**1.**

a) Describe a scenario where a distributed computing system could significantly improve the processing time of data compared to a single processor system.

b) Discuss the challenges and design considerations that must be addressed when migrating from a centralized computing architecture to a distributed system architecture in a healthcare data analysis company.

**2.**

a) Explain how logical clocks could be implemented in a distributed system used for managing a multi-city bus reservation system.

b) Propose an efficient implementation of vector clocks in a distributed social media platform to ensure that posts and comments are synchronized in the correct causal order.

**3.**

a) Describe how snapshot recording algorithms for FIFO channels can be utilized in a distributed banking transaction system.

b) Analyze the necessary and sufficient conditions for consistent global snapshots in a distributed stock trading system, and how these conditions affect system performance during high volatility.

**4.**

a) Illustrate the use of group communication in a distributed gaming environment where players interact in real-time.

b) Design a termination detection protocol using distributed snapshots for a complex multi-agent simulation system where agents dynamically enter and exit the simulation.

**5.**

a) Discuss how Lamport’s algorithm can be applied to manage access to a distributed database system among multiple users.

b) Compare and contrast the efficiency of Raymond’s tree-based algorithm versus Suzuki-Kasami’s broadcast-based algorithm in a large-scale distributed file system.

**6.**

a) Explain how the Chandy-Misra-Haas deadlock detection algorithm for the AND model can be applied in a distributed manufacturing control system.

b) Propose a strategy for resolving deadlocks in a distributed system that controls access to multiple satellite communication channels.

**7.**

a) Describe a scenario in which Byzantine agreement is necessary in a distributed system used for inter-bank transactions.

b) Analyze the implications of achieving consensus in a distributed cryptocurrency exchange system in the presence of faulty or malicious nodes.

**8.**

a) Explain how data indexing in P2P networks can improve search efficiency in a distributed file-sharing application.

b) Discuss the design challenges and security concerns in a P2P network used for a decentralized video streaming service and propose methods to mitigate these risks.

**9.**

a) Explain how causal order of messages can be ensured in a distributed system used for online classroom sessions where messages need to reflect the sequence of lecture content delivery.

b) Design an application-level multicast protocol for a distributed event management system that handles thousands of events, ensuring total order of event-related messages with optimal efficiency.

**10.**

a) Illustrate the use of Ricart-Agrawala’s algorithm in a distributed system managing access to a limited number of electric vehicle charging stations.

b) Evaluate the performance of Maekawa’s algorithm in a distributed cloud environment where instances dynamically request and release multiple resources, discussing potential bottlenecks and improvements.

**11.**

a) Describe a scenario where Chandy-Misra-Haas deadlock detection for the OR model could be used in a distributed system coordinating multiple autonomous vehicles at an intersection.

b) Develop a comprehensive deadlock resolution strategy for a distributed airline booking system where multiple agents may hold and wait for seats on various flights simultaneously.

**12.**

a) Explain how an agreement in a failure-free system can be quickly achieved in a distributed sensor network monitoring environmental conditions across various geographic locations.

b) Design a robust consensus protocol for a distributed blockchain system that operates in a message-passing synchronous system with possible node failures, ensuring that the system maintains integrity and prevents double-spending.

**13.**

a) Discuss the role of structured overlays, specifically CHORD DHT, in improving data retrieval in a peer-to-peer file-sharing application.

b) Address the security concerns in a peer-to-peer network used for a decentralized identity verification system, proposing techniques to enhance security against sybil attacks and data tampering.

**14.**

a) Demonstrate how snapshot recording algorithms for non-FIFO channels can be implemented in a distributed multi-player gaming system where players interact and make moves in real-time.

b) Design a system model and define an algorithm for recording consistent global snapshots in a distributed cryptocurrency mining operation that must handle asynchronous and out-of-order transactions.

15.

a) Describe how Suzuki-Kasami’s broadcast-based algorithm could be used to synchronize access to a distributed ledger in a financial blockchain network.

b) Propose modifications to Raymond’s tree-based algorithm to enhance its performance and fault tolerance in a distributed content delivery network that dynamically adjusts to changes in traffic and node failures.

1.

a) A distributed computing system can drastically improve data processing times in scenarios such as climate simulation, where large data sets are divided and processed in parallel across multiple processors, reducing overall computation time compared to a single processor system which would process data sequentially. b) When migrating to a distributed system in healthcare data analysis, challenges include ensuring data consistency and security across multiple nodes, scalability to handle varying loads, and fault tolerance to prevent data loss. Design considerations must address data partitioning, network latency, and the choice of synchronous versus asynchronous communication.

a) Logical clocks in a multi-city bus reservation system could be implemented by assigning a unique, incrementally increasing timestamp to every reservation event, ensuring that the sequence of reservations is maintained accurately across different servers handling different cities. b) For a distributed social media platform, an efficient vector clock implementation would involve each node maintaining a vector of timestamps, representing the causal history of events. This ensures that posts and comments are displayed in the correct causal order, respecting the dependencies such as a comment on a post appearing after the post itself.

a) In a distributed banking transaction system, snapshot recording algorithms for FIFO channels could be used to capture a consistent state of transactions across different branches without halting the system, allowing for real-time auditing and consistency checks. b) The necessary conditions for consistent global snapshots in a distributed stock trading system include ensuring that all transactions are either fully captured or not at all in the snapshot and that snapshots are recorded in a manner that reflects the causal and temporal order of trades, especially during high market volatility which may affect the performance due to the increased frequency of snapshot recording.

a) Group communication in a distributed gaming environment facilitates real-time interaction among players by efficiently managing messages such as player movements and actions, ensuring that all players have a consistent view of the game state despite differing network conditions. b) A termination detection protocol using distributed snapshots could involve each agent in the simulation system periodically recording its state and checking if all agents are in a non-active state, effectively detecting the end of a simulation cycle, even as agents dynamically enter and exit.

a) Lamport’s algorithm could manage access to a distributed database by assigning timestamps to each transaction request, ensuring that all database operations are performed in a globally agreed order, thus maintaining data consistency and preventing conflicts. b) Raymond’s tree-based algorithm and Suzuki-Kasami’s broadcast-based algorithm both manage resource allocation in distributed systems, but Raymond's is more suited for systems with a structured network topology, enhancing efficiency and scalability, while Suzuki-Kasami, being broadcast-based, is better for environments where such a structure is absent, albeit at potentially higher communication costs.

a) In a distributed manufacturing control system, the Chandy-Misra-Haas deadlock detection algorithm for the AND model helps identify deadlock situations among interdependent manufacturing processes and resources by propagating probe messages that track resource requests and grants. b) A strategy for resolving deadlocks in a system controlling multiple satellite communication channels could involve priority-based timeouts, where lower-priority tasks are automatically aborted and restarted later, thereby freeing up resources for higher-priority tasks.

a) Byzantine agreement is crucial in inter-bank transactions where even a single node’s faulty or malicious behavior could corrupt the financial data. It ensures that all honest nodes reach consensus on transaction records despite the presence of faults. b) In a distributed cryptocurrency exchange, achieving consensus among nodes, especially in the presence of faulty or malicious ones, is essential for maintaining the integrity and security of transactions, impacting trust and reliability of the exchange under adversarial conditions.

a) Data indexing in P2P networks can improve search efficiency in file-sharing applications by allowing users to quickly locate files through a distributed hash table (DHT), which reduces the load on any single node and speeds up the search process. b) The main design challenges in a P2P network for decentralized video streaming include handling large volumes of data transfers securely and efficiently. Techniques to mitigate risks include using encrypted connections, robust authentication mechanisms, and redundant data paths to ensure data integrity and availability.

a) Ensuring the causal order of messages in a distributed online classroom can be achieved by implementing logical clocks that assign timestamps to messages based on their order of occurrence, ensuring that the flow of lecture content is consistent and logically sequenced. b) An application-level multicast protocol for a distributed event management system would involve organizing nodes in a structured overlay network, where messages about events are multicast in a total ordered manner using vector clocks, optimizing message delivery paths to reduce latency and bandwidth consumption.

a) Ricart-Agrawala’s algorithm can manage access to a limited number of electric vehicle charging stations in a distributed system by ensuring that only one vehicle accesses a charging station at a time, based on a mutual exclusion protocol that is both fair and efficient. b) Maekawa’s algorithm, being a subset of the mutual exclusion algorithms in distributed systems, might face bottlenecks due to its reliance on a fixed-size subset of coordinators for each request, leading to potential delays; enhancements could include dynamic coordinator selection to balance the load more effectively.

a) Chandy-Misra-Haas deadlock detection for the OR model could effectively manage a scenario where multiple autonomous vehicles at an intersection coordinate to pass without interference, detecting potential deadlocks dynamically as vehicles request and release control over intersecting paths. b) In a distributed airline booking system, a comprehensive deadlock resolution strategy could include the implementation of a timeout policy where agents release seats if a booking is not completed within a specified time, coupled with a priority queue based on booking time to manage seat allocations efficiently.

a) Quick agreement in a failure-free distributed sensor network can be achieved using algorithms like flooding consensus, where sensor nodes rapidly propagate their readings to reach a consensus on environmental conditions, optimizing for speed and minimal message overhead. b) A robust consensus protocol for a blockchain operating in a message-passing synchronous system might involve using modified Byzantine Fault Tolerance mechanisms to ensure that all transactions are agreed upon by a majority of nodes before being committed, enhancing security and consistency even in the face of node failures.

a) The CHORD DHT (Distributed Hash Table) plays a crucial role in peer-to-peer file-sharing applications by providing a scalable, efficient method for data retrieval where nodes can find data within a bounded number of steps, significantly reducing lookup time. b) Addressing security concerns in a decentralized identity verification system includes implementing robust mechanisms against sybil attacks, such as requiring cryptographic proofs of identity or stakes, and employing data integrity checks to prevent tampering.

a) Snapshot recording algorithms for non-FIFO channels in a multiplayer gaming system can be implemented by having each player's game client maintain a local state vector that records the game events chronologically and communicates with other clients to synchronize game state in real-time. b) A system model for recording consistent global snapshots in a distributed cryptocurrency mining operation would involve asynchronously capturing snapshots of transactions at various nodes, with algorithms ensuring that all recorded transactions are consistent across the network, even when nodes receive transactions out of order.

a) Suzuki-Kasami’s broadcast-based algorithm could synchronize access to a distributed ledger in a financial blockchain network by allowing each node to broadcast requests and receive permissions from all other nodes, ensuring that all transactions are globally ordered and conflict-free. b) Enhancements to Raymond’s tree-based algorithm in a distributed content delivery network could include dynamic adjustment of the tree structure in response to node failures or traffic changes, improving fault tolerance and response time by localizing the impact of changes.