Devi Ahilya University, Indore, India Institute of Engineering & Technology				III Year B.E. Computer Engineering (Full Time)			
Subject Code & Name	**Instructions Hours per Week**			**Credits**			
5CERC1: Theory of Computation	**L**	**T**	**P**	**L**	**T**	**P**	**Total**
	3	**1**	**0**	**3**	**1**	**0**	**4**
Duration of Theory Paper: 3 Hours							

Learning Objectives:

1. Understand the foundations of computation**,** formal languages, grammars, and automata theory.
2. Analyze and design finite automata and regular expressions for modeling computational problems.
3. Explain the role of nondeterminism, minimization, and language properties in regular languages.
4. Comprehend context-free grammars, pushdown automata**,** and their applications in parsing.
5. Understand the limits of computation through Turing machines, computability, and undecidability.

Prerequisite:

Students are expected to have prior knowledge of Discrete Mathematics (sets, relations, functions, and basic proof techniques), Data Structures (stacks, queues, trees), and fundamental programming concepts to effectively understand automata, formal languages, and computability concepts.

Course Outcomes (CO) and Program Outcomes (PO) Mapping:

CO No.	Course Outcome	Program Outcomes (PO)
CO 1	Apply concepts of formal languages, regular expressions, and finite automata to model and analyze computational problems.	PO 1, PO 2, PO 3
CO 2	Analyze deterministic and nondeterministic finite automata, and determine language properties using closure properties, minimization techniques, and pumping lemma**.**	PO 1, PO 2, PO 3
CO 3	Design and transform context-free grammars**,** and apply normal forms and parsing algorithms for syntactic analysis.	PO 1, PO 2, PO 3
CO 4	Explain and compare pushdown automata and context-free languages**,** and evaluate language acceptance using appropriate PDA models**.**	PO 1, PO 2, PO 3, PO 4
CO 5	Analyze Turing machines and computability theory**,** and assess the decidability and undecidability of computational problems.	PO 1, PO 2, PO 3, PO 4, PO 12

CO-PO Relationship Matrix:

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	1									
CO2	3	3	2									
CO3	2	3	3									
CO4	2	3	2	1								
CO5	3	3	2	1								2

COURSE CONTENTS

UNIT-I: **Finite Automata and Regular Languages**
Motivation for studying theory of computation, Notion of formal languages and grammars, Kleene's Closure, Regular Expressions and Regular languages, closure properties of regular languages, Finite Automata. Finite Automata with output: Mealy and Moore machines, applications.

UNIT-II: Nondeterminism and Minimization
Nondeterministic Finite Automata, Acceptance condition. Kleene's Theorem, Myhill-Nerode relations, Minimization Algorithm, Non-Regular languages, Pumping Lemma for regular languages.

UNIT-III: Grammars and Context-Free Languages
Grammars and Chomsky Hierarchy, Context-Free Grammars, Context-Free Languages (CFLs), Inherent Ambiguity of CFLs, closure properties of CFLs, Eliminating useless symbols; null-productions; and unit productions, Chomsky Normal Form, Greibach Normal Form, Cock-Younger-Kasami (CYK) Algorithm, Applications to Parsing.

UNIT-IV: Pushdown Automata
Pushdown Automata (PDAs), PDAs vs CFLs. Deterministic PDAs and CFLs, applications, notion of acceptance for PDAs: acceptance by final states, and by empty stack; the equivalence of the two notions, Proof that CFGs generate the same class of languages that PDAs accept, Pumping Lemma for CFLs.

UNIT-V: Turing Machines and Computability
Introduction to Turing Machines, Configurations, Halting vs Looping, Turing computability, Nondeterministic, multitape and other versions of Turing machines. Church`s thesis, Universal Turing Machines, Linear Bounded Automata (LBAs) and context-sensitive languages, Recursive and Recursively enumerable languages, Undecidability of Halting Problem and unsolvable problems about Turing Machines, the diagonalization language and proof that it is not Recursively enumerable.

BOOKS RECOMMENDED:

[1] Daniel I.A. Cohen, Introduction to Computer Theory, John Wiley, 1990
[2] John C. Martin, Introduction to Languages and the Theory of Computation, 3/e Tata McGraw Hill, 2005

[3] J.E. Hopcroft and J.D.Ullman, Introduction to Automata, Languages and Computation, Narosa Publishing House, 1995
[4] J.E. Hopcroft, Rajeev Motwani and J.D.Ullman, Introduction to Automata, Languages and Computation, Pearson Education, Asia, 2002
[5] H.R. Lewis and C.H. Papadimitrou, Elements of the Theory of Computation, Prentice Hall Inc., 1999
[6] M. Sipser, Introduction to the Theory of Computation, Brooks/Cole Thomson Learning, 1996
[7] Zohar Manna, Mathematical Theory of Computation, McGraw Hill, 1997