Institut für Informatik III
Universität Bonn
HD Dr. R. Hinze

Römerstraße 164
D–53117 Bonn
26. April 2006

Übungen zu „Prinzipien von Programmiersprachen“

Blatt 2

Aufgabe 7. Die Alternative **if** lässt sich wie folgt verallgemeinern.

$$\begin{array}{lll} e & ::= \cdots & \text{-- } \textit{Ausdrücke:} \\ & \mid \ \mathbf{if}\ g\ \mathbf{end} & \text{-- erweiterte Alternative} \end{array}$$

Der Rumpf einer erweiterten Alternative ist eine Folge von sogenannten bewachten Ausdrücken.

$$\begin{array}{lll} g & ::= \epsilon & \text{-- leerer Rumpf} \\ & \mid \ e_1 \Rightarrow e_2 & \text{-- bewachter Ausdruck} \\ & \mid \ g_1 \shortmid g_2 & \text{-- Sequenz von bewachten Ausdrücken} \end{array}$$

Der Ausdruck e_1 in $e_1 \Rightarrow e_2$ ist ein Boolescher Ausdruck und heißt *Wächter*. Eine erweiterte Alternative wird ausgewertet, indem die Wächter von oben nach unten (bzw. von links nach rechts) ausgewertet werden, bis der erste Wächter zu *true* auswertet. Anschließend wird der korrespondierende Ausdruck auf der rechten Seite des Pfeils '$\Rightarrow$' ausgewertet; dieser Wert ist auch der Wert der erweiterten Alternative.

1. Definiere die statische Semantik der erweiterten Alternative, indem Du geeignete Typregeln aufstellst.

2. Definiere die dynamische Semantik der erweiterten Alternative, indem Du geeignete Auswertungsregeln aufstellst.

In der Vorlesung haben wir die Semantik von Bindungen mit Hilfe der *Substitutionssemantik* erklärt.

$$\frac{e_1 \Downarrow v_1 \qquad e_2[x \mapsto v_1] \Downarrow v}{\mathbf{let\ val}\ x = e_1\ \mathbf{in}\ e_2 \Downarrow v}$$

Die Semantik ist von einer realen Implementierung weit entfernt, da Terme auch bei Wertebindungen wiederholt ausgerechnet werden (siehe Aufgabe 8). Typischerweise werden die Werte von Variablen nicht substituiert, sondern in einer geeigneten Datenstruktur (Stack oder Heap) nachgeschlagen. Die folgende *Umgebungssemantik* modelliert diese Technik.
Eine *Umgebung* ρ ist eine stackartig organisierte Folge von Bindungen.

$$\begin{array}{ll} \rho & ::= \emptyset \\ & \mid \ \rho, x \Downarrow v \end{array}$$

Der Wert einer Variablen wird in der Umgebung nachgeschlagen.

$$\frac{}{\rho, x \Downarrow v \vdash x \Downarrow v} \qquad \frac{\rho \vdash x \Downarrow v \qquad x \neq y}{\rho, y \Downarrow w \vdash x \Downarrow v}$$

Die Auswertungsrelation ist nunmehr dreistellig: $\rho \vdash e \Downarrow v$ bedeutet "in der Umgebung ρ wertet e zu v aus".
Wertebindungen legen den berechneten Wert in der Umgebung ab.

$$\frac{\rho \vdash e_1 \Downarrow v_1 \qquad \rho, x \Downarrow v_1 \vdash e_2 \Downarrow v}{\rho \vdash \mathbf{let\ val}\ x = e_1\ \mathbf{in}\ e_2 \Downarrow v}$$

Aufgabe 8. Werte den folgenden Term mit Hilfe der Subsitutionssemantik aus.

$$\mathbf{let\ val}\ x = \mathit{succ}\ (\mathit{succ}\ \mathit{zero})\ \mathbf{in}\ \mathit{add}\ (x, x)$$

Nehme an, dass *add* wie *zero* und *succ* vordefiniert ist. Was fällt Dir auf? *Hinweis:* der Beweisbaum kann auch linear notiert werden, indem die Koten systematisch durchnumeriert werden.

Aufgabe 9. Zeige die Äquivalenz von Substitutionssemantik und Umgebungssemantik.

$$\emptyset, x_1 \Downarrow v_1, \ldots, x_n \Downarrow v_n \vdash e \Downarrow v \iff e[x_n \mapsto v_n] \cdots [x_1 \mapsto v_1] \Downarrow v$$

Beschränke Dich auf Boolesche Ausdrücke und Wertebindungen.
Aufgabe 10. Harry Hacker hat die Umgebungssemantik um folgende Regeln für die Auswertung von Abstraktionen und Applikationen erweitert.

$$\frac{}{\rho \vdash \mathbf{fun}\ (\mathbf{val}\ x : \tau) \Rightarrow e \Downarrow \mathbf{fun}\ (\mathbf{val}\ x : \tau) \Rightarrow e}$$

$$\frac{\rho \vdash e_1 \Downarrow \mathbf{fun}\ (\mathbf{val}\ x : \tau) \Rightarrow e \qquad \rho \vdash e_2 \Downarrow v_2 \qquad \rho, x \Downarrow v_2 \vdash e \Downarrow v}{\rho \vdash e_1\ e_2 \Downarrow v}$$

Was ist von den Regeln zu halten? *Hinweis:* Bestimme die Bedeutung des folgenden Programms und vergleiche das Ergebnis mit dem der Substitutionssemantik.

$$\begin{array}{l} \mathbf{let\ val}\ f = (\mathbf{fun}\ (\mathbf{val}\ x : \mathit{Nat}) \Rightarrow \mathbf{fun}\ (y : \mathit{Nat}) \Rightarrow x)\ 3 \\ \mathbf{in}\ \ \mathbf{let\ val}\ x = 5 \\ \quad\ \ \mathbf{in}\ \ f\ x \end{array}$$