Übungsblatt 7

Abgabe der Lösungen: Do, 27.06., 12:00 im Briefkasten am blauen Hochhaus

PRÄSENZAUFGABEN

Übung 1 Typprüfung einfach getypter Terme

Wir lesen $\Gamma \vdash s : \alpha$ wie folgt: "Dem λ-Term s lässt sich unter der Annahme, dass eventuellen freien Variablen von s die durch Γ gegebenen Typen zugeordnet sind, der Typ α zuweisen". Zeigen Sie, dass die folgenden Aussagen zutreffen, indem Sie jeweils eine korrekte Typinferenz angeben.

1. $x : \mathbf{int}, add : \mathbf{int} \to \mathbf{int} \to \mathbf{int} \vdash \lambda y. add\ x\ (add\ x\ y) : \mathbf{int} \to \mathbf{int}$.
2. $\vdash \lambda x\ y\ .\ x : \alpha \to \beta \to \alpha$, für alle Typen α und β.

Übung 2 Inferenz von Prinzipaltypen

Es sei t ein λ-Term und Γ ein Kontext; wir sagen, dass α der *Prinzipaltyp* (engl. *principal type*) von t ist, wenn i) $\Gamma \vdash t : \alpha$ und ii) α allgemeiner ist als jeder Typ β mit $\Gamma \vdash t : \beta$, d.h. wenn jedes solche β sich durch Substitution von Typvariablen aus α erzeugen lässt. Beispielsweise ist $a \to b \to a$ (für Typvariablen a und b) der Prinzipaltyp von $\lambda x\ y\ .\ x$. Leiten Sie den Prinzipaltyp der folgenden λ-Terme in dem jeweils gegebenen Kontext her:

1. $\Gamma = \emptyset$, $t = \lambda x\ y\ z\ .\ x\ (y\ z)$.
2. $\Gamma = \{add : \mathbf{int} \to \mathbf{int} \to \mathbf{int}, length : \mathbf{string} \to \mathbf{int}\}$, $t = \lambda x\ .\ add\ (length\ x)$

Übung 3 δ-Reduktion falsch gemacht

Der Präprozessor von C erlaubt das Definieren von *Makros* mit Variablen mittels der folgenden Syntax:

#define SOME_MACRO(X1,...,Xn) BODY

Die Expansion eines solchen Makros entspricht einer Anwendung der folgendenen simplen Reduktionsregel:

$$\mathit{SOME\_MACRO(X1,...,Xn)} \to_0 \mathit{BODY}$$

wobei *BODY* als Term ohne Variablenbindung, also als Term im Sinne des Kapitels über Termersetzungssysteme angesehen wird, so dass Substitution in *BODY* einfach Variablen durch Terme ersetzt. Das heißt, dass nun – im Gegensatz zu den δ-Regeln aus Blatt 5, Übung 2.2 – keine komplexe Umbenennung gebundener Variablen stattfindet. Betrachten wir also das folgende Makro, das die Werte von zwei **int**-Variablen vertauscht:

#define SWAP_INT(x,y) **do**{**int** aux = x; x = y; y = aux;} **while**(0)

Zeigen Sie anhand eines Beispiels, dass *SWAP_INT* aufgrund der naiven Substitution in der Expansion von C-Makros auf subtile Weise nicht korrekt funktionsfähig ist.

HAUSAUFGABEN

Übung 4 Typprüfung einfach getypter Terme II (4 Punkte)

Wir interpretieren $\Gamma \vdash s : \alpha$ wie in Übung 1. Zeigen Sie, dass die folgenden Aussagen zutreffen, indem Sie jeweils eine korrekte Typinferenz angeben.

1. $\mathit{length} : \mathbf{string} \to \mathbf{int}, \mathit{name} : \mathbf{person} \to \mathbf{string} \vdash \lambda x.\mathit{length}\ (\mathit{name}\ x) : \mathbf{person} \to \mathbf{int}$
2. $\vdash \lambda f\ x\ y\ .\ f\ y\ x : (\mathbf{int} \to \mathbf{char} \to \mathbf{string}) \to (\mathbf{char} \to \mathbf{int} \to \mathbf{string})$
3. $\vdash \lambda f\ x\ y\ .\ f\ y\ x : (\alpha \to \beta \to \gamma) \to (\beta \to \alpha \to \gamma)$, für alle Typen α, β und γ.

Übung 5 Inferenz von Prinzipaltypen II (6 Punkte)

Verwenden Sie den Algorithmus nach Hindley/Milner (konkret: Berechnen Sie entsprechende Menge PT und lösen Sie das Unifikationsproblem), um den Prinzipaltyp der folgenden λ-Terme in dem jeweils gegebenen Kontext herzuleiten:

1. $\Gamma = \{\mathit{succ} : \mathbf{int} \to \mathbf{int}\}$, $t = \lceil 2 \rceil \mathit{succ}$.
2. $\Gamma = \{s : \mathbf{string}\}$, $t = \lambda x\, y.\ y\, (\lambda z.\ xsz)$

Geben Sie Ihre Zwischenschritte bei der Berechnung der Mengen $PT(\Gamma; t; \alpha)$ sowie die ermittelten mgu an!

Übung 6 Type Inhabitation und untypisierbare Terme (8 Punkte)

1. Das zur Typinferenz symmetrische Problem ist das Problem der *type inhabitation*, d.h. das Problem, einen λ-Term eines gegebenen Typs zu finden, falls ein solcher Term existiert. Im Folgenden bezeichnen p, q und r Typvariablen.

   Finden Sie für jeden der folgenden Typen α einen λ-Term s, so dass $\vdash s : \alpha$.

   (a) $p \to p$
   (b) $p \to (q \to p)$
   (c) $(p \to (q \to r)) \to (p \to q) \to p \to r$
   (d) $((p \to q) \to r) \to q \to r$

   Bemerkung: Prüfen Sie, ob die Typen der vorherigen Teilaufgabe – als *aussagenlogische Formeln* interpretiert – aussagenlogische Tautologien sind. In der Vorlesung erfahren Sie den genauen Zusammenhang.

2. Man kann jedoch auch Typen formulieren, für die es keinen passenden Term im einfach getypten λ-Kalkül gibt. Zeigen Sie, dass es keinen Term t geben kann, sodass

$$\vdash t : (p \to p) \to q \to p.$$

3. Im Gegenzug dazu gibt es auch Terme, denen man keinen Typ zuordnen kann. Verwenden Sie das Inversionslemma aus der Vorlesung, um zu zeigen, dass:
   (a) $\nvdash \lambda x.xx : \alpha$, für jeden Typ α.
   (b) $y : \mathbf{char} \nvdash \lambda x.yx : \alpha$, für jeden Typ α.