

# Computer Science & Engineering 413

## Programming Languages and Their Implementation

**Instructor: Stuart Reges**

Email: [reges@cs.washington.edu](mailto:reges@cs.washington.edu)

Phone: 685-9138

Office: Gates Center (CSE2), room 305

Office hours: Wednesdays 3-5 pm

### Textbook

There is no required textbook.

### Course Overview

The prerequisite for this course is CSE373 and we will assume that you come to the course with significant experience in either Java or C++. We will build on that experience by examining three languages that provide an interesting contrast to what you already know. The goal is to give you a much broader view of programming and to teach you the basic terminology that we use to distinguish different programming languages. For example, the following table shows the three languages we will study along with Java/C++ categorized by programming paradigm (object-oriented vs functional) and by the kind of type system it uses (static vs dynamic):

|                   | Object-oriented | Functional |
|-------------------|-----------------|------------|
| Statically typed  | Java/C++        | OCaml      |
| Dynamically typed | Ruby            | Scheme     |

### Computer Access/Software

We will be using the Ed digital platform for completing homework and other programming that you might want to do in OCaml and Ruby. We will have you install a special software environment for Scheme.

### Grading

You will be expected to complete a variety of programming assignments for this course, to take two exams, and to complete various short programming exercises. The resulting scores will be combined according to the following weightings:

|     |   |
|-----|---|
| 50% | weekly homework assignments               |
| 15% | midterm (Friday, 5/2/25, 1:30-2:20)       |
| 35% | final exam (Monday, 6/9/25, 2:30-4:20 pm) |

Contact us in the first two weeks of the quarter if you have a conflict with these dates and times. Using the weightings above, each student's scores will be turned into an overall score ranging from 0 to 100 percent. These will be turned into grades as follows:

|     |              |     |              |
|-----|--------------|-----|--------------|
| 90% | at least 3.5 | 70% | at least 1.5 |
| 80% | at least 2.5 | 60% | at least 0.7 |

### Course Web Page

Information about the course will be kept at <http://cs.uw.edu/413>. Links to course handouts will be kept on this page along with useful links to other class resources.

### Religious Accommodations

See <https://registrar.washington.edu/staffandfaculty/religious-accommodations-policy/>.

## Indigenous Land Acknowledgement

I acknowledge that by the labor theory of property the Coast Salish people can claim historical ownership of almost none of the land currently occupied by the University of Washington.

## Late Policy

Each assignment will list its due date. Most will be due on Thursdays at 11 pm. Each student will have a total of ten “free” late days (a late day is 24 hours of lateness). There are no partial days, so assignments are either on time, 1 day late, 2 days late, etc. Because of this generous policy, students will not be granted extensions for assignments unless they have highly extenuating circumstances. Once a student has used up all free late days, each successive late day will result in a loss of 2 points. No assignment will be accepted more than 4 days after its due date.

We will grade only one version of any given program. If you make multiple submissions for an assignment, we will grade the last version submitted. If you submit a version that you later decide you do not want to have graded, you must warn your TA not to grade that version and to wait for a later submission from you.

## Policy on Collaboration

You are to complete programming assignments individually. You may discuss the assignment in general terms with other students including a discussion of how to approach the problem, but the code you write must be your own. The intent is to allow you to get some help when you are stuck, but this help should be limited and should never involve details of how to code a solution. **You must abide by the following:**

- You may **not** work as a partner with another student on an assignment.
- You may **not** show another student your solution to an assignment.
- You may **not** have another person (current student, former student, tutor, friend, anyone) “walk you through” how to solve an assignment.
- You may **not** post your homework solution code online to ask others for help. This includes public message boards, forums, file sharing sites and services, or any other online system
- You are **not** to examine online solutions that you might find on the web.

Under our policy, a student who gives inappropriate help is equally guilty with one who receives it. Instead of providing such help, refer other students to class resources (lecture examples, section, or emailing a TA or instructor). You must not share your solution and ideas with others. You must also ensure that your work is not copied by others by not leaving it in public places, emailing it others, posting it on the web, etc.