Formale Grundlagen der Informatik 3

Prof. Dr. Christoph Walther / Visar Januzaj, Nathan Wasser

Technische Universität Darmstadt — Wintersemester 2011/12

Übung 4

Version 1 vom 13.12.2011

Hinweis: Im Folgenden bezeichnet $>$ die übliche (transitive) Ordnung auf den natürlichen Zahlen.

Aufgabe 4.1 (Fundierte Mengen)

Beweisen oder widerlegen Sie die Fundiertheit der folgenden Mengen:

1. $(\mathbb{Z}, >_{\mathsf{abs}})$ mit $n >_{\mathsf{abs}} m$ genau dann, wenn $|n| > |m|$
2. $(\mathcal{C}_{\mathtt{list}[\mathbb{N}]}, \triangleright)$ mit $k \triangleright l$ genau dann, wenn $k \neq \varnothing \wedge \big(l = \varnothing \vee \mathtt{hd}(k) > \mathtt{hd}(l) \vee \mathtt{tl}(k) \triangleright \mathtt{tl}(l)\big)$
3. $(\mathbb{N} \times \mathbb{N}, \succ)$ mit $(n_1, m_1) \succ (n_2, m_2)$ genau dann, wenn $n_1 > n_2$ oder $n_1 = n_2$ und $m_1 > m_2$.

Aufgabe 4.2 (Relationen)

Betrachten Sie die folgende Relationenbeschreibung:

$$\begin{aligned} R := \{ & \langle \{?0(x)\}, \emptyset \rangle, \\ & \langle \{\neg ?0(x), ?0(y)\}, \emptyset \rangle, \\ & \langle \{\neg ?0(x), \neg ?0(y)\}, \{\{x/x, y/y - x\}, \{x/x - y, y/y\}\} \rangle \} \end{aligned}$$

Dabei ist die Prozedur „-“ wie folgt definiert:

```
function [infixl] -(a, b : ℕ) : ℕ <=
if ?0(a)
  then 0
  else if ?0(b)
         then a
         else ⁻(a) - ⁻(b)
       end_if
end_if
```

(a) Bestimmen Sie die Menge $pre_{>_{R,xy}}\big((42, 23)\big)$.

(b) Geben Sie alle mit $(5, 3)$ startenden $>_{R,xy}$-Ketten an.

Aufgabe 4.3 (Minimale Elemente und Vorgänger)

Bestimmen Sie für jede der folgenden fundierten Mengen $(M, \succ)$ die minimalen Elemente $min_{\succ}(M)$. Geben Sie außerdem für jedes nicht-minimale Element $m \in M \setminus min_{\succ}(M)$ die Anzahl der Vorgänger $|pre_{\succ}(m)|$ an.

1. $(\mathbb{N}, >)$
2. $(\mathcal{C}_{\mathtt{list}[\mathbb{N}]}, \gtrdot)$ mit $k \gtrdot l$ genau dann, wenn $k = n :: l$ für ein $n \in \mathbb{N}$.
3. $(\mathcal{C}_{\mathtt{list}[\mathbb{N}]}, >_{\mathsf{len}})$ mit $k >_{\mathsf{len}} l$ genau dann, wenn $|k| > |l|$. Dabei ist $|k|$ die Länge der Liste k.

Aufgabe 4.4 (Strukturelle Ordnung)

Geben Sie für die folgenden Datentypen `struc` jeweils die Relationenbeschreibung $R_{\texttt{struc}}$ an.

1. ```
   structure pair[@A, @B] <=
     pair(fst : @A, snd : @B)
   ```
2. ```
   structure tree[@V] <=
     tip,
     node(value : @V, left : tree[@V], right : tree[@V])
   ```
3. ```
   structure EXPR <=
     VAR#(index : nat),
     EXPR0(e-op0 : nat),
     EXPR2(e-op2 : nat, arg1 : EXPR, arg2 : EXPR)
   ```

Aufgabe 4.5 (Rekursion und Induktion)

Betrachten Sie die folgenden Prozeduren `p`:

1. ```
   function f(n : ℕ) : ℕ <=
   if ?0(n)
     then 1
     else if ?0(pred(n))
            then 1
            else f(pred(n)) + f(pred(pred(n)))
          end_if
   end_if
   ```
2. ```
   function g(n, m : ℕ) : ℕ <=
   if ?0(n)
     then 0
     else if ?0(m)
            then n
            else g(pred(n), pred(m))
          end_if
   end_if
   ```
3. ```
   function h(k : list[nat]) : ℕ <=
   if ?ø(k)
     then ⋆
     else if ?ø(tl(k))
            then hd(k)
            else if hd(k) > hd(tl(k))
                   then h(tl(k))
                   else h(hd(k) :: tl(tl(k)))
                 end_if
          end_if
   end_if
   ```

- Geben Sie zu jeder Prozedur `p` die zusammengesetzten Relationenbeschreibungen $R_{\texttt{p}}$, sowie die Induktionsaxiome $\mathit{IndAx}_{\texttt{p}(\ldots)>0,R_{\texttt{p}}}$ an.
- Begründen Sie jeweils die Fundiertheit der Relationenbeschreibungen $R_{\texttt{p}}$. Ein formaler Beweis ist nicht notwendig.