Programming Skills: Functional Programming and Haskell
CSCI-541/CSCI-641
Term 20191
Handout 1
August 26, 2019

Syllabus

General Information

Instructor: Matthew Fluet
E-mail: mtf@cs.rit.edu
Office hours: Mon. 3:30pm – 4:30pm; GOL-3555
Wed. 2:30pm – 4:30pm; GOL-3555
Thr. 9:00am – 10:30am; GOL-3555
or by appointment

Lectures: Section 01  MWF 11:00am – 11:50am; GOL-2690

Website: http://www.cs.rit.edu/~mtf/teaching/20191/psfp
http://mycourses.rit.edu

Course Description

The goal of this course is to introduce the students to a programming paradigm and an appropriate programming language chosen from those that are currently important or that show high promise of becoming important. A significant portion of the learning curve occurs through programming assignments with exemplary solutions discussed later in class. The instructor will post specifics prior to registration. With the approval of the program coordinator, the course can be taken for credit more than once, provided each instance deals with a different paradigm and language. [CSCI-641: A term project involving independent investigation is also required. Note: students who complete CSCI-541 may not take CSCI-641 for credit.]

This course instance is “Functional Programming and Haskell”.

Enrollment Requirements

- CSCI-541
  - Prerequisites: CSCI-344 or equivalent course.
- CSCI-641
  - Prerequisites: (CSCI-603 or CSCI-602) and (CSCI-605 or CSCI-604) or equivalent courses with grades of B or better or successful completion of CSCI-344.

Course Goals

In each offering of the course, students will be introduced to a new programming paradigm and a language that supports it. Both are chosen from technologies that are currently important or that show promise to become important.
A practitioner often has to become proficient in a new language and/or paradigm. The Programming Skills courses are designed to facilitate this learning process.

The course goals allow the faculty to react to current trends in software development while still maintaining a critical perspective.

A student should be allowed to retake this course as long as the topics are different from the other time(s) she/he has taken it. Undergraduate students can “mix and match” undergraduate and graduate offerings as long as the topics are different from the other time(s) she/he has taken them. CSCI-541 and CSCI-641 are co-listed.

Instances of this course are classified as belonging to the Languages and Tools (LTS) cluster.

Course and Program Outcomes

Course learning outcomes:

- Students will be able to explain and apply nearly all features of the chosen language (depending on its size) and will be able to use at least the fundamental aspects of the development and runtime environment.
  
  Program outcome(s): 2, 5 (CSCI-541); 2 (CSCI-641)
  Evaluation: Code of programming assignment solutions, examination

- Students will be able to explain the implications of the chosen paradigm and leverage its concepts.
  
  Program outcome(s): 2, 5 (CSCI-541); 2, 4 (CSCI-641)
  Evaluation: Design of programming assignment solutions, examination

- Students will be able to acquire and leverage new skills on their own, given some direction.
  
  Program outcome(s): 2, 5 (CSCI-541); 2, 4 (CSCI-641)
  Evaluation: Design of programming assignment solutions, examination

Program Outcomes:

- (CS Undergraduate Program Outcome 2) Demonstrate fluency in high-level programming languages, environments, and tools for computing.

- (CS Undergraduate Program Outcome 5) Demonstrate advanced knowledge of a selected area within the computer science discipline.

- (CS Graduate Program Outcome 2) Demonstrate a depth of knowledge in a selected area in the discipline.

- (CS Graduate Program Outcome 4) Pursue professional positions or further graduate studies.

Grades

Grades will be assigned based on the following grading scheme:

<table>
<thead>
<tr>
<th></th>
<th>CSCI-541</th>
<th>CSCI-641</th>
</tr>
</thead>
<tbody>
<tr>
<td>Attendance &amp; Participation</td>
<td>5%</td>
<td>4%</td>
</tr>
<tr>
<td>Quizzes &amp; Recitations</td>
<td>15%</td>
<td>12%</td>
</tr>
<tr>
<td>Homework Assignments (∼ 8)</td>
<td>70%</td>
<td>60%</td>
</tr>
<tr>
<td>Research Paper Review/Presentation</td>
<td>10%</td>
<td>8%</td>
</tr>
<tr>
<td>Project</td>
<td></td>
<td>16%</td>
</tr>
</tbody>
</table>
Final letter grades will be assigned based on the following grading scale:

<table>
<thead>
<tr>
<th>Letter grade</th>
<th>Numeric grade</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>(93, ∞)</td>
</tr>
<tr>
<td>A−</td>
<td>[90, 93)</td>
</tr>
<tr>
<td>B+</td>
<td>[87, 90)</td>
</tr>
<tr>
<td>B</td>
<td>[83, 87)</td>
</tr>
<tr>
<td>B−</td>
<td>[80, 83)</td>
</tr>
<tr>
<td>C+</td>
<td>[77, 80)</td>
</tr>
<tr>
<td>C</td>
<td>[73, 77)</td>
</tr>
<tr>
<td>C−</td>
<td>[70, 73)</td>
</tr>
<tr>
<td>D</td>
<td>[60, 70)</td>
</tr>
<tr>
<td>F</td>
<td>[0, 60)</td>
</tr>
</tbody>
</table>

Important Dates

Oct. 21 (Mon.): Project Proposal
Nov. 11 (Mon.): Research Paper Selection
Nov. 27 (Wed.): Project Checkpoint
Dec. 3 (Tue.): Research Paper Summary
Dec. 09 (Mon.) & Dec. 18 (Wed.): Research Paper Presentation
Dec. 16 (Mon.) – Dec. 18 (Wed.): Project Demo

Text Books

Required:
Title: Programming in Haskell (2nd edition)
Author: Graham Hutton
Publisher: Cambridge University Press
ISBN: 978-1316626221
Website: [http://www.cs.nott.ac.uk/~pszgmh/pih.html](http://www.cs.nott.ac.uk/~pszgmh/pih.html)
Course Policies

Attendance & Participation

Students are required to attend and expected to participate in class. Participation means being an engaged student: asking and answering questions, not simply attending class.

The use of cell phones and audio players is prohibited during class. If you must take a phone call, please leave the classroom immediately and do not return until you have ended the phone call.

The use of a laptop (or notebook or netbook) computer is permitted during class only for the purpose of taking notes. Persistent use of a laptop for other activities will result in 0 credit for your Attendance & Participation grade.

Assigned readings should be completed before the lecture section. You are responsible for the material in assigned readings, whether covered during lecture or not.

Late Policy

Assignments are to be submitted on time. However, to accommodate the occasional difficulty with meeting an assignment due date, each student begins the term with four “extension tokens.” By spending an extension token, you will receive a 24-hour extension on a single assignment. To spend an extension token, you must e-mail the instructor before the assignment is due; you cannot spend an extension token after an assignment’s due date has passed. You may spend at most one extension token on a single assignment. After spending four extension tokens, late assignments will not be accepted. The instructor will not discuss or answer questions about an assignment after the assignment’s due date has passed.

Regrading

After a graded exam or assignment has been returned, you have one week to bring any questions about grading to the instructor’s attention. No grade adjustments will be made after this time.

Academic Integrity

As with all courses, the RIT Honor Code and the Student Academic Integrity Policy apply. See the Department of Computer Science’s statement on Student Academic Integrity for more details.

In this course, all submitted work must be your own work (i.e., written or programmed by you alone, unless explicitly stated otherwise) and must include acknowledgments of any collaborators or sources (other than course text books or handouts) used to produce your submission.

You are encouraged to discuss course material with other students. Discussion of assignments is also allowed, but sharing solutions or code is not allowed.

Common Course Policies

See the Department of Computer Science’s Common Course Policies for more details about rescheduling an exam, course withdrawal, disability services, and academic
integrity.

Disclaimer

The instructor reserves the right to make any changes to the syllabus deemed necessary throughout the course. Minor changes, such as assignment due dates, will be announced orally during class and posted on the course mailing list and home page. Major changes, such as grading percentages, will additionally be provided in writing.

Acknowledgements

Portions of this course material based upon similar courses offered at University of Pennsylvania (Stephanie Weirich), University of Pennsylvania (Joachim Breitner, Noam Zilberstein, Richard Eisenberg), RIT (Arthur Nunes-Harwitt).