CS8603 DISTRIBUTED SYSTEMS

IMPORTANT QUESTIONS AND QUESTION BANK

UNIT I – INTRODUCTION

2-Marks

1. Define distributed systems?
2. What is middleware?
3. Illustrate coupling?
4. Describe parallelism/concurrency in distributed systems?
5. Compare centralized and distributed system?
6. State the various classes of multiprocessor/multicomputer operating system?
7. Compare shared memory Vs distributed shared memory?
8. Compose different forms of load balancing?
9. Analyze briefly on ubiquitous computing?
10. Illustrate five reasons why to build distributed System?

Part-B

1. Design in detail about application domains where distributed system is applied?
2. Write a brief note on the key algorithmic challenges in distributed computing?
3. Discuss the primitives for distributed communication?
4. Explain about the synchronous versus asynchronous executions in a message-passing system with examples?
5. Identify and explain the basic properties of scalar time?
6. Summarize NTP for synchronizing system of physical clocks in distributed system?
7. Point out in detail the recent trends in distributed Systems? What are the design issues to be considered in designing distributed system? Explain in detail about each of them?
8. List and explain the basic properties of vector time?
9. A user arrives at a railway station that she has never visited before, carrying a PDA that is capable of wireless networking. Suggest how the user could be provided with information about the local services and amenities at that station, without entering the station's name or attributes. What technical challenges must be overcome? Discuss in detail?
10. What are the processing modes of flynn taxonomy? Examine various MIMD architectures in terms of coupling?
11. Design the requirements and aspects needed for reliable and fault tolerant distributed systems?

12. Show that all events on the surface of the past cone of an event are message send events. Likewise, show that all events on the surface of the future cone of an event are message receive events?
13. Draw the omega and butterfly networks for n = 16 inputs and outputs? Elaborate the functions need to address while designing a distributed computing system?
14. Write a brief note on the key algorithmic challenges in distributed computing?
15. Explain the practical applicability of the load –balancing approach a scheduling scheme for the following types of distributed systems:
    a) A LAN-based distributed system
    b) A WAN-based distributed system
    c) A distributed system based on the processor-pool model.
    d) A distributed system based on the workstation-server model

UNIT II - MESSAGE ORDERING & SNAPSHOTS

2-Marks

1. What are the message ordering paradigms?
2. Compare closed group Vs open group algorithm?
3. State crown criterion theorem?
4. Explain message broadcast?
5. Define time stamp?
6. Discuss multiway rendezvous and binary rendezvous?
7. What are the characteristics of multicast communication?
8. Differentiate multicasting Vs unicasting?
9. Evaluate the criteria that must be met by a causal ordering protocol?
10. What are the necessary conditions to satisfy the consistent global state?

Part -B

1. Design FIFO and non-FIFO executions? Discuss the causally ordered execution?
2. Show with an equivalent timing diagram of a synchronous execution on an asynchronous system?
3. Show with an equivalent timing diagram of a asynchronous execution on a synchronous system?
4. Illustrate realizable with synchronous communication (RSC) execution?
5. Explain the hierarchy of execution classes? Examine the crown test to determine the existence of cyclic dependencies among messages?
6. Explain the channels to simulate an execution using asynchronous primitives on a synchronous system?

7. Analyse the channels to simulate an execution using synchronous primitives on an asynchronous system?
8. Explain a simple algorithm defined by Bagrodia?
9. Explain chandy and lamport algorithm?
10. Examine the two possible executions of the snapshot algorithm for money transfer?
11. Examine the necessary and sufficient conditions for causal ordering/
12. Analyze in detail about the centralized algorithm to implement total order and causal order of messages?
13. Discuss in detail about the distributed algorithm to implement total order and causal order of messages?
14. Create a simplified implementation of synchronous order. Develop the for the process Pi ,1 ≤ i≤ n?
15. Consider a distributed system where every node has its physical clock and all physical clocks are perfectly synchronized. Develop an algorithm to record global state assuming the communication network is reliable?

UNIT III DISTRIBUTED MUTEX & DEADLOCK

2-Marks

1. What are the three basic approaches for implementing distributed mutual exclusion?
2. Explain idle token?
3. Discuss the conditions for maekawa's algorithm?
4. List the three types of messages for Deadlock handling?
5. What is deadlock?
6. Define the two design issues for suzuki–kasami's?
7. How ricart–agrawala algorithm achieves mutual exclusion?
8. Explain maekawa's algorithm achieves mutual exclusion?
9. Express in diagram the wait for graph (WFG)?
10. Develop the facts of global state detection-based deadlock detection?

Part-B

1. Explain about the Lamport distributed mutual exclusion algorithm?
2. Illustrate with a case study explain ricart–agrawala algorithm?
3. Discuss the issues in deadlock detection? state example of a WFG?
4. Examine suzuki–kasami's broadcast algorithm?
5. What is deadlock? explain the models of deadlocks?
6. Formulate the mitchell and merritt's algorithm for the single resource model?

7. Describe the distributed deadlock detection algorithms in detail?
8. Conclude in brief about knapp's classification of distributed deadlock detection algorithm?
9. Briefly describe about the chandy–misra–haas algorithm for the AND model?
10. Express with neat sketch and explain chandy–misra–haas algorithm for the OR model?
11. Show that in the ricart–agrawala algorithm the critical section is accessed in increasing order of timestamp. Does the same hold in maekawa's algorithm?
12. What is the purpose of a REPLY message in lamport's algorithm? Note that it is not necessary that a site must always return a REPLY message in response to a REQUEST message. State the condition under which a site does not have to return REPLY message. Also, give the new message complexity per critical section execution in this case?
13. Suppose all the processes in the system are assigned priorities which can be used to totally order the processes. Modify chand y*et al.*'s algorithm for the AND model so that when a process detects a deadlock, it also knows the lowest priority deadlocked process?
14. Consider the following simple approach to handle deadlocks in distributed systems by using "time-outs": a process that has waited for a specified period for a resource declares that it is deadlocked and aborts to resolve the deadlock. Explain what are the shortcomings of using this method?

UNIT IV RECOVERY & CONSENSUS

2-Marks

1. Describe local check pointing?
2. What is meant by "outside world process (OWP)."?
3. Point out the phases of min-process check pointing algorithms?
4. Define rollback recovery?
5. Compare coordinated check pointing versus uncoordinated check pointing?
6. List the categories of checkpoint-based rollback-recovery techniques?
7. Give the use of piggybacking?
8. Mulate the different types of messages?
9. Point out the phases of min-process check pointing algorithms?
10. Explain the two types of communication-induced check pointing?

Part-B

1. What is rollback? and explain the several types of messages for rollback.?
2. Examine briefly about global states with examples?
3. Describe the issues involved in a failure recovery with the help of a distributed computation?
4. Elaborate the various checkpoint-based rollback-recovery techniques?
5. Describe the pessimistic logging, optimistic logging and casual logging?
6. Demonstrate in detail about the juang–venkatesan algorithm for asynchronous check pointing and recovery?
7. Discuss in detail about some assumptions underlying the study of agreement algorithms?
8. Explain agreement in (message-passing) synchronous systems with failures?
9. Give byzantine agreement tree algorithm and illustrate with an example?
10. Analyze on phase-king algorithm for consensus?
11. Design a system model of distributed system consisting of four processes and explain the interactions with the outside world?
12. Explain with examples of consistent and inconsistent states of a distributed system?
13. Consider the following simple check pointing algorithm. A process takes a local checkpoint right after sending a message. Create that the last checkpoint at all processes will always be consistent. What are the trade-offs with this method?
14. Give and analyse a rigorous proof of the impossibility of a min process, nonblocking check pointing algorithm?

UNIT V - P2P & DISTRIBUTED SHARED MEMORY

2-Marks

1. Define churn?
2. Classify the characteristics of peer to peer system?
3. What are the performance features of P2P systems?
4. List the P2P overlay and its types?
5. What are the two steps involved in chord protocol?
6. List the three core components of a CAN design?
7. Analyze the three basic operations which supports CAN?
8. Measure the properties of weak consistency?
9. Discuss the two instructions to perform hardware support for mutual exclusion?
10. Show how to provide barrier synchronization in release consistency?

Part-B

1. Explain the structured overlays and unstructured overlays in distributed indexing?
2. Examine the chord protocol with simple key lookup algorithm?
3. Illustrate in detail about A scalable object location algorithm in chord?
4. Discuss on managing churn in chord?
5. Point out tapestry P2P overlay network and its routing with an example?

6. Discuss the CAN maintenance and CAN optimizations?
7. State about the consistency models: entry consistency, weak consistency, and release consistency?
8. Summarize in detail how node insertion and node deletion are applied in tapestry?
9. Describe lamport's bakery algorithm lamport's WRWR mechanism and fast mutual exclusion?
10. User 'A' in delhi wishes to send a file for printing to user 'B' in florida, whose system is connected to a printer; while user 'C' from tokyo wants to save a video file in the hard disk of user 'D' in london. Analyze and discuss the required peer-to-peer network architecture?
11. Evaluate a formal proof to justify the correctness of algorithm that implements sequential consistency using local read operations?
12. Develop a detailed implementation of causal consistency, and provide a correctness argument for your implementation?
13. Examine the steps for the query: lookup (K8) initiated at node 28, are shown in Figure for simple key lookup algorithm