Compiler Design
CS702
Contracts: 3L
Credits- 3

Module I
Introduction to Compiling [2L]
Compilers, Analysis-synthesis model , The phases of the compiler, Cousins of the compiler.

Lexical Analysis [5L]
The role of the lexical analyzer, Tokens, Patterns, Lexemes, Input buffering, Specifications of a token, Recognition of tokens, Finite automata, From a regular expression to an NFA, From a regular expression to NFA, From a regular expression to DFA, Design of a lexical analyzer generator (Lex).

Module II
Syntax Analysis [8L]
The role of a parser, Context free grammars, Writing a grammar, Top down Parsing, Non-recursive Predictive parsing (LL), Bottom up parsing, Handles, Viable prefixes, Operator precedence parsing, LR parsers (SLR, LALR), Parser generators (YACC). Error Recovery strategies for different parsing techniques.

Syntax directed translation [4L]
Syntax directed definitions, Construction of syntax trees, Bottom-up evaluation of S attributed definitions, L attributed definitions, Bottom-up evaluation of inherited attributes.

Module III
Type checking [3L]
Type systems, Specification of a simple type checker, Equivalence of type expressions, Type conversions

Run time environments [4L]
Source language issues (Activation trees, Control stack, scope of declaration, Binding of names), Storage organization (Subdivision of run-time memory, Activation records), Storage allocation strategies, Parameter passing (call by value, call by reference, copy restore, call by name), Symbol tables, dynamic storage allocation techniques.

Module IV
Intermediate code generation [3L]
Intermediate languages, Graphical representation, Three-address code, Implementation of three address statements (Quadruples, Triples, Indirect triples).

Code optimization [4L]
Introduction, Basic blocks & flow graphs, Transformation of basic blocks, Dag representation of basic blocks, The principle sources of optimization, Loops in flow graph, Peephole optimization.

Code generations [3L]
Issues in the design of code generator, a simple code generator, Register allocation & assignment.

Text books:
1. Aho, Sethi, Ullman - "Compiler Principles, Techniques and Tools" - Pearson Education.
2. Holub - "Compiler Design in C" - PHI
3.Tremblay and Sorenson Compiler Writing-McgrawHill International .
4. Chattopadhyay , S- Compiler Design (PHI)