COMP 3007: Programming Paradigms
Course Outline for Winter 2015 (subject to change, last updated December 2014)

Most recent outline available online:
https://docs.google.com/document/d/1qE_BR6b45ewW2XpCt-RP7c7ytenXbSRPYZxyi12QIt4/edit?usp=sharing

Course Details

4:00 - 5:30pm
Mondays and Wednesdays
MC 5050

No classes on February 16 / 18 due to Reading Week.

Instructor

Gail Carmichael
Office: 5326 HP
gailcarmichael@cmail.carleton.ca

Office hours:
10:30am - 12:00pm
Mondays and Wednesdays
5326 HP

Teaching Assistants

Office hours will be held in 1170 HP starting the week of January 12.

<table>
<thead>
<tr>
<th>Name</th>
<th>Email</th>
<th>Office Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>David Robillard</td>
<td><a href="mailto:DavidRobillard@cmail.carleton.ca">DavidRobillard@cmail.carleton.ca</a></td>
<td>TBA</td>
</tr>
<tr>
<td>Peter Simonyi</td>
<td><a href="mailto:psimonyi@connect.carleton.ca">psimonyi@connect.carleton.ca</a></td>
<td>TBA</td>
</tr>
<tr>
<td>Zachary Dawson</td>
<td><a href="mailto:ZackDawson@cmail.carleton.ca">ZackDawson@cmail.carleton.ca</a></td>
<td>TBA</td>
</tr>
</tbody>
</table>
Course Description

In this course, you will spend most of the time learning functional programming with Scheme, after which you will be introduced to logical programming with Prolog.

Why learn functional programming? As this page via Washington University explains:

While programming with a functional language is still fundamentally similar to programming with any other type of language (examples of others being imperative or logic), it represents programs and algorithms through distinct forms of abstraction and gives you a new toolset with which to solve programming problems. Additionally, many of the techniques of functional programming are beginning to permeate new mainstream languages, so taking the time now to develop a thorough understanding of them is an investment which will pay great dividends.

This tends to be a “get it” or “not” kind of course, so to help you succeed, we will use interactive techniques such as Poll Everywhere and live coding. You are strongly encouraged to bring at least a mobile device and preferably your laptop to each class.

Calendar description: An introduction to alternative programming paradigms such as functional, constraint-based, concurrent, and logic programming.

Learning Objectives

By the end of the course, the successful student will be able to:

- describe what functional and logic programming are, and how they differ from other more familiar paradigms
- understand the basics of Scheme, including:
  - the substitution and environment models for expression evaluation
  - lambda expressions and related forms, such as let
  - conditional expressions
  - character and string manipulation
  - basic input/output
  - mutable data
- more deeply understand recursion in Scheme, including:
  - how both recursive and iterative processes can arise from recursive procedures
- build and manipulate lists of data using pairs in Scheme
- use lists to build and manipulate more complex abstractions in Scheme, including:
  - rational numbers
  - binary search trees
  - polynomials
- objects
- streams
- create part of a Scheme metacircular evaluator
- understand the basics of Prolog, including:
  - facts, rules, goals and questions
  - arithmetic
  - relational operators
  - simple I/O
  - scanning and backtracking
- use structures and lists in Prolog
- control backtracking in Prolog
- solve common problems in Prolog, including:
  - sorting
  - shortest path
- create a simple adventure game with Prolog

Assessment

You must pass the final exam in order to pass the course.

<table>
<thead>
<tr>
<th>Assignments</th>
<th>30%</th>
</tr>
</thead>
<tbody>
<tr>
<td>(6, bi-weekly)</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Quizzes</th>
<th>30%</th>
</tr>
</thead>
<tbody>
<tr>
<td>(best 2 of 3, Jan 28, Feb 25, Mar 18)</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Final Exam</th>
<th>40%</th>
</tr>
</thead>
<tbody>
<tr>
<td>(to be announced)</td>
<td></td>
</tr>
</tbody>
</table>

Assignments

You must complete your assignments independently and submit them on time via cuLearn. Late assignments will not be accepted. You may speak with TA’s or the instructor if you need help. Do not post solutions to the discussion forums.

Quizzes

Instead of one or two larger midterms, we will have three smaller quizzes in class. This will provide you with more rapid feedback about your understanding and better prepare you for the final exam.
Final Exam

The time and place as well as the format of the final exam will be announced later in the term. Do not make travel plans until the dates are known as no advance exams will be given.

Attendance

Although notes and resources will be available online, it is expected that you will attend class. In assignments and tests, you will be responsible for code examples, discussions, activities, and so on that you can only see if you attend and participate in lectures.

Textbooks

We will use freely available resources online as well as Programming in Prolog: Using the ISO Standard (William Clocksin and Christopher S. Mellish).

Software and Hardware Requirements

When writing Scheme code, you will use the Racket environment: http://racket-lang.org/

For Prolog assignments, you will use SWI-Prolog: http://www.swi-prolog.org/

SCS Computer Accounts

Any student taking an SCS course qualifies to have an SCS account. SCS accounts can be created at the following URL: http://www.scs.carleton.ca/newacct. SCS students can access one of the designated labs for your course.

The labs are operational 7 days a week 24 hours per day, please be advised that the building will be closed overnight, Mon. - Fri. 23:00 - 8:00 and on weekends from 17:00 - 8:00. Technical support is available in room HP5161 Monday to Friday from 9:00 until 17:00.

All SCS account related information is accessible at the following URL: http://www.scs.carleton.ca/nethelp.
Undergraduate Academic Advisor

The Undergraduate Advisor for the School of Computer Science is available in Room 5302C HP, by telephone at 520-2600, ext. 4364 or by email at undergraduate_advisor@scs.carleton.ca.

The undergraduate advisor can assist with information about prerequisites and preclusions, course substitutions/equivalencies, understanding your academic audit and the remaining requirements for graduation. The undergraduate advisor will also refer students to appropriate resources such as the Science Student Success Centre, Learning Support Services and the Writing Tutorial Services.

University Policies

Student Academic Integrity Policy

Every student should be familiar with the Carleton University student academic integrity policy. A student found in violation of academic integrity standards may be awarded penalties which range from a reprimand to receiving a grade of F in the course or even being expelled from the program or University. Some examples of offences are: plagiarism and unauthorized co-operation or collaboration. Information on this policy may be found in the Undergraduate Calendar.

Plagiarism

As defined by Senate, "plagiarism is presenting, whether intentional or not, the ideas, expression of ideas or work of others as one's own". Such reported offences will be reviewed by the office of the Dean of Science.

Unauthorized Co-operation or Collaboration

*Senate policy states that "to ensure fairness and equity in assessment of term work, students shall not co-operate or collaborate in the completion of an academic assignment, in whole or in part, when the instructor has indicated that the assignment is to be completed on an individual basis". Please refer to the course outline statement or the instructor concerning this issue.*

Academic Accommodations for Students with Disabilities

The Paul Menton Centre for Students with Disabilities (PMC) provides services to students with Learning Disabilities (LD), psychiatric/mental health disabilities, Attention Deficit
Hyperactivity Disorder (ADHD), Autism Spectrum Disorders (ASD), chronic medical conditions, and impairments in mobility, hearing, and vision.

If you have a disability requiring academic accommodations in this course, please contact PMC at 613-520-6608 or pmc@carleton.ca for a formal evaluation. If you are already registered with the PMC, contact your PMC coordinator to send me your Letter of Accommodation at the beginning of the term, and no later than two weeks before the first in-class scheduled test or exam requiring accommodation (if applicable).

After requesting accommodation from PMC, meet with me to ensure accommodation arrangements are made. Please consult the PMC website for the deadline to request accommodations for the formally-scheduled exam (if applicable) at http://www2.carleton.ca/pmc/new-and-current-students/dates-and-deadlines

**Religious Obligation**

Write to me with any requests for academic accommodation during the first two weeks of class, or as soon as possible after the need for accommodation is known to exist. For more details visit the Equity Services website: [http://www2.carleton.ca/equity/](http://www2.carleton.ca/equity/)

**Pregnancy Obligation**

Write to me with any requests for academic accommodation during the first two weeks of class, or as soon as possible after the need for accommodation is known to exist. For more details visit the Equity Services website: [http://www2.carleton.ca/equity/](http://www2.carleton.ca/equity/)

**Medical Certificate**

The following is a link to the official medical certificate accepted by Carleton University for the deferral of final examinations or assignments in undergraduate courses. To access the form, please go to [http://www.carleton.ca/registrar/forms](http://www.carleton.ca/registrar/forms)