**(IT/T/315) Principles of Compiler Design**

**First Class Test**

Compiler structure: analysis-synthesis 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, Regular Expression (RE), Non Deterministic Finite Automata (NFA), Deterministic Finite Automata (DFA), Conversion from RE to NFA, Conversion of NFA to DFA, Application of Finite Automata in Lexical Analysis.

Syntax analysis: Context Free Grammar (CFG), derivation, Parse Tree, ambiguity, top down parsing, Left Recursion Removal, Left Factoring, Predictive parsing, Bottom up parsing.

**Second Class Test Syllabus**

Syntax Analysis: LR parsers (SLR, LR(1), LALR)

Intermediate code generation: intermediate representations, translation of declarations, assignments, control flow, Boolean expressions, procedure calls, records, arrays, Implementation issues.

Run time system: storage organization, activation tree, activation record, symbol table.

Code optimization: source of optimizations, Basic Block, Local and Global Optimizations.