

# CS 457/557: Functional Programming (Winter, 2023)

**Time:** Tuesday/Thursday 10:00-11:50

**Classroom:** Fourth Avenue Building (FAB) 47

**Credit Hours:** 4

**Instructor:** Yao Li, Ph.D.

**Office:** Fourth Avenue Building (FAB) 115-07

**Office Hours:**

- Tuesday/Wednesday 13:30-14:30 (in person or via Zoom)
- By appointment: <https://calendar.app.google/L9jrmE48HUWrqTfP8> (in person or via Zoom)

**Prerequisite:**

- CS 302: Programming Methodologies and Software Implementation

**Course Website:**

- Course website: <https://lastland.github.io/courses/2023-01-12-fp.html>
- Canvas: to be published

## Description

If you are paying attention to the recent evolution of today's mainstream programming languages, you will find an emergence of features adapted from functional programming languages. Why is this happening? What is special about functional programming? Is there something that one can learn from functional programming paradigm that can make you a better programmer?

In this course, you will learn the key ideas behind functional programming that make it powerful, via a programming language called Haskell. I will show you how functional languages can help you write readable, modular, and efficient code.

No prior knowledge of Haskell or functional programming is required to take this class.

## Major Topics

- Abstraction, abstraction, and abstraction
- Algebraic data types and recursion
- Polymorphism
- Higher-order functions
- Universal property
- Equational reasoning

- Monads and applicative functors
- Lazy evaluation
- Type checking
- Property-based testing

## Goals

- Ability to solve programming problems with a functional language
- Basic understanding of theoretical foundations of the functional paradigm

Upon the successful completion of this class, students will be able to:

- Describe the key characteristics of the functional paradigm.
- Use a functional language to write simple programs manipulating lists and trees.
- Use higher order functions, including maps, folds, and composition, to perform tasks on lists.
- Define algebraic datatypes to model useful domains.
- Define and use appropriate higher order functions on a variety of algebraic data types.
- Use first-class functions as data values.
- Design and implement functional abstract data types.
- Prove properties of simple functional programs using equational reasoning and induction.
- Explain and use polymorphic functions and data types, and interpret type error messages.
- Write functional programs that perform I/O, using either monadic or direct methods.
- Explain the differences between eager and lazy evaluation, and their significance.
- Write programs using simple infinite data structures.
- Explain the basic implementation considerations for a functional language.
- Design and implement a non-trivial application program in a functional language.

## Textbook

No textbook required. However, if you are interested in getting a textbook for reference, the course materials are mostly based on *Programming in Haskell* by Graham Hutton and *Real-World Haskell* by Bryan O’Sullivan, Don Stewart, and John Goerzen.

## Other Resource

- Hoogle: <https://hoogle.haskell.org/>
- Haskell Wiki: <https://wiki.haskell.org/Haskell>
- Haskell-Cafe mailing list: <https://www.haskell.org/mailling-lists/>
- IRC channels: <https://www.haskell.org/irc/>

- Discourse: <https://discourse.haskell.org/>
- StackOverflow: <https://stackoverflow.com/questions/tagged/haskell>
- Reddit: <https://www.reddit.com/r/haskell/>
- Others: <https://www.haskell.org/community/>

## Remote Options

Lectures are delivered in person. However, **please stay home if you are sick**. Just let me know and I will send you a Zoom link for attending the class remotely.

## Assignments and Project

- Weekly assignments in the first 6 weeks. The assignments are mostly programming assignments that focus on small but nontrivial problems. They are usually released on every Thursday and due on the next Thursday.
- One final project in the last 4 weeks. You can work on the project by yourself or form a team with one other person who is taking the class. You can choose from some pre-selected topics provided by the instructor and propose your own topic.

## Individual work and collaboration

You are encouraged to discuss the assignments with anyone, but you must derive a solution on your own and you are expected to be able to clearly explain your code on request. One thing you can do is to take a walk or work on something else for one hour after discussing the assignments with your classmates and before working on your assignments.

You can either work on the final project by yourself, or collaborate with at most one other person who is taking the class. Every member of a team is jointly responsible for the team's submissions, and should be able to explain them on request. Again, you are encouraged to discuss your work with anyone (including people outside your group), but you must derive a solution on your own.

## Skipping or delaying assignments

You can skip one assignment for free. You do *not* need to provide a reason. I only ask you to email me either before or not too far after the submission deadline to let me know that you would like to use this option.

In addition, you can also delay one assignment by one week for free with no penalty. Again, you do *not* need to provide a reason. Please email me either before or not too far after the submission deadline to let me know that you would like to use this option.

## **Grading**

- 60% Assignments
  - only the best 5 assignments are counted in your final grades
- 40% Final project

## **Title IX Reporting Obligations**

Portland State is committed to fostering a safe, productive learning environment. Title IX and our school policy prohibit gender or sex-based discrimination and sexual misconduct (including harassment, domestic and dating violence, sexual assault, and stalking). We expect a culture of professionalism and mutual respect in our department and class. You may report any incident of discrimination or discriminatory harassment, including sexual harassment, to either the Office of Equity and Compliance (<https://www.pdx.edu/diversity/equity-compliance>) or the Office of the Dean of Student Life (<https://www.pdx.edu/student-life/dean-of-student-life>).

Please be aware that members of the faculty have the responsibility to report any instances of sexual harassment, sexual violence and/or other forms of prohibited discrimination to PSU's Title IX Coordinator, the Office of Equity and Compliance or the Dean of Student Life and cannot keep information confidential. If you would rather share information about sexual harassment or sexual violence to a confidential employee who does not have this reporting responsibility, you can contact a confidential advocate at 503-725-5672 or by scheduling on-line (<https://psuwrc.youcanbook.me>) or another confidential employee found on the sexual misconduct resource webpage (<https://www.pdx.edu/sexual-assault/get-help>).