Fachbereich 6 · Mathematik
Theoretische Informatik

Kurt Sieber

Theorie der Programmierung (WS 2003/04)

Übungsblatt 1

Da die theoretischen Grundlagen für die folgenden Aufgaben noch nicht vollständig in der Vorlesung vorgestellt wurden, sollten Sie bei der Lösung zunächst auf Ihre Intuition für Typen und funktionale Programmierung zurückgreifen. Wenn eine solche Intuition auch nicht vorhanden ist, versuchen Sie sie mit Hilfe der Aufgaben zu entwickeln.

Aufgabe 1
Welche der folgenden λ-Abstraktionen sind wohlgetypt? Welche Funktionen berechnen sie?

a. $\lambda x : \mathbf{bool}.\, \mathbf{if}\ x\ \mathbf{then}\ \mathit{true}\ \mathbf{else}\ \mathit{false}$

b. $\lambda x : \mathbf{bool}.\, \mathbf{if}\ x\ \mathbf{then}\ \mathit{false}\ \mathbf{else}\ \mathit{true}$

c. $\lambda x : \mathbf{bool}.\, \lambda y : \mathbf{bool}.\, \mathbf{if}\ x\ \mathbf{then}\ x\ \mathbf{else}\ y$

d. $\lambda x : \mathbf{bool}.\, \lambda y : \mathbf{bool}.\, \mathbf{if}\ x\ \mathbf{then}\ y\ \mathbf{else}\ x$

e. $\lambda x : \mathbf{int}.\, \lambda y : \mathbf{int}.\, = (\mathbf{mod}(x, y), 0)$

f. $\lambda f : \mathbf{int} \to \mathbf{int}.\, \lambda g : \mathbf{int} \to \mathbf{int}.\, \lambda x : \mathbf{int}.\, f\,(g\,x)$

Aufgabe 2
Die Funktionen *square* und *twice* seien (wie in der Vorlesung) deklariert durch

$$\begin{array}{l} \mathbf{val}\ \mathit{square} = \lambda x : \mathbf{int}.\, * (x, x) \\ \mathbf{val}\ \mathit{twice} = \lambda f : \mathbf{int} \to \mathbf{int}.\, \lambda x : \mathbf{int}.\, f(f x) \end{array}$$

Welche der folgenden Applikationen sind dann wohlgetypt? Welche Ergebnisse liefern sie?

a. $\mathit{square}\,(\mathit{square}\,3)$

b. $(\mathit{square}\ \mathit{square})\,3$

c. $\mathit{square}\ \mathit{square}$

d. $\mathit{twice}\,(\mathit{square}\,3)$

e. $(\mathit{twice}\ \mathit{square})\,3$

f. $\mathit{twice}\ \mathit{square}$

g. $\mathit{twice}\,(\mathit{twice}\ \mathit{square})$

h. $(\mathit{twice}\ \mathit{twice})\ \mathit{square}$

i. $\mathit{twice}\ \mathit{twice}$

Aufgabe 3
Jede rationale Zahl a/b lässt sich als Paar (a, b) ganzer Zahlen darstellen. Definieren Sie auf dieser Darstellung die folgenden Funktionen.

a. Addition

b. Multiplikation

c. Test auf Gleichheit

d. Test auf "<"

Ist es sinnvoll, *alle* Paare $(a, b) \in \mathit{Int} \times \mathit{Int}$ als Darstellungen zu verwenden? Stört es, dass die Darstellung nicht eindeutig ist?

Aufgabe 4
Eine Menge $M \subseteq \mathit{Int}$ kann man auch als Funktion von *Int* nach *Bool* auffassen, nämlich als die sogenannte charakteristische Funktion

$$c_M : \mathit{Int} \to \mathit{Bool}$$
$$c_M(n) = \begin{cases} \mathit{true} & \text{falls } n \in M \\ \mathit{false} & \text{falls } n \notin M \end{cases}$$

In diesem Sinne lässt sich z.B. die Menge der negativen ganzen Zahlen darstellen durch $\lambda x : \mathbf{int}. < (x, 0)$ oder die Menge der geraden Zahlen durch $\lambda x : \mathbf{int}. = (\mathbf{mod}\,(x, 2), 0)$. Auf dieser Mengendarstellung lassen sich nun die typischen Mengenoperationen implementieren. Definieren Sie im einzelnen:

a. eine Funktion *member*, die testet, ob eine ganze Zahl in einer Menge liegt;

b. eine Funktion *complement*, die das Komplement einer Menge berechnet;

c. Funktionen *union* und *intersection*, die die Vereinigung und den Durchschnitt zweier Mengen berechnen;

d. eine Funktion *preimage*, die das Urbild $f^{-1}(M)$ einer Menge M unter einer Funktion f berechnet.

Überlegen Sie sich stets als erstes, welchen Typ Ihre Funktion haben sollte. Sehen Sie einen Zusammenhang zur Vorlesung GTI? Kann man auch das Bild $f(M)$ einer Menge M unter einer Funktion f berechnen?

Dienstags findet die Vorlesung jetzt in **H-C 3310** statt.

Übungsblätter finden sich auf

`http://www.informatik.uni-siegen.de/~sieber/public/2003_WS_TP/Main.html`