CST402	DISTRIBUTED COMPUTING	CATEGORY	L	T	P	CREDIT	YEAR OF INTRODUCTION
		PCC	2	1	0	3	2019

Preamble: The purpose of this course is to understand the system models, algorithms and protocols that allow computers to communicate and coordinate their actions to solve a problem. This course helps the learner to understand the distributed computation model and various concepts like global state, termination detection, mutual exclusion, deadlock detection, shared memory, failure recovery, consensus, file system. It helps the learners to develop solutions to problems in distributed computing environment.

Prerequisite: Basic knowledge in data structures and operating systems.

Course Outcomes: After the completion of the course the student will be able to

CO1	Summarize various aspects of distributed computation model and logical time. **(Cognitive Knowledge Level: Understand)**
CO2	Illustrate election algorithm, global snapshot algorithm and termination detection algorithm. **(Cognitive Knowledge Level: Apply)**
CO3	Compare token based, non-token based and quorum based mutual exclusion algorithms. **(Cognitive Knowledge Level: Understand)**
CO4	Recognize the significance of deadlock detection and shared memory in distributed systems. **(Cognitive Knowledge Level: Understand)**
CO5	Explain the concepts of failure recovery and consensus. **(Cognitive Knowledge Level: Understand)**
CO6	Illustrate distributed file system architectures. **(Cognitive Knowledge Level: Understand)**

Mapping of course outcomes with program outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	✔	✔	✔									✔
CO2	✔	✔	✔	✔								✔
CO3	✔	✔	✔									✔
CO4	✔	✔	✔									✔
CO5	✔	✔	✔									✔
CO6	✔	✔	✔									✔

Abstract POs defined by National Board of Accreditation			
PO#	**Broad PO**	**PO#**	**Broad PO**
PO1	Engineering Knowledge	**PO7**	Environment and Sustainability
PO2	Problem Analysis	**PO8**	Ethics
PO3	Design/Development of solutions	**PO9**	Individual and team work
PO4	Conduct investigations of complex problems	**PO10**	Communication
PO5	Modern tool usage	**PO11**	Project Management and Finance
PO6	The Engineer and Society	**PO12**	Life long learning

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination Marks (%)
	Test 1 (%)	Test 2 (%)	
Remember	30	30	30
Understand	50	50	50
Apply	20	20	20
Analyze			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3

Continuous Internal Evaluation Pattern:

Attendance	**10 marks**
Continuous Assessment Tests(Average of Internal Tests1&2)	**25 marks**
Continuous Assessment Assignment	**15 marks**

Internal Examination Pattern

Each of the two internal examinations has to be conducted out of 50 marks. First series test shall be preferably conducted after completing the first half of the syllabus and the second series test shall be preferably conducted after completing remaining part of the syllabus. There will be two parts: Part A and Part B. Part A contains 5 questions (preferably, 2 questions each from the completed modules and 1 question from the partly completed module), having 3 marks for each question adding up to 15 marks for part A. Students should answer all questions from Part A. Part B contains 7 questions (preferably, 3 questions each from the completed modules and 1 question from the partly completed module), each with 7 marks. Out of the 7 questions, a student should answer any 5.

End Semester Examination Pattern:

There will be two parts; Part A and Part B. Part A contains 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 full questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carries 14 marks.

Syllabus

Module – 1 (Distributed systems basics and Computation model)

Distributed System – Definition, Relation to computer system components, Motivation, Primitives for distributed communication, Design issues, Challenges and applications. A model of distributed computations – Distributed program, Model of distributed executions, Models of communication networks, Global state of a distributed system, Cuts of a distributed computation, Past and future cones of an event, Models of process communications.

Module – 2 (Election algorithm, Global state and Termination detection)

Logical time – A framework for a system of logical clocks, Scalar time, Vector time. Leader election algorithm – Bully algorithm, Ring algorithm. Global state and snapshot recording algorithms – System model and definitions, Snapshot algorithm for FIFO channels – Chandy Lamport algorithm. Termination detection – System model of a distributed computation, Termination detection using distributed snapshots, Termination detection by weight throwing,Spanning-tree-based algorithm.

Module – 3 (Mutual exclusion and Deadlock detection)

Distributed mutual exclusion algorithms – System model, Requirements of mutual exclusion algorithm. Lamport's algorithm, Ricart–Agrawala algorithm, Quorum-based mutual exclusion algorithms – Maekawa's algorithm. Token-based algorithm – Suzuki–Kasami's broadcast algorithm. Deadlock detection in distributed systems – System model, Deadlock handling strategies, Issues in deadlock detection, Models of deadlocks.

Module – 4 (Distributed shared memory and Failure recovery)

Distributed shared memory – Abstraction and advantages. Shared memory mutual exclusion – Lamport's bakery algorithm. Check pointing and rollback recovery – System model, consistent and inconsistent states, different types of messages, Issues in failure recovery, checkpoint based recovery, log based roll back recovery.

Module – 5 (Consensus and Distributed file system)

Consensus and agreement algorithms – Assumptions, The Byzantine agreement and other problems, Agreement in (message-passing) synchronous systems with failures – Consensus algorithm for crash failures. Distributed file system – File service architecture, Case studies: Sun Network File System, Andrew File System, Google File System.

(Note: Proof of correctness and performance analysis are not expected for any of the algorithms in the syllabus).

Text Books

1. Ajay D. Kshemkalyani and Mukesh Singhal, Distributed Computing: Principles, Algorithms, and Systems, Cambridge University Press, 2011.

Reference Books

1. George Coulouris, Jean Dollimore, Tim Kindberg and Gordon Blair. Distributed Systems: Concepts and Design, Addison Wesley, Fifth edition.
2. Kai Hwang, Geoffrey C Fox, Jack J Dongarra, Distributed and Cloud Computing – From Parallel Processing to the Internet of Things, Morgan Kaufmann Publishers, 2012.
3. Sukumar Ghosh, Distributed Systems: An Algorithmic Approach, CRC Press, Second edition, 2015.
4. Maarten Van Steen, Andrew S. Tanenbaum, Distributed Systems, Prentice Hall of India,Third edition, 2017.
5. Randy Chow and Theodore Johnson, Distributed Operating Systems and Algorithm Analysis, Pearson Education India, First edition, 2009.
6. Valmir C. Barbosa, An Introduction to Distributed Algorithms, MIT Press, 2003.

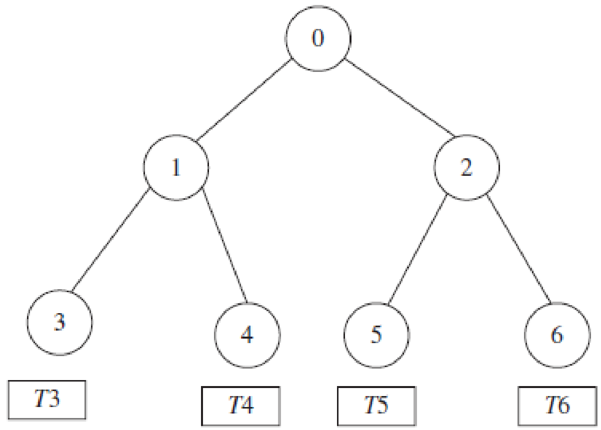
Course Level Assessment Questions

Course Outcome1 (CO1):

1. Define logical clock and explain the implementation of the logical clock.
2. Explain different forms of load balancing.

Course Outcome 2(CO2):

1. Apply ring-based leader election algorithm with 10 processes in the worst-performing case. Count the number of messages needed.
2. Apply spanning tree-based termination detection algorithm in the following scenario. The nodes are processes 0 to 6. Leaf nodes 3, 4, 5, and 6 are each given tokens T3, T4, T5 and T6 respectively. Leaf nodes 3, 4, 5 and 6 terminate in the order, but before terminating node 5,it sends a message to node 1.



Course Outcome 3(CO3):

1. What are the requirements of mutual exclusion algorithms?
2. Illustrate Suzuki- Kasami's broadcast algorithm.

Course Outcome 4(CO4):

1. Compare different models of deadlocks.
2. Illustrate the detailed abstraction of distributed shared memory and interaction with application processes.

Course Outcome 5(CO5):

1. Explain how consensus problem differs from the Byzantine agreement problem.
2. Classify different log based roll back recovery techniques.

Course Outcome 6 (CO6):

1. Explain the directory service and its interface operations in a file service architecture.
2. Describe the architecture of Google file system.

Model Question Paper

QP CODE:

Reg No: _______________

Name: _________________ **PAGES : 4**

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

EIGHTH SEMESTER B.TECH DEGREE EXAMINATION, MONTH & YEAR

Course Code: CST402

Course Name: Distributed Computing

Max.Marks:100 **Duration: 3 Hours**

PART A

Answer All Questions. Each Question Carries 3 Marks

1. Identify any three distributed applications and for each application, determine which all motivating factors are important for building an application over a distributed system.
2. Assume that the surface of the past cone form a consistent cut. Does it mean that all events on the surface of the past cone are always concurrent? Demonstrate with the help of an example.
3. Specify the issues in recording a global state.
4. Explain the rules used to update clocks in scalar time representation.
5. Describe how quorum-based mutual exclusion algorithms differ from the other categories of mutual exclusion algorithms.
6. Explain with example, how wait-for-graphs can be used in deadlock detection.
7. List any three advantages of using distributed shared memory.
8. Define the no-orphans consistency condition.

9. Define Byzantine agreement problem.

10. Differentiate between whole file serving and whole file caching in Andrew file system **(10x3=30)**

Part B

(Answer any one question from each module. Each question carries 14 Marks)

11. (a) Explain the three different models of the service provided by communication networks. **(6)**

(b) Explain how the causal dependency between events in distributed execution is defined using Lamport's happened before relationship. **(8)**

OR

12. (a) Address the various strategies that can be adopted to satisfy the requirements of a reliable and fault tolerant distributed system. **(6)**

(b) Which are the different versions of send and receive primitives for distributed communication? Explain. **(8)**

13. (a) Illustrate bully algorithm for electing a new leader. Does the algorithm meet liveness and safety conditions? **(7)**

(b) Clearly mentioning assumptions, explain the rules of termination detection using distributed snapshots. **(7)**

OR

14. (a) In Chandy-Lamport algorithm for recording global snapshots, explain how the recorded local snapshots can be put together to create the global snapshot. Can multiple processes initiate the algorithm concurrently? **(6)**

(b) Illustrate the working of spanning tree based termination detection algorithm. **(8)**

15. (a) Explain and illustrate Lamport's mutual exclusion algorithm. **(8)**

(b) Discuss the three types of messages required for deadlock handling in Maekawa's algorithm. Explain how Maekawa's algorithm handles deadlocks. **(6)**

OR

16. (a) Explain and illustrate Ricart- Agrawala algorithm for achieving mutual exclusion. **(8)**

(b) Explain any three different models of deadlock. **(6)**

17. (a) What are the issues in failure recovery? Illustrate with suitable examples. **(7)**

(b) Show that Lamport's Bakery algorithm for shared memory mutual exclusion, satisfy the three requirements of critical section problem. **(7)**

OR

18. (a) Differentiate consistent and inconsistent states with examples. **(4)**

(b) What is check point-based rollback-recovery? Explain the three classifications of check point-based rollback-recovery. **(10)**

19. (a) Explain consensus algorithm for crash failures under synchronous systems. **(6)**

(b) Summarize distributed file system requirements. **(8)**

OR

20. (a) Differentiate Andrew file system and NFS. **(4)**

(b) Explain Sun NFS architecture with diagram. **(10)**

Teaching Plan

No	Contents	No. of Lecture Hours (35 hours)
Module – 1(Distributed systems basics and Computation model) (7 hours)		
1.1	Distributed System – Definition, Relation to computer system components	1 hour
1.2	Primitives for distributed communication.	1 hour
1.3	Design issues, challenges and applications.	1 hour
1.4	Design issues, challenges and applications.	1 hour
1.5	A model of distributed computations – Distributed program, Model of distributed executions	1 hour
1.6	Models of communication networks, Global state of a distributed system, Cuts of a distributed computation	1 hour
1.7	Cuts of a distributed computation, Past and future cones of an event, Models of process communications.	1 hour
Module – 2 (Election algorithm, Global state and Termination detection) (8 hours)		
2.1	Logical time – A framework for a system of logical clocks, Scalar time	1 hour
2.2	Vector time.	1 hour
2.3*	Leader election algorithm – Bully Algorithm, Ring Algorithm	1 hour
2.4	Global state and snapshot recording algorithms – System model and definitions	1 hour
2.5*	Snapshot algorithm for FIFO channels – Chandy Lamport algorithm.	1 hour
2.6	Termination detection – System model of a distributed computation	1 hour
2.7*	Termination detection using distributed snapshots	1 hour
2.8*	Termination detection by weight throwing, Spanning tree-based algorithm	1 hour
Module – 3 (Mutual exclusion and Deadlock detection) (6 hours)		
3.1*	Distributed mutual exclusion algorithms – System model, Lamport's algorithm	1 hour

3.2*	Ricart–Agrawala algorithm	1 hour
3.3*	Quorum-based mutual exclusion algorithms – Maekawa's algorithm	1 hour
3.4*	Token-based algorithm – Suzuki–Kasami's broadcast algorithm.	1 hour
3.5	Deadlock detection in distributed systems – System model, Deadlock handling strategies, Issues in deadlock detection	1 hour
3.6	Models of deadlocks	1 hour
Module – 4 (Distributed shared memory and Failure recovery) (7 hours)		
4.1	Distributed shared memory – Abstraction and advantages.	1 hour
4.2*	Shared memory mutual exclusion – Lamport's bakery algorithm.	1 hour
4.3	Checkpointing and rollback recovery – System model, consistent and inconsistent states	1 hour
4.4	different types of messages, Issues in failure recovery	1 hour
4.5	checkpoint based recovery	1 hour
4.6	log based roll back recovery.	1 hour
4.7	log based roll back recovery.	1 hour
Module – 5(Consensus and Distributedfile system) (7 hours)		
5.1	Consensus and agreement algorithms – Assumptions, The Byzantine agreement and other problems	1 hour
5.2	Agreement in (message-passing) synchronous systems with failures – Consensus algorithm for crash failures	1 hour
5.3*	Agreement in (message-passing) synchronous systems with failures – Consensus algorithm for crash failures	1 hour
5.4	Distributed File System – File Service Architecture	1 hour
5.5	Case Studies: Sun Network File System	1 hour
5.6	Andrew File System	1 hour
5.7	Google File System.	1 hour

* **Proof of correctness and performance analysis are not expected for this algorithm**.