Formal Languages and Automata Theory

Course title & Code	Credits	Credit distribution of the course			Eligibility criteria	Pre- requisite of the Course (if any)
		Lecture	Tutorial	Practical/ Practice		
Formal Languages and Automata Theory	4	3	0	1	Class XII Pass	NA

COURSE OBJECTIVE

- To learn about fundamental concepts of finite automata and formal language
- To enhance student's ability to understand and solve mathematical proofs for computation and algorithm
- To learn about deterministic and non- deterministic machines.
- To design grammars and recognizers for different formal languages

COURSE OUTCOME

- Students will have a clear understanding of abstract models of computation.
- Students will be able to analyze and design the finite automata, pushdown automata, formal language and language.
- Students will be able to apply mathematical and formal techniques for solving problems in computer science.

SYLLABUS

UNIT I **(6 Hours)**

Introductionto Finite Automata: The central concepts of Automata theory; Deterministic finite automata; Nondeterministic finite automata. Applications of finite automata, Finite automata with Epsilon transitions.

UNIT II **(12 Hours)**

Finite Automata and RegularExpressions: Applications of Regular Expressions; Regular languages; Proving languages not to be regular languages; Closure properties of regular languages; Decision properties of regular languages; Equivalence and minimization of automata.

UNIT III **(9 Hours)**

Context—free grammar: Parse trees; Applications; Ambiguity in grammars and Languages. Definition of the Pushdown automata; the languages of a PDA; Equivalence of PDA's and CFG's.

UNIT IV **(9 Hours)**

Deterministic PushdownAutomata: Normal forms for CFGs; The pumping lemma for CFGs; Closure properties of CFLs.

UNIT V **(9 Hours)**

Turing Machine: Programming techniques for Turing Machines, Extensions to the basics Turning machines, Turning machines and computers.

REFERENCES

1. John E. Hopcroft, Rajeev Motwani, Jeffrey D.Ullman: Introduction to Automata Theory, Languages and Computation, 3rd Edition, Pearson Education, 2011.

2. John C Martin: Introduction to Languages and Automata Theory, 3rd Edition, Tata McGraw- Hill, 2007.

3. Daniel L.A. Cohen: Introduction to Computer Theory, 2nd Edition, John Wiley & Sons, 2009.

4. Thomas A. Sudkamp: An Introduction to the Theory of Computer Science, Languages and Machines, 3rd Edition,Pearson Education, 2006.

PRACTICAL COMPONENT (IF ANY)

Implement the following in C/C++:

LIST OF PRACTICALS (30 Hours)

1. Simulate a Deterministic Finite Automaton (DFA) that accepts strings with three consecutive 1's as a substring.
2. Simulate a Nondeterministic Finite Automaton (NFA) for strings ending with "01".
3. Check if a given string belongs to a regular language using a regular expression.
4. Minimize a DFA for a given regular language.
5. Simulate a Pushdown Automaton (PDA) for the language 0^n1^n where $n \geq 1$.
6. Generate a parse tree for a given context-free grammar (CFG).
7. Simulate a Deterministic Pushdown Automaton (DPDA) for a deterministic CFL.
8. Convert a CFG into its Chomsky Normal Form (CNF).
9. Simulate a Turing Machine that computes the 2's complement of a binary number.
10. Simulate a Turing Machine that recognizes palindromes in a binary string.