Programming Lanaugages (0) Roadmap

Kenjiro Taura

Objectives of programming languages

▶ easy to learn

- ▶ easy to get programs right
- \blacktriangleright execute fast
- ▶ safe (avoid disaster)

The course objectives

- get how different programming languages approach these goals differently
- ► topics
 - ► types
 - ▶ code reusability (generics, subtyping, inheritance, etc.)
 - memory management/safety
 - ▶ performance
 - building compilers
- the main course work: you choose a language from below and do course work in it
 - ► Go
 - Julia
 - ► OCaml
 - Rust

The course format

- ▶ after a few weeks, we group students
- each group will be four students, each working on a different language
- we discuss approaches to the above objectives taken by different languages, within and across groups
- you are expected to engage in these discussions and other activities (not just to listen to talks and get things done)

Evaluation

- ▶ small coding-centric assignments (a few times)
- reflective essay (every week, until the end of the next day)
- ▶ participations (esp. in discussions)
- a final report (building a simple C compiler by default; other options are possible)

no exams

assignments / reports subject for evaluation/grading are indicated as assignments in UTOL

Reflective essay

- every week, you write a short reflective essay that expresses such things as
 - what you have learned (conceptualize/internalize experiences)
 - what came through *your* mind while listening to the talk and working on assignments
 - how you worked on the exercise (where you struggled, how you got help, how useful was AI, etc.)

Today

- answer a survey on your programming language experiences and the preferred (natural) language
- ▶ play with the Jupyter environment
 - choose a language you work on (for today)
 - write a few programs in it
- practice submission (submit pl00_intro in Jupyter and UTOL (Assignment 1))
- ▶ work on assignment pl01_basics
- ▶ and share your answers!

About AI (ChatGPT, Copilot, ...)

- generally OK to use it for coding exercises and technical assignments
- do not use it for reflective essays (obvious. it's about you)
- ▶ AI solves many basic coding problems esp. in early weeks
- ▶ basic coding problems are still given for
 - ► fun,
 - ▶ learning main *ideas* behind language design, and
 - prerequisite for discussing implementation (memory management, compilers, etc.)

main takeaway: you don't have to be struggled by minor/syntactic errors (AI will fix your mistake); you instead focus on ideas/concepts