



Programmiersprachen II

Hausaufgabe 8 – WS 16

Tübingen, 20. Dezember 2016

Abgabe Geben Sie diese Hausaufgabe bis Donnerstag den 12. Januar 2017 ab. Entweder bis 12:00 per Email an Philipp Schuster (philipp.schuster@uni-tuebingen.de) oder zu Beginn der Übung auf Papier.

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: Join

Wir betrachten folgende Menge von Typen:

$\langle type \rangle ::= \text{'Top'} \mid \langle type \rangle \rightarrow \langle type \rangle \mid \{l_i : \langle type \rangle_i\}$

Wir betrachten folgende Regeln der Subtyp-Relation:

S-TOP
 $\vdash S <: \text{Top}$

S-ARROW
 $\frac{\vdash T_1 <: S_1 \quad \vdash S_2 <: T_2}{\vdash S_1 \rightarrow S_2 <: T_1 \rightarrow T_2}$

S-RCD
 $\frac{\{l_i \mid i \in 1..n\} \subseteq \{k_j \mid j \in 1..m\} \quad \text{for each } k_j = l_i, \vdash S_j <: T_i}{\vdash \{k_j : S_j^{j \in 1..m}\} <: \{l_i : T_i^{i \in 1..n}\}}$

Was ist jeweils das Join der folgenden Paare von Typen:

1. $\{a : \text{Top}\}$ und Top
2. $\{a : \text{Top}, b : \text{Top}\} \rightarrow (\{x : \text{Top}\} \rightarrow \{z : \text{Top}\})$ und $\{a : \text{Top}\} \rightarrow (\{y : \text{Top}\} \rightarrow \{z : \text{Top}\})$
3. $(\{a : \text{Top}, b : \text{Top}\} \rightarrow \{x : \text{Top}\}) \rightarrow \{z : \text{Top}\}$ und $(\{a : \text{Top}\} \rightarrow \{y : \text{Top}\}) \rightarrow \{z : \text{Top}\}$

Aufgabe 2: Meet

Was ist jeweils das Meet der Paare von Typen aus Aufgabe 2.

Aufgabe 3: Typchecker für eine Sprache mit Subtyping

Implementieren Sie den in der Vorlesung vorgestellten algorithmischen Typchecker für die in der Vorlesung vorgestellte Sprache mit Subtyping in der Programmiersprache Ihrer Wahl. Beispiele für die Repräsentation der Terme und Typen in Haskell und Java befinden sich auf der Webseite. Erweitern Sie optional die Implementation der Sprache um ein if-then-else Konstrukt. Finden Sie beim Typchecken von if-then-else den kleinsten Supertyp.