

# Semantik und Programmverifikation

Prof. Dr. Christoph Walther / Simon Siegler  
Technische Universität Darmstadt — Wintersemester 2008/09

## Übungsblatt 3

---

### Aufgabe 3.1 (Rekursionsrelation)

Beweisen Sie die Terminierung des folgenden Algorithmus und geben Sie an, welche Funktion dieser berechnet.

```
function  $g(x, y, z : \mathbb{N}) : \mathbb{N} \Leftarrow$   
  if  $z = 0$   
    then if  $x = 0$   
          then 0  
          else  $g(x - 1, y, y)$   
        fi  
    else  $g(x, y, z - 1) + 1$   
  fi  
end
```

Hinweis: Zeigen Sie die Terminierung, indem Sie eine Rekursionsrelation  $>_g$  definieren und deren Fundiertheit nachweisen (vgl. Beispiele 1.4.1 und 1.4.2 im Buch).

### Aufgabe 3.2 (funktionales Programmieren)

Schreiben Sie ein funktionales Programm  $P \in \mathcal{FP}$ , das

- eine Funktionsprozedur  $\text{function } \text{exp}(x, y : \text{nat}) : \text{nat} \Leftarrow \dots$  zur Berechnung der Exponentialfunktion und
- eine Funktionsprozedur  $\text{function } \text{prim}(x : \text{nat}) : \text{bool} \Leftarrow \dots$  zur Überprüfung, ob eine natürliche Zahl prim ist,

enthält.

### Aufgabe 3.3 (Auswertungsrelation der Basismaschine)

- Vervollständigen Sie den Beweis zu Satz 2.2.1 (i), in dem Sie im Induktionsschritt die Fälle  $t = \text{pred}(t_1)$  und  $t = \text{eq}_{\text{nat}}(t_1, t_2)$  betrachten.
- Vervollständigen Sie den Beweis zu Satz 2.2.1 (ii), in dem Sie im Induktionsschritt die Fälle  $t = \text{pred}(t_1)$  und  $t = \text{if}_s(b, p_1, p_2)$  betrachten.

### Aufgabe 3.4 (Auswertungsrelation für Programme)

Beweisen Sie die folgende Behauptung:

Für jede Regel der Form

$$\frac{f(t_1, \dots, t_j, \dots, t_n)}{f(t_1, \dots, r_j, \dots, t_n)}, \text{ falls } t_j \longrightarrow_P r_j$$

im Auswertungskalkül für funktionale Programme gilt

$$f(t_1, \dots, t, \dots, t_n) \longrightarrow_P^* f(t_1, \dots, r, \dots, t_n), \text{ falls } t \longrightarrow_P^* r.$$

**Aufgabe 3.5** (Auswertungsfolgen)

Das Programm  $P$  bestehe aus den folgenden Funktionsprozeduren:

**function**  $zero(x : nat) : bool \Leftarrow eq(x, 0)$

**function**  $plus(x, y : nat) : nat \Leftarrow$   
 $if_{nat}(zero(x), y, succ(plus(pred(x), y)))$

**function**  $times(x, y : nat) : nat \Leftarrow$   
 $if_{nat}(zero(x), 0, plus(times(pred(x), y), y))$

**function**  $inner(x : nat) : nat \Leftarrow inner(succ(x))$

**function**  $outer(x : nat) : nat \Leftarrow succ(outer(x))$

Berechnen Sie durch Angabe der Auswertungsfolgen die Ergebnisse von

- (a)  $eval_P(times(0, outer(0)))$ ,
- (b)  $eval_P(times(outer(0), 0))$ ,
- (c)  $eval_P(times(succ(0), outer(0)))$ ,
- (d)  $eval_P(eq_{nat}(inner(0), outer(0)))$ ,
- (e)  $eval_P(if_{nat}(eq_{nat}(inner(0), inner(0)), 0, 0))$ .