Friedrich-Alexander-Universität
Technische Fakultät

Theorie der Programmierung

Übung 09 - der (ungetypte) λ-Kalkül

Leon Vatthauer

4. Juli 2025

Der (ungetypte) λ-Kalkül

Terme und Konventionen

λ-Terme

$$t ::= x \mid t_1 t_2 \mid \lambda x.t \qquad (x \in V)$$

Der (ungetypte) λ-Kalkül

Terme und Konventionen

λ-Terme

$$t ::= x \mid t_1 t_2 \mid \lambda x.t \qquad (x \in V)$$

Konventionen

- Applikation ist *links-assoziativ*: $x\ y\ z = (x\ y)\ z$
- Abstraktion reicht so weit wie möglich (Vgl. Quantoren in GLoIn)
- Aufeinanderfolgende Abstraktionen werden zusammengefasst: $\lambda x.\lambda y.\lambda z.y\ x = \lambda x\ y\ z.y\ x$

Der (ungetypte) λ-Kalkül

Freie Variablen und Substitution

Freie Variablen

Sei t ein λ-Term, $FV(t)$ ist dann definiert durch:

$$\begin{aligned} FV(x) \quad &= \{x\} \quad (\text{für } x \in V) \\ FV(t\ s) \quad &= FV(t) \cup FV(s) \\ FV(\lambda x.t) &= FV(t) \setminus \{x\} \end{aligned}$$

Der (ungetypte) λ-Kalkül

Freie Variablen und Substitution

Freie Variablen

Sei t ein λ-Term, $FV(t)$ ist dann definiert durch:

$$\begin{aligned} FV(x) &= \{x\} \quad (\text{für } x \in V) \\ FV(t\ s) &= FV(t) \cup FV(s) \\ FV(\lambda x.t) &= FV(t) \setminus \{x\} \end{aligned}$$

Substitution

Eine Substitution ist eine Abbildung $\sigma : V_0 \to T(V)$, wobei $V_0 \subseteq V$ (*endliche* Teilmenge).
Die Anwendung einer Substitution auf λ-Terme ist ebenfalls rekursiv definiert:

$$\begin{aligned} x\sigma &= \sigma(x) \\ (t\ s)\sigma &= (t\sigma)\ (s\sigma) \\ (\lambda x.t)\sigma &= \lambda y.(t\sigma') \end{aligned}$$

mit y frisch, also $y \notin FV(\sigma(z))$ für alle $z \in FV(\lambda x.t)$ und $\sigma' = \sigma[x \mapsto y]$

Der (ungetypte) λ-Kalkül

α-Äquivalenz und β-Reduktion

α-Äquivalenz

Zwei Terme t_1, t_2 heißen α-Äquivalent, wenn sie durch Umbenennung gebundener Variablen auseinander hervorgehen. Formal:

$$\lambda x.t =_\alpha \lambda y.t[y/x] \quad \text{wenn } y \notin FV(t)$$

Der (ungetypte) λ-Kalkül

α-Äquivalenz und β-Reduktion

α-Äquivalenz

Zwei Terme t_1, t_2 heißen α-Äquivalent, wenn sie durch Umbenennung gebundener Variablen auseinander hervorgehen. Formal:

$$\lambda x.t =_\alpha \lambda y.t[y/x] \quad \text{wenn } y \notin FV(t)$$

β-Reduktion

Die β-Reduktion modelliert das Ausrechnen einer Funktionsanwendung, z.B: $(\lambda x.3 + x)\ 5 \to_\beta 3 + 5$
Die Einschrittreduktion $\to_\beta$ ist:

$$C((\lambda x.t)\ s) \to_\beta C(t[s/x])$$

Aufgabe 1.1

α-Äquivalenz und β-Reduktion

Entscheiden Sie für jedes der folgenden Paare von λ-Termen, ob die Terme jeweils α-Äquivalent zueinander sind:

(a) $\lambda x\ y.x\ y =_\alpha^? \lambda u\ v.u\ v$

(b) $\lambda x\ y.x\ y =_\alpha^? \lambda u\ v.v\ u$

(c) $(\lambda x.x\ x)(\lambda y.y\ y) =_\alpha^? (\lambda x.x\ x)(\lambda x.x\ x)$

α-Äquivalenz

Zwei Terme t_1, t_2 heißen α-Äquivalent, wenn sie durch Umbenennung gebundener Variablen auseinander hervorgehen. Formal:

$$\lambda x.t =_\alpha \lambda y.t[y/x] \quad \text{wenn } y \notin FV(t)$$

Aufgabe 1.2

α-Äquivalenz und β-Reduktion

Entscheiden Sie in jedem der folgenden Fälle, ob der jeweilige Reduktionsschritt eine zulässige β-Reduktion darstellt:

(a) $(\lambda x\ y\ z.x\ y\ z)(\lambda y.y\ y) \rightarrow_\beta^? \lambda y\ z.(\lambda y.y\ y)\ y\ z$

(b) $(\lambda x\ y\ z.x\ y\ z)(y\ y) \rightarrow_\beta^? \lambda y\ z.(y\ y)\ y\ z$

(c) $(\lambda x\ y\ z.x\ y\ z)(y\ y) \rightarrow_\beta^? \lambda y\ z.(u\ u)\ y\ z$

(d) $(\lambda x\ y\ z.x\ y\ ((\lambda u.u\ x)(y\ y)))\ u\ v \rightarrow_\beta^? (\lambda x\ y\ z.x\ y\ ((y\ y)\ x))\ u\ v$

β-Reduktion

Die β-Reduktion modelliert das Ausrechnen einer Funktionsanwendung, z.B: $(\lambda x.3 + x)\ 5 \rightarrow_\beta 3 + 5$

Die Einschrittreduktion $\rightarrow_\beta$ ist:

$$C((\lambda x.t)\ s) \rightarrow_\beta C(t[s/x])$$

Aufgabe 2

β-Normalformen

Wir definieren die folgenden drei λ-Terme:

$$flip = \lambda f\ x\ y.f\ y\ x \qquad const = \lambda x\ y.x \qquad twice = \lambda f\ x.f\ (f\ x)$$

1. Beschreiben Sie die Funktionsweise dieser drei Funktionen in eigenen Worten.

Aufgabe 2

β-Normalformen

Wir definieren die folgenden drei λ-Terme:

$$flip = \lambda f\ x\ y.f\ y\ x \qquad const = \lambda x\ y.x \qquad twice = \lambda f\ x.f\ (f\ x)$$

1. Beschreiben Sie die Funktionsweise dieser drei Funktionen in eigenen Worten.
2. Ermitteln Sie die β-Normalformen der folgenden Terme:
   (a) $flip\ const\ twice$
   (b) $twice\ flip$

Aufgabe 2

β-Normalformen

Wir definieren die folgenden drei λ-Terme:

$$flip = \lambda f\ x\ y.f\ y\ x \qquad const = \lambda x\ y.x \qquad twice = \lambda f\ x.f\ (f\ x)$$

1. Beschreiben Sie die Funktionsweise dieser drei Funktionen in eigenen Worten.
2. Ermitteln Sie die β-Normalformen der folgenden Terme:
   (a) $flip\ const\ twice$
   (b) $twice\ flip$

β-Reduktion

Die β-Reduktion modelliert das Ausrechnen einer Funktionsanwendung, z.B: $(\lambda x.3 + x)\ 5 \rightarrow_\beta 3 + 5$
Die Einschrittreduktion $\rightarrow_\beta$ ist:

$$C((\lambda x.t)\ s) \rightarrow_\beta C(t[s/x])$$