Informatik I, Blatt Nr. 7, Abgabe: 4.2.2010 um 11 Uhr
http://proglang.informatik.uni-freiburg.de/teaching/info1/2009/

Hinweis zum λ-Kalkül

In diesem Aufgabenblatt sind machen Klammern durch eckige Klammern ersetzt. Dies dient nur der Übersichtlichkeit. Es wird anstelle $(\lambda(x)e)$ aus der Vorlesung $[\lambda(x)e]$ geschrieben. Dann lässt sich die Applikation $(e_1\ e_2)$ leichter von dem Lambda-Term unterscheiden und die Klammern sind leichter zuzuordnen.

Im folgenden sind $w, x, y, z \in V$ Variablen.

1 Aufgabe

[3+5 Punkte]

Sei die Menge der freien Variablen definiert durch

$$\mathrm{free}(e) := \begin{cases} \{v\} & \text{falls } e = v \\ \mathrm{free}(e_1) \cup \mathrm{free}(e_2) & \text{falls } e = (e_1\ e_2) \\ \mathrm{free}(e_1) \backslash \{x\} & \text{falls } e = [\lambda(x)e_1] \end{cases}$$

und die Menge der gebundenen Variablen durch

$$\mathrm{bound}(e) := \begin{cases} \emptyset & \text{falls } e = v \\ \mathrm{bound}(e_1) \cup \mathrm{bound}(e_2) & \text{falls } e = (e_1\ e_2) \\ \mathrm{bound}(e_1) \cup \{x\} & \text{falls } e = [\lambda(x)e_1] \end{cases}$$

a. Seien w, x, y, z Variablen. Geben Sie für die folgenden Terme die Menge der freien und gebundenen Variablen an:

$$[\lambda(x)[\lambda(y)[\lambda(y)(y\ w)]]]$$
$$((w\ [\lambda(x)w])\ ([\lambda(y)x]\ [\lambda(x)x]))$$
$$([\lambda(w)x]\ [\lambda(x)x])$$

b. Geben Sie eine induktive Definition der Menge $\mathrm{var}(e)$ der in einem Lambda-Term e vorkommenden Variablen an, ohne dazu free oder bound zu benutzen. Beweisen Sie nun durch Induktion über den Aufbau der λ-Terme oder widerlegen Sie durch Gegenbeispiel:

$$\mathrm{free}(e) \cup \mathrm{bound}(e) = \mathrm{var}(e)$$
$$\mathrm{free}(e) \cap \mathrm{bound}(e) = \emptyset$$

Abgabe: Papier.

2 Aufgabe

[2 Punkte]

Sei

$$e = [\lambda(x)(z\ [\lambda(z)((w\ x)\ z)])]$$

Berechnen Sie folgende Substitutionen und geben Sie alle Zwischenschritte an:

a. $e\,[z \mapsto [\lambda(x)y]]$

b. *(gestrichen)*

Abgabe: Papier.

3 Aufgabe

[2 Punkte]

Der Lambda-Ausdruck

$$[\lambda(w)(z\ ([\lambda(x)x]\ w))]$$

kann mit einem β-Reduktionsschritt zu seiner β-Normalform reduziert werden. Leiten Sie nur mit den Kompatibilitätsregeln (wie CLAM, CAPPL etc.) und der grundlegenden β-Reduktionsregel

$$([\lambda(x)e]\ f) \rightarrow_\beta e\,[x \mapsto f]$$

diesen Reduktionsschritt her. *Abgabe:* Papier.

4 Aufgabe

[4 Punkte]

Die $\rightarrow_\beta$-Relation ist eine binäre Relation auf E_λ und kann daher als Graph dargestellt werden. Zeichnen Sie für jeden der folgenden Lambda-Ausdrücke den Graphen der von dem Ausdruck ausgehenden möglichen β-Reduktionen bis hin zur β-Normalform.

$$([\lambda(x)[\lambda(y)x]]\ ([\lambda(w)w]\ z))$$
$$([\lambda(z)z]\ ([\lambda(x)(x\ x)]\ [\lambda(x)(x\ x)]))$$

Abgabe: Papier.

5 Aufgabe

[4 Punkte]

Reduzieren Sie

$$(((\text{If True})\ (\text{Succ}\ \lfloor 1 \rfloor))\ (\text{Pred}\ \lfloor 4 \rfloor))$$

mit den in der Vorlesung vorgestellten Definitionen für Pred, If, True und Church-Numerale zu einer β-Normalform. Heben Sie in jedem β-Reduktionsschritt den Redex hervor.

Lassen Sie dabei den syntaktischen Zucker wie Pred und $\lfloor n \rfloor$ so lange wie möglich intakt, damit Ihr Tutor (dem spätestens jetzt die Augen schmerzen) es leichter hat.
Abgabe: Papier.

Changelog

2010-01-22: 2b gestrichen, in 5 wurde Pred$\lfloor 1 \rfloor$ zu Succ$\lfloor 1 \rfloor$ geändert.