THEORY OF COMPUTATION

IV Semester: CSE / IT

Course Code	Category	Hours / Week			Credits	Maximum Marks		
		L	T	P	C	CIA	SEE	Total
AIT002	Foundation	3	-	-	3	30	70	100
Contact Classes: 45	Tutorial Classes: Nil	Practical Classes: Nil				Total Classes: 45		

OBJECTIVES:

The course should enable the students to:

I. Introduce and study abstract, mathematical models of computation and use them to solve Computational problems
II. Understand the relationship between formal languages in Chomsky's hierarchy and different Machines.
III. Analyze and explain the behavior of push-down automata.
IV. Understand the limits and capacities of Turing's machines to recognize languages.

COURSE LEARNING OUTCOMES:

Students, who complete the course, will have demonstrated the ability to do the following:

1. Use the definitions and notations for sets, relations and functions in defining and study Finite Automata.
2. Knowledge on formal languages and Kleene's Theorem to intend programming languages.
3. Construct deterministic and nondeterministic finite state automata (DFA and NFA) for solving simple decision problems.
4. Perform conversions between nondeterministic finite automata and deterministic finite automata and regular expressions and finite state automata to gain knowledge about formal proofs in computer science.
5. Knowledge on recursive definitions of regular languages, regular expressions and the use of regular expressions to represent regular languages.
6. Detailed knowledge on the relationship between regular expressions and finite automata.
7. Identify that few languages are not regular by using Pumping lemma.
8. Knowledge on Left Linear grammar, Right Linear grammars and converting grammars into Finite Automata.
9. Understand the fundamental role played by Context-Free Grammars (CFG) in designing formal computer languages with simple examples.
10. Knowledge on Context-Free Grammars so that able to prove properties of Context-Free Grammars.
11. Identify relationship between regular languages and context-free grammars.
12. Use the pumping lemma for Context Free Languages to show that a language is not context-free.
13. Understand the equivalence between Context-Free Grammars and Non-deterministic Pushdown Automata.
14. Understand deterministic Pushdown Automata to parse formal language strings by using (i) top down or (ii) bottom up techniques.
15. Knowledge on converting Context-Free Grammars into pushdown automata to identify the acceptance of a string by the Context Free Language.
16. Understand the path processing computation using Turing Machines (Deterministic and Non-Deterministic) and Church-Turing Thesis in computers.
17. Knowledge on non-halting Turing Machine accepted by Recursively Enumerable Languages.

18. Understand the power of the Turing Machine, as an abstract automaton, that describes computation, effectively and efficiently.
19. Theory of Computation is important in programming language design, parsers, web-scrappers, Natural Language Processing (NLP), and is at the heart of modern compiler architectures.
20. Process the knowledge and skills for employability and to succeed in national and international level competitive exams.

UNIT-I	**FINITE AUTOMATA**	**Classes: 9**

Fundamentals: Alphabet, strings, language, operations; Introduction to finite automata: The central concepts of automata theory, deterministic finite automata, nondeterministic finite automata, an application of finite automata, finite automata with epsilon transitions.

UNIT-II	**REGULAR LANGUAGES**	**Classes: 9**

Regular sets, regular expressions, identity rules, constructing finite automata for a given regular expressions, conversion of finite automata to regular expressions, pumping lemma of regular sets, closure properties of regular sets (proofs not required), regular grammars-right linear and left linear grammars, equivalence between regular linear grammar and finite automata, inter conversion.

UNIT-III	**CONTEXT FREE GRAMMARS**	**Classes: 10**

Context free grammars and languages: Context free grammar, derivation trees, sentential forms, right most and leftmost derivation of strings, applications.

Ambiguity in context free grammars, minimization of context free grammars, Chomsky normal form, Greibach normal form, pumping lemma for context free languages, enumeration of properties of context free language (proofs omitted).

UNIT-IV	**PUSHDOWN AUTOMATA**	**Classes: 8**

Pushdown automata, definition, model, acceptance of context free language, acceptance by final state and acceptance by empty stack and its equivalence, equivalence of context free language and pushdown automata, inter conversion. (Proofs not required). Introduction to deterministic context free languages and deterministic pushdown automata.

UNIT-V	**TURING MACHINE**	**Classes: 9**

Turing machine: Turing machine, definition, model, design of Turing machine, computable functions, recursively enumerable languages, Church's hypothesis, counter machine, types of Turing machines (proofs not required), linear bounded automata and context sensitive language, Chomsky hierarchy of languages.

Text Books:

1. John E. Hopcroft, Rajeev Motwani, Jeffrey D.Ullman, "Introduction to Automata, Theory, Languages and Computation", Pearson Education, 3rd Edition, 2007.

Reference Books:

1. John C Martin, "Introduction to Languages and Automata Theory", Tata McGraw-Hill, 3rd Edition,

2007.
2. Daniel I.A. Cohen, “Introduction to Computer Theory”, John Wiley & Sons, 2nd Edition, 2004.

Web References:

1. https://www.tutorialspoint.com/automata_theory/index.htm
2. https://www.iitg.ernet.in/dgoswami/Flat-Notes.pdf

E-Text Books:

https://freefundkenotes.files.wordpress.com/2014/02/toc-klp-mishra.pdf

MOOC Courses:

1. http://nptel.ac.in/courses/111103016/
2. http://nptel.ac.in/courses/106106049/
3. http://onlinevideolecture.com/?course_id=1312
4. http://www.nptelvideos.in/2012/11/theory-of-computation.html