

Principles of Programming Languages - Video course

Prerequisites: Data Structures, and an introductory course in Computer Architecture.

Course objective:

To introduce the major programming paradigms, and the principles and techniques involved in design and implementation of modern programming languages. To introduce frameworks for specifying and reasoning about programming languages.

Course contents:

Notions of syntax and semantics of programming languages; introduction to operational/natural semantics of functional and imperative languages. Data abstractions and control constructs; block-structure and scope, principles of abstraction, qualification and correspondence; parameter passing mechanisms; runtime structure and operating environment; practical and implementation issues in run-time systems and environment; abstract machines; features of functional and imperative languages; the untyped and simply-typed Lambda calculus' type systems for programming languages including simple types and polymorphism; objects, classes and inheritance in object-oriented languages.

Course Organization (topics and hours)

Abstract Syntax and Semantics (algebraic)	4	
Operational Semantics (functional paradigm, logic paradigm)	5	5
Block Structure and Control (imperative paradigm)	4	
Principles of Abstraction, Qualification, Correspondence	4	
Abstract Machines and Runtime structures		
	4	
Parameter Passing and Runtime stacks		
	5	
Type Systems and Type checking/inference		5
Object-oriented systems		
	5	
Lambda Calculus		
	5	

Brief description of laboratory activities:

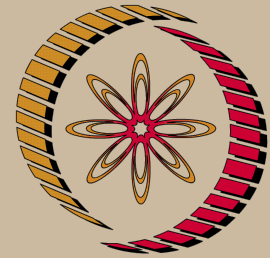
Programming exercises will include use of tools for lexical analysis and parsing; representation of abstract syntax; abstract machines for imperative and functional languages; translations from high-level languages to abstract machine instructions; implementation of run-time structures and parameter-passing; type checking and type-inference; term reduction.

Suggested texts and reference materials:

1. *Programming Language Pragmatics*. Michael Scott, Morgan Kaufmann, 2000.
2. *Essentials of Programming Languages*. Friedman, Wand and Haynes, Prentice-Hall International (PHI), 1998.
3. *Principles of Programming Languages*. Tennant. PHI, 1981.

Resources required for the course:

Modern language compilers (SML, Prolog, Java, ...)



NP-TEL

NPTEL

<http://nptel.ac.in>

Computer
Science and
Engineering

Coordinators:

Prof. S. Arun Kumar

Department of Computer Science
and Engineering IIT Delhi

PCs for programming exercises, preferably running Linux.

A joint venture by IISc and IITs, funded by MHRD, Govt of India

<http://nptel.ac.in>