Fachbereich Informatik

Programmiersprachen und Softwaretechnik

Prof. Dr. Klaus Ostermann

Responsible for the lab
Philipp Schuster
philipp.schuster@uni-tuebingen.de

EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Programming Languages 2

Homework 10 – WS 18 Tübingen, 10. Januar 2019

In order to be admitted to the exam, you have to successfully submit your homework every week, except for 2 weeks. A successful submission is one where you get at least 1 point.

Handin   Please submit this homework until Thursday, January 17, either via email to Philipp Schuster (philipp.schuster@uni-tuebingen.de) before 12:00, or on paper at the beginning of the lab.

Groups   You can work in groups of up to 2 people. Please include the names and Matrikelnummern of all group members in your submission.

Points   For each of the Tasks you get between 0 and 2 points for a total of 6 points. You get:
  1 point, if your submission shows that you tried to solve the task.
  2 points, if your submission is mostly correct.

Task 1: Type Reconstruction

Find suitable types for the question marks in the following program:

$\lambda x :?_1.\lambda f :?_2.\,\text{if}(\text{iszero}\,x)\,\text{then}\,x\,\text{else}\,f\;\;x$

Task 2: Weakest Unifier

Consider the following grammar of types with unification variables T_i:

$\langle \textit{type} \rangle ::= \text{Num} \mid \langle \textit{type} \rangle \rightarrow \langle \textit{type} \rangle \mid T_i$

Find the weakest (most general) unifier (if it exists) of the following pairs of types with unification variables:

1. $\text{Num} \rightarrow T_1$ and $T_2 \rightarrow \text{Num}$

2. Num and $T_1 \rightarrow T_1$

3. T_1 and T_2

Task 3: Implement Type Reconstruction

Implement the type reconstruction algorithm from the lecture for lambda calculus extended with numbers and addition but not recursion in a programming language of your choice. Do not try to write a parser but assume a representation of terms in memory. Examples for a representation of terms and types in Scala are on the website.