

**AUTOMATA THEORY AND COMPUTABILITY**  
(Effective from the academic year 2018 -2019)

**SEMESTER – V**

<b>Subject Code</b>	18CS54	<b>CIE Marks</b>	40
<b>Number of Contact Hours/Week</b>	3:0:0	<b>SEE Marks</b>	60
<b>Total Number of Contact Hours</b>	40	<b>Exam Hours</b>	3 Hrs

**CREDITS –3**

**Course Learning Objectives:** This course will enable students to:

- Identify different Formal language Classes and their Relationships
- Design Grammars and Recognizers for different formal languages
- Prove or disprove theorems in automata theory using their properties
- Determine the decidability and intractability of Computational problems

**Module 1**

	<b>Contact Hours</b>
<b>Why study the Theory of Computation, Languages and Strings:</b> Strings, Languages. A Language Hierarchy, Computation, <del>Finite State machines (FSM)</del> ; Deterministic FSM, Regular languages, Designing FSM, Nondeterministic FSMs, From FSMs to Operational Systems, Simulators for FSMs, Minimizing FSMs, Canonical form of Regular languages, Finite State Transducers, Bidirectional Transducers. <b>Textbook 1: Ch 1,2, 3,4, 5.1 to 5.10</b> <b>RBT: L1, L2</b>	08

**Module 2**

<b>Regular Expressions (RE):</b> what is a RE?, Kleene's theorem. Applications of REs. Manipulating and Simplifying REs. Regular Grammars: Definition, Regular Grammars and Regular languages. Regular Languages (RL) and Non-regular Languages: How many RLs, To show that a language is regular, Closure properties of RLs, to show some languages are not RLs. <b>Textbook 1: Ch 6, 7, 8: 6.1 to 6.4, 7.1, 7.2, 8.1 to 8.4</b> <b>RBT: L1, L2, L3</b>	08
--	----

**Module 3**

<b>Context-Free Grammars(CFG):</b> Introduction to Rewrite Systems and Grammars. CFGs and languages, designing CFGs, simplifying CFGs, proving that a Grammar is correct, Derivation and Parse trees, Ambiguity, Normal Forms. Pushdown Automata (PDA): Definition of non-deterministic PDA, Deterministic and Non-deterministic PDAs, Non-determinism and Halting, alternative equivalent definitions of a PDA, alternatives that are not equivalent to PDA. <b>Textbook 1: Ch 11, 12: 11.1 to 11.8, 12.1, 12.2, 12.4, 12.5, 12.6</b> <b>RBT: L1, L2, L3</b>	08
---	----

**Module 4**

<b>Algorithms and Decision Procedures for CFLs:</b> Decidable questions, Un-decidable questions. <b>Turing Machine:</b> Turing machine model, Representation, Language acceptability by TM, design of TM, Techniques for TM construction. Variants of Turing Machines (TM), The model of Linear Bounded automata. <b>Textbook 1: Ch 14: 14.1, 14.2, Textbook 2: Ch 9.1 to 9.8</b> <b>RBT: L1, L2, L3</b>	08
--	----

**Module 5**

<b>Decidability:</b> Definition of an algorithm, decidability, decidable languages, Undecidable languages, halting problem of TM, Post correspondence problem. Complexity: Growth rate of functions, the classes of P and NP, Quantum Computation: quantum computers, Church-Turing thesis. <b>Applications:</b> G.1 Defining syntax of programming language, Appendix J: Security <b>Textbook 2: 10.1 to 10.7, 12.1, 12.2, 12.8, 12.8.1, 12.8.2</b>	08
---	----

**Textbook 1: Appendix: G.1(only), J.1 & J.2**

DRT, TIT, TIT, TIT	
<b>Module 5</b>	10
<p><b>Transaction Processing:</b> Introduction to Transaction Processing, Transaction and System concepts, Desirable properties of Transactions, Characterizing schedules based on recoverability, Characterizing schedules based on Serializability, Transaction support in SQL. <b>Concurrency Control in Databases:</b> Two-phase locking techniques for Concurrency control, Concurrency control based on Timestamp ordering, Multiversion Concurrency control techniques, Validation Concurrency control techniques, Granularity of Data Access and Multiple Concurrency Locking, Introduction to Database Recovery.</p> <p><b>Protocols:</b> Recovery Concepts, NO-UNDO/REDO recovery based on Deferred update, Recovery techniques based on immediate update, Shadow paging, Database backup and recovery from catastrophic failures</p> <p><b>Textbook 1:</b> 20.1 to 20.6, 21.1 to 21.7, 22.1 to 22.4, 22.7.</p> <p><b>RBT:</b> L1, L2, L3</p>	
<p><b>Course Outcomes:</b> The student will be able to :</p> <ul style="list-style-type: none"> <li>Identify, analyze and define database objects, enforce integrity constraints on a database using RDBMS.</li> <li>Use Structured Query Language (SQL) for database manipulation.</li> <li>Design and build simple database systems</li> <li>Develop application to interact with databases.</li> </ul>	
<p><b>Question Paper Pattern:</b></p> <ul style="list-style-type: none"> <li>The question paper will have ten questions.</li> <li>Each full Question consisting of 20 marks</li> <li>There will be 2 full questions (with a maximum of four sub questions) from each module.</li> <li>Each full question will have sub questions covering all the topics under a module.</li> <li>The students will have to answer 5 full questions, selecting one full question from each module.</li> </ul>	
<p><b>Textbooks:</b></p> <ol style="list-style-type: none"> <li>Fundamentals of Database Systems, RamezElmasri and Shamkant B. Navathe, 7th Edition, 2017, Pearson.</li> <li>Database management systems, Ramakrishnan, and Gehrke, 3<sup>rd</sup> Edition, 2014, McGraw Hill</li> </ol>	
<p><b>Reference Books:</b></p> <ol style="list-style-type: none"> <li>SilberschatzKorth and Sudharshan, Database System Concepts, 6<sup>th</sup> Edition, Mc-GrawHill, 2013.</li> <li>Coronel, Morris, and Rob, Database Principles Fundamentals of Design, Implementation and Management, Cengage Learning 2012.</li> </ol>	

  
 Head of the Department  
 Dept. of Artificial Intelligence & Machine Learning  
 Alva's Institute of Engineering and Technology  
 Shobhavana Campus, Mijar  
 Moodubidre 574 225, D.K. Karnataka, India