

3rd Semester	MCA03002	Compiler Design	L-T-P 3-0-0	3 CREDITS
--------------------------------	-----------------	------------------------	------------------------	----------------------

Module- I: (8 Periods)

Compiler Structure: Model of compilation, various phases of a compiler. Lexical analysis: Interface with input parser and symbol table, token, lexeme and patterns, difficulties in lexical analysis, input buffering. Specification of tokens. Regular grammar & language definition.

Module- II: (12 Periods)

Syntax Analysis: Grammar, Parsing, ambiguity, top down parsing, recursive descent parsing, transformation on the grammars, predictive parsing LL(1) grammar, Nor LL(1) grammar, Bottom up parsing, operator precedence grammars, LR parsers (SLR, CLR, LALR).

Module- III: (10 Periods)

Syntax directed definitions: Inherited and synthesized attributes, dependency graph, evaluation order, bottom up and top down evaluation of attributes, L- and S-attributed definitions. Type checking: type: type system, type expressions, structural and name equivalence of types, type conversion. Run time system: storage organization, activation tree, activation record, parameter passing symbol table, dynamic storage allocation.

Module- IV: (10 Periods)

Intermediate code generation: intermediate code representation techniques. Intermediate Code generation for control flow, function call, Boolean expressions and procedure calls. Code optimization: source of optimizations, optimization of basic blocks, loops, global dataflow analysis, solution to iterative dataflow equations, code improving transformations, dealing with aliases, data flow analysis of structured flow graphs.

Module- V: (10 Periods)

Code generation and instruction selection: Issues, basic blocks and flow graphs, register allocation, code generation, DAG representation of programs, code generation from DAGS, peep hole optimization. Symbol table management: Data structure for symbol table organization. Error Handling and recovery.

Books:

1. K. C. Louden, "Compiler Construction, Principle and Practice", Cengage Publication
2. Alfred V. Aho, Ravi Sethi, and Ullman, "Compilers Priciples, Techniques and Tools", Pearson Publication
3. V.Raghvan, "Principles of Compiler Design", TMH Publication
4. Levine, Mason and Brown, "Lex & Yacc", O' Reilly Publication