Home
Schedule
Assignments
Resources

Piazza
Canvas
fp@cs
Frank Pfenning

15-814 Types and Programming Languages

Fall 2019
Frank Pfenning
TuTh 10:30-11:50
GHC 4215
12 units
First lecture will be Tue Sep 3

This graduate course provides an introduction to programming languages viewed through the lens of their type structure.

Prerequisites: This is an introductory graduate course with no formal prerequisites, but an exposure to various forms of mathematical induction will be helpful. Enterprising undergraduates and masters students are welcome to attend this course.

Prior Versions of This Course

- Fall 2018
- Fall 2017, Karl Crary
- Fall 2015, Robert Harper

Class Material

Schedule	Lecture notes and additional readings
Assignments	Homework assignments and due dates
Resources	Links to other resources

Course Information

Lectures	Tu Th 10:30-11:50, GHC 4215
Instructor	Frank Pfenning, fp@cs Office Hours **Fri 1:00pm-2:00pm, GHC 6017**
Teaching Assistant	Farzaneh Derakhshan, fderakhs@andrew Office Hours **Mon 2:30pm-3:30pm, WeH 5302**
Course Communication	piazza.com/cmu/fall2019/15814
Textbook and Notes	Robert Harper, Practical Foundations for Programming Languages (Second Edition), Cambridge University Press, April 2016. Additional notes will be posted on the schedule page.

Credit	12 units
Grading	60% Homework, 15% Midterm, 25% Final
Homework	Homework assignments are posted on the assignments page.
Midterm	**Thu Oct 17**, in class. Closed book.
Final	**Fri Dec 13, 5:30pm-8:30pm, TEP 2611** Closed book.
Home	http://www.cs.cmu.edu/~fp/courses/15814-f19/

Learning objectives: After taking this course, students will be able to

- define programming languages via their type system and operational semantics
- draw from a rich set of type constructors to capture essential properties of computational phenomena
- state and prove the preservation and progress theorems or exhibit counterexamples
- recognize and avoid common fallacies in proofs and language design
- write small programs to illustrate the expressive power and limitations of a variety of type constructors
- state and prove properties of individual programs based on their semantics or exhibit counterexamples
- critique programming languages and language constructs based on the mathematical properties they may or may not satisfy
- appreciate the deep philosophical and mathematical underpinnings of programming language design

Core topics:

- Static and dynamic semantics
- Preservation and progress
- Hypothetical judgments and substitution
- Propositions as types, natural deduction, sequent calculus
- The untyped lambda-calculus
- Functions, eager and lazy products, sums
- Recursive types
- Parametric polymorphism, data abstraction, existential types
- K machine, S machine, substructural operational semantics
- Message-passing concurrency, session types

[**Home** | Schedule | Assignments | Resources]

fp@cs
Frank Pfenning

Home
Schedule
Assignments
Resources

Piazza
Canvas
fp@cs
Frank Pfenning

15-814 Types and Programming Languages Assignments

- Assignments generally are given out Tuesday and are due the following Tuesday, but the schedule may vary.
- Unless otherwise noted, assignments are individual assignments. They must represent your own work. You may consult any public resources such as papers, reports, or lecture notes, but you must explicitly credit such external resources in your answers. You are not permitted to draw upon assignments or solutions from previous instances of this course or other courses in preparing your work. Carnegie Mellon's policy on academic integrity applies to this course.
- Some problems will be marked as subject to the **Whiteboard Policy**:

  > Once you have made an honest effort to solve a whiteboard problem on your own, you are free to discuss that problem with your classmates. At the end of your discussion, any record of the discussion must be erased and you must wait a few hours before writing your solution on your own.
  > Practically, this means:
  > - If you work at a board, erase the board. Do not take pictures of the board.
  > - If you write on pieces of paper throw them out at the end of the discussion.
  > - Do not discuss problems over Slack, email, WhatsApp, etc.
  >
  > Please follow the spirit and the letter of this policy: its goals are to help you learn, to ensure that you understand the work you submit, and to ensure that your work is your own.

- Homeworks may require small implementations, or simply write-ups with LaTeX or with pencil and paper.
- If you want to typeset your answers, some sample LaTeX lecture notes and necessary style file will be posted in the misc/latex/ directory or are available in the homework handouts.
- Emphasis is on **correctness** and **elegance**. Some assignments may be very difficult. If you cannot do them, write down your thoughts, but never hand back an incorrect "proof".
- Written homeworks are due at 11:59pm on the due date. They should be submitted as PDFs via Canvas.
- We will try our best to return graded homework during the lecture following the due date.
- Assignment hand-in and grades are tracked on Canvas.

Out	Assignment	Reference	Due	Sample Solution

Sep	10	The Untyped λ-Calculus	(LaTeX source)	**Tue**	**Sep**	**17**	(sample solution)
Sep	17	The Simply-Typed λ-Calculus	(LaTeX source)	**Tue**	**Sep**	**24**	(sample solution)
Sep	24	Nontermination	(LaTeX source)	**Tue**	**Oct**	**1**	(sample solution)
Oct	1	Lazy Pairs and Type Isomorphisms	(LaTeX source)	**Tue**	**Oct**	**7**	(sample solution)
Oct	9	Mutual Recursion	(LaTeX source)	**Tue**	**Oct**	**15**	(sample solution)
Oct	22	The K Machine	(LaTeX source)	**Tue**	**Oct**	**29**	(sample solution)
Oct	29	Propositions as Types	(LaTeX source)	**Tue**	**Nov**	**5**	(sample solution)
Nov	6	Parametricity	(LaTeX source)	**Tue**	**Nov**	**12**	(sample solution)
Nov	20	Lazy Records	(LaTeX source)	**Tue**	**Nov**	**26**	(sample solution)
Nov	27	Linearity	(LaTeX source)	**Fri**	**Dec**	**6**	(sample solution)

[Home | Schedule | **Assignments** | Resources]

fp@cs
Frank Pfenning

[Home](#)
Schedule
[Assignments](#)
[Resources](#)

[Piazza](#)
[Canvas](#)
fp@cs
[Frank Pfenning](#)

15-814 Types and Programming Languages
Schedule

- Lectures are Tuesday and Thursday, 10:30pm-11:50pm, GHC 4215
- The lecture notes provide additional reading material
  They complement, but do not replace the lecture or the textbook
- The schedule is subject to change throughout the semester
- Chapter and exercise references are to the textbook
- Further references can be found in the lecture notes and textbook

Date			Lecture Notes	Additional Reading	Due
Tue	Aug	27	**No Lecture** (CSD IC)		
Thu	Aug	29	**No Lecture** (CSD IC)		
Tue	Sep	3	The Lambda Calculus	Ch 21.1-21.2	
Thu	Sep	5	Recursion	Ch 21.2, lam.sml	
Tue	Sep	10	Simple Types	Ch 8.1-8.2	
Thu	Sep	12	Representation Theorems		
Tue	Sep	17	Subject Reduction		Asst 1
Thu	Sep	19	From λ-Calculus to Programming Languages	Ch 8.1-2, Ch 6	
Tue	Sep	24	Products	Ch 10.1-2	Asst 2
Thu	Sep	26	Sums	Ch 11.1-2	
Tue	Oct	1	Recursive Types	Ch 20, Ch 21.4	Asst 3
Thu	Oct	3	Elaboration		
Tue	Oct	8	Exceptions	Ch 29	Asst 4
Thu	Oct	11	The K Machine	Ch 28, code (in Haskell)	
Tue	Oct	15	Bisimulation		Asst 5
Thu	Oct	17	**Midterm Exam**	(sample solution) F'18 midterm (sample solution)	

Tue	Oct	22	Propositions as Types	Ch 12	
Thu	Oct	24	Parametric Polymorphism	Ch 16	
Tue	Oct	29	Parametricity	Ch 16.3, Ch 48	Asst 6
Thu	Oct	31	Data Abstraction		
Tue	Nov	5	Representation Independence		Asst 7
Thu	Nov	7	Shared Memory Concurrency		
Tue	Nov	12	Negative Types		Asst 8
Thu	Nov	14	Memory Safety		
Tue	Nov	19	Mutable Memory		
Thu	Nov	21	Linear Types		
Tue	Nov	26	**No Class** (Reading Day)		Asst 9
Thu	Nov	28	**No Class** (Thanksgiving Break)		
Tue	Dec	3	Message-Passing Concurrency	[PP19, Sec 1-4]	
Thu	Dec	5	Review	(15-814 PL Awards)	Asst 10
Fri	Dec	13	**Final Exam** 5:30pm-8:30pm, TEP 2611	(sample solution)	

[Home | **Schedule** | Assignments | Resources]

fp@cs
Frank Pfenning