CSCI 334:
Principles of Programming Languages

Lecture 1: Course intro

Instructor: Dan Barowy
Williams

1

Topics

What this course is **about**

What to **expect** in this course

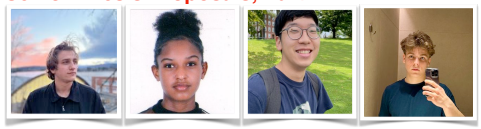
A **small assignment** over the weekend

2

We're going to talk about three things today. What the course is about, what to expect during the semester, and then I'm going to prime you for a small assignment over the weekend. The assignment should not take you more than an hour.

Announcements

- Class field trip to WCMA on **Tuesday, Feb 6**
  Be sure to leave your bags, food, coats, etc. in the WCMA coat room. Please bring pencil and paper.
- First Colloquium this **Friday, Feb 2**
  Senior Thesis Proposals, Part 1



3

Every week:

1. Readings to do **before coming to class**.
2. In-class quiz **on Thursday**.
3. Lab **due Monday at 10pm**.

4

Readings will be listed both on the course website and in each lab handout. Labs are due every Sunday.

Your to-dos

1. Lab 0, **due Monday 2/5**.
   Grade scale:
   A if it is turned in.
   Otherwise… a different grade.
2. Be sure to do the assigned readings (fun and short!) **before next class**.
3. If you plan to use lab computers, check that you can login **today**.
   Only Lida/Kelsey can help you with account problems. They are well-adjusted people who work 9am-5pm during the work week.

5

So, things coming up…

If you don't remember your login…

Email csaccounts@williams.edu

6

If you can't find Kelsey (in TCL 312) or Lida, you can also email them.

If you don’t know/remember the TCL 312/TBL 301 door code…

7

Ask a friend in the CS department.

Why do I study programming languages?

8

A Bicycle for the Mind

9

A short video of Steve Jobs.

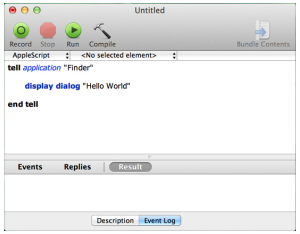
A Bicycle for the Mind

A computer is a key part of this “bicycle”—but not all of it.

10

His bicycle analogy is a good one, but the “bicycle” is more than just the computer hardware.

A Bicycle for the Mind



The other key part is a programming language.

11

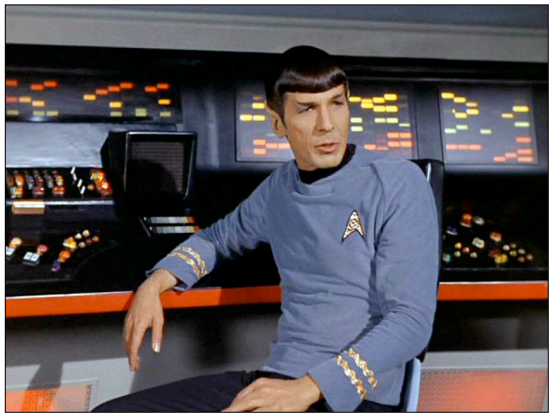
The other part, perhaps the most important part, is the programming language. Here’s a really interesting one called AppleScript. You probably don’t know AppleScript, but you might be able to tell me what it does anyway!

AppleScript

William R. Cook (1963-2021)

12

AppleScript was created by Will Cook, a very influential programming languages researcher who, sadly, passed away recently.



13

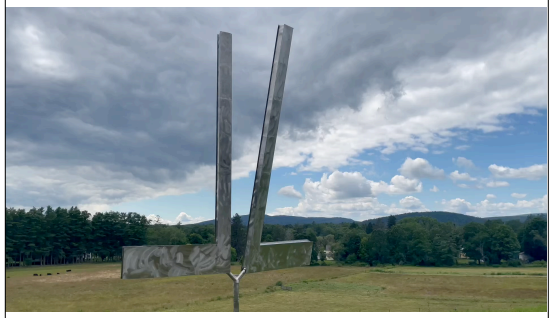
If you've never studied programming language design before, you might think that it is the domain of alien, logical beings. It is certainly true that computers are rigidly logical and their manner of communicating is foreign to us humans.

14

But many of the good PL designers I know are actually very creative, often artistic people, and they want to be able to communicate with a computer in ways that are more natural to humans. They want computers to be a vehicle for their creativity. They spend a lot of time thinking about good interfaces, because that's what a programming language is: a human interface to a computer.

15

When you get this combination right, a logical but expressive human interface, working with a computer can feel like magic.



16

This harmony of engineering and art is not a new idea. The best engineered objects engender both of these qualities. Here's the sister to a sculpture you may have seen before (this one is at Naumkeag in Stockbridge, MA). Notice that these gigantic, sharp blades gracefully dance in the air. What they DON'T do is swing around wildly, cutting people in half! That's because an engineer, George Rickey, built them.

17

Before WWII, George Rickey was a painter. He was drafted into the US Army in 1942, where he became a gunnery technician. After the war, he drew on his new skill in machining to create kinetic sculptures.

FlashFill

18

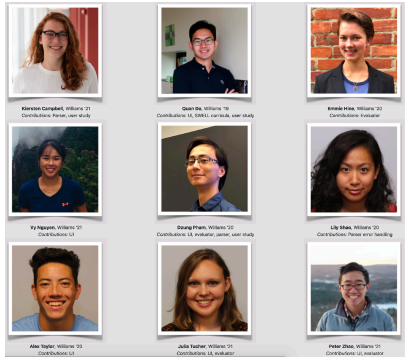
Coming back to the world of programming languages, let me give you some examples that, for me, feel like magic. The first is a tool called FlashFill that comes with Microsoft Excel. When FlashFill was released in 2011, it blew people's minds.

SWELL

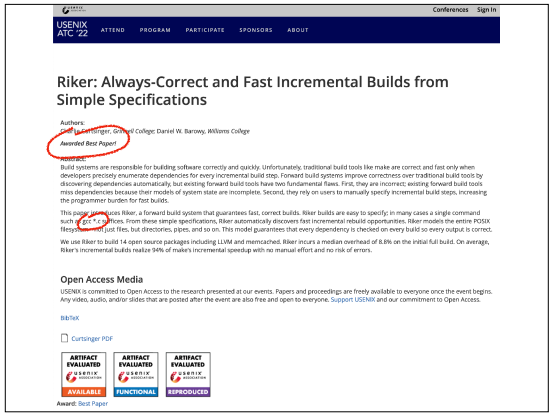
19

Here's a project that I worked on with a number of Williams students. SWELL is a programming language for teaching middle-schoolers how to code. https://swell-lang.org.

Swell



20



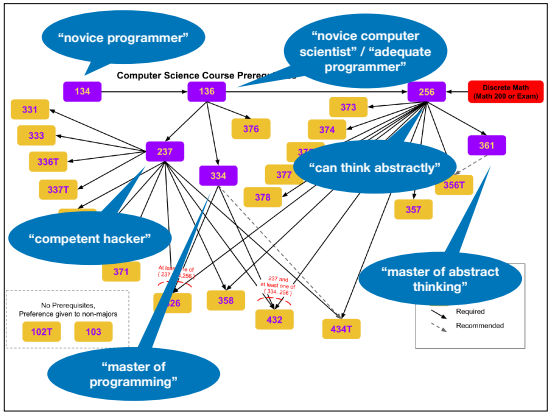
21

And here's another more recent one. If you've ever had to write a Makefile, know now that you don't actually have to. A smart language tool can figure it out for you. This one really felt like magic to me when we got it working. The funny thing is: we knew it was going to work before we coded any of it up, because PL theory told us that it was possible.

Class outcomes

1. Speak the **language of languages**
   a. understand the role of a **language model**
   b. evaluate **fitness** of language for **purpose**
   c. **rapidly learn** new languages
2. Add tools to your **mental toolbox**
   a. techniques for **clear thinking**
   b. become a **much better programmer**

22



23

You might have wondered what the purpose of each class is in our curriculum. Wonder no more! This class is about mastering the key abstractions that will make you an excellent programmer. Side note: in my first year of grad school, my two lab partners were Williams alums. What they knew made me feel incredibly inadequate. I am very honored to have joined this department to carry on that tradition.

Class outcomes

1. Speak the **language of languages**
   a. understand the role of a **language model**
   b. evaluate **fitness** of language for **purpose**
   c. **rapidly learn** new languages
2. Add tools to your **mental toolbox**
   a. techniques for **clear thinking**
   b. become a **much better programmer**
3. **Be your favorite class!**

24

Administrivia

25

OK, now for the boring but important stuff about what to expect in this class.

Syllabus

26

Mentor Check-ins

Goal: to give you the opportunity to **grow** as a **computer scientist**.

Meet with your assigned mentor **5x** during semester.

27

Every week, you will do two things only: do the reading and do the labs. I want this course to be logistically simple. Be sure to do the readings before coming to class, because we will be doing class activities that assume you have read the materials.

Grades

Midterms:	35%
Final Project:	20%
Mentor Check-ins:	5%
Lab assignments:	30%
In-Class Quizzes:	10%

28

If you

1. submit your work **on time**,
2. pay attention to **feedback**, and
3. diligently **correct your mistakes**,

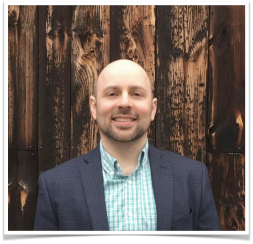
you have **little to worry about** in this class.

29

Help

30

Me



31

Our Wonderful TAs


Valeria


Milo

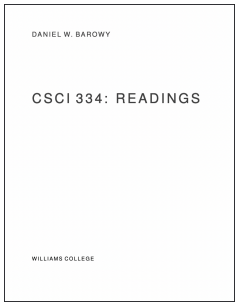

Simon S


Simon J


Jess

32

Course packet



33

BTW, I wrote most of the course packet to ensure that it covered ONLY the things you need to know in this class to complete your final project. So there is no fluff, and I worked really hard (all summer actually) trying to make it as fun and easy a read as possible. Also, if you find serious errors in the text, I will reward you.

Course readings

Probably the most important part of the class.

34

Course readings

In-class quizzes, usually on Thursday.

35

Every week, we will do class activities that help you reinforce what you learned in the book. Take advantage of these by doing the reading before class. You participation grade is binary. If you are here and you do it you get an A.

Course readings

```
$ dotnet fsi

Microsoft (R) F# Interactive version 12.5.0.0 for F# 7.0
Copyright (c) Microsoft Corporation. All Rights Reserved.

For help type #help;;

> let xs = [1; 2; 3; 4]
- let ys = List.fold (fun acc x -> acc + x) 0
- ;;
val xs: int list = [1; 2; 3; 4]
val ys: (int list -> int)
```

The readings are designed to be **active**.

Follow along with the activities in the book.

You'll get much more out of them!

36

Type the code into your computer as you read. It is worth it!

Anonymous grading

37

In case you did not know, I always grade anonymously. I think it's important not to play favorites, but I am human, and it is easy to do. Grading anonymously prevents me from doing that. So be sure to omit your name from all of your lab solutions. I know who work belongs to because git will tell me when I need to know.

Homework late policy:

Late Work

Without prior arrangement, you are expected to turn in all assignments in a timely manner to receive full credit. Nevertheless, I understand that sometimes events conspire to make on-time homework assignment a challenge. Each student may use a maximum of **three free late days** during the course of the semester. A late day permits you to hand in a lab up to 24 hours late, without penalty, no questions asked. **You may use at most one late day on any given assignment.**

To take a late day, be sure to fill out the late day form **before the assignment is due** (https://forms.gle/nPh4rvt8gZUxm4Xo9). Late assignments will be penalized at a rate of **20% per day**.

38

You have three free late days. But you have to tell me when you plan to use them, otherwise I will grade your work prematurely. Please fill out the form. If you have an accommodation or are submitting late for some other reason, also please use the form.

Something even better:
resubmissions

Resubmissions

You may find that occasionally, you do not do as well on an assignment as you had hoped. That's OK! Revisiting a mistake is one of the best ways to learn. To encourage you to engage in this practice, you are permitted to resubmit **two** assignments during the semester. This policy includes labs 0–7 and the two midterm exams but not labs 8-10 or the final project.

A resubmission allows you to earn back **up to 50% of the missing points**. For example, if you received a 75% on an assignment, you may earn up to 87.5% upon resubmission.

Resubmissions must be submitted in the following manner:

1. They must be submitted before the end of the final exam reading period.
2. They must include both the original work and the new submission.
3. They must be accompanied with a typed document, written in plain language, that explains, for every correction:
   (a) what the error was in the original work,
   (b) how you fixed the error, and
   (c) why the new version is correct.

Please note that resubmissions must be typed or they will not be accepted. Detailed instructions for submitting a resubmission will be distributed via a separate handout.

39

You also get two "redos." If you've taken a class with me before, you probably know about these.

Honor Code

40

Solutions to assignments **should not be posted in any public forum**, including public git (e.g., GitHub, GitLab, etc) repositories. Students taking our courses should not be looking for solutions, but tempting them by making solutions available is inappropriate. This applies not just to the semester you are taking the course, but to the future as well.

41

COVID, etc.

42

git

https://aslan.barowy.net

43

We will be using a different git repository this semester. This is a little bit of an experiment on my part, but from your perspective, it should look familiar. Nothing really new here.

Course website

https://williams-cs.github.io/cs334-s24-www/

44

Course Organization

First half

- language models
- theoretical foundations
- functional vs. imperative languages
- new ways of thinking

45

Course Organization

First half

46

Course Organization

Second half

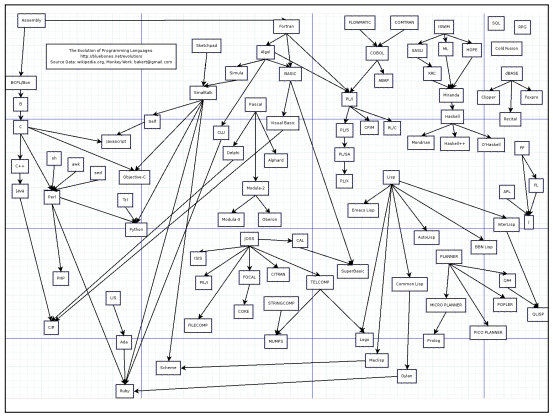
- language architecture
- object oriented programming
- projects!

47

Course Organization

Second half

48



49

We could go broad and talk about the zoo of programming languages

---

You are going to get your hands dirty

50

But I prefer to go deep.

I always want you to succeed

Our notions for success may not always be the same, but I promise you: I do not assign busy work.

51

I always care about what you think

52

1. Optional feedback on assignments (for bonus credit!)
2. Optional, anonymous feedback form on course website

Something new: `push-checker`

53

`push-checker`

54

- To trigger:

```
$ git commit -m "check"
$ git push
```

- A checklist will appear in your repo's **issues** tab.
- You get **1 attempt per day**.

  (lab 0 gets 10 attempts per day)

Recap & Next Class

Today:

- Course goals
- Course structure

Next class:

- WCMA!