

MCA-18 Formal Languages and Automata

Unit 1: Introduction To Finite Automata

Alphabets and languages- Finite Representation of Languages. Deterministic Finite Automata – Non- deterministic Finite Automata – Equivalence of Deterministic and Non-Finite Automata – Properties of the Languages Accepted by Finite Automata – Finite Automata and Regular Expressions – Proofs those Languages Are and Are Not Regular.

Unit 2: Context Free Languages

Context –Free Grammar – Regular Languages and Context-Free Grammar – Pushdown Automata – Pushdown Automata and Context-Free Grammar – Properties of Context-Free Languages – Closure Properties – Periodicity Properties – Determinism and Parsing – Deterministic Pushdown Automata and Context – Free Languages – Top- down Parsing – Bottom – Up parsing.

Unit 3: Turing machines

The Definition of Turing Machine – Computing with Turing Machines – Combining Turing Machines – some Examples of More Powerful Turing Machines, Acceptability and Decidability

Unit 4: Church' Thesis

Church's Thesis – The Primitive Recursive functions – Godelization – The μ -Recursive Functions – Turing – Computability of the μ -Recursive functions – Universal Turing Machines.

Unit 5: Uncomputability

The Halting Problem – Turing-Enumerability, Turing – Acceptability, and Turing - Decidability – Unsolved problems about Turing machines and - Recursive Functions - Post's correspondence problem.

Suggested readings:

1. Alfred V Aho , Jeffrey D. Ullman, "Principles of Compiler Design", Narosa
2. A.V. Aho, R. Sethi and J.D Ullman, "Compiler: principle, Techniques and Tools", AW
3. H.C. Holub "Compiler Design in C", Prentice Hall Inc.
4. Apple, "Modern Computer Implementation in C: Basic Design", Cambridge press