

# Welcome to Programming Languages!

## Principles of Programming Languages

Colorado School of Mines

<https://lambda.mines.edu>

## Sumner

(and you should call me "Sumner")

### **Absolutely not my name:**

- Jonathan (I don't go by my first name ever; call me Sumner)
- Dr. Evans (I am NOT a doctor)
- Mr. Evans (Wait, my dad isn't here)
- Professor Sumner (Insulting to actual professors)

# What is this course?

- **Programming Language Design & Implementation:**
  - What makes a good programming language?
  - What are the common trade-offs in programming language design?
  - What techniques are used to implement programming languages?
  - We're even going to implement our own programming languages!
- **Theoretical Foundations of Programming Languages**

# Why take a PL course?

- **If you're going to be a software engineer:**
  - As we work in a rapidly evolving industry, you'll be able to learn new languages quicker, and make the right choices for your next software design
  - You'll learn practical skills, such as parsing complex inputs, even if the skills aren't applied to making a programming language
- **If you're going to be a computer scientist:**
  - Through programming languages, you will experience a very practical application of computational theory
  - We will cover the mathematical foundations of programming languages
- **If you're going to do something else:**

Quote from a student a few semesters ago

*"I didn't realize how useful PL would be until I wrote an assembler for my internship this past summer!"*

# What goes into this PL course?

## ■ Programming:

- Python (serving as a multi-paradigm OO-language)
- Racket (serving as a first step into language oriented programming)
- One language of your choice (Language Explore Project)
- *Many more*

## ■ Theory:

- Parsing
- Typing systems
- Memory management
- Lambda calculus
- Regular expressions and finite state machines
- *Much more*

# Assignments & Projects

*Homework:* You'll be given both programming and theory-related homework assignments

*Explore Project:* You'll be given the chance to study a language of your choice. You will submit some example programs that you wrote in this language and give a short presentation on it

*SlytherLisp:* You'll implement a programming language interpreter for a Scheme-like programming language over the course of the semester



<https://piazza.com/mines/spring2019/csci400/home>

# Expectations

- **Prerequisites:** CSCI 262 (Data Structures) and CSCI 306 (Software Engineering)<sup>1</sup>
- Basic Linux skills are a *must*; CSCI 274 recommended
- All of your code is expected to run on the machines in the ALAMODE (BB 136) Linux lab
- Familiarity with C (or C++) will be helpful

---

<sup>1</sup>If you have prior experience in Java or other OOP languages, I may (at my own discretion) waive this prerequisite. Please contact me if you are interested in pursuing this option.

# Linux Support

The **Mines Linux Users Group** can help you install, setup, and learn about Linux. They have meetings open to the campus Thursdays at 6 PM.

They also have Linux Help Sessions at 5 PM every Thursday before the main meeting. Your instructor happens to be the Linux Help Guru.

Sign up for the mailing list for more info:  
<https://lug.mines.edu/maillinglist>.

# Learning Groups

You may notice you are seated with a Learning Group:

- Please sit with them each lecture
- You will be given assignments to complete with your group outside of lecture
- In class, you will often share your findings with your group members
- We will change groups every few weeks

# Grading Policy

This course is worth 1000 points. The points are allocated as follows:

- 1 Two Exams.** 170 points each. Total 340 points.
- 2 Two Projects.** Total 350 points.
- 3 Three Homework Assignments.** Total 250 points.
- 4 Learning Group Participation.** 60 points.

The Syllabus has a much more detailed explanation of this breakdown.

The course uses the plus/minus grading scale.

# Late Policy

You can turn in homework and deliverables on projects using the **slip day system**. Here's how it works:

- 1 You currently have 8 slip days.
- 2 For each 24-hours you turn in an assignment late, it will cost you one slip day.
- 3 Note the number of slip days you are spending when you turn in.
- 4 You cannot spend more than 5 slip days on a single assignment without asking for instructor permission.

# Textbooks and Other Resources

- **The Racket Guide:**  
<https://docs.racket-lang.org/guide/index.html>
- **Beautiful Racket:** <https://beautifulracket.com/> (online, honor system payment)
- **Structure and Interpretation of Computer Programs:**  
<https://mitpress.mit.edu/sicp> (online, free)
- Other readings may be requested by instructor throughout the semester