# DM22 Programming Languages

Rolf Fagerberg

Spring 2006

# 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 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

- Richard Bird: *Introduction to Functional Programming using Haskell*, 2nd edition.
- A book on Prolog (still to be chosen).

## Exam

Written exam, 4 hours, 13-scale.

## Compulsory Projects

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

## Hours

Lectures: Tuesdays 10-12 in U20.
Exercise classes: Wednesdays 8–10 in U20.