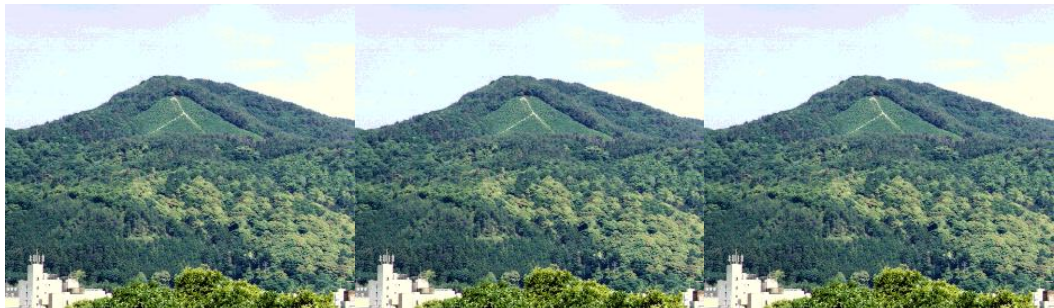


Parallel Functional Programming

Stephen A. Edwards

Columbia University

Fall 2026



Instructor



Prof. Stephen A. Edwards

sedwards@cs.columbia.edu

<http://www.cs.columbia.edu/~sedwards/>

462 Computer Science Building

Haskell

```
primes = filterPrime [2..]  
  where filterPrime (p:xs) =  
        p : filterPrime [x | x <- xs, x `mod` p /= 0]
```

Sieve of Eratosthenes

Purely Functional · Declarative · Lazy · Statically Type-Inferred · Parallel

Sequential Haskell in the first half · Parallel in the second half

Prerequisites

Data structures (COMS W3134, W3137, or equivalent)

- ▶ You must be fluent in at least one programming language



- ▶ You must dream about lists and trees



- ▶ You do **not** need prior experience in a *functional* programming language; that's what this course is for

Goals

- ▶ Understand the functional programming style:
recursion and pattern matching on recursive data structures
- ▶ Be able to express algorithms in a pure functional setting
- ▶ Structure data with Haskell's type system:
algebraic data types, parametric polymorphism, type classes, monads
- ▶ Understand Haskell's lazy evaluation strategy
- ▶ Understand parallel algorithm challenges:
discretization, load balancing
- ▶ Express parallel scheduling policies through Haskell's laziness
- ▶ Understand how to use all of these to produce efficient parallel code

Assignments and Grading

30 % Homework Assignments

10 % Project Proposal

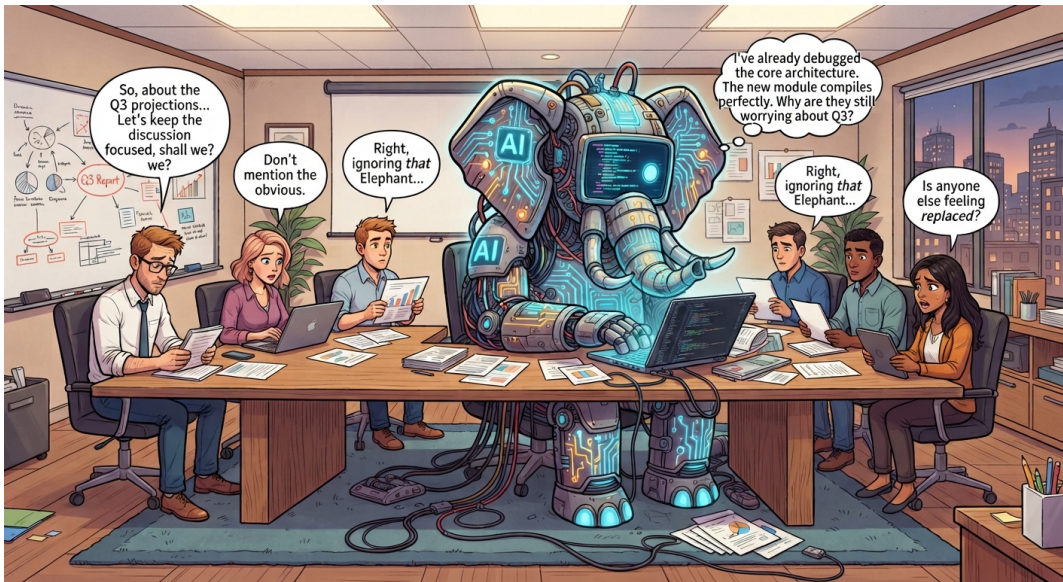
20 % Project Review

40 % Final Project

This is a coding[†] class

[†]More precisely, mostly debugging,
with a little bit of bugging





Source: Google Gemini (AI)

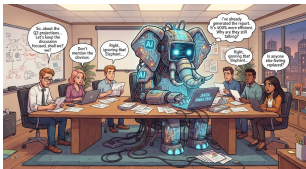
Policies

- ▶ Homework graded orally, 1 bit/student/assignment
- ▶ For each assignment, you must convince us in 5 minutes during office hours that you understand what you submitted
- ▶ You must submit all your AI chat logs in addition to solutions
- ▶ Project proposals, review, and final demonstrations will be “oral defenses” in front of the instructor, outside of lecture time. You need to convince me you know what your code is doing
- ▶ Final project defenses will take place during finals week

The AI Elephant in the Room

<https://gemini.google.com/app/6473c26317c9feb3>

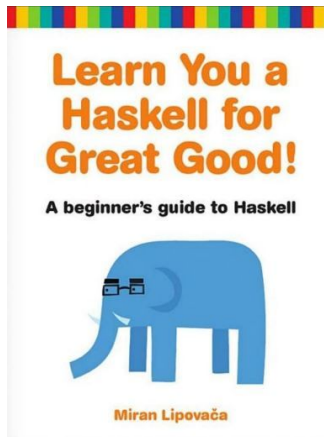
User prompt: You are a comic illustrator. Draw an image consisting of an "elephant in the room" where the elephant is AI.



User prompt: Focus the context more on programming. Remove "data analysis" from the elephant's laptop.



Recommended Texts



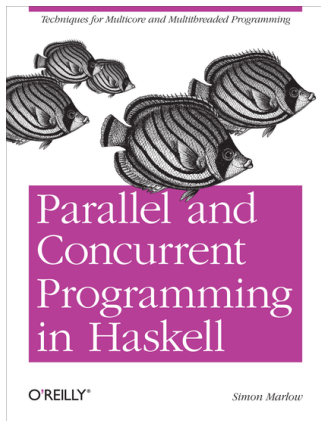
Miran Lipovača.

Learn You a Haskell for Great Good!
No Starch Press, 2001.

<https://learnyouahaskell.github.io/>

Excellent introductory text. We will be following it for roughly the first half of the class.

Recommended Texts

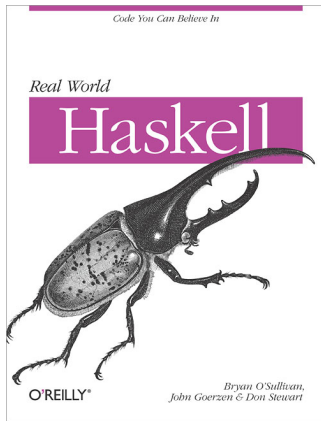


Simon Marlow.
Parallel and Concurrent Programing in Haskell.
O'Reilly, 2013.

<https://simonmar.github.io/pages/pcph.html>

Like its title says. Assumes a reasonable understanding of Haskell. We will be following it for the second half of the class.

Recommended Texts

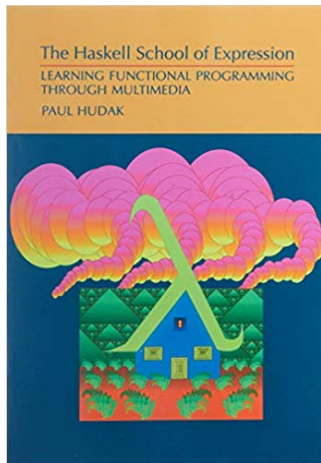


Bryan O'Sullivan, Don Stewart, and John Goerzen.
Real World Haskell.
O'Reilly, 2009.

<http://book.realworldhaskell.org/>

Also an introductory text on Haskell that starts at the beginning, it quickly focuses on practical, real-world aspects of writing Haskell programs, such as elaborate I/O, and interfacing with external libraries.

Recommended Texts

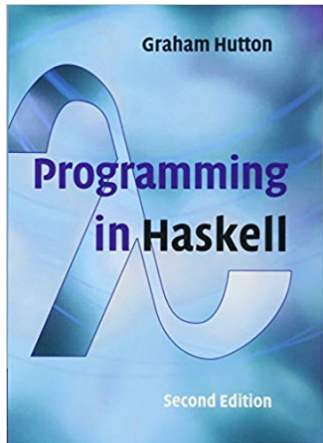


Paul Hudak.
The Haskell School of Expression.
Cambridge University Press, 2000.

<http://www.cs.yale.edu/homes/hudak/SOE/>

An idiosyncratic approach to learning Haskell based on multimedia (graphics, animation, and sound) ultimately leading to domain-specific languages.

Recommended Texts

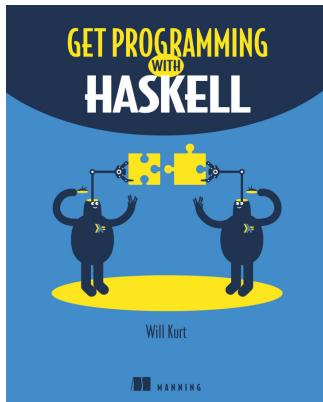


Graham Hutton.
Programming in Haskell.
Second Edition, Cambridge University Press, 2016.

<http://www.cs.nott.ac.uk/~pszgmh/pih.html>

Another introductory Haskell text, this one written by
a professor from the University of Nottingham

Recommended Texts



Will Kurt.
Get Programming with Haskell.
Manning, 2018.

[https://www.manning.com/books/
get-programming-with-haskell](https://www.manning.com/books/get-programming-with-haskell)

Another introductory Haskell text, written more like a textbook