

# Softwareparadigmen SS 2017, Übungsblatt 3

**Abgabe:** 24. Mai 2017, bis 16:00 Uhr vor dem Sekretariat IST, Inffeldgasse 16b, 2. OG

## Beispiel 1 (5,5 P.)

Gegeben sei der Datentyp *IntegerQueue*, mit dem es möglich ist, Ganzzahlen in einer Warteschlange zu speichern.

Formale Definition des Datentyps *IntegerQueue* = (A, F, P, C):

- Wertebereich  $A = \{\langle \rangle\} \cup \{\langle x_1, \dots, x_n \rangle \mid x_i \in \mathbb{Z} \wedge n \in \mathbb{N} \wedge n \geq 1\}$
- Funktionen:
  1.  $f_1$ :  

$$\begin{aligned} enqueue(Q, x) &= \langle x \rangle, & if \ (Q == \langle \rangle) \\ &= \langle x, x_1, \dots, x_n \rangle, & if \ (Q == \langle x_1, \dots, x_n \rangle) \end{aligned}$$
  2.  $f_2$ :  

$$\begin{aligned} head(Q) &= \perp, & if \ (Q == \langle \rangle) \\ &= x_n, & if \ (Q == \langle x_1, \dots, x_n \rangle, n \geq 1) \end{aligned}$$
  3.  $f_3$ :  

$$\begin{aligned} dequeue(Q) &= \perp, & if \ (Q == \langle \rangle) \\ &= \langle \rangle, & if \ (Q == \langle x \rangle) \\ &= \langle x_1, \dots, x_{n-1} \rangle, & if \ (Q == \langle x_1, \dots, x_n \rangle, n \geq 2) \end{aligned}$$
- Prädikat:
  1.  $p_1$ :  

$$isEmptyQueue(Q) = (Q == \langle \rangle)$$
- Konstanten:
  1.  $c_1$ :  
 $emptyQueue = \langle \rangle$
  2.  $c_2$ :  
 $error = \perp$   
 Diese Konstante repräsentiert einen Fehler, zum Beispiel wenn ein Element aus einer leeren Queue entfernt werden soll.
  3.  $c_n$ :  
 $\dots, -2, -1, 0, 1, 2, \dots$   
 Konstanten für Integer Zahlen.

- (a) Geben Sie eine Codierung des Datentyps *IntegerQueue* in den Datentyp *Listen + Integer* an.
- (b) Beweisen Sie dass die Codierungseigenschaft erfüllt ist.

**Beispiel 2** (4 P.)

Gegeben ist folgendes AL-Programm. Hier steht  $\alpha$  für ein beliebiges Statement, das jedoch weder lesend noch schreibend auf die Variable  $i$  zugreifen darf.

```

1 begin
2   i := 0;
3   while(i < 2) do
4     begin
5       i := i + 1;
6        $\alpha$ 
7     end
8 end

```

- Ersetzen Sie  $\alpha$  durch das konkrete Statement  $a := a + a$ . Verwenden Sie die Interpretationsfunktion für AL-Programme, um das Programm auszuwerten. Der Wert von  $a$  soll anfänglich 42 sein. Geben Sie auch die Zwischenschritte sowie das resultierende  $\omega$  Environment an.
- Beweisen Sie unter Verwendung der Interpretationsfunktion für AL-Programme: Das oben stehende AL-Programm ist für alle  $\alpha$  ident mit dem folgenden AL-Programm.

```

1 begin
2   begin
3      $\alpha$  ;
4      $\alpha$ 
5   end ;
6   i := 2
7 end

```

**Beispiel 3** (3 P.)

Interpretieren Sie die gegebenen prädikatenlogischen Ausdrücke über dem angegebenen Datentyp und bestimmen Sie deren semantischen Status (gültig, erfüllbar, unerfüllbar). Geben Sie die Zwischenschritte für  $I_{PL}$  an, sowie die Belegung durch die  $\omega$  Environments. Die Interpretation der Terme ( $I_T$ ) dürfen Sie abkürzen. Beachten Sie, dass für die Operatoren folgende Ordnung der Bindungsstärke (= operator precedence) gilt:

- $\neg$
- $\wedge$
- $\vee$
- $\rightarrow$

- Datentyp der ganzen Zahlen  $\forall x \text{ eq?}(\text{times}(x, y), 0)$
- Datentyp der Listen  $\exists l \text{ eq?}(\text{first}(l), l) \vee \text{eq?}(\text{first}(l), a)$
- Datentyp der ganzen Zahlen  $\forall x \forall y \text{ eq?}(x, y) \rightarrow (\text{ge?}(x, y) \wedge \text{ge?}(y, x))$

**Viel Erfolg!**