DM22 Programming Languages

Rolf Fagerberg

Spring 2005

Programming Languages

Imperative

Assignment, sequence, decision, iteration, sub-routine/ procedure/function/method, variables hold state.

Object-Oriented

Same + well-developed modularisation tools.

Functional

No assignments, no state. Only functions and expressions. High level of abstraction. Easier to prove behavior. Program text close to math.

Logic

Declarative, logic based. Facts, rules, goals.

Special purpose

Scripting, web, macros, concurrency,. . .

Why Study Programming Languages?

- New ways to express ideas in programming.
- New perspectives on known ways.
- Raised level of abstraction.
- Choose right language for task/problem domain.
- Pick up new languages more easily.

The language determines what can be (easily) expressed

$$\Downarrow$$

it influences how you think about programming.

Programming Language Courses

Two options:

- Compare a large number of languages for similar and different features (language phylogeny).
- Learn to program seriously in a few languages complementing your current knowledge.

DM22 uses $\boxed{\text{second option.}}$

Teaches progamming in Haskell (functional) and Prolog (logic).

Related courses: DM17 (language syntax, computational power of languages), DM18 (language semantics, compilation/language implementation).

DM22 Formally

Literature

- Simon Thompson: *The Craft of Functional Programming*, 2nd edition.
- W.F. Clocksin and C.S. Mellish: *Programming in Prolog*, 5th edition (or 4th).

Exam

Written exam, 4 hours, 13-scale.

Compulsory Projects

One in Haskell, one in Prolog, pass/fail.

Hours

Lectures: Mondays 14-16 in U9.
Examinatorier: Thursdays 8–10 in U49C *or* Fridays 12-14 in U29.