

4. Übung zur Vorlesung „Prinzipien von Programmiersprachen“ Wintersemester 2008/2009

Abgabe: 2. Dezember 2008 in der Vorlesung

Aufgabe 13

(Präsenzaufgabe)

In der Vorlesung wurden mehrere Parameterübergabemechanismen eingeführt. Die folgende Prozedur soll die Werte von zwei Variablen x und y vertauschen.

```
procedure swap(x, y: integer);  
var  
    temp: integer;  
begin  
    temp:= x;  
    x := y;  
    y := temp;  
end;
```

Die Variablen i , $a[1]$ und $a[2]$ sollen die Werte 1, 2 und 3 besitzen. Wie lauten die Werte von i , $a[1]$ und $a[2]$ nach dem Aufruf von `swap(a[i], i)`, falls

- x und y per Wertaufruf übergeben werden,
- x per Wertaufruf und y per Referenzaufruf übergeben wird,
- x und y per Referenzaufruf übergeben werden?

Aufgabe 14

In der Vorlesung wurde die Semantik des Schleifenkonstrukts `while (B) S;` mit Hilfe von Inferenzregeln spezifiziert. Geben Sie analog dazu Inferenzregeln an, die die Semantik der folgenden Schleifenkonstrukte spezifizieren:

- `do S; while (B);`
- `for (init_stat; bool_expr; incr_stat) S;`
- `loop {S1; if (B) break; S2};` wobei in S_1 und S_2 keine `break`-Anweisungen vorkommen sollen.

Aufgabe 15

Geben Sie eine Ableitung von $\langle E, M \rangle p(2); \langle E, M' \rangle$ an, wobei

- a) $E = p : \text{proc}(n : \text{int}, S, E')$
- b) $S = \text{if } (n > 1) \ p(n-1); j = n * j;$
- c) $E' = j : (\text{ref1}, \text{int})$
- d) $M = \{\text{ref1} \mapsto 1\}.$

Wie sieht der Speicher M' nach der Abarbeitung von $p(2);$ aus? Die Herleitungen für $\overset{\text{lookup}}{\vdash}, \overset{R}{\vdash}$ und $\overset{L}{\vdash}$ dürfen weggelassen werden.

Aufgabe 16

Implementieren Sie das Spiel „Game of Life“. Die Welt, in der das Spiel stattfindet, ist ein Schachbrett mit 10×10 Feldern. Jedes Feld beherbergt eine Zelle, die entweder lebt oder tot ist. Der Zustand einer Zelle kann sich von Generation zu Generation ändern. Die Änderungen vollziehen sich nach den folgenden Regeln:

- a) Die Nachbarn einer Zelle sind die Zellen, die sich in einem vertikal, horizontal oder diagonal angrenzenden Feld befinden.
- b) Falls eine Zelle lebt und maximal nur eine lebende Nachbarzelle besitzt, so stirbt die Zelle bis zur nächsten Generation an Einsamkeit.
- c) Falls eine Zelle lebt und mehr als drei lebende Nachbarzellen besitzt, so stirbt die Zelle bis zur nächsten Generation an Überbevölkerung.
- d) Falls eine Zelle zwei oder drei lebende Nachbarn besitzt, so bleibt die Zelle bis zur nächsten Generation am Leben.
- e) Falls eine Zelle tot ist und genau drei lebende Nachbarn besitzt, so wird sie zum Leben erweckt. Andernfalls bleibt die Zelle tot.
- f) Das Gebären und das Sterben von Zellen einer Generation findet zur gleichen Zeit statt.

Stellen Sie den Zustand der Welt durch ein zweidimensionales Feld dar. Das Programm soll 10 Generationen berechnen und ausgeben. Verwenden Sie hierfür die Standardmethoden `System.out.print` und `System.out.println`. Die lebenden Zellen können beispielsweise durch ein `*` in ihrem Feld dargestellt werden, während die Felder der toten Zellen leer bleiben.

In der ersten Generation leben die Zellen an den Positionen (3,3), (4,3), (5,3), (6,4), (7,4) und (8,4).