CSCI 256: Principles of Programming Languages

Syllabus – Spring 2010

Ben Coleman
coleman@cs.moravian.edu
214 PPHAC

Office Hours: M-F 10:10 – 11:10
or by appointment
Office Phone: 610-625-7781

Course Description

A study of the features of programming languages and of the methods used to specify and translate them. Topics covered include block structure, naming, procedure invocations and parameter passage, data types, data accessing, syntactic analysis, and correspondence of source language and object language constructs.

Goals

Upon completion of this course, a successful student will be able to:

- Describe the important features of the four major paradigms of programming languages.
- Implement a top-down parser for a simple language.
- Identify the language design decisions for various programming languages.
- Describe the implementation of key language features including control structures, subprograms, and object-oriented concepts.
- Implement classic concurrent patterns in Java.

Required Text

In addition to the following text, supplementary readings will be given periodically during the semester.

- Concepts of Programming Languages, ninth edition, by Robert W. Sebesta

You should expect to spend about an hour before each class session working through the readings. This means reading the text for detail and working to learn vocabulary – not just skimming through the material before class.

Course Policies

- **Responsibilities** – Your attendance is expected at each class meeting. You are also responsible for the contents of reading assignments, handouts, class activities, and class email.
- **Late Policy** – I understand that life sometimes gets in the way of getting work done. Consequently, late assignments will be accepted without penalty in the class after the assignment was due. However, this policy should not be used as a crutch, and if you frequently use it I will deduct from your grade. After the next class session, late work will not be accepted unless there are exceptional circumstances.
- **Extensions** – In a similar vein, I am generous with extensions on work if you approach me *before* the day the assignment is due.
- **Academic Honesty** – Except on tests, you are *encouraged* to discuss the material and work with other students in the course. Specifically, on homework and programming assignments you may discuss any portion of the assignment with your fellow students. This policy does not allow you to copy another student's work verbatim – you must produce your own code or write-up of the material. Work together to learn the concepts, but keep in mind that you are ultimately responsible for the material on the tests.
- **Disabilities** – If you have a disability that may affect your performance in this course, please contact me immediately to discuss academic accommodations.

Graded Material

- **Homework** – The goal of homework problems is for you to practice using the current course content and to explore the topics in more detail. Problems will be assigned nearly every class session and will be due the next class.
- **Projects** – Three projects will be assigned during the semester. Each will have a programming component as well as some form of written submission. Specifications for each assignment will be distributed in class.
- **Tests** – Two tests will be given during the semester on Wednesday, February 17 and Monday April 12. You may only re-schedule a test for college-approved absences or documented illness. In either case, you must contact me *before* the beginning of the test.
- **Final** – The final will be cumulative and will be given in-class on Wednesday, May 5 at 1:30 p.m. Any change to the final exam schedule must be approved by both me and the dean of students.

Grade Determination

Homework	25%
Programming Assignments	25%
Tests	30%
Final	20%

All grades will be computed on the standard scale using plusses and minuses.

Course Outline

Date	Reading	Topic
M Jan 18		• Day 1 Activities
W Jan 20	• Ch 1: pp. 2-33	• Language Design Issues
F Jan 22	• 15.1-15.4: pp. 656-664	• Functional Programming
M Jan 25	• 15.5: pp. 664-682	• Scheme
W Jan 27	• Handout	• Deep Recursion in Scheme
F Jan 29	• Handout	• Advanced Scheme
M Feb 1	• 3.1-3.3: pp. 116-134	• Syntax and Grammars
W Feb 3	• 3.4: pp. 134-141	• Attribute Grammars
F Feb 5	• 3.5.3: pp. 150-164	• Axiomatic Semantics
M Feb 8	• 4.1-4.2: pp. 172-182	• Lexical Analysis
W Feb 10	• 4.3-4.4: pp. 182-194	• Top-Down Parsing
F Feb 12	• 4.5: pp. 194-204	• Bottom-Up Parsing
M Feb 15	• 5.1-5.4: pp. 208-225	• Names, Variables, and Bindings
W Feb 17		• Test #1
F Feb 19	• 5.5-5.6: pp. 225-235	• Scope and Lifetime
M Feb 22	• 5.7-5.8: pp. 237-240	• Reference Environments and Constants
W Feb 24	• 6.1-6.6: pp. 248-279	• Primitives and Arrays
F Feb 26	• 6.7: pp. 282-304	• Records, Points, and Reference Types
M Mar 1	• 6.10-6.13: pp. 304-314	• Type Checking and Strong Typing
W Mar 3	• Ch 7: pp. 320-344	• Expressions and Assignment Statements
F Mar 5		• "Slip Day"
M Mar 7 – F Mar 12		• Spring Break
M Mar 15	• Ch 8: pp. 350-385	• Control Structures
W Mar 17	• 9.1-9.4: pp. 392-406	• Sub-Program Basics
F Mar 19	• 9.5: pp. 406-427	• Parameter Passing Modes
M Mar 22	• 9.6-9.11: pp. 427-443	• Overloading and Generic Sub-Programs
W Mar 24	• 10.1-10.3: pp. 450-462	• Implementation of Sub-Programs
F Mar 26	• 10.4: pp. 462-470	• Nested Sub-Programs
M Mar 29	• Ch 11: pp. 482-518	• Abstract Types
W Mar 31	• 12.1-12.9: pp. 524-560	• Object-Oriented Programming

Date	Reading	Topic
F Apr 2	• 12.10: pp. 560-564	• Implementing Objects
M Apr 5		• Easter Break
W Apr 7	• 13.1-13.5: pp. 570-586	• Concurrency
F Apr 9	• 13.6-13.9: pp. 587-611	• Language Concurrency
M Apr 12		• Test #2
W Apr 14	• Handout	• Classic Concurrency Examples
F Apr 16	• Handout	• Classic Concurrency Examples
M Apr 19	• 14.1: pp. 616-622	• Exception Handling
W Apr 21	• 14.2-14.4: pp. 622-643	• Language Examples of Exceptions
F Apr 23	• 14.5-14.6: pp. 643-650	• Event Handling and Java
M Apr 26	• Handout	• Swing
W Apr 28	• Handout	• More Swing
F Apr 30	• Handout	• Swing and Concurrency

The details of this syllabus and schedule are subject to change based on our progress through the material.