Theorie der Programmierung

Braindump SS 2016

Aufgabe 1

Wir definieren ein Termersetzungssystem über der aus zwei binären Funktionssymbolen / und $\circ$ (in Infixnotation geschrieben) bestehenden Signatur Σ durch:

$$
\begin{aligned}
(a/b) \circ (a/c) &\to_0 b \circ (c/a) \\
(a \circ a)/b &\to_0 b \circ b \\
a/(b/c) &\to_0 a/(b \circ c)
\end{aligned}
$$

1. Zeigen Sie mittels Polynomordnungen, dass das System stark normalisierend ist.
2. Ist das System konfluent? Geben Sie einen Beweis bzw. ein Gegenbeispiel an.

Aufgabe 2

Wir erinnern an einige im λ-Kalkül definierte Funktionen:

$$
\begin{aligned}
\mathit{len} &= \lambda\, \ell.\ \ell \mathit{zero}\ (\lambda\, x.\ \mathit{succ}) \\
\mathit{cons} &= \lambda\, x\ \ell.\ \lambda\, u\ f.\ f x (\ell u f) \\
\mathit{nil} &= \lambda u\ f.\ u
\end{aligned}
$$

1. Geben Sie die ersten vier $\beta\delta$-Reduktionsschritte des Terms $\mathit{len}(\mathit{cons}\ \mathit{two}\ \mathit{nil})$ unter a) normaler und b) applikativer Reduktion an. Markieren Sie durch Unterstreichen in jedem Schritt den zu reduzierenden Redex.
2. Gegeben sei

   $$s = \text{ let } (f = \lambda; t.\ t\ 2) \text{ in } (f\ \mathit{add})(\mathit{len}\ (f\ \mathit{cons}\ \mathit{nil}))$$

   und der Kontext

   $$\Gamma = \{2 : \mathbb{N},\ \mathit{add} : \mathbb{N} \to \mathbb{N} \to \mathbb{N},\ \mathit{len} : \forall a.\ \mathbb{L}\ a \to \mathbb{N},\ \mathit{nil} : \forall a.\ \mathbb{L}\ a,\ \mathit{cons} : \forall a.\ a \to \mathbb{L}\ a \to \mathbb{L}\ a\},$$

   wobei $\mathbb{N}$ der Typ der natürlichen Zahlen und $\mathbb{L}\ a$ der Typ der Listen mit Element vom Typ a ist. Finden Sie den Prinzipialtyp $\mathrm{PT}(\Gamma; s; \alpha)$.

Aufgabe 3

Wir betrachten den Datentyp der Listen $List\ a$:

$$\begin{aligned}&\text{data } List\ a \text{ where}\\ &\quad nil:\ () \ \to\ List\ a\\ &\quad cons:\ a\ \to\ List\ a \to\ List\ a.\end{aligned}$$

Zusätzlich seien die folgenden Funktionen auf Listen definiert:

$$\begin{aligned}&concat:\ List\ a \to\ List\ a \to\ List\ a \text{ where}\\ &\quad concat\ nil\ ys = ys\\ &\quad concat\ (cons\ x\ xs)\ ys = cons\ x\ (concat\ xs\ ys)\end{aligned}$$

$$\begin{aligned}&flatten:\ List\ (List\ a) \to\ List\ a \text{ where}\\ &\quad flatten\ nil = nil\\ &\quad flatten\ (cons\ xs\ xss) = concat\ xs\ (flatten\ xss)\end{aligned}$$

$$\begin{aligned}&map:\ (a \to b) \to List\ a \to\ List\ b \text{ where}\\ &\quad map\ f\ nil\ = nil\\ &\quad map\ f\ (cons\ x\ xs)\ = cons\ (f\ x)(map\ f\ xs)\end{aligned}$$

1. Zeigen Sie mittels struktureller Induktion, dass

   $$flatten(map\ (map\ f)\ xss) = map\ f\ (flatten\ xss)$$

   für alle f, xss gilt.

2. Eine Liste natürlicher Zahlen $s:\ List\ Int$ ist *steep*, wenn der jeweils nächste Int größer ist als der vorherige, d.h. die Liste der natürlichen Zahlen ist aufsteigend sortiert.
   Schreiben Sie eine Funktion $steep's$, die $True$ zurückgibt, wenn s $steep$ ist. Hierfür ist folgendes Gerüst zu verwenden:

   $$\begin{aligned}&steep' = snd.\ fold\ c\ g \text{ where}\\ &\quad c = \qquad\qquad\qquad\qquad\qquad :: (Nat, Bool)\\ &\quad g = \qquad\qquad\qquad :: Nat \to (Nat,\ Bool) \to (Nat,\ Bool)\end{aligned}$$

Aufgabe 4

Wir betrachten den Kodatentyp $IntStream$:

$$\begin{aligned}&\text{codata } IntStream \text{ where}\\ &\quad head: IntStream \to Int\\ &\quad tail: IntStream \to IntStream.\end{aligned}$$

Gegeben sind weiterhin folgende auf $IntStream$ definierte Funktionen:

$$\begin{aligned}head(alt\ a\ b) &= a\\ tail(alt\ a\ b) &= alt\ b\ a\\ head(const\ i) &= i\\ tail(const\ i) &= const\ i\\ head(s \boxplus t) &= (head\ s) + (head\ t)\\ tail(s \boxplus t) &= (tail\ s) \boxplus (tail\ t)\\ head(psum\ s) &= heads\\ tail(psum\ s) &= (const(head\ s)) \boxplus (psum(tail\ s))\end{aligned}$$

1. Definieren Sie den Begriff $Bisimulation$ über dem Typ $IntStream$.
2. Zeigen Sie mittels Koinduktion:

   $$psum\ (alt\ 1\ (-1)) = alt\ 1\ 0$$

   Hinweis: Sie dürfen ohne Beweis verwenden, dass gilt

   $$\begin{aligned}(const\ a) \boxplus (const\ b) &= const(a+b)\\ r \boxplus (s \boxplus t) &= (r \boxplus s) \boxplus t\\ (r \boxplus s) \boxplus t &= r \boxplus (s \boxplus t)\end{aligned}$$

Aufgabe 5

Gegeben sei die Sprache

$$L = \{z \bullet z^R \mid z \in \{0,1\}^*\},$$

dabei ist z^R die Spiegelung des Wortes z. Ist beispielsweise $z = 1100$, so ist $z^R = 0011$.

Ist L regulär? Wenn ja, geben Sie einen regulären Ausdruck für L an und erläutern Sie, warum dieser L definiert. Andernfalls zeigen Sie mittels des Pumping-Lemma, dass L nicht regulär ist.