

Advanced Functional Programming

Software Engineering Chair and Programming Systems Lab

Small-group work

Divide into groups and discuss the following questions. You don't have to answer the questions in the given order, so pick your favorite questions.

1. Two key questions for any scientific paper are: what problem do the authors solve? What is their contribution? To answer the latter, think about what you can do now but could not do before.
2. Is overloading in Haskell resolved at compile time or at run time?
3. In an object-oriented language a method can take arguments of a certain class or any subclass. This is also known as polymorphism; is it ad-hoc, parametric, or neither of these?
4. Identify parametric and ad-hoc polymorphism in Java or C++.
5. A polymorphic function in a functional language is independent of the structure of its argument: the length of a list can be computed no matter whether the list holds numbers or trees. How is this implemented in ML or Haskell? How in C++?
6. In Java every object carries a vector of methods. This poses a problem for binary methods like `a == b` because either `a.equals(b)` could be meant or `b.equals(a)`. Does this problem exist with type classes, too?

The plenary session

1. Recorders from individual groups will present their groups' conclusions.
2. Class discussion. We will discuss the answers together.

Homework Assignment

1. Read *Higher-Order Functions for Parsing* by Graham Hutton, Journal of Functional Programming, 2(3), pages 323–343, 1992.
2. Summarize the paper *in your own words* on one page. Put your name and student ID on your summary and drop off a printout at office 326/45 until Monday, November 14th at noon (12am). If the door is closed, slide your printout under the door. No Emails.