Programming Paradigms

Introduction (Part 1)
About Me: Michael Pradel

- Since 9/2019: Full Professor at University of Stuttgart

- Before
  - Studies at TU Dresden, ECP (Paris), and EPFL (Lausanne)
  - PhD at ETH Zurich, Switzerland
  - Postdoctoral researcher at UC Berkeley, USA
  - Assistant Professor at TU Darmstadt
  - Sabbatical at Facebook, Menlo Park, USA
About the Software Lab

- My research group since 2014
- Focus: Tools and techniques for building **reliable**, **efficient**, and **secure** software
  - Program testing and analysis
  - Machine learning, security
- Thesis and job opportunities
Overview

■ Motivation
  □ What the course is about
  □ Why it is interesting
  □ How it can help you

■ Organization
  □ Exercises
  □ Grading

■ Introduction
  □ Programming languages:
    History, paradigms, compilation, interpretation
The Role of Programming

- **Programming**: Essential form of expression for a computer scientist
  - "The limits of my language mean the limits of my world." (Ludwig Wittgenstein)

- Programming languages determine what algorithms and ideas you can express
Goal of this Course

Understand how programming languages (PLs) work

- How are languages defined?
- What language design choices exist?
- How are languages implemented?
Why Learn About PLs?

Enables you to

- choose right PL for a specific purpose
- choose among alternative ways to express things
- make best use of tools (e.g., debuggers, IDEs, analysis tools)
- understand obscure language features
- simulate useful features in languages that lack them
Concepts vs. Languages

This course is not about

- All details of a specific language
- A systematic walk through a set of languages

Instead, this course is about

- Concepts underlying many languages
- Various languages as examples
Isn’t Knowing {Pick a PL} Enough?

- **Complex systems**: Built in various languages
  - E.g., Facebook: Wild mix of languages covering various language paradigms

- **New languages** arrive regularly (and old ones fade away)
Isn’t Knowing \{Pick a PL\} Enough?

Complex systems: Built in various languages

- E.g., Facebook: Wild mix of languages covering various language paradigms
- New languages arrive regularly (and old ones fade away)

![TIOBE Programming Community Index](source: www.tiobe.com)