

BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE, PILANI(Raj.)
INSTRUCTION DIVISION
FIRST SEMESTER 2015-2016
Course Handout (Part II)

Date: 03-08-2015

In addition to part I (General Handout for all courses appended to the time table) this portion gives further specific details regarding the course.

Course No: CS F301/ IS F301
Course Title: Principles of Programming Languages
Instructor-in-charge: Vandana Agarwal (vandana@pilani.bits-pilani.ac.in)
Instructors: Saiyedul Islam <sislam@pilani.bits-pilani.ac.in>,
Mohit Sati <mohit.sati@pilani.bits-pilani.ac.in>,
Priyanka Gaur <p2014410@pilani.bits-pilani.ac.in>
Course Website: nalanda.bits-pilani.ac.in (Learning Management System)

Course Description

The course covers features of programming languages and introduces the main programming paradigms. It covers, in detail, the semantics of the features of programming languages –Control Abstraction, Data Types and Data Abstraction, Scope and Parameter passing and Concurrency related features. It covers various aspects of runtime environments like global and local data, code, function call stacks, dynamically allocated data, runtime features for exceptions and threads. Introduction to programming paradigms. Functional paradigm – formal elements of lambda calculus, introduction to syntax of common functional programming languages and programming exercises that explore the functional paradigm. Logic programming paradigm - formal elements of logic programming and programming tasks that explore the logic paradigm. Scripting as a paradigm. Domain specific languages.

Text and Reference Books:

[T1]. Ravi Sethi , "Programming Languages: Concepts and Constructs" 2nd Edition by Addison Wesley.

[R1]. Robert W. Sebesta, "Concepts of Programming Languages", 2nd Edition by The Benjamin/Cummings Publishing Company, Inc.

[R2]. Aho, Lam, Sethi and Ullman, "Compilers Principles, Techniques, and Tools". Pearson Education. Low Price Edition. 2004

Lecture Schedule

Lecture Numbers	Topic	Reading References
1	Introduction and Motivation	
2	Language Paradigms, Imperative vs. Declarative Styles of Programming; Programming Languages that support these styles, Compilers. Features of a Programming Language, compilers	Chapter 1 (T1 and R1)
3-4	Language Description: Syntactic Structure,	Chapter 2(T1)
5-7	Control Abstraction: Structured programming, Loops and jumps	Chapter 3 (T1)
8-9	Types: Data Representation, Primitive and Structured Data types, ADTs; Type checking vs. Type Inferencing, Type Equivalence and	Chapter 4 (T1)

	Subtyping.	
10-11	Data Layout models: Primitive Data, Structured Data (Arrays/Lists, Records/Structures, Variants/Unions), Objects (Simple objects and classes, Inheritance Models – Single vs. Multiple, Interfaces and abstract/virtual classes).	Class Notes
12-13	Basic Runtime Environments: Code vs. Data, Global vs. Local Data, Functions and Call Stacks, Dynamically allocated data and heaps;	Chapter 7 (R2)
14-16	Procedures: Introduction to procedures, Recursion, Parameter Passing Methods, Call-by-Value, Call-by-Reference, Call by Value Result. Scope rules for names, static and dynamic scope rules, nested scope, Activation Records, Lexical Scope, Dangling Pointers, Tail Recursion Elimination	Chapter 5 (T1)
17-18	Object Oriented Abstraction: Object Oriented Programming Paradigm and features, Class hierarchy, Inheritance, Information hiding,	Chapter 7 (T1)
19-20	Functional Programming: Introduction and basic elements of Functional Programming	Chapter 8 (T1)
21-22	Functional Programming: Lists and associated Operations, Function Declaration, Higher Order Functions, Polymorphism, Data Types	Chapter 9, 10 (T1)
23-25	Logic Programming: Relations, First Order Logic, Logic Programming and Horn-Clause Programming, Unification, Deduction and Search as a strategy for deduction, Indexing, Pruning, Definite Clause Grammars.	Chapter 11(T1)
26	Concurrency and Distribution: Threads, Synchronization features (semaphores/ monitors/ locks), Shared Memory programming.	Chapter 12 (T1)
27-28	Complex Runtime Environments: Exception handling and non local jumps, Threads and cactus stacks	Chapter 12 (R1)

5. Evaluation Components

Component	Mode	Duration	Date	Weight
Mid Semester Test	Closed Book	90min	6/10 8:00 - 9:30 AM	30%
Quiz	Closed Book	50min	TBA	15%
Assignments	Take home			10%
Comprehensive	Closed / Partially Open Book	3 hours	3/12 FN	45%

6. Notices: All notices concerning this course will be put on the **CSIS notice board** OR the course website as appropriate.

7. Makeup Policy:

- **Permission of the Instructor-in-Charge is required** to take a make-up

- **Make-up applications must be given to the Instructor-in-charge personally.**
- *A make-up test shall be granted only in genuine cases where - in the Instructor's judgment - the student would be physically unable to appear for the test.*
- In case of an unanticipated illness preventing a student from appearing for a test, the student must present a Medical Certificate from BITS hospital.
- In case of unanticipated absence for a test due to a trip out of Pilani, the student must present a letter from his/her Warden or the Chief Warden certifying such absence and the reason(s).
- Requests for make-up for the comprehensive examination – under any circumstances – can only be made to Dean, Instruction Division.

Instructor-in-charge

CS F301/IS F301