Friedrich-Alexander-Universität
Technische Fakultät

Theorie der Programmierung

Übung 07 - Der einfach getypte λ-Kalkül

Leon Vatthauer

12. Juni 2023

Der einfach getypte λ-Kalkül ($\lambda \rightarrow$)

Definitionen

Typen

Sei **V** eine Menge von *Typvariablen* a, b, etc. und **B** eine Menge von Basistypen, etwa **Bool**, **Int**. Die Grammatik für Typen $\alpha, \beta, \ldots$ ist dann

$$\alpha, \beta ::= a \mid \mathbf{b} \mid \alpha \rightarrow \beta \qquad (\mathbf{b} \in \mathbf{B}, a \in \mathbf{V})$$

Notation

Im Gegensatz zur Applikation von λ-Termen, welche links-assoziativ ist (also *add* 5 3 = (*add* 5) 3), ist der Funktionspfeil rechts-assoziativ, also $\alpha \rightarrow \beta \rightarrow \gamma = \alpha \rightarrow (\beta \rightarrow \gamma)$

Kontext

Ein *(Typ-)Kontext* ist eine Menge

$$\Gamma = \{x_1 : \alpha_1; \ldots; x_n : \alpha_n\}$$

Der einfach getypte λ-Kalkül ($\lambda \to$)

Definitionen

Typisierung

Wir lesen $\Gamma \vdash t : \alpha$ als „im Kontext Γ hat der Term t den Typ α“ und definieren diese Relation wie folgt:

$$(\mathrm{Ax})\ \frac{}{\Gamma \vdash x : \alpha}\ (x : \alpha \in \Gamma) \qquad (\to_i)\ \frac{\Gamma[x \mapsto \alpha] \vdash t : \beta}{\Gamma \vdash \lambda x.t : \alpha \to \beta}$$

$$(\to_e)\ \frac{\Gamma \vdash t : \alpha \to \beta \qquad \Gamma \vdash s : \alpha}{\Gamma \vdash t\ s : \beta}$$

Inversionslemma

1. Wenn $\Gamma \vdash x : \alpha$, dann $x : \alpha \in \Gamma$
2. Wenn $\Gamma \vdash t\ s : \beta$, dann existiert α mit $\Gamma \vdash t : \alpha \to \beta$ und $\Gamma \vdash s : \alpha$
3. Wenn $\Gamma \vdash \lambda x.t : \gamma$, dann hat γ die Form $\gamma = \alpha \to \beta$ und $\Gamma[x \mapsto \alpha] \vdash t : \beta$

Aufgabe 1

Typprüfung einfach getypter Terme

Zeigen Sie, dass die folgenden Aussagen zutreffen, indem Sie jeweils eine korrekte Typherleitung angeben.

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

Typisierung

Wir lesen $\Gamma \vdash t : \alpha$ als „im Kontext Γ hat der Term t den Typ α“ und definieren diese Relation wie folgt:

$$(\mathrm{Ax})\ \frac{}{\Gamma \vdash x : \alpha}\ (x : \alpha \in \Gamma) \qquad (\to_i)\ \frac{\Gamma[x \mapsto \alpha] \vdash t : \beta}{\Gamma \vdash \lambda x.t : \alpha \to \beta}$$

$$(\to_e)\ \frac{\Gamma \vdash t : \alpha \to \beta \qquad \Gamma \vdash s : \alpha}{\Gamma \vdash t\ s : \beta}$$

Algorithmus W nach Hindley/Milner

Prinzipaltyp

Es sei t ein λ-Term und Γ ein Kontext; wir sagen, dass α der *Prinzipaltyp* (engl. *principal type*) von t ist, wenn

1. $\Gamma \vdash t : \alpha$
2. 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 $\lambda xy.x$.

Algorithmus W nach Hindley/Milner

Wir definieren rekursiv eine Menge $PT(\Gamma; t; \alpha)$ von Typgleichungen, so dass der mgu $\sigma = \mathbf{mgu}(PT(\Gamma; t; \alpha))$ die allgemeinste Lösung von $\Gamma \vdash t : \alpha$ liefert (wenn eine Lösung existiert).

1. $PT(\Gamma;\ x;\ \alpha) = \{\alpha \doteq \beta \mid x : \beta \in \Gamma\}$
2. $PT(\Gamma;\ \lambda x.t;\ \alpha) = PT((\Gamma[x \mapsto a]);\ t;\ b) \cup \{a \to b \doteq \alpha\}$, mit a, b frisch
3. $PT(\Gamma;\ t\ s;\ \alpha) = PT(\Gamma;\ t;\ a \to \alpha) \cup PT(\Gamma;\ s;\ a)$, mit a frisch

Aufgabe 2

Inferenz von Prinzipaltypen

Leiten Sie den Prinzipaltyp der folgenden λ-Terme in dem jeweils gegebenen Kontext her

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

Algorithmus W nach Hindley/Milner

Wir definieren rekursiv eine Menge $PT(\Gamma; t; \alpha)$ von Typgleichungen, so dass der mgu $\sigma = \mathbf{mgu}(PT(\Gamma; t; \alpha))$ die allgemeinste Lösung von $\Gamma \vdash t : \alpha$ liefert (wenn eine Lösung existiert).

1. $PT(\Gamma;\ x;\ \alpha) = \{\alpha \doteq \beta \mid x : \beta \in \Gamma\}$
2. $PT(\Gamma;\ \lambda x.t;\ \alpha) = PT((\Gamma[x \mapsto a]);\ t;\ b) \cup \{a \to b \doteq \alpha\}$, mit a, b frisch
3. $PT(\Gamma;\ t\ s;\ \alpha) = PT(\Gamma;\ t;\ a \to \alpha) \cup PT(\Gamma;\ s;\ a)$, mit a frisch

Aufgabe 3.1

Type Inhabitation und untypisierbare Terme

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, sodass $\vdash s : \alpha$.

(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) $((p \to q) \to r) \to q \to r$

Aufgabe 3.2

Type Inhabitation und untypisierbare Terme

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) $\not\vdash \lambda x.x\ x : \alpha$, für jeden Typ α.

(b) $y : \mathsf{char} \not\vdash \lambda x.y\ x : \alpha$, für jeden Typ α.

Inversionslemma

1. Wenn $\Gamma \vdash x : \alpha$, dann $x : \alpha \in \Gamma$
2. Wenn $\Gamma \vdash t\ s : \beta$, dann existiert α mit $\Gamma \vdash t : \alpha \to \beta$ und $\Gamma \vdash s : \alpha$
3. Wenn $\Gamma \vdash \lambda x.t : \gamma$, dann hat γ die Form $\gamma = \alpha \to \beta$ und $\Gamma[x \mapsto \alpha] \vdash t : \beta$