Prof. Dr. Manfred Schmidt-Schauß
Künstliche Intelligenz/Softwaretechnologie

Fachbereich Informatik (20)
Johann Wolfgang Goethe-Universität Frankfurt am Main

Funktionale Programmierung I

Sommersemester 2000

Übungsblatt Nr. 9

Abgabe: Dienstag 06. Juni 2000 vor der Vorlesung

Aufgabe 1 (20 Punkte)

Gelten die folgenden Äquivalenzen für endliche bzw. unendliche Listen? Beweisen Sie Ihre Aussagen mit Hilfe des in der Vorlesung vorgestellten Prinzips der Induktion über Datentypen.

$$\begin{aligned} \texttt{append}\ (\texttt{take}\ n\ xs)\ (\texttt{drop}\ n\ xs) &\sim_c\ xs \\ \texttt{map}\ f\ (\texttt{map}\ g\ xs) &\sim_c\ \texttt{map}\ (f\ .\ g)\ xs \end{aligned}$$

Verwenden Sie dabei neben den bekannten Funktionen `map` und `append` die Definitionen

$$\begin{aligned} \texttt{take}\ 0\ xs &= [] & \texttt{drop}\ 0\ xs &= xs \\ \texttt{take}\ n\ [] &= [] & \texttt{drop}\ n\ [] &= [] \\ \texttt{take}\ n\ (x : xs) &= x : (\texttt{take}\ (n-1)\ xs) & \texttt{drop}\ n\ (x : xs) &= \texttt{drop}\ (n-1)\ xs \end{aligned}$$

Aufgabe 2 (10 Punkte)

Übersetzen Sie die folgenden verschränkt rekursiven Superkombinatordefinitionen nach KFP.

$$\begin{aligned} \texttt{f}\ n\ m &= \texttt{case}\ n\ (0\ \texttt{->}\ 0)\ (S\ n'\ \texttt{->}\ \texttt{f}\ n'\ (\texttt{g}\ n'\ m)) \\ \texttt{g}\ n\ m &= \texttt{case}\ m\ (0\ \texttt{->}\ 1)\ (S\ m'\ \texttt{->}\ \texttt{f}\ n\ m') \end{aligned}$$

Aufgabe 3 (10 Punkte)

In der Vorlesung wurden sowohl die sog. Church-Zahlen für die Darstellung der natürlichen Zahlen durch Funktionen als auch die mögliche Erweiterung auf ganze Zahlen vorgestellt. Definieren Sie in KFP+ die Rechenoperationen `PLUS`, `TIMES` und `EXPT` auf *ganzen* Churchzahlen, ohne Konstruktoren oder `case` zu benutzen.

Aufgabe 4 (10 Punkte)

Übersetzen Sie die KFP-Ausdrücke $(\lambda x.(xx))\ (\lambda x.(xx))$ sowie $\lambda f.(\lambda x.f(xx))(\lambda x.f(xx))$ in den SKI-Kombinatorkalkül und zeigen Sie, daß der I-Kombinator redundant ist.