

## PRINCIPLES OF COMPILER DESIGN

### (Professional Elective-VI)

Course Code: 19IT1163

L	T	P	C
3	0	0	3

**Pre-requisites:** Formal Languages and Automata Theory

#### **COURSE OUTCOMES:**

At the end of the course, the student will be able to:

**CO1:** Identify tokens and lexemes for the given set of instructions.

**CO2:** Construct parsing tables for a given grammar.

**CO3:** Develop Syntax Directed Translation Schemes.

**CO4:** Model SDD's using Intermediate Representations

**CO5:** Develop algorithms to generate code for a target machine

#### **UNIT- I**

**(10 LECTURES)**

##### **INTRODUCTION TO COMPILING:**

Overview of Compilers, Phases of a Compiler, Bootstrapping.

##### **LEXICAL ANALYSIS:**

The Role of Lexical Analyzer, Input Buffering, Specification of Tokens, Regular Expressions, Regular Definitions- Recognition of Tokens, A language for specifying Lexical Analyzers (LEX).

**Learning Outcomes:** At the end of the module, the student will be able to

1. Summarize the phases of a compiler (L2)
2. Determine the tokens and lexemes in a given input (L3)
3. Construct Regular Expressions for specifying tokens (L3)

#### **UNIT-II**

**(10 LECTURES)**

##### **SYNTAX ANALYSIS:**

The role of the Parser, Context-free Grammars, Elimination of Left Recursion, Left factoring a grammar.

##### **TOP-DOWN PARSING:**

Recursive descent Parsing, First and Follow, Predictive Parsing, LL (1) Grammars.

##### **BOTTOM-UP PARSING:**

Shift-Reduce Parser, LR Parsers-SLR, Canonical LR, LALR, Operator Precedence Parser, Parser Generator (YACC).

**Learning Outcomes:** At the end of the module, the student will be able to

1. Apply rules to make the grammar ready for parsing (L3)
2. Construct Predictive Parsing table for the given grammar (L3)
3. Construct various LR Parsing tables for a given grammar (L3)

#### **UNIT-III**

**(10 LECTURES)**

##### **SYNTAX-DIRECTED TRANSLATION:**

Syntax-Directed Definition, S-Attributed SDD, L-Attributed SDD, Translation Schemes.

##### **TYPE CHECKING:**

Type Systems, Specification of a Simple type checker, Equivalence of Type Expressions, Type Conversions, Overloading of functions and operators.

**Learning Outcomes:** At the end of the module, the student will be able to

1. Classify S-Attributed SDD and L-Attributed SDD (L2)
2. Develop type system for a simple language (L3)
3. Describe two kinds of type conversions (L2)

**UNIT-IV****(10 LECTURES)****RUN-TIME ENVIRONMENTS:**

Source Language Issues, Storage Organization, Storage Allocation Strategies, Blocks, Access Links, Procedure Parameters, Displays, Parameter Passing, Symbol Tables.

**INTERMEDIATE CODE GENERATION:**

Intermediate Languages- Graphical Representations, Three address code, Implementations

**Learning Outcomes:** At the end of the module, the student will be able to

1. Summarize various storage allocation strategies (L2)
2. Explain various parameter passing methods (L2)
3. Make use of Quadruple, Triple and Indirect Triple representations to represent three address code. (L3)

**UNIT- V****(10 LECTURES)****CODE OPTIMIZATION:**

Introduction, Principle sources of optimization

**CODE GENERATION:**

Issues in the Design of a Code Generator, The Target Language, Basic Blocks and Flow Graphs, A Simple Code Generator, Peephole optimization

**Learning Outcomes:** At the end of the module, the student will be able to

1. Make use of function preserving optimizations and loop optimizations on a given code (L3)
2. Describe basic blocks and flow graphs (L2)
3. Apply rules to design a simple code generator (L3)

**TEXT BOOKS:**

1. Alfred V Aho, Monica S Lam, Ravi Sethi, Jeffrey D. Ullman, “*Compilers- Principles Techniques, and Tool*”, Pearson Education India, 2<sup>nd</sup> Edition, 2014.

**REFERENCE BOOKS:**

1. V. Raghavan, “*Principles of Compiler Design*”, 1<sup>st</sup> Edition, McGraw Hill Education, 2017.
2. Alfred V Aho, Ravi Sethi, Jeffrey D. Ullman, “*Compilers- Principles Techniques, and Tool*”, 2<sup>nd</sup> Edition, Pearson Education, 2008.
3. Kenneth C. Louden, “*Compiler Construction Design*”, 2<sup>nd</sup> Edition, Cengage, 2010.

**WEB REFERENCES:**

1. [https://swayam.gov.in/nd1\\_noc20\\_cs13/preview](https://swayam.gov.in/nd1_noc20_cs13/preview)