

CS 558: Software Foundations

Fall 2017

Matthew R. Lakin

August 22, 2017

Course Information

Lectures

Tuesdays and Thursdays 11:00–12:15, in Mechanical Engineering room 214.

Instructor

Matthew Lakin.

Email: mlakin@cs.unm.edu

Office hours: Tuesdays and Thursdays 10–11am.

Office: CEC 2041G.

Note that CEC 2041G is located within the “Center for Biomedical Engineering” administrative office suite on the second floor of the Centennial Engineering Center.

Teaching assistant

Phillip Kay.

Email: pkay@unm.edu

Office hours: Mondays and Wednesdays 12:30am–1:30pm.

Office hours location: ECE building lobby / common area.

Course topics and format

The course is an informal introduction to the theory used to describe and define programming languages and to guide their implementation. Our approach is type-based, in the spirit of our

textbook, Pierce's Types and Programming Languages (TAPL). As a prelude, the course offers a brief overview of functional programming techniques and of programming language features found in the purely functional programming language Haskell.

The course is intended for first-year graduate students, but advanced undergraduates are welcome as well. No specific courses are prerequisites, but programming experience and mathematical maturity are necessary. Experience with functional programming (at the level of UNM CS357) is very helpful.

The course will provide students with the background they need for CS550 (usually offered in the Spring).

The course consists of lectures, homework assignments (primarily programming based), two mid-term examinations, and a final examination.

Assignments

There will be two in-class midterm exams, and a final exam covering the entire course. Several programming assignments may be given: in the early part of the course these tasks will be drawn from the general domains of mathematics, science, and engineering, to practice programming skills; in the later part of the course the tasks will correspond to implementation of programming language theory. Several short written homework assignments may be given to consolidate lecture material; they may take the form of short algebraic proofs of program fragment equivalence, or consideration of small language extensions.

Textbooks

- Benjamin C. Pierce, Types and Programming Languages, MIT Press, 2002, ISBN-10: 0262162091.
- Miran Lipovača, Learn You a Haskell for Great Good, No Starch Press, 2011, ISBN-13: 978-1-59327-283-8. Accompanying webpage: <http://learnyouahaskell.com>.

Grading

You are expected to attend class regularly, read any assigned reading before class, and participate in class discussion. The grade will be determined as follows:

- Homeworks: 50%
- Exams: 50% (15% for each midterm, 20% for the final)

Note that no requests for grade changes will be considered after the final day of classes.

Communication

The Loboweb email list functionality will be used for administrative announcements. Lecture notes and homework assignments will be uploaded to the UNM Learn page for the class.

Topics

The topics covered in class will be a subset of the following:

- Topics in functional programming
 - functional programming and Haskell introduction
 - prelude types and classes
 - functions and list comprehensions; unit testing; literate programming; interactive programs
 - recursive and higher-order functions
 - declaring types and classes
 - lists in depth: map, filter, and their algebraic laws
 - lists in depth: foldr, scanr, and their algebraic laws
 - trees with folds, binary heap trees, rose trees
 - efficiency: accumulating parameters, tupling, fusion, and deforestation
 - modules and abstract data types
 - lazy evaluation and infinite data structures; approximation ordering; cyclic structures; streams
 - monads
 - metaprogramming in a functional programming language
- Topics in programming language semantics
 - syntax
 - operational semantics
 - lambda calculus syntax and reduction
 - programming in the lambda calculus
 - combinators and combinator reduction
 - types
 - simply typed lambda calculus

- simple extensions (ascription; let-bindings; records; variants; recursion)
- references
- exceptions
- subtyping
- recursive types
- type reconstruction
- unification
- universal polymorphism
- program transformations

UNM statement of compliance with ADA

Qualified students with disabilities needing appropriate academic adjustments should contact the instructor as soon as possible to ensure their needs are met in a timely manner. Handouts are available in alternative accessible formats upon request.

UNM statement regarding Title IX

In an effort to meet obligations under Title IX, UNM faculty, Teaching Assistants, and Graduate Assistants are considered “responsible employees” by the Department of Education (see pg 15 - <http://www2.ed.gov/about/offices/list/ocr/docs/qa-201404-title-ix.pdf>). This designation requires that any report of gender discrimination which includes sexual harassment, sexual misconduct and sexual violence made to a faculty member, TA, or GA must be reported to the Title IX Coordinator at the Office of Equal Opportunity (<http://oeo.unm.edu>). For more information on the campus policy regarding sexual misconduct, see: <https://policy.unm.edu/university-policies/2000/2740.html>