

Übungen zu „Semantik von Programmiersprachen“, WS 2004/05

Nr. 1, Besprechung mündlicher Aufgaben: 1. November 2004 in der Übung,
Abgabe der Hausaufgaben: 3. November 2004 vor der Vorlesung

Hinweise:

- Schriftliche Aufgaben sollen vor der Vorlesung abgegeben werden (Abgabe in Gruppen bis zu zwei Personen möglich).
 - Mündliche Aufgaben sollen zum angegebenen Tutoriumstermin vorbereitet werden, sie werden im Tutorium präsentiert und besprochen.
 - Die Präsentation von mündlichen und schriftlichen Aufgaben im Tutorium ist Bedingung für den Erwerb eines Leistungsnachweises. Bitte achten Sie selbst (mit) darauf, dass Sie diese Bedingung erfüllen können. Da im ersten Tutorium keine schriftlichen Aufgaben besprochen werden, enthält dieses Blatt mehr mündliche Aufgaben.
-

Mündliche Aufgaben

- 1.1 (a) Zeichnen Sie den abstrakten Syntaxbaum des IMP-Programms aus Aufg. 1.7.
(b) Schreiben Sie ein IMP-Programm, das zu zwei gegebenen Werten X und Y deren größten gemeinsamen Teiler $\text{ggt}(X, Y)$ berechnet.
- 1.2 Geben Sie analog zur Vorgehensweise der Vorlesung eine Semantik für Ausdrücke an, welche Zeichenketten (Strings) als Werte besitzen:
- (a) Legen Sie eine sinnvolle Menge von Grundelementen und -operationen fest.
(b) Definieren Sie eine Menge von Ausdrücken über diesen Grundoperationen.
(c) Legen Sie eine Auswertungsrelation fest.
- 1.3 (a) Erweitern Sie die Sprache IMP um ein Konstrukt

repeat S until b

analog zum gleichnamigen Konstrukt in Pascal, indem Sie entsprechende semantische Regeln angeben. (Benutzen Sie dabei *nicht* das **while**-Konstrukt.)

- (b) Zeigen Sie:

$(\text{repeat } S \text{ until } b) \sim S; \text{ if } b \text{ then skip else } (\text{repeat } S \text{ until } b);$

- 1.4 (a) Definieren Sie formal, was es im Sinne der operationellen Semantik bedeutet, dass eine Anweisung (in einem Zustand σ und allgemein) *terminiert* bzw. *divergiert*.
 (b) Betrachten Sie die folgenden Anweisungen:

```

while   $\neg(X = 1)$   do ( $Y := Y * X; X := X - 1$ )
while    $1 \leq X$    do ( $Y := Y * X; X := X - 1$ )
while   true     do skip

```

Begründen Sie im Sinne Ihrer Definitionen, welche der Anweisungen terminieren und welche divergieren.

- 1.5 Sei Δ ein Alphabet (d.h. eine endliche, nicht-leere Menge). Eine Zeichenkette über Δ ist eine Folge $a_1 \dots a_n$ von Symbolen $a_j \in \Delta$ mit $0 \leq j \leq n, n \geq 0$. Die Anzahl n der Symbole einer Zeichenkette bezeichnet man als Länge der Zeichenkette. Die leere Zeichenkette hat die Länge 0. Zwei Zeichenketten u und v können zu der Zeichenkette uv konkateniert werden.

Zeigen Sie

- (a) mittels vollständiger Induktion
 (b) durch einen Widerspruchsbeweis

die folgende

Behauptung: Es existiert keine Zeichenkette u , für die $au = ub$ mit zwei verschiedenen Symbolen a und b aus Δ .

Schriftliche Aufgaben

- 1.6 Alternative Auswertungsstrategien für Boolesche Ausdrücke

4 Punkte

- (a) **Sequentielle Auswertung:** Geben Sie Auswertungsregeln für Boolesche Ausdrücke der Form $b_1 \wedge b_2$ und $b_1 \vee b_2$ an, welche in einem Ausdruck der Gestalt **false** $\wedge b_2$ und **true** $\vee b_2$ den zweiten Teilausdruck nicht auswerten, weil das Gesamtergebnis unabhängig von dessen Wert ist.
 (b) **Parallele Auswertung:** Geben Sie Auswertungsregeln an, welche einen Booleschen Ausdruck der Form $b_1 \vee b_2$ dann zu **true** auswerten, wenn sich b_1 oder b_2 zu **true** auswerten lässt, und in diesem Fall b_2 bzw. b_1 nicht auswerten.

- 1.7 Erstellen Sie den Ableitungsbaum für die operationelle Semantik der Anweisung

5 Punkte

$$Z := 0; \text{ while } Y \leq X \text{ do } (Z := Z + 1; X := X - Y)$$

in einem Zustand $\sigma \in \Sigma$ mit $\sigma(X) = 17$ und $\sigma(Y) = 5$.

- 1.8 Zeigen Sie mittels struktureller Induktion, dass die Auswertung von arithmetischen Ausdrücken immer terminiert, d.h. dass für alle Ausdrücke $a \in \mathbf{AExp}$ und Zustände $\sigma \in \Sigma$ ein $m \in \mathbf{N}$ existiert mit $\langle a, \sigma \rangle \rightarrow m$.

3 Punkte