

---

---

**SCHEMES AND SYLLABUS**  
**PROGRAMMING LANGUAGES**

Subject Code: 10CS666  
Hours/Week : 04  
Total Hours : 52

I.A. Marks : 25  
Exam Hours: 03  
Exam Marks: 100

**PART - A**

**UNIT – 1** 7 Hours  
Introduction; Names, Scopes, and Bindings: The art of language design; Programming language spectrum; Why study programming languages? Compilation and interpretation; Programming environments. Names, scope, and bindings: The notion of binding time; Object lifetime and storage management; Scope rules; Implementing scope; The meaning of names within a scope; The binding of referencing environments; Macro expansion.

**UNIT – 2** 7 Hours  
Control Flow: Expression evaluation; Structured and unstructured flow Sequencing; Selection; Iteration; Recursion; Non-determinacy

**UNIT – 3** 6 Hours  
Data Types: Type systems; Type checking; Records and variants; Arrays; Strings; Sets; Pointers and recursive types; Lists; Files and Input/Output; Equality testing and assignment.

**UNIT – 4** 6 Hours  
Subroutines and Control Abstraction: Review of stack layout; Calling sequences; Parameter passing; Generic subroutines and modules; Exception handling; Coroutines; Events.

**PART – B**

**UNIT – 5** 6 Hours  
Data Abstraction and Object Orientation: Object oriented programming; Encapsulation and Inheritance; Initialization and finalization; Dynamic method binding; Multiple inheritance; Object oriented programming revisited.

**UNIT – 6** 7 Hours  
Functional Languages, and Logic Languages: Functional Languages: Origins; Concepts; A review/overview of scheme; Evaluation order revisited; Higher-order functions; Functional programming in perspective. Logic Languages: Concepts; Prolog; Logic programming in perspective.

**UNIT – 7** 6 Hours  
Concurrency: Background and motivation; Concurrency programming fundamentals; Implementing synchronization; Language-level mechanisms; Message passing.

## UNIT – 8

7 Hours

Run-Time Program Management: Virtual machines; Late binding of machine code; Inspection/introspection.

**Text Books:**

1. Michael L. Scott: Programming Language Pragmatics, 3Edition, Elsevier, 2009.  
(Chapters 1.1 to 1.5, 3.1 to 3.7, 6 excluding the sections on CD, 7 excluding the ML type system, 8, 9, 10 excluding the sections on CD, 11 excluding the sections on CD, 12, 15. Note: Text Boxestitled Design & Implementation are excluded)

**Reference Books:**

1. Ravi Sethi: Programming languages Concepts and Constructs, 2 Edition, Pearson Education, 1996.
2. R Sebesta: Concepts of Programming Languages, 8 Edition, Pearson Education, 2008.
3. Allen Tucker, Robert Nonan: Programming Languages, Principles Paradigms, 2nd Edition, Tata McGraw-Hill, 2007.

**TABLE OF CONTENT**

1. Introduction; Names, Scopes, and Bindings	5-12
2. Control Flow	14-20
3. Data Types	22-38
4. Subroutines and Control Abstraction	40-60
5. Data Abstraction and Object Orientation	62-67
6. Functional Languages, and Logic Languages	69-80
7. Concurrency	82-93
8. Run-Time Program Management	95-96