I.Übungsblatt zur Vorlesung
“Semantik von Programmiersprachen” SS 97

Prof. Gert Smolka und Martin Müller

Ausgabe: 17.4.1997 Abgabe: 24.4.1997

Aufgabe I.1(Terminierung (2 Punkte)) Betrachten Sie die **Imp**-Anweisung $c = \textbf{while true do } c'$. Argumentieren Sie über die Struktur der Ableitungen, daß, für alle Zustände σ, kein Zustand σ' existiert, so daß $\langle c, \sigma\rangle \to \sigma'$.

Aufgabe I.2(Deterministische Auswertung (2 Punkte)) Zeigen Sie durch strukturelle Induktion, daß die Auswertung arithmetischer Ausdrücke in **Imp** deterministisch ist, d.h., daß für alle $a \in \mathbf{AExp}$ und $\sigma \in \Sigma$ genau ein $n \in \mathbf{N}$ existiert so daß $\langle a, \sigma\rangle \to n$.

Aufgabe I.3(Mathematische Induktion (2 Punkte)) Ein *Wort* sei eine Folge von Symbolen $a \in A$. Ein Wort der Form $a_1 a_2 \ldots a_n$ hat *Länge* n. Leere Worte sind erlaubt und haben Länge 0. Verkettung von Worten s_1 und s_2 liefert das Wort $s_1 s_2$. Zeigen Sie mit mathematischer Induktion, daß es kein Wort s gibt, so daß $as = sb$, wenn $a, b \in A$ verschiedene Symbole sind.

Aufgabe I.4(Arithmetische Ausdrücke mit Zuweisungen (6 Punkte)) Arithmetische Ausdrücke $a \in \mathbf{AEZ}$ mit Zuweisungen seien durch die folgende abstrakte Syntax gegeben:

$$\begin{array}{rcl} a & ::= & n \mid X \mid a_1 \circ a_2 \mid X \leftarrow a \\ X & \in & \mathbf{Loc} \end{array}$$

Das Ergebnis n der Auswertung von $X \leftarrow a$ stimme mit demjenigen von a überein. Die Auswertung von $X \leftarrow a$ habe jedoch zusätzlich den Seiteneffekt der Zuweisung $X := a$. Solche arithmetischen Ausdrücke finden Sie zum Beispiel in der Sprache C.

1. Geben Sie eine Ergebnissemantik $\to \subseteq (\mathbf{AEZ} \times \Sigma) \times (\mathbf{N} \times \Sigma)$ für arithmetische Ausdrücke mit Zuweisungen an, die von links nach rechts auswertet. Zum Beispiel soll $\langle (X \leftarrow 2) * X, \sigma\rangle \to \langle 4, \sigma[2/X]\rangle$ für alle Zustände σ gelten.
2. Was würde passieren, wenn wir die parallele Auswertung (bzw. die Auswertung bei Bedarf) um arithmetische Ausdrücke mit Zuweisung erweitern würden (informell)?
3. Die Sprache **Imp*** sei definiert wie **Imp**, wobei die arithmetischen Ausdrücke **AExp** ersetzt sind durch **AEZ** wie oben definiert. Die booleschen Ausdrücke $b \in \mathbf{BEZ}$ von **Imp*** sind definiert wie zuvor, allerdings unter Rückgriff auf **AEZ** anstelle von **AExp**. Die Auswertung arithmetischer und boolescher Ausdrücke in **Imp*** soll von links nach rechts erfolgen. Geben Sie eine vollständige Ergebnissemantik für **Imp*** an. Beachten Sie dabei, daß die Auswertungsrelation für boolesche Ausdrücke in **Imp*** nun eine Relation vom Typ $(\mathbf{BEZ} \times \boldsymbol{\Sigma}) \times (\mathbf{T} \times \boldsymbol{\Sigma})$ sein muß.

Aufgabe I.5(Einschritt-Semantik Teil A (5 Punkte)) Arithmetische Ausdrücke $a \in$ **AE** ohne Lokationen sind durch die folgende abstrakte Syntax gegeben:

$$\begin{array}{rcl} \circ & ::= & + \mid * \mid - \\ a & ::= & n \mid a_1 \circ a_2 \\ n & \in & \mathbf{N} \end{array}$$

In der Vorlesung wurde eine *Ergebnissemantik* für solche Ausdrücke vorgestellt. Darunter verstehen wir eine operationale Semantik $\rightarrow \subseteq \mathbf{AE} \times \mathbf{N}$, die Ausdrücke mit ihrem Ergebnis in Beziehung setzt. In dieser Aufgabe diskutieren wir verschiedene *Einschritt-Semantiken*. Die parallele Einschritt-Semantik $\rightarrow_p \subseteq \mathbf{AE} \times \mathbf{AE}$ ist die kleinste Relation, die die folgenden Regeln und Axiome erfüllt:

$$\frac{a_0 \rightarrow_p a_0'}{a_0 \circ a_1 \rightarrow_p a_0' \circ a_1} \qquad \frac{a_1 \rightarrow_p a_1'}{a_0 \circ a_1 \rightarrow_p a_0 \circ a_1'}$$

$$n_1 \circ n_2 \rightarrow_p n \quad \text{falls } n = n_1 \circ n_2 \text{ über den ganzen Zahlen}$$

(Die Notation $a \;{}_p\!\leftarrow a'$ stehe im folgenden für $a' \rightarrow_p a$.)

1. Argumentieren Sie, daß $\rightarrow_p$ *terminiert*, d.h. daß keine unendliche Berechnung $a_0 \rightarrow_p a_1 \rightarrow_p \ldots$ existiert.

2. Zeigen Sie, daß $\rightarrow_p$ *uniform konfluent* ist: Falls $a_1 \;{}_p\!\leftarrow a \rightarrow_p a_2$, dann gilt $a_1 = a_2$ oder es existiert a' mit $a_1 \rightarrow_p a' \;{}_p\!\leftarrow a_2$. Hinweis: Strukturelle Induktion über a_1.

3. Definieren Sie eine Einschritt-Semantik $\rightarrow_s \subseteq \mathbf{AE} \times \mathbf{AE}$, die arithmetische Ausdrücke sequentiell von links nach rechts auswertet.

4. (1 Zusatzpunkt) Beweisen Sie (1) durch Regelinduktion.

5. (2 Zusatzpunkte) Zeigen Sie, daß für alle $a \in \mathbf{AE}$ ein eindeutig bestimmtes n existiert mit $a \rightarrow_p^* n$.

Aufgabe I.6(Auswertung bei Bedarf (2 Punkte)) Wir definieren boolesche Ausdrücke $b \in \mathbf{BE}$ ohne Lokationen durch folgende abstrakte Syntax. Als arithmetische Ausdrücke nehmen wir dabei $a \in \mathbf{AE}$ an.

$$b \;::=\; \mathbf{true} \mid \mathbf{false} \mid \neg b \mid b_1 \wedge b_2 \mid b_1 \vee b_2 \mid a_1 \leq a_2$$

Definieren Sie eine Einschritt-Semantik, die boolesche Ausdrücke sequentiell von links nach rechts auswertet. Dabei sollen rechts stehende Argumente nur bei Bedarf berechnet werden. Zum Beispiel wird b zur Auswertung von **false** $\wedge\, b$ nicht benötigt.