MCA- C203 Theory of Computer Science

L	T	P	C
3	1	0	4

Course objective:

1. Develop a formal notation for strings, languages and machines.
2. Design finite automata to accept a set of strings of a language.
3. Prove that a given language is regular and apply the closure properties of languages.
4. Design context free grammars to generate strings from a context free language and convert them into normal forms.
5. Prove equivalence of languages accepted by Push Down Automata and languages generated by context free grammars
6. Identify the hierarchy of formal languages, grammars and machines.
7. Distinguish between computability and non-computability and Decidability and undecidability

Course outcomes:

1. Write a formal notation for strings, languages and machines.
2. Design finite automata to accept a set of strings of a language.
3. For a given language determine whether the given language is regular or not.
4. Design context free grammars to generate strings of context free language.
5. Determine equivalence of languages accepted by Push Down Automata and languages generated by context free grammars
6. Write the hierarchy of formal languages, grammars and machines.
7. Distinguish between computability and non-computability and Decidability and undecidability.

Introduction to Languages; Recursive Definitions; Regular Expressions; Finite Automata; Transition Graphs; Kleene's Theorem;

Non- Deterministic Finite Automata, Finite Automata with Output - Moore and Mealy machines, Equivalence of Moore and Mealy machines; Regular Languages; Non-regular Languages; Decidability.

Context-Free Grammars. Trees; Regular Grammars; Chomsky's Normal Form; Pushdown Automata; Context- Free Languages; Non- Context- Free Languages; Parsing; Decidability for CFG - Emptiness, Finiteness and Membership questions about CFG.

Turing Machines (TM); Post Machines (PM), simulating a PM on TM.

Recursively Enumerable Languages; Encoding of Turing Machines; Phrase Structure Grammar; Context Sensitive Grammar; Defining the Computer and Computable Functions; Church's Thesis; Halting Problem for Turing Machines.

Recommended Books:

1. Cohen, Daniel I.A., Introduction to Computer Theory, John Wiley & Sons
2. K. L. P. Mishra and N. Chandrasekaran, Theory of Computer Science: Automata Languages and Computation, PHI

HEAD
Department of Computer Science
Gurukul Kangri Vishwavidyalaya
Haridwar (UK) - 249404