BANGALORE INSTITUTE OF TECHNOLOGY

K. R. Road, V. V. Puram, Bengaluru – 560 004

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

COURSE OBJECTIVES AND OUTCOMES 2020-24

Course Title : Automata Theory & Computability	**Course Code : 18CS54**
Number of Contact Hours/Week : 03	**Exam Hours : 03**
Total Number of Contact Hours. : 40	**Exam Marks : 60**

Prerequisites

1. Data Structures and Applications (18CS32)
2. Discrete Mathematical Structures (18CS36)

Course Learning Objectives

This course will enable students to

CLO1: Introduce core concepts in Automata and Theory of Computation.

CLO2: Identify different Formal Language Classes and their Relationships.

CLO3: Build Grammars and Recognizers for different formal languages.

CLO4: Prove or disprove theorems in automata theory using their properties.

CLO5: Determine the decidability and intractability of Computational problems.

Course Outcomes

The students should be able to

CO1: Acquire fundamental understanding of the core concepts in automata theory and Theory of Computation.

CO2: Learn how to translate between different models of Computation. (e.g., Deterministic and Non-deterministic and Software models).

CO3: Build Grammars and Automata (recognizers) for different language classes and become knowledgeable about restricted models of Computation (Regular, Context Free) and their relative powers.

CO4: Develop skills in formal reasoning and reduction of a problem to a formal model, with an emphasis on semantic precision and conciseness.

CO5: Classify a problem with respect to different models of Computation.

CO-PO-PSO MAPPING

	PO1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO1 0	PO1 1	PO1 2		PSO 1	PSO 2
CO1	2	1													
CO2	1	2			3									2	1
CO3			3		3							1		2	1
CO4		3										1			
CO5		3										1			

CO-PO MAPPING WITH JUSTIFICATION

18CS54	CO1	PO1	2	Apply the knowledge of Mathematics to understand the fundamentals of Automata Theory & Computation.
		PO2	1	The knowledge of Mathematical Principles help the students to apply the same to identify and analyze the engineering problems.
	CO2	PO1	1	Apply the fundamental knowledge of Mathematics to follow the procedures to translate between different models of computation such as NDFSM to DFSM.
		PO2	2	Enable's to analyse the requirement to verify/check to the advantage of translating from one model to another i.e. NDFSM to DFSM.
		PO5	3	Able to illustrate Software Models to translate between different models of computation.
	CO3	PO3	3	Able to build grammars and automata for different models of computation such as Language Recognizers, Communication Protocols and so on.
		PO5	3	Able to exhibit Simulators such as JFLAP to build grammars and automata for different models of computation.
		PO12	1	Constant learning will help to build different types of models for complex real world problems.
	CO4	PO2	3	Enables to analyze the problems before starting the formal reasons such as to prove and disprove theorems in automata theory.
		PO12	1	Constant learning will help to state new theorems for automata theory.
	CO5	PO2	3	Enables to identify a problem with respect to different models of computation in automata theory based on knowledge acquired in the course.

		PO12	**1**	Constant learning will help to classify complex real world problems with respect to different models of computation.

CO-PSO MAPPING WITH JUSTIFICATION

18CS54	**CO2**	**PSO1**	**2**	Graduates of the program will have the ability to translate between different models of computation which can be used in various applications such as Web Applications, Lexical Analyser and so on.
		PSO2	**1**	Graduates of the program will have the ability to take up research or higher studies in Machine Learning, Artificial Intelligence and so on.
	CO3	**PSO1**	**2**	Building of different models of automata enables graduates of the program to employable in IT industry.
		PSO2	**1**	Graduates of the program will have the ability to take up research or higher studies in Machine Learning, Artificial Intelligence and so on.

Faculty In-charges

Course Coordinator **Module Coordinator** **IQAC** **Programme Coordinator**