CSE 114A: Fall 2021

Foundations of Programming Languages

Lecture 1: Course Overview

Owen Arden
UC Santa Cruz

A Programming Language

- Two variables
  - `x, y`
- Three operations
  - `x++`
  - `x--`
  - `(x=0)? L1:L2;`

```
L1: x++;
    y--;
    (y=0)?L2:L1
L2: …
```

Fact: This is “equivalent to” to every PL!

Good luck writing quicksort

… or Windows, Google, Spotify!

So why study PL ?

Programming language

shapes

Programming thought

So why study PL ?

Language affects how:

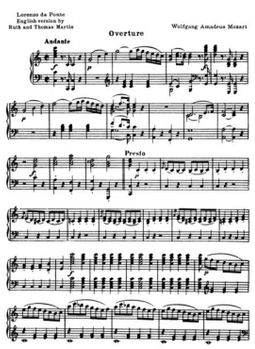
- Ideas are expressed
- Computation is expressed

Course Goals

"Free your mind"

-Morpheus

Learn New Languages/Constructs

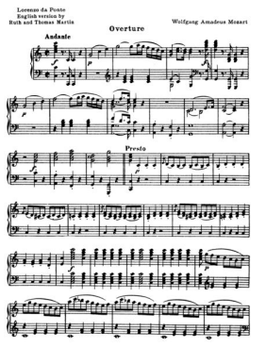


New ways to:

- describe
- organize
- think about

computation

Goal: Enable you to Program



- Readable
- Correct
- Extendable
- Modifiable
- Reusable

Goal: How to learn new PLs

No Java (C#) 15 (10) years ago
AJAX? Python? Ruby? Erlang? F#?...

Learn the anatomy of a PL
- Fundamental building blocks
- Different guises in different PLs

Re-learn the PLs you already know

Goal: How to design new PLs

…“who, me ?”

Buried in every extensible system is a PL

- Emacs, Android: Lisp
- Word, Powerpoint: Macros, VBScript
- Unreal: UnrealScript (Game Scripting)
- Facebook: FBML, FBJS
- SQL, Renderman, LaTeX, XML …

Enables you to choose right PL

"…but isn't that decided by
- libraries,
- standards,
- and my boss ?"

Yes.

My goal: educate tomorrow's tech leaders & bosses, so you'll make informed choices

---

Speaking of Right and Wrong...

Imperative Programming

x = x+1

WTF?

x = x+1

Imperative = Mutation

Imperative = Mutation

Bad!

Don’t take my word for it

John Carmack
Creator of FPS: Doom, Quake,...

Don’t take my word for it

Tim Sweeney (Epic, Creator of UNREAL)

“In a concurrent world, imperative is the wrong default”

Functional Programming

Functional Programming ?

No Assignment.

No Mutation.

No Loops.

OMG! Who uses FP?!

So, Who Uses FP ?

Google

MapReduce

So, Who Uses FP ?

Microsoft

Linq, F#

So, Who Uses FP ?

facebook

Erlang

So, Who Uses FP ?

twitter

Scala

So, Who Uses FP ?

Wall Street

(all of the above)

So, Who Uses FP ?

…CSE 116

Course Mechanics

Mechanics

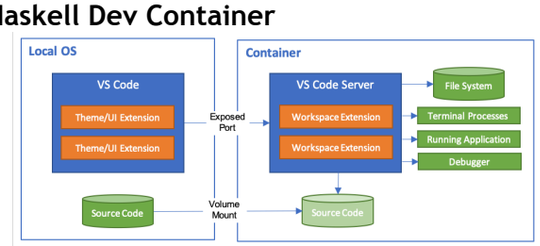
Course website:
https://ucsc-cse-114a.github.io/fall21/

Course texts (optional):

- An Introduction to Functional Programming Through Lambda Calculus by Greg Michaelson. Free pre-print.
- Thinking Functionally with Haskell by Richard Bird. Available online (free via library).
- Programming in Haskell (2nd ed.) by Graham Hutton.
- Real World Haskell by Bryan O'Sullivan. Available online (free via library).
- Learn You a Haskell for Great Good by Miran Lipovača. Available free online
- Write You a Haskell by Stephen Diehl. (incomplete, but useful) Available free online

Mechanics

Haskell Dev Container

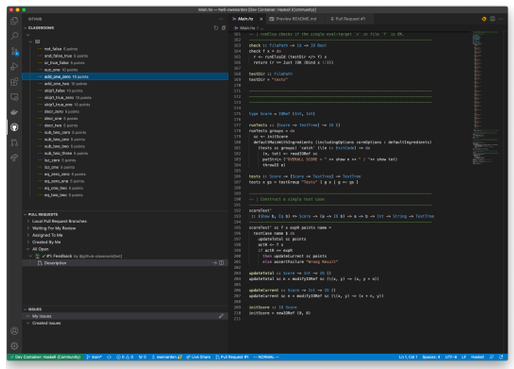


- **https://github.com/UCSC-CSE-114A/cs114a-devcontainer**

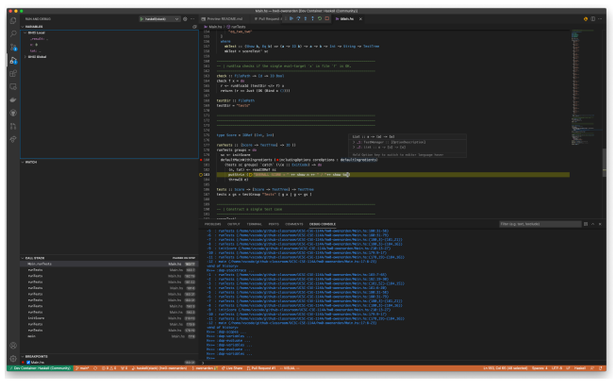
Recommended IDE: VS Code

- New this year, legit IDE setup for Haskell!
  - Devcontainer: A Haskell dev environment is built in a container and VS Code automatically mounts the container volume
  - Also some integrations with Git and GitHub Classroom

VS Code



VS Code



Peer Instruction (ish)

Peer Instruction

- Make class interactive
  - Help YOU and ME understand whats tricky
- Respond to in-class quizzes
  - 5% of your grade
  - Respond to 75% questions
- Bring laptop/phone if you have one

In Class Exercises

1. Solo Vote: Think for yourself, select answer
2. Discuss: Analyze Problem with neighbors
   - Practice analyzing, talking about tricky notions
   - Reach consensus
   - Have questions, raise your hand!
3. Group Vote: Everyone in group votes
4. Class-wide Discussion:
   - What did you find easy/hard?
   - Questions from here show up in exams

Requirements and Grading

- In-Class Exercises: 5%
- Midterm: 30%
- Programming Assignments (6): 30%
- Final: 35%

Two hints/rumors:

1. Lot of work
2. Don't worry (too much) about grade

Note: Regrades must be requested ***in person*** *within two weeks of receiving grade*

Resources

- Online lecture notes
- Readings and exercises
- Webcasts:
  - User: cse-116-1
  - Pass: lambda
- Pay attention to lecture and section!
- Do assignments yourself (+partner)!

Ask for help!

- Lots of help available, will be adding more soon. (watch website)
- Lab sessions 4 days/wk with tutors to help with assignments
- Discussion sections with TAs to help with lecture concepts

Programming Assignments

All assignments are managed through GitHub Classroom (link on course page).

- **You must *push* your submitted code.**

Deadline Extension:

- Four “late days”, used as “whole unit”
- 5 mins late = 1 late day
- Plan ahead, **no other extensions**

See course webpage for HW deadlines

Programming Assignments

Unfamiliar languages
+ Unfamiliar environments

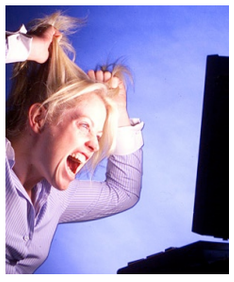
Start Early!

Weekly Programming Assignments

Scoring = Test suite

No Compile, No Score

Weekly Programming Assignments



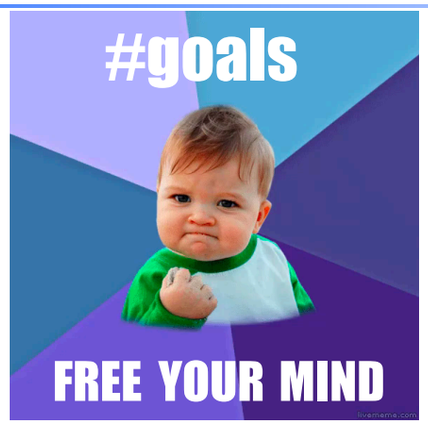
Forget Java, C, C++ …
… other 20th century PLs

Don't complain
… that Haskell is hard
… that Haskell is @!%@#

Immerse yourself in new language

It is not.

Immerse yourself in new language



Word from our sponsor …

- Programming Assignments done **ALONE** or in **(official) groups of two** (as permitted)
- We use plagiarism detection software
  - MOSS is fantastic, plagiarize at your own risk
- **Zero Tolerance**
  - offenders punished ruthlessly
- Please see academic integrity statement:
  - https://ue.ucsc.edu/academic-misconduct.html