Mathematics and Computer Science Department

CSCI 170: Analysis of Programming Languages
Fall Semester 2011
http://www.cs.clarku.edu/~jmagee/csci170/

Class Meetings @ BP 326
Mon, Thu 2:50 pm - 4:05pm

John Magee, Instructor
jmagee@clarku.edu
Always include “CSCI170” in the subject.

Office hours @ BP 332:
Tue 2:45 pm – 4:00 pm;
Fri 11:00am – 12:30pm
and by appointment

Teaching Assistant

Ryan Osbaldeston
rosbaldeston@clarku.edu
Always include “CSCI170” in the subject.

Tutoring Hours: TBA,
and by appointment
Tutoring hours held at CS lab, BP 310.

Course Goals
To understand the principles underlying the design and implementation of programming languages.
We will study the principles involved in specifying language syntax and semantics, drawing on specific
examples from various programming languages, look at some of the basic principles of compilers and
interpreters, and examine the main alternative approaches to programming language design
(imperative, functional, object-oriented, etc.). We will pay particular attention to the imperative and
functional approach, since practical experience will be gained by writing part of a Scheme interpreter
in C.

Major topics of this course will include:

- Introduction to C
- Scheme
- Syntax and Parsing
- Semantics
- Abstract Data Types
- Control
- Object Oriented Programming
- Formal Semantics
- Functional Programming
- Logic Programming
Prerequisites

Prerequisites for this course require a grade of C or better in CSCI 121 – Data Structures and MATH 114 – Discrete Mathematics.

Books

*Programming Languages, Principles and Practice, 3rd Edition*
by Kenneth C. Louden and Kenneth A. Lambert

Other online readings and tutorials will be posted to the schedule page.

Software

For the applied parts of the course, we will be programming using the C language.
All of these are available in the CS computer lab (BP 310):

- Netbeans IDE (for C and C++ development)
- Linux programming tools (gcc, make, etc.)

Grading

The following percentages are tentative and may be changed at my discretion at any time:

<table>
<thead>
<tr>
<th>Component</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Programming Assignments/Projects</td>
<td>40%</td>
</tr>
<tr>
<td>Midterm Exam</td>
<td>25%</td>
</tr>
<tr>
<td>Final Exam</td>
<td>35%</td>
</tr>
</tbody>
</table>

The Project

Writing an interpreter might sound like a difficult task, but (at least for Scheme) it is really not that hard if you think about it the right way (and that's why we're here!). You will be given the basic input/output procedures on which the interpreter can be built. The project will be broken down into a set of five smaller, manageable assignments (detailed information will be given shortly). Grades are based on performance (does it work?), style, design and documentation.

Withdrawing from the Course

If you feel that you want to drop or withdraw from the class, please come talk to me about it as early as possible; I want to help you succeed, but you need to ask for help.

Add/Drop period ends Wednesday September 7, 2011 at 11:00pm
The last date to withdraw and receive a “W” grade is Friday, November 4, 2011.
Policies and Miscellaneous

The official administrative business of this class will be conducted by email. Grade questions/disputes, explanation of absence, etc. will be processed via email so that we both have a written record of what was agreed. Feel free to discuss in person and follow up with email for the record.

Attendance and discussion/asking questions are expected and will be reflected in your grade. If you must be absent, please email me in advance to let me know why you won’t be in class, and to let me know what you will do to keep up with the assignments. CSCI 170 is not a correspondence course. Inadequate attendance is grounds for a grade of F.

Assignments are due on the date stated on the homework assignment (to be posted on web).

- Assignments received within the next class will be accepted with a 10% penalty.
- Assignments received within 2 classes of the deadline will be accepted with a 20% penalty.
- Assignments received more than 2 classes past the deadline will not be accepted or graded.

Plan your work accordingly, and work on all assignments as soon as they are given so you can ask questions in class and get assistance in the labs and tutoring hours.

Students are responsible for ensuring that assignments are correctly submitted. If you have a question or problem, seek help from CSCI170 staff immediately.

No special make-up work will be accepted after the end of the semester. Don’t even ask. In the event of a documented major medical problem, a grade of Incomplete will be given pending the submission of complete work. However, make up work “to improve one’s grade” will not be accepted.

It is the student’s responsibility to retain all papers and exams that have been graded and returned. Should these original documents not be available in the event of a grade dispute, I will need to defer to the own records.

Grades are not negotiable. Don’t even ask – just do the work and you’ll get the grade you deserve. Of course, please bring any clerical grading errors to my attention by email and I will gladly fix them.
Plagiarism, Collaboration, and Collusion

All CSCI170 homework assignments are independent work.

It is the student’s responsibility to know and understand the Clark University Academic Integrity policy, which is within the Academic Advising Handbook (The Blue Book) available at the Academic Advising Center.

In addition to the definition of plagiarism in the handbook, with respect to CSCI170, plagiarism is specifically defined to include (but is not limited to) the following:

- collaboration on the solutions/code you write
- copying any part of someone else’s assignment/program, even if you have permission and/or have modified the code
- sharing or giving your assignment/code or even a subset of your assignment/code to another student to review
- reviewing another student’s solution (including from past semesters)
- reviewing solutions on the internet

It is my policy to use automatic plagiarism detection software, and suspicious similarities will be uncovered. The University takes acts of cheating and plagiarism very seriously; violators may be suspended or fail the course.

What is acceptable cooperation?
Cooperation is recommended in understanding programming concepts and system features. You are encouraged to discuss the homework problem statements and expected output, and to seek and receive help with C, Scheme, Linux, the Netbeans IDE, and other tools.

However, each student must write his or her own solution/code and other deliverables independently.

Acknowledge Help Received
To avoid any misconception about academic conduct, you should acknowledge the names of anybody you discuss assignments with.