310245: THEORY OF COMPUTATION

Teaching Scheme Examination Scheme

Lectures: 3 Hrs/Week Theory: 100 marks

Objectives:

• Study abstract computing models

• Learn about the theory of computability and complexity.

Prerequisites:

• Discrete Structures

• Data Structures and Algorithms

Unit I: Basic Mathematical Objects

Sets, Logic, functions, Relations,

Languages: Languages in abstract, defining languages, Kleene closure.

Recursive Definitions: New method for defining languages, important languages.

Finite Automata: An Informal Picture of FA, Deterministic Finite Automaton (DFA): How a DFA

processes Strings, Simpler Notations for DFA, Extending the transition function to strings, the

language of DFA, Non-deterministic Finite Automaton (NFA): NFA, Extended transition function,

the language of an NFA, Equivalence of NFA and DFA, FA with e-transitions: Use of e-transitions,

NFA with e, e-closures, Extended transitions and languages for e-NFA, Eliminating €-transitionsCon

version of NFA with e to NFA without e, Conversion of NFA without e to DFA, Conversion

of NFA with 6 to DFA (direct method), FA with output: Moore and Mealy machines -Definition,

models, inter-conversion. (6 hrs.)

Unit II: Regular Expressions (RE) and Languages

 Regular Expressions - Operators of RE, Building RE, Precedence of operators, Algebraic

laws for RE, Arden's Theorem, FA and RE: DFA to RE, RE to DFA (RE to s-NFA & e-NFA to

DFA and RE to DFA-direct method), FA limitations, Properties of Regular Languages: pumping lemma for regular languages, closure and decision properties of regular languages, Equivalence and minimization of automata, Application of RE: Regular expressions in Unix, GREP utilities of Unix, Lexical analysis and finding patterns in text. (6 hrs.)

Unit III: Context Free Grammars (CFG) and Languages

Context Free Grammar- Definition, derivations, languages of a grammar, sentential form, Parse Tree- inference, derivation and parse tree, from inference to tree, Ambiguity in grammars and languages: removal of ambiguity, inherent ambiguity, Properties of CFL- Normal forms- Chomsky Normal Form and Greibach Normal Form(GNF), Eliminating unit productions, useless production, useless symbols, and e-productions, Regular Grammar - definition, left linear and right linear Regular Grammar, Regular Grammar and Finite Automata, FA to RG and RG to FA, Interconversion between left linear and right linear regular grammar. The pumping lemma for CFL, Closure properties of CFL, Decision properties of CFL, Chomsky Hierarchy, Application of CFG: Parser, Markup languages, XML and Document Type Definitions. (6 hrs.)

Unit IV: Push Down Automata (PDA)

Definition, The Language of PDA, Equivalence of PDA's and CFG- CFG to PDA, PDA to CFG, Deterministic Push Down Automata (DPDA) - Regular language and DPDA, DPDA and CFL, DPDA and ambiguous grammar, Non-deterministic Push Down Automata (NPDA). (6 hrs.)

Unit V: Turing Machine 11

Problems that computer cannot solve, The Turing Machine(TM)-Notation, the language of TM, TM and Halting, Programming techniques to TM, Extensions to basic TM, TM and Computers.

Post Machine: Introduction to Post Machines, Comparison between FA, PDA, Post Machine and TM (6 hrs.)

Unit VI: Recursively enumerable languages

Recursively Enumerable and Recursive, Enumerating a Language, More General Grammars Context-Sensitive Languages and the Chomsky Hierarchy, Not All Languages are Recursively Enumerable.

Un-decidability: A Language that is not recursively enumerable, An un-decidable problem that is RE, Post Correspondence Problem, Other Undecidable Problems. (6 hrs.)

Text books:

1. Hopcroft J., Mptwani R., Ullman J., "Introduction to Automata Theory, Languages and Computations", Third edition, Pearson Education Asia.

2. John C Martin. "Introduction to Language and Theory of Computation", Third edition, Tata McGraw- Hill

3. Daniel Cohen., "Introduction to Computer Theory", Second edition, Wiley Publications (india).

References Books:

1. Lewis H., Papadimitriou C., "Elements of Theory of Computation", Second edition, Pearson

2. Moret B., "The Theory of Computation", Pearson Education Asia

3. Mishra K., Chandrasekaran N., 'Theory of Computer Science (Automata, Languages and Computation)", Second Edition, Prentice Hall of India