Compiler Design

UNIT 1: Introduction to Compilers:

Structure of a compiler – Lexical Analysis – Role of Lexical Analyzer – Input Buffering – Specification of Tokens – Recognition of Tokens – Lex – Finite Automata – Regular Expressions to Automata – Minimizing DFA.

Unit II : Syntax Analysis:

Role of Parser – Grammars – Error Handling – Context-free grammars – Writing a grammar, Top-Down Parsing – General Strategies Recursive Descent Parser – Predictive Parser-LL(1) Parser-Shift Reduce Parser-LR Parser-LR (0) Item Construction of SLR Parsing Table - Introduction to LALR Parser – Error Handling and Recovery in Syntax Analyzer-YACC.

Unit III : Intermediate Code Generation:

Syntax Directed Definitions, Evaluation Orders for Syntax Directed Definitions, Intermediate Languages: Syntax Tree, Three Address Code, Types and Declarations, Translation of Expressions, Type Checking.

Unit IV: Run-Time Environment and Code Generation:

Storage Organization, Stack Allocation Space, Access to Non-local Data on the Stack, Heap Management – Issues in Code Generation – Design of a simple Code Generator.

Unit V: Code Optimization:

Principal Sources of Optimization – Peep-hole optimization – DAG- Optimization of Basic Blocks-Global Data Flow Analysis – Efficient Data Flow Algorithm.