**CA20108A** **(3L + 1 T hrs./week)**

# FORMAL LANGUAGES AND AUTOMATA THEORY

No. of questions to be set: Total 5 questions will be given from Unit I & Unit II.
No. of questions to be answered: All questions to be answered.

**Objective:** The central objective of the course is to provide learners with a detailed understanding of the mathematical models of the machines and their evolution through requirement generation and advancement in languages.

**Pre-requisites:** Computer Programming concepts and Discrete Structures for Computer Science.

**Course Outcomes (CO):**

| | |
|---|---|
| CO1 | By the completion of the course the students will be able to define a system and recognize the behaviour of a system. |
| CO2 | Students will able to convert Finite Automata to regular expression. |
| CO3 | Students will able to understand and can check equivalence of CFL and PDA |

## UNIT-I

### Introduction [4 Hrs.]

Mathematical preliminaries: Sets, Logic, Functions, Relations, Languages.
Definitions: Language, Grammar, Automata, Relation between language, Grammar and automata, Importance of automata theory.

### Finite automata [5 Hrs.]

Informal introduction: Drawing examples from everyday life to bring out the essence of finite automata, Finiteness and its importance in automata theory.
Deterministic finite automata: Definition, Processing strings, Transition functions, Language of a DFA, Nondeterministic finite automata: Non-determinism, Definition, Extended transition functions, Language of a NFA, Equivalence of DFA and NFA, Kleene's theorem, Epsilon transitions, Applications of Finite automata in text search.

### Regular expressions and regular languages [4 Hrs.]

Memory required to recognize a language, Regular expressions, Regular expression to finite automata, Finite automata to regular expression, Algebraic laws for regular expressions, Applications of regular expressions, Criterion for regularity, Regular languages.

### Properties of regular languages [3 Hrs.]

Pigeonhole principle, Pumping lemma for regular languages, Closure properties, Testing membership of regular languages, Equivalence of automata.

### Context Free Grammars and Languages [4 Hrs.]

Definition, Leftmost and rightmost grammars, Parse trees, Ambiguity: Ambiguous grammar, Removing ambiguity. Normal forms, Applications of context free grammars: Parsers.

## UNIT-II

**Pushdown automata and context free languages [7 Hrs.]**
Definition of pushdown automata, Representing pushdown automata, Acceptance by pushdown automata: By final state, By empty stack, Deterministic pushdown automata, Equivalence of pushdown automata and context free grammars, Pumping lemma for context free languages, Closure properties of context free languages, Testing membership of context free, Decision problems for context free languages.

**Turing machines [6 Hrs.]**
Definition, Language of a Turing machine, Programming Turing machines, The Church-Turing thesis, A simple programming language, Extensions of the basic Turing machine.

**Recursively enumerable languages [2 Hrs.]**
Definition, Enumeration, Chomsky hierarchy.

**Undecidability [3 Hrs.]**
The halting problem, The post correspondence problem, Time and space complexity of Turing machines, Complexity classes.

**Language learning [2 Hrs.]**
Learning framework, Inductive inference, Grammar induction.

**Text Books:**
1. John. E. Hopcroft, Rajeev Motwani, Jeffry. Ullman, Introduction to Automata Theory, Languages and Computation, Pearson Education.
2. John Martin, Introduction to Languages and the Theory of Computation, Tata McGraw Hill.

**Reference Books:**
1. Peter Linz, An Introduction to Formal Languages and Automata, Narosa.
2. James. L. Hein, Discrete Structures, Logic and Computability, Narosa.
3. Partha Niyogi, The Computational Nature of Language Learning and Evolution, PHI.
4. Zvi Kohavi and Niraj K. Jha, Switching and Finite Automata theory, Tata McGraw Hill.