Grundlagen der Programmierung

SS 05
Prof. Dr. K. Madlener
Lösungshinweise zu Übungsblatt 2

Aufgabe 2.1. ad 1: ρ i st kein Endomorphismus. Dies kann man aus der Identität

$$\rho(uv) = \rho(v)\rho(u)$$

für alle $u, v \in \Sigma^*$ folgern. ρ ist offensichtlich injektiv und surjektiv. Beachte

$$\rho(\rho(w)) = w$$

für alle $w \in \Sigma^*$.

ad 2: Das Kalkül K ist gegeben durch folgende Regeln für alle $a \in \Sigma$, $x \in \Sigma^*$

$$\frac{}{\varepsilon}\,, \quad \frac{}{a}\,, \quad \frac{x}{axa}$$

Bezeichne P_K die Menge aller in K ableitbaren Wörter. Es ist folgendes zu zeigen:

a) $P_K \subseteq P$
b) $P \subseteq P_K$

Zeige: $P_K \subseteq P$
(Induktion über die Länge von Ableitungen).
Sei u aus einem Axiom ableitbar.
$\rightsquigarrow u = \varepsilon$ oder $u = a$, $a \in \Sigma$.
Klar: $\varepsilon, a \in P$.
Sei nun $u \in \Sigma^*$ ein Wort, das durch eine echte Regel gewonnen wurde, wobei für die Prämisse v die Behauptung, also $v \in P$, gegolten hat.
$\rightsquigarrow u = ava$ mit $a \in \Sigma$ und es gilt

$$\begin{aligned}\rho(ava) &= \rho(va)\rho(a)\\ &= \rho(a)\rho(v)\rho(a)\\ &= a\rho(v)a\\ &= ava\end{aligned}$$

$\rightsquigarrow u$ ist Palindrom, also $u \in P$.
Insgesamt: $P_K \subseteq P$.

Zeige: $P \subseteq P_K$
(Induktion über die natürlichen Zahlen).
Zeige dazu: Für jedes $n \in \mathbb{N}$ sind alle Palindrome u mit $|u| \leq n$ in K ableitbar.

$n = 0$: Es gibt nur ein Wort u mit $|u| = 0$, nämlich $u = \varepsilon$. Dies ist offensichtlich in K ableitbar und ein Palindrom.

$n = 1$: Jeder Buchstabe $a \in \Sigma$ ist ein Palindrom und in K ableitbar.

$n \to n+1$: Sei $n \geq 1$ und u mit $|u| = n+1$ ein Palindrom. Dann lässt sich u darstellen als

$$u = ava$$

mit $a \in \Sigma$, $v \in \Sigma^*$.

Offensichtlich ist v ein Palindrom, $|v| < |u|$ und somit $|v| \leq n$, und somit die Ind. Vor. auf v anwendbar.
Somit ist v in K ableitbar, d. h.

$$\vdash_K v$$

Anwenden der Regel $\dfrac{v}{ava}$ liefert $\vdash_K ava$.
Also ist u in K ableitbar.
Insgesamt: $P \subseteq P_K$
q. e. d

Aufgabe 2.2. Lösung
ad(1):
Sei $R := \Gamma_{val1}(\emptyset)$.
Zeige: R ist Relation, aber nicht Graph einer Funktion.
Es gilt $R \subseteq A \times N$, R ist somit Relation(trivial).
Betrachte nun den Ausdruck

$$a + b * b$$

der in $AExp_1$ ableitbar ist. Es existieren u.a. folgende Ableitungen:
(1) $(a, b, a + b, a + b * b)$
(2) $(a, b, b * b, a + b * b)$
In Val_1 lassen sich diese Ableitungen erweitern zu
(1) $((a, 1), (b, 2), (a + b, 3), (a + b * b, 6))$
(2) $((a, 1), (b, 2), (b * b, 4), (a + b * b, 5))$
Hinter Auswertung (1) steckt Berechnung $(1 + 2) * 2$, hinter (2) Berechnung $1 + (2 * 2)$. In R findet sich also sowohl $(a + b * b, 6)$ als auch $(a + b * b, 5)$. Somit ist R nicht Graph einer Funktion.
Im $AExp_2$ werden nun Klammern eingefügt, die in Auswertungskalkül Val_2 berücksichtigt werden. Durch die Klammerung wird die Eindeutigkeit des Kalküls garantiert.

Zur Erinnerung: Ein Kalkül $K = (A, R)$ mit Objektmenge A und Regel-Relation $R \subseteq A^* \times A$ heißt *eindeutig*, wenn R^{-1} Graph einer Funktion $R^{-1} : A \to A^*$ ist, d.h. für alle $a \in A$ existiert höchstens ein $\overline{a} \in A^*$ mit $(\overline{a}, a) \in R$.

Dies soll nun gezeigt werden. Hierzu verwenden wir strukturelle Induktion über den Aufbau von arithmetischen Ausdrücken in $AExp_2$.
Es gilt: Ist $\alpha \in \Gamma_{AExp2}(\emptyset)$ und $k(\alpha, i)$ die Anzahl der "(" minus der Anzahl der ")" in den ersten i Buchstaben von α, so gilt

(1) $k(\alpha, i) \geq 1$ für $1 \leq i < |\alpha|$ und
(2) $k(\alpha, |\alpha|) = 0$.

(ohne Beweis.)

Weiter gilt: Ist $\gamma \in \Gamma_{AExp_2}(\emptyset)$, so hat γ genau eine eindeutige Darstellung der Form

(1) $\gamma = a$ oder
(2) $\gamma = b$ oder
(3) $\gamma = (\alpha + \beta)$ mit $\alpha, \beta \in \Gamma_{AExp_2}(\emptyset)$ oder
(4) $\gamma = (\alpha * \beta)$ mit $\alpha, \beta \in \Gamma_{AExp_2}(\emptyset)$.

Beweis: Dass γ eine solche Darstellung hat, ist klar (betrachte den letzten Ableitungsschritt). Es bleibt die Eindeutigkeit zu zeigen.
1. Fall: $\gamma = a$. Dann ist natürlich $\gamma \neq b$, $\gamma \neq (\alpha + \beta)$ und $\gamma \neq (\alpha * \beta)$.
2. Fall: $\gamma = b$ geht analog.
3. Fall: $\gamma = (\alpha + \beta)$. Natürlich ist hier $\gamma \neq a$ und $\gamma \neq b$. Eine andere Darstellung von γ muss also die Form $\gamma = (\alpha' \circ \beta')$ mit $\alpha', \beta' \in \Gamma_{Val_2}(\emptyset)$ und $\circ \in \{+, *\}$ haben.
Zunächst zeigen wir $|\alpha| = |\alpha'|$.
Angenommen $|\alpha| \neq |\alpha'|$. Sei O. B. d. A. $|\alpha| < |\alpha'|$. Dann folgt

$$\begin{aligned} k(\alpha', |\alpha|) &= k((\alpha', |\alpha| + 1) - 1 \\ &= k(\gamma, |\alpha| + 1) - 1 \\ &= k((\alpha, |\alpha| + 1) - 1 \\ &= k(\alpha, |\alpha|) = 0. \end{aligned}$$

Also gilt $|\alpha'| = |\alpha|$.. Widerspruch. Beachte, dass in der ersten bzw. der dritten Gleichung das Wort $(\alpha'$ bzw. $(\alpha$ als erstes Argument für k gemeint ist!
Da das Ergebnis beim Entfernen der ersten (letzten) n Buchstaben von γ eindeutig bestimmt ist, gilt $\alpha = \alpha'$ und $\beta = \beta'$. Beachte: $(\alpha + \beta) = (\alpha' \circ \beta')$, d.h. das Wort $(\alpha + \beta)$ stimmt mit dem Wort $(\alpha' \circ \beta')$ überein, insbesondere auf den ersten (letzten) n Buchstaben.
4. Fall: $\gamma = (\alpha * \beta)$ geht analog.

Angenommen, es gibt $\gamma \in \Gamma_{AExp_2}(\emptyset)$ und $n, m \in \mathbb{N}$ mit $(\gamma, n), (\gamma, m) \in \Gamma_{Val_2}(\emptyset)$ und $n \neq m$. Dann gibt es auch ein solches γ mit minimaler Länge, d.h. für alle $\beta \in \Gamma_{AExp_2}(\emptyset)$ mit $|\beta| < |\gamma|$ folgt aus $(\beta, m_1), (\beta, m_2) \in \Gamma_{Val_2}(\emptyset)$, dass $m_1 = m_2$ gilt.
Durch Fallunterscheidung folgt $m = n$, was ein Widerspruch zur Annahme ist:
1. Fall: $\gamma = a$. Dann muss $n = 1 = m$ sein.
2. Fall: $\gamma = b$. Dann muss $n = 2 = m$ sein.
3. Fall: $\gamma = (\alpha + \beta)$. Dann muss wegen der Eindeutigkeit der Darstellung von γ sowohl (γ, n), als auch (γ, m) durch die dritte Regel von Val_2 erzeugt worden sein, d.h. es gibt $n_1, n_2, m_1, m_2 \in \mathbb{N}$ mit $(\alpha, n_1), (\beta, n_2), (\alpha, m_1), (\beta, m_2) \in \Gamma_{Val_2}(\emptyset)$ und $m_1 + m_2 = m$ und $n_1 + n_2 = n$. Weil γ minimal gewählt wurde, gilt $n_1 = m_1$ und $n_2 = m_2$. Daraus folgt $m = m_1 + m_2 = n_1 + n_2 = n$.
4. Fall: $\gamma = (\alpha * \beta)$ geht analog.

ad (2):

Sei $g_0 = \emptyset$ und $g_{i+1}(x) = \begin{cases} 1 & x = a \\ 2 & x = b \\ g_i(\alpha) \circ_{\mathbb{N}} g_i(\beta) & x = (\alpha \circ \beta), \circ \in \{+, *\} \end{cases}$

Mit Induktion über den natürlichen Zahlen wird nun $g_i \sqsubseteq g_{i+1}$ gezeigt.
Für $i = 0$ gilt die Behauptung offensichtlich.
Sei nun $i \geq 0$ und $x \in \mathrm{dom}(g_i)$. Für $x = a$ und $x = b$ gilt offensichtlich $g_i(x) = g_{i+1}(x)$. Sei nun $x = \alpha \circ \beta$. Dann gilt

$$\begin{aligned} g_i((\alpha \circ \beta)) &= g_{i-1}(\alpha) \circ_{\mathbb{N}} g_{i-1}(\beta) && \text{(Rekursionsgleichung)} \\ &= g_i(\alpha) \circ_{\mathbb{N}} g_i(\beta) && \text{(Induktionsvoraussetzung)} \\ &= g_{i+1}((\alpha \circ \beta)) && \text{(Rekursionsgleichung)} \end{aligned}$$

Insgesamt gilt somit $\mathrm{dom}(g_i) \subseteq \mathrm{dom}(g_{i+1})$ und für alle $x \in \mathrm{dom}(g_i)$ die Gleichung $g_i(x) = g_{i+1}(x)$, d.h. $g_i \sqsubseteq g_{i+1}$.

Es gilt

$$g = \bigcup \{g_i : i \in \mathrm{Nat}\}.$$

Funktion g erfüllt die Rekursionsgleichung und ist bzgl. $\sqsubseteq$ die kleinste Lösung dieser Gleichung.

Wir zeigen, dass $g = f$ gilt.
Zunächst stellen wir fest, dass g und f totale Funktionen sind. Dies kann mit struktureller Induktion über dem Kalkül $AExp_2$ gezeigt werden. Ebenfalls mit struktureller Induktion über dem Kalkül $AExp_2$ zeigen wir nun die Gleichheit von f und g. Für $x = a$ und $x = b$ ist die Behauptung offensichtlich. Sei also $x = (\alpha \circ \beta)$. Es gilt

$$\begin{aligned} f((\alpha \circ \beta)) &= f(\alpha) \circ_{\mathbb{N}} f(\beta) && \text{(Kalkül)} \\ &= g(\alpha) \circ_{\mathbb{N}} g(\beta) && \text{(Induktionsvoraussetzung)} \\ &= g((\alpha \circ \beta)) && \text{(Definition von g)} \end{aligned}$$

Somit gilt $f = g$. Also ist f bzgl. $\sqsubseteq$ die kleinste Lösung der angegebenen Rekursionsgleichung.

Aufgabe 2.3. ad (1): Signatur $(\{nat\}, \{0 :\to nat,\ succ : nat \to nat,\ + : nat \times nat \to nat, * : nat \times nat \to nat,\ <: nat \times nat\})$.
Algebra A ist gegeben durch
$nat_A = \{0, 1\}$
$0_A = 0$
$succ_A(x) = x$
$x +_A y = \begin{cases} 1 & x = 0 \text{ und } y = 1 \\ 0 & \text{sonst} \end{cases}$
$x *_A y = x +_A y$
$<_A = \{(0, 1)\}$
ad (2):
$\{nat, index, array\}$
$const : nat \to array$
$get : array, index \to nat$

$set : array, index, nat \to array$
$1, 2, \ldots, N : index$

$1_A = 1_{\mathbb{N}}$
$\vdots$
$N_A = N_{\mathbb{N}}$
$index_A = \{1_A \ldots N_A\}$
$array = (nat_A)^{index_A}$
$const_A(i)(j) = i$
$get(a, i) = a(i)$
$set_A(a, i, n)(j) = \begin{cases} a(j) & i \neq j \\ n & \text{sonst} \end{cases}$

Beachte: $set_A(a, i, n)$ liefert ein Feld (array). Dieses Feld wird modelliert durch eine Funktion, die Indizes aus $index_A$ auf Elemente aus nat_A abbildet. Diese Funktion kann also insbesondere an einer Stelle $j \in index_A$ ausgewertet werden.

Informationen zur Vorlesung:
http://www-madlener.informatik.uni-kl.de/ag-madlener/teaching/ss2005/gdp/gdp.html