Computer Science 4010/5010 (Section 001)
Programming Languages
Tennessee Technological University Department of Computer Science
MWF 2:30-3:25 p.m. in Bruner 404

Instructor: Martha Kosa
Office: Bruner 214
Office Phone: 372-3579
Email: mjkosa@tn-tech.edu
URL: http://www.csc.tntech.edu/~mjkosa/
Office Hours: 3:30-4:30 MWF, 10-11 T, 2:00-4:30 TR
However, if you see the light on in my office at other times, please knock and I will try to help you. You can also call or email to make an appointment.

Textbooks: Programming Language Pragmatics, by Michael L. Scott, Morgan Kaufmann, 2009, and Seven Languages in Seven Weeks, by Bruce A. Tate, The Pragmatic Bookshelf, 2010, plus course notes as necessary.

Catalog Description: Lec. 3, Credit 3. Concepts distinguishing modern programming languages with emphasis on language design, implementation, and run-time behavior.

Purpose: The purpose of this course is to gain an appreciation and understanding of the ideas behind high-level programming languages and practice with different paradigms (imperative, functional, logic, object-oriented).

Prerequisites: CSC 2710 and CSC 3410.

Major Teaching Methods: Lectures, demonstrations, discussions

Course Requirements: Four in-class tests (tentatively scheduled for Sept. 18, Oct. 9, Nov. 2, and Nov. 23) and a final examination (Thurs., Dec. 10, at 1:00 p.m.) will each constitute 14% of your grade. 30% will come from several programming assignments, possible in-class quizzes, and possible homework assignments. Students in CSC 5010 may be required to complete more involved assignments and/or test questions.

Grading Scale: The standard 10-point scale (90+ is an A, etc.) applies.

Attendance Policy: You are expected to attend every class. You are responsible for all assignments and material covered during all class meetings whether you are present or not.

Students with a disability requiring accommodations should contact the Office of Disability Services (ODS). An Accommodation Request (AR) should be completed as soon as possible, preferably by the end of the first week of the course. The ODS is located in the Roaden University Center, Room 112; phone 372-6119.

Should normal classroom activities be disrupted by a flu outbreak, the format for this course may be modified to enable completion. In that event, you will be given new instructions for continuation of the course.
Projected Course Topics

1. Motivation for studying programming languages
2. Programming Language Syntax
   a. Specification
   b. Scanning
   c. Parsing
3. Names, Scopes, Bindings
   a. Binding time
   b. Object lifetime and storage management
   c. Scope rules and implementation
   d. Meaning of names within a scope
4. Functional Programming Languages
   a. History
   b. Concepts
   c. Example Languages (such as Scheme, ML, Haskell, Scala)
   d. Evaluation order
   e. Higher-Order Functions
   f. Theoretical Foundations
5. Data Abstraction and Object Orientation
   a. Object-Oriented Programming
   b. Encapsulation and Inheritance
   c. Initialization and Finalization
   d. Dynamic Method Binding
   e. Multiple Inheritance
6. Logic Programming Languages
   a. Concepts
   b. Example Languages (such as Prolog)
   c. Theoretical Foundations
   d. Implementation issues
7. Scripting Languages
   a. Definition
   b. Problem Domains
   c. Web Scripting
   d. Innovative Features

List of Objectives

1. Improve ability to learn new programming languages quickly.
2. Use a variety of programming languages from different paradigms, such as Scheme, ML, Scala, and Prolog, with some proficiency to solve programming problems.
3. Understand how several programming language features are implemented.
4. Compare and contrast different programming language paradigms.