
Principles of Programming Languages

| | | | | | |
|------------------------|---|--------|--------|-------|-------|
| Instructor | Prof. Dan Barowy | | | | |
| Office | TCL 307 | | | | |
| Email | dbarowy@cs.williams.edu | | | | |
| Phone | 597-2807 | | | | |
| Office Hours | Tuesdays, 1:30pm–2:30pm and Wednesdays, 10:00am–noon or by appointment in TCL 307 | | | | |
| Homework Help Sessions | Thursdays, 7:00–9:00pm in TCL 206 | | | | |
| TA Hours | M | Tu | W | Th | F |
| | 7–9pm | 7–11pm | 7–11pm | 7–9pm | 6–8pm |
| | in TCL 312 (UNIX Lab) | | | | |
| Lectures | Section 1: Tuesday and Thursday 9:55am–11:10am Section 2: Tuesday and Thursday 11:20am–12:35pm in TCL 206 | | | | |
| Web Page | http://www.cs.williams.edu/~dbarowy/cs334s18 | | | | |

Readings

(Required) *Concepts in Programming Languages*, John C. Mitchell.
Additional readings will be posted on the web site.

Course Objectives

Although it may not seem like it at times, all programming languages are designed for humans, not for computers. As such, choosing the wrong language for the job can have a big effect on productivity. A language choice may also have important consequences for the quality and correctness of the resulting programs. A major goal of this course is to “get under the hood” of programming language design, enabling you to make informed choices about the appropriateness of a given language for a project.

At the end of this course, you should be able

1. to quickly learn an unfamiliar programming language;
2. to know, at a high level, how a programming language works;
3. to understand the tradeoffs between functional, logic, and object-oriented programming paradigms;
4. and finally, to identify the pros and cons of a language with respect to a given a programming problem.

As in other CS courses, we will discuss alternate ways of solving the same problem. Because programming languages are intrinsically tied up (and motivated by) programming problems, we will not only investigate their features, but also the engineering problems that led to their development.

“Is this on the exam?”

You should assume that all material covered in class is on the exam. I sometimes embellish a lecture with interesting history or personal anecdotes. These embellishments will not be on the exam, but I hope you enjoy them and that they give you an appreciation for the people behind the languages.

This course will also involve extensive reading on your part, including the textbook and supplemental readings. You are expected to read and understand all assigned materials, even if we do not cover it in class. All assigned material is fair game for exam questions.

Time Commitment

Expect to spend at least 10 hours per week working on assignments outside of class meetings.

Lectures

Lectures are mandatory. I expect you to attend and participate.

Tentative Schedule

Note that this schedule is subject to revision.

| Week | Date | Topic |
|---------|-----------|---|
| Week 0 | 1/31 | Intro, Halting Problem |
| Week 1 | 2/6–2/8 | Lisp and Functional Programming |
| Week 2 | 2/13–2/15 | PL Foundations |
| Week 3 | 2/20–2/22 | More Foundations and ML |
| Week 4 | 2/27–3/1 | ML and types |
| Week 5 | 3/6–3/8 | Stacks and Scope |
| Week 6 | 3/13–3/15 | Control Flow, Exceptions |
| Week 7 | 4/3–4/5 | Modularity, Abstraction, OOP |
| Week 8 | 4/10–4/12 | OOP |
| Week 9 | 4/17–4/19 | C++ and Java |
| Week 10 | 4/24–4/26 | Object-Functional Programming and Scala |
| Week 11 | 5/1–5/3 | Logic Programming |
| Week 12 | 5/8–5/10 | Domain Specific Languages |

Homework

Homework assignments are a combination of problem sets with written answers and programming assignments. For some of these you will be required to work with a partner.

Problems sets are assigned weekly, typically on Tuesday. All homework should:

- be completed no later than 11:59PM on the due date using GitHub,
- be typeset using \LaTeX (a programming language!),
- include source code for questions involving programming, and
- list any students with whom you discussed the problems (see Honor Code handout).

You will be asked to fix and resubmit homework not following these conventions. Homework will not be accepted via email.

Late Homework

Each student may use a maximum of three late days during the course of the semester. A single late day enables you to hand in the homework up to 24 hours after the original due date. You may combine up to two late days for any given assignment. Once those late days are used up, late homework will be penalized 20% per day.

Lab Resources for Homework Assignments

You are strongly advised to use the Computer Science Department's Unix "lab" computers for the programming problems. These computers are preconfigured with all of the required software. If you are not familiar with the Unix computing environment, please speak with me or the TAs as soon as possible so we can bring you up to speed on what you need to know. You may also see Mary Bailey to get your Unix password if you have forgotten it.

You may use your personal computer for assignments, but please be advised that if you do so, all technical support issues are your own personal responsibility. Requests for technical support for your personal machine will be answered with "use a lab machine."

Exams

There will be midterm and final exams covering both lectures and readings. The midterm will occur after Spring Break. The final will be a 24-hour Take Home Exam.

Grading

Grades will be determined roughly as follows:

Midterm: 25%

Final Exam: 35%

Homework and programs: 35%

Other (class participation, attendance, quizzes, etc.): 5%

Please note that "A" grades are awarded only for exceptional work. This means that, in order to receive a top grade, you must stretch yourself academically. Stretching most clearly manifests itself in the form of "challenge" problems. Challenge problems may require study beyond the material covered in lectures or readings. You can expect to see at least one challenge problem on each homework assignment. Other ways to exceed an A- might include extra points for exceptionally clear and insightful answers to non-challenge problems or other exemplary contributions to the course.