CS 583: (Advanced) Functional Programming

Course Introduction

Haskell

It’s Pure Fun!
Introduce yourself!

In class (show of hands):
- anyone new to Haskell?
- previous PL courses?
- taking DSL course concurrently?

Through TEACH ([engr.oregonstate.edu/teach](http://engr.oregonstate.edu/teach)):
- photo, name, year, group
- Haskell/FP confidence/proficiency
- programming languages you know best
- what you hope to take away from this class
- interesting fact

*(on the course web page)*
Course logistics

Course web page:
web.engr.oregonstate.edu/~walkiner/teaching/cs583-sp16

Piazza (announcements, Q&A)
piazza.com/oregonstate/spring2016/cs583

Office hours (KEC 3049):
Mon 2:30 – 3:30pm
Fri 1:00 – 2:00pm
Prerequisites

Courses that will prepare you for this course . . .

• Official: CS 581
• Ideal: CS 381, CS 589 (Fall 2015), CS 581 (pre-2015)

What I assume you know . . .

• algebraic data types
• pattern matching
• using/understanding recursion
• higher-order functions

If you know FP but not Haskell . . . spend some extra time this week to get familiar with it
A “hardcore” course in FP :-)

Planned topics:

- type-directed programming
- equational reasoning and refactoring
- higher-order abstract syntax
- type classes (functors, monads, transformers)
- managing effects (w/ monads)
- zippers and functional data structures
- phantom types, GADTs
- graph reduction (implementing lazy evaluation)
- … and more?

xkcd.com/1312
Learning strategy

In class:
• lectures
• live coding
• peer-critique workshops
• presentations (end)

Out-of class:
• occasional reading
• prepare for workshops
• functional programming project
Evaluation

- 40% midterms (2 x 20%)
- 20% workshop participation
  - submit Hasklets
  - prepare criticism
  - participate in discussion
- 40% functional programming project
  - intermediate deadlines
  - final product (code+document)
  - final presentation