CS611
Advanced Programming Languages
Radu Rugina
Administration and Overview
29 Aug 03

General Information

<table>
<thead>
<tr>
<th>When</th>
<th>MWF 10:10 - 11:00AM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Where</td>
<td>Upson Hall 211</td>
</tr>
<tr>
<td>Instructor</td>
<td>Radu Rugina (Upson 4141)</td>
</tr>
<tr>
<td></td>
<td>Siggi Cherem, Michael Clarkson, Brian Hackett</td>
</tr>
<tr>
<td>Teaching Assistants</td>
<td></td>
</tr>
<tr>
<td>Course staff email</td>
<td><a href="mailto:cs611@cs.cornell.edu">cs611@cs.cornell.edu</a></td>
</tr>
<tr>
<td>News group</td>
<td>cornell.class.cs611</td>
</tr>
<tr>
<td>Web page</td>
<td>courses.cs.cornell.edu/cs611</td>
</tr>
</tbody>
</table>

Workload

- 5-6 homeworks (about half with programming requirements, in ML)
- Scribe 2-3 lectures (in pairs)
- One prelim: tentatively 10/28, 7:00-9:30 PM
- Final exam: 12/15, 12:00-2:30 PM

Grading

- Homeworks: 40%
- Preliminary exam: 20%
- Final exam: 30%
- Class participation: 10%

Policy for Homeworks/Exams

- NOT done in groups
  - do your own work
- Late homeworks increasingly penalized
  - Penalty linearly increasing
  - x days late = (x * 10)% penalty

Course Contents

- What the course IS about:
  formal semantics of programming languages
  - Formal frameworks for describing and reasoning about the computation in a program
  - Tools for formally analyzing and proving properties about the programs
  - Formalism ⇒ lots of math!
- What the course IS NOT about:
  - Survey of modern programming languages or of advanced language constructs
  - Study of implementation techniques (see cs 412)
Advises

- CS 411 offered this fall
  - Covers the material for the CS Qual Exam
- Undergraduates advised to take CS 411 first
- Graduate students also have the CS 411 option
- Warning: taking CS 611 and CS 681 concurrently requires a significant amount of work

Overview

- Semantics of programming languages
  1. Semantics of a simple language
  2. Semantics of other language constructs
  3. Static and abstract semantics

Part I: Basic PL Semantics

- Three main ways to describe semantics:
  - Operational semantics
  - Denotational semantics
  - Axiomatic Semantics
- Discuss relation between different semantics
- Use IMP = simple imperative language
  - Only integers and booleans
  - Only assignments, if statements, and while statements

Part II: Language Features

- Functions
  - REC = IMP + functions, recursion
  - Lambda calculus, higher order functions
- Continuations
- Scope
- References
- Exceptions, etc.

Part III: Static/Abstract Semantics

- Types
  - Typed lambda calculus
  - Recursive types, subtyping
  - Type inference
- Abstract interpretation
  - Iterative solution
  - Widening operators
  - Formal correctness

Textbooks

- Textbooks:
  - Glynn Winskel: The formal semantics of programming languages
  - Carl Gunter: Semantics of programming languages
  - John Mitchell: Foundation of programming languages
  - Benjamin Pierce: Types and programming languages
  - Flemming Nielson, Rias Nielson, Chris Hankin: Principles of program analysis
- Books are on reserve in Engineering Library