Programming Languages 2

Homework 13 – WS 18

Tübingen, 31. 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, February 7, 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: Equivalence of types

Show that the types \( \forall X :: \ast \cdot ((\lambda Y :: \ast \cdot Y) X) \rightarrow X \) and \( \forall X :: \ast \cdot X \rightarrow X \) are equivalent. Draw a derivation tree for the relation \( \equiv \) from the lecture.

Task 2: Programming on the type level

We encode natural numbers on the type level in System F omega as repeated application of a given type constructor. To abbreviate types of this kind we define \( \text{TypeNat} := (\ast \Rightarrow \ast) \Rightarrow (\ast \Rightarrow \ast) \). Define a type \( \text{Add} \) that performs addition on two natural numbers encoded on the type level. The type should have kind \( \text{TypeNat} \Rightarrow \text{TypeNat} \Rightarrow \text{TypeNat} \).

Task 3: Well-kindness

Show that your type \( \text{Add} \) from Task 1 has the required kind. In other words, draw a derivation tree for \( \vdash \text{Add} :: \text{TypeNat} \Rightarrow \text{TypeNat} \Rightarrow \text{TypeNat} \).