Spring 2013

CS 7100: Advanced Programming Languages

Krishnaprasad Thirunarayan
Wright State University - Main Campus, t.k.prasad@wright.edu

CS 7100 Advanced Programming Languages

- **Instructor:** T. K. Prasad
- **Phone No.:** (937)-775-5109
- **Email:** t.k.prasad@wright.edu
- **Home Page**: http://knoesis.wright.edu/tkprasad

- **Quarter:** Spring, 2013
- **Class Hrs:** MW, 4:40-6pm, 103 Bio. Sci.
- **Office Hrs:**MW, 3:30-4:30pm (395 Joshi) (or by appointment)

Course Objectives

To provide a solid foundation for studying advanced topics in Programming Language Specification and Design.
Specifically, the student learning outcomes include:

- Demonstration of knowledge of programming language design
- Creation and understanding algebraic specification of abstract data types
- Developing/modifying interpreter-based specification (operational semantics) of programming languages
- Demonstration of knowledge of attribute grammar framework and axiomatic-basis for computer programming
- Familiarity with highlights of contemporary, multi-paradigm languages

Prerequisites CS 3180/5180 Comparative Languages

Course Description

This course introduces concepts related to the specification and design of high-level programming languages. It discusses different programming paradigms, algebraic specification and implementation of data types, and develops interpreters in Scheme for specifying operationally the various programming language features/constructs (spanning simple expression language to class-based object-oriented language). It also introduces attribute grammar framework that is convenient for automatic translation and axiomatic semantics formalism that assists in program verification. The programming assignments will be coded in Scheme using Racket IDE.

Course Load

The course load includes homeworks and programming assignments worth 30 points, a midterm exam worth 30 points, and a final exam worth 40 points. Normally, the exams are open notes and open book.

Reference

1. Friedman, Wand and Haynes: Essentials of Programming Languages, 2nd (preferred) or 3rd Edition. MIT Press, 2001. ISBN 0-262-06217-8 (code.zip)
2. R. Kent Dybvig: The Scheme Programming Language, 4th Edition. The MIT Press, 2009.

1. Guttag, J.V., "Abstract Data Types and the Development of Data Structures," CACM, vol. 20, No. 6, June 1977, pp. 396-404.
2. Chapter 1 of Guttag, J. V., et al, Larch: Languages and Tools for Formal Specification, Springer-Verlag, NY, 1993.
3. H. Abelson and G. J. Sussman, Structure and Interpretation of Computer Programs, 2nd Ed., MIT Press, 1996.
4. M. Felleisen, R. B. Findler, M. Flatt, and S. Krishnamurthi, How to Design Programs, MIT Press, 2002.
5. Scheme : Language Reference Manual
6. The Teaching About Programming Languages Project
7. Racket Download Site (http://racket-lang.org/)

Grading

The letter grades will be assigned using the following scale: A[90-100], B[80-90), C[70-80), D[60-70), and F[0-60). However, I reserve the right to adjust the scale somewhat to utilize the gaps in the distribution.

Class Schedule and Syllabus

	Topics with links to Lecture Notes	Addl. Readings (EOPL-2nd ed)
Class 1	Evolution of Programming Languages	Turing Awards
Class 2	Why specify?	
Class 3	Scheme Metalanguage	Chap 1.1, 1.2
Class 4	Abstract Data Types: Algebraic Specs	Chap 2
Class 5	(continue)	
Class 6	Programming Paradigms	
Class 7	Abstract Syntax and its Representation	Chap 2
Class 8	Interpreter for a Simple Expression Language	Chap 3
Class 9	User-Defined Functions; Scoping	Chap 1.3, 3
Class 10	Implementing Recursion	Chap 3

Class 11	Closures and Streams	
Class 12	**Midterm Exam** (February 13)	
Class 13	Imperative Programming : Assignment	Chap 3
Class 14	Interpreter for an Object-Oriented Language	Chap 5
Class 15	(oopl.ps)	
Class 16	Introduction to Attribute Grammars	
Class 17	(continue)	
Class 18	Introduction to Axiomatic Semantics	
Class19	(continue)	
Class 20	Case Study: Specification of COOL	
Class 21	(continue)	
Class 22	Case Studies: Specification of Java	(Old Course)
Class 23	(continue)	
Class 24	Case Study: Design of multi-paradigm languages	
Class 25	(continue)	
	Final Exam (April 22, 5:45pm-7:45pm)	

Old Exams (Spring 2011)

- Midterm (pdf).
- Final (pdf).

Assignments (Spring 2013)

- Assignment 1 .
- Assignment 2 .
- Assignment 3 . (asg3.ppt)
- Assignment 4.
