DM509 Programming Languages

Rolf Fagerberg

Fall 2006, 2nd quarter

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

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 second option.

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

Related courses: DM517 (language syntax, computational power of languages), DM516 (language semantics, compilation/language implementation).

DM22 Formally

Literature

- R. Bird: *Introduction to Functional Programming using Haskell*, 2nd edition.
- W.F. Clocksin and C.S. Mellish: *Programming in Prolog*, 5th edition.

Exam

Written exam, 4 hours, 13-scale.

Compulsory Projects

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

Hours

Mondays 10–12 in U14.
Tuesdays 12–14 in U14.
Thursdays 10–12 in U24.
(Week two: Friday 10–12 in U14 instead of Tuesday)