

## Hinweise zur Abgabe

Bitte reichen Sie Ihre Abgaben bis zum 29.1.2009 um 11 Uhr ein. Abgaben in elektronischer Form schicken Sie **per Email** an **Ihren** Tutor. Abgaben in Papierform werfen Sie bitte in den **Briefkasten** Ihrer Übungsgruppe im Geb. 051 im Erdgeschoss. Bei jeder Aufgabe ist angegeben, ob Sie elektronisch oder auf Papier abgegeben werden muss.

Bei allen Aufgaben, die Sie per Mail abgeben, müssen Sie sich an die Namenskonventionen der Aufgaben halten. Dies gilt sowohl für die Dateinamen der Abgabe, als auch für Namen von Funktionen. Bitte geben Sie bei der elektronischen Abgabe nur eine Zip-Datei ab. Diese muss alle in den Aufgaben angegebenen `.scm` Dateien (DrScheme) enthalten. Alle Dateien müssen sich in der Zip-Datei in einem Ordner befinden. Der Name dieses Ordners muss Ihrem Loginnamen für den Rechnerpool des Instituts für Informatik entsprechen. Geben Sie unter keinen Umständen Worddokumente usw. ab!

Achten Sie bei der Papierabgabe darauf, dass jedes Blatt Papier Ihrer Abgabe Ihren Namen, Ihre Übungsgruppe, die Blattnummer und den Namen Ihres Tutors trägt. Falls Ihre Papierabgabe aus mehreren Seiten besteht, tackern Sie die Blätter.

Sie können DrScheme im Pool verwenden (starten mit `drscheme`). Achten Sie darauf, dass Sie jeweils das richtige Sprachlevel ausgewählt haben!

## Hinweis zum $\lambda$ -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.

## 1 Aufgabe

*[3+2+5 Punkte]*

Sei die Menge der freien Variablen definiert durch

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

und die Menge der gebundenen Variablen durch

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

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

$$\begin{aligned} & [\lambda(x)(x [\lambda(y)z])] \\ & ([\lambda(z)[\lambda(y)z]] z) \\ & (([\lambda(x)(x x)] [\lambda(c)(z z)]) c) \end{aligned}$$

- b. Geben Sie eine Definition  $\text{var}(e)$  für alle Variablen eines Lambda Terms an, ohne  $\text{free}$  und  $\text{bound}$  zu verwenden.

c. Beweisen Sie durch Induktion über den Aufbau der  $\lambda$ -Terme oder widerlegen Sie:

$$\begin{aligned}\text{free}(e) \cup \text{bound}(e) &= \text{var}(e) \\ \text{free}(e) \cap \text{bound}(e) &= \emptyset\end{aligned}$$

d. **Bonus:** (2 Punkte) Substituieren Sie bei allen Termen der ersten Teilaufgabe die Variable  $z$  durch  $([\lambda(v)v]e)$ . Arbeiten Sie die Substitution mit Mitteln der Syntax oder der Semantik auf den Lambda Termen? Begründen Sie Ihre Antwort.

*Abgabe:* Papier.

## 2 Aufgabe

[6 Punkte]

Schreiben Sie alle Redexe der folgenden Lambda Terme auf und werten Sie die Terme, falls dies möglich ist, bis zur Normalform aus.

$$\begin{aligned} & (([\lambda(y)(y y)] [\lambda(c)(z z)]) c) \\ & (([\lambda(z)[\lambda(x)z]] z) c) \\ & (([\lambda(x)[\lambda(z)z]] z) c) \\ & ([\lambda(y)(y [\lambda(y)z])] ([\lambda(a)a] [\lambda(b)z])) \end{aligned}$$

*Abgabe:* Papier.

## 3 Aufgabe

[4 Punkte]

Reduzieren Sie

$$(\text{Pred } [2])$$

mit den in der Vorlesung vorgestellten Definitionen für Pred und Church-Numeralen zu einer  $\beta$ -Normalform.

*Abgabe:* Papier.