

IT 704 Elective –I (IT- 713- Automata and Compiler Design)

Unit I: Introduction: Alphabets, Strings and Languages; Automata and Grammars, Deterministic finite Automata (DFA)-Formal Definition, Simplified notation: State transition graph, Transition table, Language of DFA, Nondeterministic finite Automata (NFA), Equivalence of NFA and DFA, Minimization of Finite Automata, Regular Expressions, Arden's theorem.

Unit II: Compiler Structure: Compilers and Translators, Various Phases of Compiler, Pass Structure of Compiler, Bootstrapping of Compiler. Lexical Analysis: The role of Lexical Analyzer, A simple approach to the design of Lexical Analyzer, Implementation of Lexical Analyzer. The Syntactic Specification of Programming Languages: CFG, Derivation and Parse tree, Ambiguity, Capabilities of CFG. Basic Parsing Techniques: Top-Down parsers with backtracking, Recursive Descent Parsers, Predictive Parsers,

Unit III: Bottom-up Parsers, Shift-Reduce Parsing, Operator Precedence Parsers, LR parsers (SLR, Canonical LR, LALR) Syntax Analyzer Generator: YACC, Intermediate Code Generation: Different Intermediate forms: three address code, Quadruples & Triples. Syntax Directed translation mechanism and attributed definition. Translation of Declaration, Assignment, Control flow, Boolean expression, Array References in arithmetic expressions, procedure calls, case statements, postfix translation.

Unit IV: Run Time Memory Management: Static and Dynamic storage allocation, stack based memory allocation schemes, Symbol Table management Error Detection and Recovery: Lexical phase errors, Syntactic phase errors, Semantic errors.

Unit V: Code Optimization and Code Generation: Local optimization, Loop optimization, Peephole optimization, Basic blocks and flow graphs, DAG, Data flow analyzer, Machine Model, Order of evaluation, Register allocation and code selection

References:-

- Louden, "Compiler construction", Cengage learning .
- Alfred V Aho, Jeffrey D. Ullman, "Principles of Compiler Design", Narosa.
- A.V. Aho, R. Sethi and J.D Ullman, "Compiler: principle, Techniques and Tools", AW.
- Michal Sipser, "Theory of Computation", Cengage learning.
- H.C. Holub, "Compiler Design in C", Prentice Hall Inc.
- Hopcroft, Ullman, "Introduction to Automata Theory, Languages and Computation", Pearson Education.
- K.L.P. Mishra and N.Chandrasekaran, "Theory of Computer Science : Automata, Languages and Computation", PHI.