Technische Universität München
Institut für Informatik
Prof. Manfred Broy
A. Gruler, B. Spanfelner

WS 2006/07

13. November 2006
Übungsblatt 3

Übungen zu „Formale Methoden“

Aufgabe 1 Induktionsbeweise über *Int*

Betrachten Sie die Funktion *sub*: *Int*, *Int* → *Int* gegeben durch

$$sub(x, y) = \begin{cases} x, & \text{falls } y = 0; \\ sub(x - 1, y - 1), & \text{falls } y > 0; \\ sub(x + 1, y + 1), & \text{falls } y < 0 \end{cases}$$

Eine entsprechende Spezifikation der Sorte *Int* finden Sie auf der Seite zur Vorlesung.

(a) Beweisen Sie zunächst

$$sub(x, y - 1) = sub(x + 1, y)$$

für alle $x, y \in Int$ durch Induktion.

(b) Beweisen Sie nun

$$sub(x, sub(y, z)) = sub(x + z, y)$$

für alle $x, y, z \in Int$, $0 \leq z \leq y$, ebenfalls durch Induktion.

Aufgabe 2 Algebraische Spezifikation von Mengen

In dieser Aufgabe betrachten wir Mengen von beliebigen (Daten-)Elementen gleichen Typs.

(a) Welche prinzipiell unterschiedlichen Arten von Mengen gibt es?

(b) Geben Sie für die identifizierten Arten jeweils eine entsprechende algebraische Spezifikation an. Worin unterscheiden sich die Spezifikationen?

Aufgabe 3 (Hausaufgabe) Induktionsbeweis über Int
(6 Punkte; Abgabe bis spätestens 20.11.2006 zu Beginn der Übung.)

Gilt die Gleichung von Aufgabe 1(b)

$$sub(x, sub(y, z)) = sub(x + z, y)$$

für alle $x, y, z \in Int$? Beweisen Sie Ihre Aussage durch Induktion.

Aufgabe 4 (Hausaufgabe) Induktionsbeweis über Nat
(8 Punkte; Abgabe bis spätestens 20.11.2006 zu Beginn der Übung.)

Beweisen Sie die folgende Aussage über der Spezifikation Nat der natürlichen Zahlen mittels Induktion. Kennzeichnen Sie dabei die jeweiligen Teilschritte des Beweises, so dass das Induktionsschema klar ersichtlich wird.

$$\Big(\ \neg\big(\mathsf{iszero}(x)\big)\ \Big) \quad\rightarrow\quad \Big(\ \mathsf{succ}\big(\mathsf{pred}(x)\big) = x\ \Big)$$

Das Symbol $\rightarrow$ steht dabei für die logische Implikation, also

$$(x \rightarrow y) \ \equiv\ (\neg x \vee y) \quad , \quad x, y \in Bool$$

Gehen Sie für die Lösung von den gegebenen Spezifikationen für Nat bzw. $Bool$ aus.

```
SPEC NAT =
{ based_on BOOL,
  sort Nat,

  0 :                Nat,
  succ, pred :       Nat → Nat,
  iszero :           Nat → Bool,
  _ + _ , _ * _ :    Nat, Nat → Nat,      Infix

  Nat generated_by 0, succ,

  iszero(0) = true,
  iszero(succ(x)) = false,

  pred(succ(x)) = x,

  0 + y = y,
  succ(x) + y = succ(x + y),
  0 * y = 0,
  succ(x) * y = y + (x * y),
}
```

$$
\begin{array}{|l|}
\hline
\textsf{SPEC BOOL} = \\
\{\ \textsf{sort Bool}, \\
\\
\quad \begin{array}{lll}
\textsf{true}, \textsf{false} : & \textsf{Bool}, & \\
\neg : & \textsf{Bool} \to \textsf{Bool}, & \textit{Präfix} \\
\_ \vee \_ \ , \quad \_ \wedge \_ \ : & \textsf{Bool}, \textsf{Bool} \to \textsf{Bool}, & \textit{Infix}
\end{array} \\
\\
\quad \textsf{Bool generated\_by true, false}, \\
\\
\quad \textsf{not}\ (\textsf{true} = \textsf{false}), \\
\\
\quad \neg(\textsf{true}) = \textsf{false}, \\
\quad \neg(\textsf{false}) = \textsf{true}, \\
\quad \neg(\neg(x)) = x, \\
\\
\quad (\textsf{false} \vee x) = (x \vee \textsf{false}) = x, \\
\quad (\textsf{true} \vee x) = (x \vee \textsf{true}) = \textsf{true}, \\
\quad (\textsf{true} \wedge x) = (x \wedge \textsf{true}) = x, \\
\quad (\textsf{false} \wedge x) = (x \wedge \textsf{false}) = \textsf{false} \\
\} \\
\hline
\end{array}
$$