

**BIRLA INSTITUTE OF TECHNOLOGY & SCIENCE, PILANI**

**WORK INTEGRATED LEARNING PROGRAMMES**

**COURSE HANDOUT**

**Part A: Content Design**

|  |  |
| --- | --- |
| **Course Title** | DISTRIBUTED COMPUTING |
| **Course ID No.** | SS ZG526 |
| **Credit Units** | 5 |
| **Authors** | Dr. CHITTRANJAN HOTA, LUCY J GUDINO (Updated by Anindya Neogi) |
| **Version No** | 2.1 |
| **Date** |  |

**Course Objectives**

|  |  |
| --- | --- |
| **No** | **Course Objective** |
| **CO1** | To learn hardware architectures for building distributed systems, and their communication models. |
| **CO2** | To learn the design aspects of various software applications that can be deployed on various distributed systems. |
| **CO3** | To provide an understanding of the complexities and resource management issues that are critical in a large distributed system. |
| **CO4** | To provide algorithmic aspects of building/designing distributed systems in domains like IoT, P2P, Cluster, Grid computing etc. |

**Text Book(s)**

|  |  |
| --- | --- |
| T1 | Ajay D. Kshemkalyani, and Mukesh Singhal “Distributed Computing: Principles, Algorithms, and Systems”, Cambridge University Press, 2008 (Reprint 2013). |

**Reference Book(s) & other resources**

|  |  |
| --- | --- |
| R1 | John F. Buford, Heather Yu, and Eng K. Lua, “P2P Networking and Applications”, Morgan Kaufmann, 2009 Elsevier Inc. |
| R2 | Kai Hwang, Geoffrey C. Fox, and Jack J. Dongarra, “Distributed and Cloud Computing: From Parallel processing to the Internet of Things”, Morgan Kaufmann, 2012 Elsevier Inc. |
| R3 | A. Tanenbaum and M. V. Steen, “Distributed Systems”, 2nd Edition, Pearson Prentice Hall |
| R4 | MongoDB |
| R5 | Cassandra |
| R6 | CouchDB |
| R7 | Dynamo |
| R8 | Paxos |

**Content Structure**

|  |  |  |
| --- | --- | --- |
| Module No | Module Title | Objectives |
| 1 | **Introduction to Distributed Computing** | This module will give an introduction to Distributed computing in terms of various hardware and software models that can be used to build a distributed computing system. It will also cover communication models appropriate for building these systems. Design issues and Challenges for building such systems will also be discussed. |
| 2 | **Logical Clocks & Vector Clocks** | To explore alternate ways of managing time (clock) in a distributed computing system as there is no centralized hardware clock available in this type of system. To study ways of building systems that can provide a global notion of time. |
| 3 | **Global state and snapshot recording algorithms** | Recording of a global state for the purpose of carrying out distributed computations is a complex task because of non-availability of Single physical memory and single physical clock in a distributed system. So, designers have to explore ways or algorithms to record or collect global state of a distributed computation. |
| 4 | **Terminology and Basic algorithms** | To study a framework in which distributed algorithms can be classified and analysed. Basic distributed graph algorithms, synchronizers, and practical graph problem are studied and analysed to give a better understanding of distributed algorithm design. |
| 5 | **Message ordering and Termination detection** | In a distributed system, message ordering plays an important role on making the system consistent or stable. The order of messages sent should match (happen prior) the order of its’ receipt in a group communication application implemented in a distributed system. This module will discuss ways to achieve various message ordering schemes. Also, the approaches for detecting termination of a distributed computation will be discussed. |
| 6 | **Distributed Mutual Exclusion & Deadlock detection** | In a distributed system when multiple entities compete for a shared resource, the access to this shared resource has to be serialized (or coordinated). This module will cover different assertion based, and tree based distributed algorithms to implement DME. Deadlocks are very common in distributed computing systems, and this module will discuss ways to detect this deadlock and resolve them. |
| 7 | **Consensus and Agreement Algorithms** | When multiple entities cooperate with each other in solving a complex function or task in a distributed system, there are instances where a majority of these entities must agree on certain decisions without which the task cannot be solved. This module will discuss ways to achieve these. |
| 8 | **Peer-to-Peer computing and Overlay graphs** | Overlay networks have been playing a dominant role in providing better services since past two decades. Peer-to-Peer computing has emerged as an important distributed application (network) in ensuring a wide variety of improved services be it in voice (Skype), file sharing (BitTorrent), digital currency (BitCoin), or Anonymous surfing (Tor) etc. This module will cover different aspects of P2P Overlay application development. |
| 9 | **Cluster Computing** | Cluster computing is also a model of carrying out distributed computation using a set of homogeneous systems or machines. This module will deal with the design aspects of Cluster computing. |
| 10 | **Consistency and replication** | Distributed systems need to provide a view of data consistency at a global level or specific to clients connecting to the system. This module will discuss replication, various types of consistency and protocols. Practical examples will be discussed through case studies of modern distributed databases. |

**Learning Outcomes:**

|  |  |
| --- | --- |
| **No** | **Learning Outcomes** |
| **LO1** | Students will be able to describe middleware platforms like RPC(Sun RPC, Java RMI, etc) for implementing communication models over distributed systems. |
| **LO2** | Students will be able to appraise/justify the need for Logical clocks and their usages in building distributed systems and its’ components. |
| **LO3** | Students will be able to explain Mutual exclusion primitives, Agreement protocols, Consistency and Deadlock handling scenarios in distributed systems. |
| **LO4** | Students will be able to describe search, storage, communication, efficiency and other related issues in paradigms like P2P, Cluster, Distributed Databases. |

**Part B: Contact Session Plan**

|  |  |
| --- | --- |
| **Academic Term** | First Semester 2024-2025 |
| **Course Title** | DISTRIBUTED COMPUTING |
| **Course No** | SS ZG526 |
| **Lead Instructor** | K.SRINIVASA RAO |

**Course Contents**

|  |  |  |
| --- | --- | --- |
| **Contact Sessions(#)** | **List of Topic Title**  **(from content structure in Course Handout)** | **Text/Ref Book/external resource** |
| 1 | **M1: Introduction to Distributed Computing** Motivation, Multiprocessor Vs Multicomputer Systems.  Distributed Communication models: Remote Procedure Call, Publish/Subscribe model, Message Queues etc.  Design issues and Challenges for building distributed computing systems. | T1 (Chap.1) |
| 2 | **M2: Logical Clocks & Vector clocks**  A framework for a system of logical clocks.  Scalar time, Vector time.  Implementation of Logical and Vector clocks, Efficient implementation of Vector clocks.  Physical Clock synchronization: NTP | T1 (Chap.3) |
| 3 | **M3: Global state and snapshot recording algorithms**  System model and definitions  Snapshot recording algorithms for FIFO channels  Snapshot recording algorithms for non-FIFO channels  Necessary and sufficient conditions for consistent global snapshots. | T1 (Chap.4) |
| 4 | **M4: Terminology and Basic algorithms**  Classifications and basic concepts  Elementary graph algorithms, Synchronizers.  Maximal Independent set, and Connected dominating set. | T1 (Chap.5) |
| 5 | **M5: Message ordering and Termination detection**  Message ordering paradigms  Group Communication  Protocols for ensuring Causal order of messages | T1 (Chap.6) |
| 6 | **M5: Message ordering and Termination detection**  Total order  Application level multicast  Termination detection using distributed snapshots  Termination detection using weight throwing  A spanning-tree based termination detection algorithm | T1 (Chap.6), T1 (Chap.7) |
| 7 | **M6: Distributed Mutual Exclusion & Deadlock detection**  Introduction and Preliminaries  Assertion based: Lamport’s algorithm, and Ricart-Agrawala’s algorithm  Assertion based: Maekawa’s algorithm | T1 (Chap.9) |
| 8 | **M6: Distributed Mutual Exclusion & Deadlock detection**  Token based: Suzuki-Kasami’s broadcast based algorithm  Token based: Raymond’s tree based algorithm  Models of distributed deadlock  Chandy-Misra-Haas deadlock detection for AND model  Chandy-Misra-Haas deadlock detection for OR model  Deadlock resolution | T1 (Chap.9), T1 (Chap.10) |
| 9 | **M7: Consensus and Agreement Algorithms**  Problem definition  The Byzantine agreement and other consensus problems  Overview of Results  Agreement in failure-free system (synchronous or asynchronous)  Agreement in (message-passing) synchronous systems with failures | T1 (Chap.14) |
| 10 | **M8: Peer-to-Peer computing and Overlay graphs**  Introduction  Data indexing and Overlays  Unstructured Overlays  Structured Overlays: CHORD DHT  Design issues of P2P overlays | T1 (Chap.18) |
| 11 | **M8: Peer-to-Peer computing and Overlay graphs**  Graph structure of Complex networks  Internet Graphs  Generalized Random graph networks  Small-world and Scale-free networks | T1 (Chap.18) |
| 12 | **M8: Peer-to-Peer computing and Overlay graphs**  Security concerns from P2P networks  Mitigating security risks in P2P networks | R1 (Chap.14) |
| 13 | **M9: Cluster Computing**  Cluster development trends  Design objectives of Computer clusters  Cluster organization and resource sharing  Node architecture and MPP packaging | R2 (Chap.2) |
| 14 | **M9: Cluster Computing**  Cluster system interconnects  Hardware, software and Middle ware support  GPU Clusters for massive parallelism  Cluster job and resource management | R2 (Chap.2) |
| 15 | **M10: Consistency in Distributed Systems**  CAP theorem and implications in modern distributed systems  Data centric consistency  Client centric consistency  Replica management  Consistency protocols  Handling failures: Paxos algorithm | R3 (Chap.7), R8 |
| 16 | **M10: Case studies – Distributed Databases**  MongoDB, Cassandra, CouchDB, Dynamo architecture | R4, R5, R6, R7 |

**Evaluation Scheme**:

Legend: EC = Evaluation Component; AN = After Noon Session; FN = Fore Noon Session

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Name** | **Type** | **Duration** | **Weight** | **Day, Date, Session, Time** |
| **EC-1** | **Quiz-1** | | **\*** | **5%** | September 1-10, 2024 |
| **Quiz-2** | | **\*** | **5%** | October 10-20, 2024 |
| **Assignment** | | **\*** | **10%** | November 1-10, 2024 |
| EC-2 | Mid-Semester Test | Closed Book | 2 hours | 30% | Saturday, 21/09/2024 (AN) |
| EC-3 | Comprehensive Exam | Open Book | 2 ½ hours | 50% | Saturday, 30/11/2024 (AN) |

**Important Information**

Syllabus for Mid-Semester Test (Closed Book): Topics in Weeks 1-7

Syllabus for Comprehensive Exam (Open Book): All topics given in plan of study

Evaluation Guidelines:

1. EC-1 consists of either two Assignments or three Quizzes. Announcements regarding the same will be made in a timely manner.
2. For Closed Book tests: No books or reference material of any kind will be permitted. Laptops/Mobiles of any kind are not allowed. Exchange of any material is not allowed.
3. For Open Book exams: Use of prescribed and reference text books, in original (not photocopies) is permitted. Class notes/slides as reference material in filed or bound form is permitted. However, loose sheets of paper will not be allowed. Use of calculators is permitted in all exams. Laptops/Mobiles of any kind are not allowed. Exchange of any material is not allowed.
4. If a student is unable to appear for the Regular Test/Exam due to genuine exigencies, the student should follow the procedure to apply for the Make-Up Test/Exam. The genuineness of the reason for absence in the Regular Exam shall be assessed prior to giving permission to appear for the Make-up Exam. Make-Up Test/Exam will be conducted only at selected exam centres on the dates to be announced later.

It shall be the responsibility of the individual student to be regular in maintaining the self-study schedule as given in the course handout, attend the lectures, and take all the prescribed evaluation components such as Assignment/Quiz, Mid-Semester Test and Comprehensive Exam according to the evaluation scheme provided in the handout.