

Wright State University

## CORE Scholar

---

Computer Science & Engineering Syllabi

College of Engineering & Computer Science

---

Fall 2007

### CS 784: Programming Languages

Krishnaprasad Thirunarayan

*Wright State University - Main Campus*, [t.k.prasad@wright.edu](mailto:t.k.prasad@wright.edu)

Follow this and additional works at: [https://corescholar.libraries.wright.edu/cecs\\_syllabi](https://corescholar.libraries.wright.edu/cecs_syllabi)



Part of the [Computer Engineering Commons](#), and the [Computer Sciences Commons](#)

---

#### Repository Citation

Thirunarayan, K. (2007). CS 784: Programming Languages. .  
[https://corescholar.libraries.wright.edu/cecs\\_syllabi/232](https://corescholar.libraries.wright.edu/cecs_syllabi/232)

This Syllabus is brought to you for free and open access by the College of Engineering & Computer Science at CORE Scholar. It has been accepted for inclusion in Computer Science & Engineering Syllabi by an authorized administrator of CORE Scholar. For more information, please contact [library-corescholar@wright.edu](mailto:library-corescholar@wright.edu).

# CS 784 Programming Languages

---

- **Instructor:** T. K. Prasad
  - **Phone No.:** (937)-775-5109
  - **Email:** [t.k.prasad@wright.edu](mailto:t.k.prasad@wright.edu)
  - **Home Page:** <http://www.cs.wright.edu/~tkprasad>
  - **Quarter:** Fall, 2007
  - **Class Hrs:** TTh, 6:05pm-7:20pm, 047 Millett
  - **Office Hrs:** TTh, 3:30pm-4pm, 5:30pm-6pm, 395 Joshi Center (or by appointment)
- 

## Course Objectives

To provide a solid foundation for studying advanced topics in Programming Language Specification and Design.

---

## Prerequisites CS 480/680 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 for specifying operationally the various programming language features/constructs. It also introduces attribute grammar formalism and axiomatic semantics briefly. The programming assignments will be coded in Scheme.

---

## 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.

---

## Text

1. Friedman, Wand and Haynes: Essentials of Programming Languages. 2nd Edition. MIT Press, 2001. ISBN 0-262-06217-8
2. R. Kent Dybvig: The Scheme Programming Language, 3rd Edition. The MIT Press, 2003.

## Reference

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. Chez Scheme Download Site (<http://www.scheme.com>)
8. DrScheme Download Site (<http://www.drscheme.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

<b>Topics with links to Lecture Notes</b>	<b>Addl. Readings</b>
<b>Class 1</b> <a href="#">Evolution of Programming Languages</a>	
<b>Class 2</b> <a href="#">Scheme Metalanguage</a>	Chap 1.1, 1.2
<b>Class 3</b> <a href="#">Abstract Data Types: Algebraic Specs</a>	Chap 2
<b>Class 4</b> (continue)	
<b>Class 5</b> <a href="#">Programming Paradigms</a>	
<b>Class 6</b> <a href="#">Abstract Syntax and its Representation</a>	Chap 2
<b>Class 7</b> <a href="#">Interpreter for a Simple Expression Language</a>	Chap 3
<b>Class 8</b> <a href="#">User-Defined Functions; Scoping</a>	Chap 1.3, 3
<b>Class 9</b> <b>Midterm Exam</b> (Oct 4)	
<b>Class 10</b> <a href="#">Implementing Recursion</a>	Chap 3
<b>Class 11</b> <a href="#">Closures and Streams</a>	
<b>Class 12</b> <a href="#">Imperative Programming : Assignment</a>	Chap 3
<b>Class 13</b> (continue)	
<b>Class 14</b> <a href="#">Interpreter for an Object-Oriented Language</a>	Chap 5
<b>Class 15</b> (oopl.ps)	
<b>Class 16</b> <a href="#">Introduction to Attribute Grammars</a>	
<b>Class 17</b> (continue)	
<b>Class 18</b> <a href="#">Introduction to Axiomatic Semantics</a>	
<b>Class 19</b> (continue)	
<b>Class 20</b> Wrap-up	

**Final Exam** (Nov 15, 8pm-10pm)

---

## Old Exams (Fall 2005)

- [Midterm \(pdf\)](#).
  - [Final \(pdf\)](#).
- 

## Assignments (Fall 2007)

- [Assignment 1](#) .
  - [Assignment 2](#) .
  - [Assignment 3](#) .
- 

*T. K. Prasad* (08/22/07 11:25:40 AM)