

Übungen zu „Grundlagen des Compilerbau“, Winter 2009/10

Nr. 10, Abgabe der Aufgaben: 3. Februar 2010 vor der Vorlesung

Aufgaben

10.1 PSA-Erweiterung

6 Punkte

Bekanntlich stellt die MA folgende Befehle: `LOAD n`, `STORE n`, `LIT n`, `JMP n`, `JPFALSE n` sowie beliebige arithmetische und Vergleichsoperationen mit der bekannten Semantik zur Verfügung. Zusätzlich werden die neuen Befehle `DUP` und `POP` mit folgender Semantik zur Verfügung gestellt:

$$\begin{aligned} \mathcal{C}[\text{DUP}](m, d : z, h) &:= (m + 1, d : z : z, h) \\ \mathcal{C}[\text{POP}](m, d : z, h) &:= (m + 1, d, h) \end{aligned}$$

- (a) Die Sprache PSA soll im Anweisungsteil Γ um `case`-Anweisungen der Gestalt / 3

```

case  E of
      0 ->  $\Gamma_0$ ;
      :
      k ->  $\Gamma_k$ 
esac

```

erweitert werden. `case`-Anweisungen ermöglichen Fallunterscheidungen. Es gilt:

$$\begin{aligned} &\mathcal{C}[\text{case } E \text{ of } 0 \rightarrow \Gamma_0; \dots k \rightarrow \Gamma_k \text{ esac}] \rho \sigma \\ &:= \begin{cases} \mathcal{C}[\Gamma_0] \rho \sigma & \text{falls } \mathcal{E}[E] \rho \sigma = 0 \\ \vdots & \vdots \\ \mathcal{C}[\Gamma_k] \rho \sigma & \text{falls } \mathcal{E}[E] \rho \sigma = k \\ \sigma & \text{sonst} \end{cases} \end{aligned}$$

Erweitern Sie die Übersetzungsfunktion für Anweisungen $ct :: \text{Cmd} \times \text{Tab} \times \text{Adr} \rightarrow \text{MA-Code}$ um eine Übersetzungsvorschrift für `case`-Anweisungen.

- (b) Die Sprache PSA soll im Anweisungsteil Γ um `loop`-Schleifen der Gestalt `loop E (Γ)` / 3
 erweitert werden. Semantisch unterscheiden sich diese von `while`-Schleifen vor allem dadurch, dass die Anzahl der Schleifendurchläufe beim Schleifeneintritt durch den Wert von E festgelegt ist und nicht durch den Schleifenrumpf beeinflusst werden kann. Formal:

$$\begin{aligned} \mathcal{C}[\text{loop } E (\Gamma)] \rho \sigma &:= f^n(\sigma) \\ \text{mit } f &:= \mathcal{C}[\Gamma] \rho; \quad n := \mathcal{E}[E] \rho \sigma \end{aligned}$$

Erweitern Sie die Übersetzungsfunktion für Anweisungen $ct :: \text{Cmd} \times \text{Tab} \times \text{Adr} \rightarrow \text{MA-Code}$ um eine Übersetzungsvorschrift für `loop`-Schleifen.

10.2 Aktivierungsblöcke und abstrakte Maschine für PSP

6 Punkte

(a) Gegeben sei der folgende *MP*-Zustand

/ 3

$$(5, 4, 8 : 3 : 10 : 7 : 4 : 3 : 27 : 9 : 8 : 3 : 54 : 1 : \dots) \in ZR$$

Ermitteln Sie den Zustand nach Ausführung der nächsten vier Befehle des folgenden *MP*-Codes:

```

      ⋮
5: LOAD(1,1)
6: ADD
7: RET
8: LOAD(0,2)
9: STORE(2,1)
10: CALL (84,1,3);
11: RET
      ⋮
    
```

(b) *Keiner* der folgenden Keller-Inhalte kann bei der Ausführung eines übersetzten PSP-Programms als Prozedurkeller der *MP*-Maschine entstehen. Finden und begründen Sie die Inkonsistenzen.

/ 3

(Welchen Kellerinhalt hat eine korrekte Blockstruktur?)

- A. 7:2:34:4:3:5:12:2:1:3:2:2:0:0:0:17
- B. 10:2:34:7:3:0:12:3:2:12:3:2:1:0:0:0:17
- C. 7:2:34:7:3:5:12:3:2:12:3:2:2:0:0:0:17