Institut für Informatik III
Universität Bonn
HD Dr. R. Hinze

Römerstraße 164
D–53117 Bonn
24. Mai 2007

Übungen zu „Prinzipien von Programmiersprachen“

Blatt 5

Aufgabe 17. Allein aus dem Typ einer polymorphen Funktion lässt sich eine Eigenschaft der Funktion ablesen. Diese Eigenschaft heißt das *freie Theorem* der Funktion. Zum Beispiel gilt für jede Funktion $m : \forall\alpha.(\alpha \to \alpha) \to ([\alpha] \to [\alpha])$

$$\mathit{map}\ r \cdot m\ \langle A\rangle\ f = m\ \langle B\rangle\ f \cdot \mathit{map}\ r \quad \Longleftarrow \quad r \cdot f = f \cdot r$$

Hat φ den polymorphen Typ $\forall\alpha.\tau\ \alpha$, dann besagt das freie Theorem von φ, dass $(\varphi\ \langle A\rangle, \varphi\ \langle B\rangle) \in \tau\ R$ für alle Relationen $R \subseteq A \times B$. Hierbei wird der Typ von φ als binäre Relation gelesen.

$$\begin{aligned}
(f, g) \in R \to S &\iff \forall a, b\ .\ (a, b) \in R \Longrightarrow (f\ a, g\ b) \in S \\
((a_1, a_2), (b_1, b_2)) \in R{\times}S &\iff (a_1, b_1) \in R \wedge (a_2, b_2) \in S \\
([a_1, \ldots, a_n], [b_1, \ldots, b_n]) \in [R] &\iff (a_1, b_1) \in R \wedge \cdots \wedge (a_n, b_n) \in R
\end{aligned}$$

Zum Beispiel gilt für m, dass $(m\ \langle A\rangle, m\ \langle B\rangle) \in (R \to R) \to ([R] \to [R])$ für alle Relationen $R \subseteq A \times B$. Spezialisieren wir die Relation R zu einer Funktion r, dann erhalten wir die obige Aussage. *Beachte:* wenn $(a, b) \in R \iff r\ a = b$, dann ist $(as, bs) \in [R] \iff \mathit{map}\ r\ as = bs$.

1. Wie lautet das freie Theorem von Funktionen des Typs

   $$\forall\alpha.\alpha \to \alpha$$

   Zeige, dass id die einzige Funktion dieses Typs ist.

2. Wie lautet das freie Theorem von Funktionen des Typs

   $$\forall\alpha.(\alpha \to \alpha) \to (\alpha \to \alpha)$$

   Wieviele Funktionen dieses Typs gibt es? Lassen sich die Funktionen charakterisieren?

Aufgabe 18. Implementiere in Scheme die Vereinigung, den Durchschnitt und die Differenz von Mengen, wobei eine Menge als geordnete Liste repräsentiert wird.

Aufgabe 19. Zuweisungen lassen sich zu Mehrfachzuweisungen verallgemeinern:

$$(e_1, \ldots, e_n) := (e'_1, \ldots, e'_n)$$

mit $n \geqslant 0$. Mit Hilfe einer Mehrfachzuweisung lässt sich zum Beispiel der Austausch von Variableninhalten einfach programmieren:

$$(x, y) := (!y, !x)$$

Ohne Mehrfachzuweisungen muss eine Hilfsvariable eingeführt werden. Formalisiere die statische Semantik von Mehrfachzuweisungen mit Hilfe von Typregeln und die dynamische Semantik mit Hilfe von Auswertungsregeln.