**GOAPAL KRISHNA COLLEGE OF ENGINEERING AND TECHNOLOGY**
**GOURAHARI VIHAR, PO: RANIPUT, JEYPORE – 764 005**

**LESSON PLAN**

**Name of the Subject: CSPC3004 COMPILER DESIGN**
**Session : 2025-26**
**Name of the Faculty:** Amod Kumar Bagh
**Semester:** 6th **Branch:** Computer Sc. & Engg.
**Semester From**: **No. of Weeks:** 10 Weeks

| WEEK | DAY | THEORY TOPICS | CLASSES |
|---|---|---|---|
| **Module-I: Introduction to Compiler and Lexical Analysis** | | | **8** |
| 1 | 1 | Language processing systems: | 1 |
| | 2 | preprocessors, compiler, assembler, interpreter, linker, bootstrap loaders | 2 |
| | 3 | | |
| | 4 | cross compilers. Structure of a compiler and the different phases. | 1 |
| 2 | 5 | Lexical Analysis: Role of the lexical analyzer, input buffering, regular expressions | 2 |
| | 6 | | |
| | 7 | construction of NFA and DFA, DFA minimization, transition diagrams, token recognition. Lexical errors and recovery, introduction to lexical analyzer generator (LEX). | 2 |
| | 8 | | |
| **Module-II: Syntax Analysis** | | | **8** |
| 3 | 9 | Role of a parser, derivation, | 1 |
| | 10 | ambiguity, left recursion, left factoring, recursive descent parsing. Bottom-up parsing: | 1 |
| | 11 | Shift-reduce parsing, operator precedence parsing, handles and handle pruning. | 2 |
| | 12 | | |
| 4 | 13 | Introduction to LR parsing: SLR, CLR, and LALR parsing tables. | 1 |
| | 14 | Parser conflicts and resolution, dangling else problem, | 1 |
| | 15 | error recovery strategies in parsing. Parser generator (YACC). | 2 |
| | 16 | | |
| **Module-III: Connecting devices:** | | | **8** |
| 5 | 17 | Semantic Analysis and Syntax-Directed Translation Semantic Analysis: Syntax-directed definitions (SDD), | 2 |
| | 18 | | |
| | 19 | evaluation of semantic rules. Translation schemes: Syntax-directed translation schemes (SDTS), | 1 |
| | 20 | implementation of S-attributed and L-attributed definitions. | 1 |
| 6 | 21 | Type checking and conversions. Intermediate Code Generation: | 1 |
| | 22 | Abstract syntax trees, three-address code, quadruples, triples, indirect triples. | 1 |

| | | | |
|---|---|---|---|
| | 23<br>24 | Translation of expressions, assignment statements, Boolean expressions, case statements, back patching, and procedure calls. | 2 |
| | | | |
| **Module-IV: (8 Hrs.)<br>Internetworking:** | | | **8** |
| 7 | 25 | Symbol Table and Runtime Environment<br>Symbol table: | 1 |
| | 26<br>27<br>28 | structure, operations (insert, lookup), scope management, activation records.<br>Runtime environment: storage organization, stack allocation, | 3 |
| 8 | 29<br>30 | heap management, access to non-local data, parameter passing | 2 |
| | 31 | mechanisms. Error handling: classification of errors, | 1 |
| | 32 | error recovery strategies. | 1 |
| **Module-V: (08 Hrs.)<br>Code Optimization and Code Generation** | | | **08** |
| 9 | 33 | Code optimization:<br>Machine-independent: constant folding, common subexpression elimination, | 1 |
| | 34 | dead code elimination, copy propagation, loop optimization, | 1 |
| | 35<br>36 | strength reduction, basic blocks and flow graphs, data flow analysis.<br>Machine-dependent: DAG-based optimization, peephole optimization, register allocation, | 2 |
| 10 | 37 | instruction scheduling, inter-procedural optimization, | 1 |
| | 38<br>39 | garbage collection using reference counting.<br>Code generation: target machine model, issues in the design of a code generator, | 2 |
| | 40 | generation of code for expressions and control structures. | 1 |