PIT6J001 COMPILER DESIGN

Module – I (16 Hrs)

Introduction: Overview and Phases of compilation.
Lexical Analysis: Non-Deterministic and Deterministic Finite Automata (NFA & DFA), Regular grammar, Regular expressions and Regular languages, Design of a Lexical Analyzer as a DFA, Lexical Analyzer generator.
Syntax Analysis: Role of a Parser, Context free grammars and Context free languages, Parse trees and derivations, Ambiguous grammar.
Top Down Parsing: Recursive descent parsing, LL (1) grammars, Non-recursive Predictive Parsing, Error reporting and Recovery.
Bottom Up Parsing: Handle pruning and shift reduces Parsing, SLR parsers and construction or SLR parsing tables, LR(1) parsers and construction of LR(1) parsing tables, LALR parsers and construction of efficient LALR parsing tables, Parsing using Ambiguous grammars, Error reporting and Recovery,Parser generator.

Module – II (08 Hrs)

Intermediate Code Generation: DAG for expressions, Three address codes - Quadruples and Triples, Types and declarations, Translation of Expressions, Array references, Type checking and Conversions, Translation of Boolean expressions and control flow statements, Back Patching, Intermediate Code Generation for Procedures.

Module – III (08 Hrs)

Code Generation: Factors involved, Registers allocation, Simple code generation using STACK Allocation, Basic blocks and flow graphs, Simple code generation using flow graphs.
Code Optimization: Objective, Peephole Optimization, Concepts of Elimination of local common sub-expressions, Redundant and un-reachable codes, Basics of flow of control optimization.

Module – IV (08 Hrs)

Run Time Environment: Storage Organizations, Static and Dynamic Storage Allocations, STACK Allocation, Handlings of activation records for calling sequences.
Syntax Directed Translation: Syntax Directed Definitions (SDD), Inherited and Synthesized Attributes, Dependency graphs, Evaluation orders for SDD, Semantic rules, Application of Syntax Directed Translation.
Symbol Table: Structure and features of symbol tables, symbol attributes and scopes.

Text Book:
1. Compilers – Principles, Techniques and Tools, A. V. Aho, M. S. Lam, R. Sethi, J. D. Ullman, Pearson.
2. Compiler Design, S. Chattopadhyay, PHI.

Reference Books:
1. Modern Compiler Design, D. Galles, Pearson Education.
2. Advanced Compiler Design & Implementation, S. S. Muchnick, Morgan Kaufmann.
3. Compiler Design in C, A. I. Holub, PHI