SRI KRISHNA INSTITUTE OF TECHNOLOGY

(Accredited by NAAC, Approved by A.I.C.T.E. New Delhi, Recognized by Govt. of Karnataka & Affiliated to V.T U., Belgaum)
#29, Chimney Hills, Hesaraghatta Main Road, Chikkabanavara Post, Bangalore- 560090

Department of Artificial Intelligence and Machine Learning

Academic Year: 2022-2023	**Semester: 5**
Course Name: Automata Theory & Computability	**Course Code: 18CS54**
Total Contact hours: 40	**Credits:3**
SEE Marks:40 ; CIE: 60	**Total Marks: 100**
Course Plan Author: Dr Geetha C Megharaj	**Date: 30-9-2022**

Course Prerequisites:
- Set Theory
- Proof Techniques.

Course Objectives: This course (18CS54) will enable students to:
- Introduce core concepts in Automata and Theory of Computation
- Identify different Formal language Classes and their Relationships
- Identify different Formal language Classes and their Relationships
- Prove or disprove theorems in automata theory using their properties
- Determine the decidability and intractability of Computational problems

Course Outcomes:

.CO Number	**Course Outcome** At the end of the course, student should be able to . . .	**Blooms' Level**
CO1	Understand the core concepts in Automata Theory and Theory of Computation.	L2
CO2	Design different automation models of Computation and conversion between the models.	L3
CO3	Design Grammars and Automata for different types of languages	L3
CO4	Understand working principles of Turing Machine.	L3
CO5	Classify a problem with respect to different models of Computation.	L3

Program Outcomes and Program Specific Outcomes

PO1. Engineering Knowledge: Apply the knowledge of mathematics, science, engineering fundamentals and an Engineering specialization to the solution of complex problems in Computer Science and Engineering.
PO2. Problem Analysis: Identify, formulate, review research literature and analyze complex Computer Science Engineering problems reaching substantiated conclusions using first principles ofmathematics, natural sciences and engineering sciences.
PO3. Design / Development of Solution: Design solutions for complex Computer Science Engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety and the cultural, societal and environmental considerations.
PO4. Conduct investigation of complex problems: Use research based knowledge and research methods including design of experiments, analysis and interpretation of data and synthesis of the information to provide valid conclusions related to Computer Science Engineering.
PO5. Modern Tool Usage: Ability to create, select, and apply appropriate techniques, resources and modern engineering and IT tools including prediction, modeling and analysis to complex Computer Science Engineering activities with an understanding of the limitations
PO6. The Engineering and Society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
PO7. Environmental and Sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts and demonstrate the knowledge of and need for sustainable development.
PO8. Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
PO9. Individual and Team Work: Function effectively as an individual and as a member or leader in diverse teams and in multidisciplinary settings.
PO10. Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations and give and receive clear instructions
PO11. Project Management and Finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one’s own work, as a member and leader in a team, tomanage projects and in multidisciplinary environments
PO12. Life-long Learning: Recognize the need for and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.
PSO1. Adapt, Contribute and Innovate ideas in the field of Artificial Intelligence and Machine Learning;
PSO2. Enrich their abilities to qualify for Employment, Higher studies and Research in various domains of Artificial Intelligence and Machine Learning such as Data Science, Computer Vision, Natural Language Processing with ethical values;
PSO3 Acquire the practical proficiency with niche technologies and open source platforms and to become Entrepreneur in the domain of Artificial Intelligence and Machine Learning

. CO – PO Mapping

Course Outcomes	**Program Outcomes**														
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	1										1	1	2	
CO2	3	2			1							1	1	3	
CO3	3	2										1	1	2	
CO4	3	2										1	1	2	
CO5	3	1										1	1	1	

Course Content (Syllabus)

Module1:	Contact Hours
Why study the Theory of Computation, Languages and Strings: Strings, Languages. A Language Hierarchy, Computation, **Finite State Machines (FSM)**: 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. Textbook 1: Ch 1,2, 3,4, 5.1 to 5.10 : **RBT: L1, L2**	08
Module2:	
Regular Expressions (RE): 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. Textbook 1: Ch 6, 7, 8: 6.1 to 6.4, 7.1, 7.2, 8.1 to 8.4 : **RBT: L1, L2, L3**	08
Module3:	
Context-Free Grammars(CFG): 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. Textbook 1: Ch 11, 12: 11.1 to 11.8, 12.1, 12.2, 12,4, 12.5, 12.6 : **RBT: L1, L2, L3**	08
Module4:	
Algorithms and Decision Procedures for CFLs: Decidable questions, Un-decidable questions. **Turing Machine**: 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. Textbook 1: Ch 14: 14.1, 14.2, Textbook 2: Ch 9.1 to 9.8 **RBT: L1, L2, L3**	08
Module5:	

Decidability: 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. **Applications:** G.1 Defining syntax of programming language, Appendix J: Security Textbook 2: 10.1 to 10.7, 12.1, 12.2, 12.8, 12.8.1, 12.8.2 Textbook 1: Appendix: G.1(only), J.1 & J.2 **RBT: L1, L2, L3**	08

Schedule of Instruction

Sl.No	Class no	Module	Topic	Text Book, Page no.	Course Outcome	Delivery mode
1	1	**Module1**	Strings, Languages. A Language Hierarchy, Computation,	(T1,8)	CO1	Lecture
2	2		**Finite State Machines (FSM)**: Deterministic FSM, Regular languages, Designing FSM	(T1,54)	CO1	Lecture
3	3		Designing FSM	(T1,63)	CO2	Lecture
4	4		Designing FSM	(T1,63)	CO2	Lecture
5	5		Nondeterministic FSMs	(T1,66).	CO2	Lecture
6	6		From FSMs to Operational Systems, Simulators for FSMs	(T1,79).	CO2	Lecture
7	7		Minimizing FSMs	(T1,82).	CO2	Lecture
8	8		Minimizing FSMs	(T1,82).	CO2	Lecture
9	9		Canonical form of Regular languages, Finite State Transducers	(T1,94).	CO2	Lecture
10	10		Bidirectional Transducers.	(T1,98).	CO2	Lecture
11	11	**Module 2**	what is a RE? and examples	(T1,128)	CO2	Lecture
12	12		Kleene"s theorem, Applications of REs	(T1,133)	CO2	Lecture
13	13		Manipulating and Simplifying REs	(T1,149)	CO2	Lecture
14	14		Manipulating and Simplifying Res	(T1,149)	CO2	Lecture
15	15		Regular Grammars: Definition, Regular Grammars and Regular languages.	(T1,155)	CO2	Lecture
16	16		Regular Grammars: Definition, Regular Grammars and Regular languages.	(T1,155)	CO2	Lecture
17	17		To show that a language is regular, Closure properties of RLs	(T1,163)	CO2	Lecture
18	18		To show some languages are not RLs.	(T1,169)	CO2	Lecture

19	21	**Module 3**	Introduction to Rewrite Systems and Grammars, CFGs and languages.	(T1,203)	CO3	Lecture
20	22		Designing CFGs	(T1,212)	CO3	Lecture
21	23		Simplifying CFGs	(T1,212)	CO3	Lecture
22	24		Proving that a Grammar is correct	(T1,215)	CO3	Lecture
23	25		Derivation and Parse trees	(T1,218)	CO3	Lecture
24	26		Ambiguity, Normal Forms	(T1,220)	CO3	Lecture
25	27		Pushdown Automata (PDA): Definition of non-deterministic PDA, Deterministic and Non-deterministic PDAs	(T1,249)	CO2	Lecture
26	28		Non-determinism and Halting	(T1,274)	CO2	Lecture
27	29		alternative equivalent definitions of a PDA	(T1,275)	CO2	Lecture
28	30		alternatives that are not equivalent to PDA.	(T1,277)	CO2	Lecture
29	31	**Module 4**	**Algorithms and Decision Procedures for CFLs**: Decidable questions, Un-decidable questions.	(T1,314)	CO4	Lecture
30	32		**Turing Machine**: Turing machine model	(T2,277)	CO4	Lecture
31	33		**Turing Machine**: Turing machine model	(T2,277)	CO4	Lecture
32	34		Representation, Language acceptability by TM	(T2,279)	CO4	Lecture
33	35		design of TM	(T2,284)	CO4	Lecture
34	36		design of TM	(T2,284)	CO4	Lecture
35	37		design of TM	(T2,284)	CO4	Lecture
36	38		Techniques for TM construction	(T2,289)	CO4	Lecture
37	39		Variants of Turing Machines (TM)	(T2,292)	CO4	Lecture
38	40		The model of Linear Bounded automata.	(T2,297)	CO4	Lecture
39	41	**Module 5**	Definition of an algorithm, decidability	(T2,309)	CO5	PPT
40	42		Decidable Languages, Undecidable Languages	(T2,311)	CO5	PPT
41	44		Halting problem of TM	(T2,314)	CO5	PPT
42	45		Post correspondence problem. Complexity: Growth rate of functions	(T2,315)	CO5	PPT
43	46		The classes of P and NP	(T2,349)	CO5	PPT

44	47		Quantum Computation: quantum computers, Church-Turing thesis.	(T1,360)	CO5	PPT
45	49		**Applications:** G.1 Defining syntax of programming language, Appendix J: Security	(T1,880)	CO5	PPT
46	50		**Applications:** G.1 Defining syntax of programming language, Appendix J: Security	(T2,948)	CO5	PPT

**L – Lecture, V- Videos or any other mode*

Textbooks	
T1	Elaine Rich, Automata, Computability and Complexity, 1st Edition, Pearson education,2012/2013
T2	K L P Mishra, N Chandrasekaran , 3rd Edition, Theory of Computer Science, PhI, 2012.
Reference books	
R1	John E Hopcroft, Rajeev Motwani, Jeffery D Ullman, Introduction to AutomataTheory, Languages, and Computation, 3rd Edition, Pearson Education, 2013.
R2	Michael Sipser : Introduction to the Theory of Computation, 3rd edition, Cengage learning,2013.
R3	John C Martin, Introduction to Languages and The Theory of Computation, 3rd Edition, Tata McGraw –Hill Publishing Company Limited, 2013
R4	Peter Linz, “An Introduction to Formal Languages and Automata”, 3rd Edition, Narosa Publishers, 1998
R5	Basavaraj S. Anami, Karibasappa K G, Formal Languages and Automata theory, Wiley India, 2012.
R6	C K Nagpal, Formal Languages and Automata Theory, Oxford University press, 2012.

Web links and Video Lectures (e-Resources):	
1	https://sites.google.com/skit.org.in/drgeetha-18cs54/home
2	https://www.tutorialspoint.com/automata_theory/index.htm
3	https://www.youtube.com/watch?v=AcuqeRo7e1M
4	https://www.youtube.com/watch?v=G_mCqJakvYk
5	https://www.youtube.com/watch?v=PvLaPKPzq2I

Assessment Schedule:

Sl. No.	Assessment type	Contents	CO	Duration In Hours	Marks	Date & Time
1	CIE Test 1	Module1	CO1, CO2	1.5	30	
2	CIE Test 2	Module2 & Module3	CO2, CO3	1.5	30	
3	CIE Test 3	Module4 & Module5	CO4, CO5	1.5	30	
4	Assignment 1	Module1	CO1, CO2		10	
5	Unit Test1, Unit Test2	Module2 & Module 3	CO2, CO3, CO4		5+5	

6	Assignment 2, QUIZ1	Module4 & Module 5	CO3, CO4		5+5	
7	**Semester End Examination**	Module1,Module2, Module3 , Module4, Module5	CO1-5	3	60	

**The sum of total marks of three tests, two assignments, and seminar will be out of 100 marks and will be scaled down to 40 marks. CIE + SEE = 40 + 60 = 100 marks

Faculty Incharge **DAC Chairman**