



Programmiersprachen II

Hausaufgabe 12 – WS 16

Tübingen, 6. Februar 2017

Abgabe Geben Sie diese Hausaufgabe bis Donnerstag den 09. Februar 2017 bis 12:00 per Email an Philipp Schuster (philipp.schuster@uni-tuebingen.de) oder auf Papier in Raum B211 ab. Es wird an diesem Tag keine Übung geben, Sie können aber Fragen per Email stellen.

Gruppen Sie können in Gruppen von bis zu 2 Personen arbeiten. Schreiben Sie in jedem Fall die Namen und Matrikelnummern aller Gruppenmitglieder mit auf die Hausaufgabe / in die Email. Wenn Sie in einer Gruppe arbeiten, achten Sie darauf, dass alle Mitglieder der Gruppe den Stoff verstehen. Nur dann sind die Hausaufgaben eine gute Vorbereitung auf die Prüfung.

Punkte Sie können für die Aufgaben dieser Woche jeweils zwischen 0 und 2 Punkten bekommen. Insgesamt also zwischen 0 und 6 Punkten. Sie bekommen für die Aufgaben jeweils:

1 Punkt, wenn Ihre Abgabe zeigt, daß Sie sich mit der Aufgabe ernsthaft beschäftigt haben.

2 Punkte, wenn Sie die Aufgabe weitgehend korrekt gelöst haben.

Um zur Klausur zugelassen zu werden müssen Sie mindestens 50% der maximal möglichen Punkte in den Hausaufgaben erreichen. Mit 60% bis 100% der möglichen Hausaufgabenpunkte erhalten Sie einen Bonus von 0% bis 20% der Klausurpunkte in der Klausur.

Aufgabe 1: Programmieren auf dem Typelevel

Wir kodieren natürliche Zahlen auf dem Typelevel in System F Omega als wiederholte Anwendung eines Typoperators, also als Typen mit Kind $(* \Rightarrow *) \Rightarrow (* \Rightarrow *)$. Als Abkürzung definieren wir diesen Kind als `TypeNat`. Geben Sie einen Typen an, der der Addition zweier so kodierter natürlicher Zahlen entspricht. Der Typ soll Kind `TypeNat \Rightarrow TypeNat \Rightarrow TypeNat` haben.

Aufgabe 2: Well-kindedness

Zeigen Sie, dass Ihr Typ `T` aus Aufgabe 1 im leeren Kontext den geforderten Kind hat. Das heißt, dass $\vdash T :: \text{TypeNat} \Rightarrow \text{TypeNat} \Rightarrow \text{TypeNat}$ gilt. Zeichnen Sie dazu einen Ableitungsbaum.

Aufgabe 3: Äquivalenz von Typen

Zeigen Sie, dass die Typen $\forall X:: * . ((\lambda Y:: * . Y) X) \rightarrow X$ und $\forall X:: * . X \rightarrow X$ äquivalent sind. Zeichnen Sie dazu einen Ableitungsbaum für die in der Vorlesung definierte Relation \equiv .