CMSC 22100: Programming Languages

The University of Chicago, Spring 2016
Adam Shaw

`http://www.classes.cs.uchicago.edu/archive/2016/spring/22100`

Welcome!

The course is directed towards learning the basic elements of programming language theory and scientific programming language design. This includes studies of the following topics:

- the lambda calculus,
- formal specification of evaluation systems (dynamic semantics),
- design of type systems (static semantics), and
- theorems about programming languages (notably, type soundness) and proof techniques.

The learning of these abstractions will be supplemented by closely related programming exercises in the Standard ML programming language ("SML"). As a secondary benefit, since SML, as much as any other programming language, embodies the achievements of programming language theory, writing SML code will serve to emphasize and clarify the nature of those achievements.

Furthermore, students will gain some introductory experience with machine-assisted proof using the interactive Coq system.

Once we have covered the foundations of programming language theory, we will explore supplemental topics, including, possibly, some of the following: parametric polymorphism, object-oriented programming, type classes, monads, and continuations.

Instructor

Adam Shaw, email: `ams@cs.uchicago.edu`, office: Ryerson 157.

Instructional Staff

Teaching Assistants: Tri Huynh, Hao Jiang. Grader: Melissa Grossbarth.

Lectures are Tuesdays and Thursdays, 12:00–1:20 in Ryerson 251. The first class meeting is on Tuesday, March 29; the last is on Tuesday, May 31.

I do not allow the use of electronic devices during lectures. They are simply too distracting.

Schedule of Topics by Week (subject to change)

Week	Topics
1	Standard ML and scientific programming language design
2	inductive proofs, evaluation semantics, normal forms
3	the untyped lambda calculus, type systems, type soundness
4	variables, the simply-typed lambda calculus
5	type classes, modules
6	*midterm I*, machine-assisted proof
7	machine-assisted proof
8	topics TBD: subtyping, Java, objects
9	topics TBD: polymorphism, module systems
10	*midterm II*

Office Hours To be announced on the web once the quarter starts.

Texts The required text is Benjamin Pierce's *Types and Programming Languages*. It is available at the Seminary Co-op Bookstore.

I will also assign readings from Benjamin Pierce *et al.'s Software Foundations*; this is available online, free of charge, at `https://www.cis.upenn.edu/~bcpierce/sf/current/index.html`.

There are various Standard ML books that might be of use to students who would like references for the language; none is required. These include Bob Harper's *Programming in Standard ML* (online at `http://www.cs.cmu.edu/~rwh/smlbook/book.pdf`), Jeffrey Ullman's *Elements of ML Programming*, and Lawrence Paulson's *ML for the Working Programmer*.

Software All the software we use in this course is available online for all common platforms. We will mainly use *sml*, the *Coq* proof assistant, and *subversion*. To install Standard ML on your computer, please take a look at `http://smlnj.org`. Coq is at `https://coq.inria.fr`.

Grading Coursework is comprised of homework exercises and exams. Regular attendance at class meetings and active participation both in person and online are not part of how you are evaluated, but both are very much appreciated by your instructor!

Each student's final grade will be computed according to the following formula: homework 50%, exams 25% each. We will scale the grades, so what precisely constitutes an A, B, *etc.* will be determined by the collective performance of the class.

Homework There will be weekly homework assignments, generally due on Wednesday nights.

Exams There will be two midterm exams, in class, on Tuesday, May 3, and Tuesday, May 31. There will be no final exam during finals week.

Late Work Deadlines in this course are rigid. Since you submit your work electronically, deadlines are enforced to the minute. Late work will not be counted. I will accept late work only in case of extraordinary circumstances.

Academic Honesty In this course, as in all your courses, you must adhere to college-wide honesty guidelines as set forth at `http://college.uchicago.edu/policies-regulations/academic-integrity-student-conduct`. The college's rules have the final say in all cases. If you ever have any questions or concerns about honesty issues, raise them with your instructor, early.

Also, note that any student who is determined to have violated academic honesty rules will not be allowed to withdraw from this course.