and data privacy concerns.
  4. **What is edge computing?**  
Edge computing processes data closer to the source (e.g., IoT devices) instead of centralized cloud servers, reducing latency and bandwidth usage.
  5. **What is the difference between vertical and horizontal scaling in cloud computing?**
    - **Vertical Scaling:** Adding more power (CPU, RAM) to an existing server.
    - **Horizontal Scaling:** Adding more servers to distribute the load.
  6. **What are cloud-native applications?**  
Cloud-native apps are designed specifically for cloud environments, using microservices, containers, and CI/CD pipelines to enhance scalability and resilience.
  7. **What is serverless computing?**  
Serverless computing allows developers to run applications without managing servers. Examples include AWS Lambda and Azure Functions.
- 

## Programming Concepts in Distributed Systems

1. **What is message passing in distributed systems?**  
Message passing is a method of inter-process communication where processes exchange information using messages instead of shared memory.
2. **Explain the difference between TCP and UDP.**
  - **TCP (Transmission Control Protocol):** Reliable, connection-oriented communication with error checking.
  - **UDP (User Datagram Protocol):** Faster, connectionless communication with no guarantee of delivery.
3. **What is a stub in RPC?**  
A stub is a client-side proxy that translates local procedure calls into network requests and sends them to the server.
4. **What is marshalling and unmarshalling in distributed systems?**
  - **Marshalling:** Converts data into a format suitable for transmission over the network.
  - **Unmarshalling:** Reconstructs the original data at the receiver's end.

5. **What is a consistency model in distributed systems?**

A consistency model defines how updates to shared data are propagated. Types include strict consistency, eventual consistency, and causal consistency.

6. **Explain the concept of failover.**

Failover is an automated process where a backup system takes over when the primary system fails, ensuring high availability.

7. **What are sockets in network programming?**

Sockets are endpoints for communication between two nodes in a network, enabling data exchange over protocols like TCP or UDP.

---

## **Cloud and Distributed System Tools**

1. **What is Kubernetes?**

Kubernetes is an open-source platform for automating container orchestration, enabling deployment, scaling, and management of containerized applications.

2. **What is Docker, and why is it used?**

Docker is a containerization tool that packages applications with their dependencies, ensuring they run consistently across different environments.

3. **What is a load balancer?**

A load balancer distributes incoming network traffic across multiple servers to improve reliability and performance.

4. **What is a CDN (Content Delivery Network)?**

A CDN caches content across multiple servers worldwide, delivering it to users from the closest server to reduce latency.

5. **What is fault tolerance, and how is it achieved in cloud computing?**

Fault tolerance ensures system reliability despite failures. Techniques include redundancy, replication, and auto-scaling.