CS 306: COMPILER DESIGN (3-0-2: 4)

Compiler Structure
Analysis-synthesis model of compilation, various phases of a compiler, tool based approach to compiler construction.

Lexical Analysis
Interface with input, parser and symbol table, token, lexeme and patterns, difficulties in lexical analysis, error reporting, and implementation. Regular definition, Transition diagrams, LEX.

Syntax Analysis
Context free grammars, ambiguity, associability, precedence, top down parsing, recursive descent parsing, transformation on the grammars, predictive parsing, Bottom up parsing, operator precedence grammars, LR parsers (SLR, LALR, LR), YACC.

Syntax Directed Translation
Inherited and synthesized attributes, dependency graph, evaluation order, bottom-up and top-down evaluation of attributes, S- and L-attributed definitions.

Intermediate Code Generation

Intermediate representations, Code generation and instruction selection: issues, basic blocks and flow graphs, register allocation, code generation, Code optimization, source of optimizations, optimization of basic blocks, loops, global dataflow analysis, solution to iterative dataflow equations.

Suggested list of Experiments:
1) Simulation of a Finite state machine to distinguish among Integers, Real Numbers & Numbers with Exponents.
2) Simulation of a Finite state Automata to recognize the tokens of various control statements.
3) Use of LEX tool to convert two digit integers to Roman.
4) Use of LEX tool to recognize the tokens and to return the token found for a C like Language.
5) Implement a lexical analyzer that reads the input one character at a time and returns to the parser the token it has found.
6) Implement a symbol table routine to determine whether an identifier lexeme has previously seen & store a new lexeme into symbol table.
7) Use of YACC tool to convert infix expressions to postfix.
8) Parsing of arithmetic and algebraic expressions and equations.
9) Implement recursive descent, predictive, operator precedence & LR parsers.
10) Generation of Syntax tree, DAG during parsing using YACC.

Text Books

1. A.V. Aho, R. Sethi and J D. Ullman : "Compilers-Principles, Techniques and Tools". Pearson Education.

References

1. Louden , Kenneth C : "Compiler Construction-Principles and Practice". Thomson.
2. Allen I. Holub: "Compiler Design in C", PHI.