Technische Universität München
Fakultät für Informatik
Dr. H. Ehler, Dr. M. Pizka,
W. Prenninger, Dr. M. Schneider

WS 2001
Lösungsvorschläge zu Blatt 6
22. November 2001

Übungen zu Einführung in die Informatik I

Aufgabe 22 Prädikatenlogische Ausdrücke über den natürlichen Zahlen (Lösungsvorschlag)

a)	y ist eine gerade Zahl	$\Leftrightarrow$	$T(2,y)$
b)	y ist eine Primzahl	$\Leftrightarrow$	$\forall\ \mathbf{nat}\ x : [T(x,y) \Rightarrow (x=1) \vee (x=y)]$
c)	x und y sind teilerfremd	$\Leftrightarrow$	$\forall\ \mathbf{nat}\ v : (T(v,x) \wedge T(v,y)) \Rightarrow v=1$
d)	z ist ggT von x und y	$\Leftrightarrow$	$T(z,x) \wedge T(z,y) \wedge (\forall\ \mathbf{nat}\ v : (T(v,x) \wedge T(v,y)) \Rightarrow v \leq z)$

Aufgabe 23 Formulierung prädikatenlogischer Ausdrücke (Lösungsvorschlag)

Bei der Lösung dieser Aufgabe verwenden wir, da wir nur Elemente aus der Menge **D** aller Drachen betrachten, anstatt $\forall x \in \mathbf{D}$ bzw. $\exists x \in \mathbf{D}$, die abkürzende Schreibweise $\forall x$ bzw. $\exists x$.

a) Es ergeben sich folgende prädikatenlogische Ausdrücke:

(i) $\forall y[\forall x\, Ki(x,y) \Rightarrow Fl(x)] \Rightarrow Gl(y)$

(ii) $\forall x\, Gr(x) \Rightarrow Fl(x)$

(iii) $\forall x[\exists y\, Ki(x,y) \wedge Gr(y)] \Rightarrow Gr(x)$

b) Eine informelle Begründung könnte folgendermaßen beschaffen sein: Aus der Tatsache (iii) folgt, dass Kinder eines grünen Drachens grün sind. Mit (ii) ergibt sich, dass die Kinder grüner Drachen fliegen können. Da ein grüner Drachen glücklich ist, wenn alle seine Kinder (das beinhaltet auch den Fall, dass ein Drache keine Kinder hat) fliegen können (i) wir folgern, dass grüne Drachen glücklich sind.

c) Ausgangspunkt der Beweisführung ist die Tatsache (iii):

$$\begin{array}{rcl}
 & & \forall x[\exists y\, Ki(x,y) \wedge Gr(y)] \Rightarrow Gr(x) \\
\text{(Definition der Implikation)} & = & \forall x\, \neg[\exists y\, Ki(x,y) \wedge Gr(y)] \vee Gr(x) \\
\text{(Bezieh. zwischen All- und Existenzquantor)} & = & \forall x[\forall y\, \neg(Ki(x,y) \wedge Gr(y))] \vee Gr(x) \\
\text{(Weglassen der Klammern)} & = & \forall x \forall y\, \neg(Ki(x,y) \wedge Gr(y)) \vee Gr(x) \\
\text{(Definition der Implikation)} & = & \forall x \forall y\, (Ki(x,y) \wedge Gr(y)) \Rightarrow Gr(x) \\
\text{(Vertauschen der Allquantoren)} & = & \forall y \forall x\, (Ki(x,y) \wedge Gr(y)) \Rightarrow Gr(x)
\end{array}$$

Wir haben bis jetzt den ersten Teil der informellen Begründung, –alle Kinder grüner Drachen sind grün–, formal begründet. Es gilt jedoch auch Tatsache (ii); wir können also weiter die Gültigkeit von

$$[\forall y \forall x\, (Ki(x,y) \wedge Gr(y)) \Rightarrow Gr(x)] \wedge [\forall x\, Gr(x) \Rightarrow Fl(x)]$$

folgern, was nach Zusammenfassen der Allquantoren de äquivalenten Ausdruck

$$\forall y \forall x\ ((Ki(x,y) \wedge Gr(y)) \Rightarrow Gr(x)) \wedge (Gr(x) \Rightarrow Fl(x))$$

ergibt. Die Implikation ist eine transitive Operation, d.h. es gilt die Tautologie: $[(a \Rightarrow b) \wedge (b \Rightarrow c)] \Rightarrow (a \Rightarrow c)$ (dies kann leicht algebraisch oder mit einer Wertetabelle gezeigt werden); somit können wir folgern:

$$\forall y \forall x (Ki(x,y) \wedge Gr(y)) \Rightarrow Fl(x). \quad (*)$$

Wir haben also gezeigt, dass die Kinder grüner Drachen fliegen können. Nun gilt zusätzlich noch die Tatsache (i), die wir mit dem Prädikat (*) zu

$$\begin{aligned} \Rightarrow \quad & [\forall y \forall x\ (Ki(x,y) \wedge Gr(y)) \Rightarrow Fl(x)] \wedge [\forall y \forall x\ (Ki(x,y) \Rightarrow Fl(x)) \Rightarrow Gl(y)] \\ = \quad & \forall y \forall x\ ((Ki(x,y) \wedge Gr(y)) \Rightarrow Fl(x)) \wedge ((Ki(x,y) \Rightarrow Fl(x)) \Rightarrow Gl(y)) \end{aligned}$$

zusammenfassen können. Mit der (etwas mühsamer, jedoch prinzipiell nicht schwierig zu beweisenden) Tautologie $[((a \wedge b) \Rightarrow c) \wedge ((a \Rightarrow c) \Rightarrow d)] \Rightarrow (b \Rightarrow d)$ folgern wir schließlich:

$$\forall y Gr(y) \Rightarrow Gl(y)$$

Das heißt, alle grünen Drachen sind glücklich!

Aufgabe 24 Interpretation von prädikatenlogischen Formeln (Lösungsvorschlag)

Wir wählen als Rechenstruktur A = **NAT** und die Sorte **m** = **nat**.

a) Wir spezifizieren die Prädikate p, q wie folgt:

$$p^{\mathbf{NAT}} : \mathbb{N}^{\perp} \to \mathbb{B}^{\perp};\ \text{für } x \in \mathbb{N} : p(x) \quad = \quad \begin{cases} \mathbf{L} & \text{, falls } x \text{ gerade} \\ \mathbf{O} & \text{, falls } x \text{ ungerade} \end{cases}$$

$$q^{\mathbf{NAT}} : \mathbb{N}^{\perp} \to \mathbb{B}^{\perp};\ \text{für } x \in \mathbb{N} : q(x) \quad = \quad \begin{cases} \mathbf{L} & \text{, falls 3 ein Teiler von } x \text{ ist} \\ \mathbf{O} & \text{, falls 3 kein Teiler von } x \text{ ist} \end{cases}$$

Dann ist sowohl die Formel t erfüllt (denn es gibt eine Zahl, die gerade ist, und es gibt eine Zahl, die durch 3 teilbar ist) als auch die Formel s (denn es gibt eine Zahl, die sowohl gerade als auch durch 3 teilbar ist). Die Formeln t und s sind also in diesem Kontext beide gültig.

b) Wir spezifizieren die Prädikate p, q wie folgt:

$$p^{\mathbf{NAT}} : \mathbb{N}^{\perp} \to \mathbb{B}^{\perp};\ \text{für } x \in \mathbb{N} : p(x) \quad = \quad \begin{cases} \mathbf{L} & \text{, falls } x \text{ gerade} \\ \mathbf{O} & \text{, falls } x \text{ ungerade} \end{cases}$$

$$q^{\mathbf{NAT}} : \mathbb{N}^{\perp} \to \mathbb{B}^{\perp};\ \text{für } x \in \mathbb{N} : q(x) \quad = \quad \begin{cases} \mathbf{L} & \text{, falls } x \text{ ungerade} \\ \mathbf{O} & \text{, falls } x \text{ gerade} \end{cases}$$

Dann ist zwar die Formel t erfüllt, aber die Formel s nicht, denn es gibt keine natürliche Zahl, die sowohl gerade als auch ungerade ist. Die Formeln t und s sind also nicht beide gültig.

Aus (b) folgt also, dass das ähnlich aussehende Formelpaar

$$(\exists\, \mathbf{m}\, x :\ p(x)) \wedge (\exists\, \mathbf{m}\, x :\ q(x)) \text{ und } \exists\, \mathbf{m}\, x : (p(x) \wedge q(x))$$

nicht äquivalent ist. Analog kann man zeigen, dass das Formelpaar

$$(\forall\,\mathbf{m}\,x:\ p(x)) \vee (\forall\,\mathbf{m}\,x:\ q(x)) \text{ und } \forall\,\mathbf{m}\,x : (p(x) \vee q(x))$$

nicht äquivalent ist.

Aufgabe 25 Ableitungen von prädikatenlogischen Formeln (Lösungsvorschlag)

a) Wir geben eine Herleitung der Formel an:

$(\Sigma, \{\forall\,\mathbf{bool}\,x : t\})$	$\vdash \forall\,\mathbf{bool}\,x : t$	wg. Regel (0)
$(\Sigma, \{\forall\,\mathbf{bool}\,x : t\})$	$\vdash t[true/x]$	wg. Regel (2), true $\neq\perp$ von Sorte **bool**
$(\Sigma, \{\forall\,\mathbf{bool}\,x : t\})$	$\vdash \exists\,\mathbf{bool}\,x : t$	wg. Regel (3), true $\neq\perp$ von Sorte **bool**
$(\Sigma, \{\})$	$\vdash (\forall\,\mathbf{bool}\,x : t) \Rightarrow (\exists\,\mathbf{bool}\,x : t)$	wg. Regel (5)

b) Die Herleitung in (a) funktioniert für beliebige Sorten **m** anstelle von **bool** nur, wenn man bei Anwendung von Regel (3) einen Term der Sorte **m** angeben kann, mit $t_1 \neq\perp$. Für eine Rechenstruktur A mit $\mathbf{m}^{\mathbf{A}} = \{\perp\}$ ist dies nicht möglich. Die Formel ist in der Tat auch für alle solchen Rechenstrukturen nicht gültig. Sie ist auch nicht herleitbar.

c) O.b.d.A. sind x, y nicht in Σ.

$(\Sigma, \{\forall\,\mathbf{m}\,x : \forall\,\mathbf{n}\,y : t\})$	$\vdash \forall\,\mathbf{m}\,x : \forall\,\mathbf{n}\,y : t$	wg. Regel (0)
$(\Sigma, \{\forall\,\mathbf{m}\,x : \forall\,\mathbf{n}\,y : t\})$	$\vdash \forall\,\mathbf{n}\,y : t$	wg. Regel (2) mit $t_1 = x$
$(\Sigma, \{\forall\,\mathbf{m}\,x : \forall\,\mathbf{n}\,y : t\})$	$\vdash t$	wg. Regel (2) mit $t_1 = y$
$(\Sigma, \{\forall\,\mathbf{m}\,x : \forall\,\mathbf{n}\,y : t\})$	$\vdash \forall\,\mathbf{m}\,x : t$	wg. Regel (1) und x ist nicht frei in $\{\forall\,\mathbf{m}\,x : \forall\,\mathbf{n}\,y : t\}$ und x ist nicht in Σ
$(\Sigma, \{\forall\,\mathbf{m}\,x : \forall\,\mathbf{n}\,y : t\})$	$\vdash \forall\,\mathbf{n}\,y : \forall\,\mathbf{m}\,x : t$	wg. Regel (1) und y ist nicht frei in $\{\forall\,\mathbf{m}\,x : \forall\,\mathbf{n}\,y : t\}$ und y ist nicht in Σ
$(\Sigma, \{\})$	$\vdash (\forall\,\mathbf{m}\,x : \forall\,\mathbf{n}\,y : t) \Rightarrow (\forall\,\mathbf{n}\,y : \forall\,\mathbf{m}\,x : t)$	wg. Regel (5)

d) Wir wählen als Rechenstruktur A = **NAT** und folgenden Spezialfall der Formel in (d) :

$$(\forall\,\mathbf{nat}\,y : \exists\,\mathbf{nat}\,x : x \geq y) \Rightarrow (\exists\,\mathbf{nat}\,x : \forall\,\mathbf{nat}\,y : x \geq y)$$

Diese gilt offensichtlich nicht in **NAT**.

e) Hier wird eine weitere Regel benötigt:
(6) Wenn $(\Sigma, H) \vdash t$ und $\Sigma = (S, F)$, $f \in F$, f nicht frei in H und t, so gilt: $(\Sigma', H) \vdash t$ für $\Sigma' = (S, F \setminus \{f\})$.

$(\Sigma, \{\exists\,\mathbf{m}\,x : \forall\,\mathbf{n}\,y : t\})$	$\vdash \exists\,\mathbf{m}\,x : \forall\,\mathbf{n}\,y : t$	wg. Regel (0)
$(\Sigma', \{\exists\,\mathbf{m}\,x : \forall\,\mathbf{n}\,y : t\})$	$\vdash \forall\,\mathbf{n}\,y : t$	wg. Regel (4) und x ist nicht frei in $\{\exists\,\mathbf{m}\,x : \forall\,\mathbf{n}\,y : t\}$ mit $\Sigma = (S, F)$, $\Sigma' = (S, F')$ und $F' = F \cup \{x\}$
$(\Sigma', \{\exists\,\mathbf{m}\,x : \forall\,\mathbf{n}\,y : t\})$	$\vdash t$	wg. Regel (2) mit $t_1 = y \neq\perp$
$(\Sigma', \{\exists\,\mathbf{m}\,x : \forall\,\mathbf{n}\,y : t\})$	$\vdash \exists\,\mathbf{m}\,x : t$	wg. Regel (3) und da $x \in F'$ gilt $t_1 = x \neq\perp$
$(\Sigma, \{\exists\,\mathbf{m}\,x : \forall\,\mathbf{n}\,y : t\})$	$\vdash \exists\,\mathbf{m}\,x : t$	wg. Regel (6)
$(\Sigma, \{\exists\,\mathbf{m}\,x : \forall\,\mathbf{n}\,y : t\})$	$\vdash \forall\,\mathbf{n}\,y : \exists\,\mathbf{m}\,x : t$	wg. Regel (1) und y ist nicht in Σ und nicht frei in $\{\exists\,\mathbf{m}\,x : \forall\,\mathbf{n}\,y : t\}$
$(\Sigma, \{\})$	$\vdash (\exists\,\mathbf{m}\,x : \forall\,\mathbf{n}\,y : t) \Rightarrow (\forall\,\mathbf{n}\,y : \exists\,\mathbf{m}\,x : t)$	wg. Regel (5)

Aufgabe 26 Die Rechenstruktur der Punkte im Koordinatensystem (Lösungsvorschlag)

a) Die gesuchte formale Spezifikation lautet:

$$\begin{array}{ll}
\mathbf{bool}^{\mathbf{K}} & = \mathbb{B}^{\perp} \\
\mathbf{int}^{\mathbf{K}} & = \mathbb{Z}^{\perp} \\
\mathbf{point}^{\mathbf{K}} & = (\mathbb{Z} \times \mathbb{Z})^{\perp} \\
\mathrm{make}^{\mathbf{K}}: & \mathbb{Z}^{\perp} \times \mathbb{Z}^{\perp} \to (\mathbb{Z} \times \mathbb{Z})^{\perp} \\
\mathrm{make}^{\mathbf{K}}(x,y) & = (x,y) \\
\mathrm{projX}^{\mathbf{K}}: & (\mathbb{Z} \times \mathbb{Z})^{\perp} \to \mathbb{Z}^{\perp} \\
\mathrm{projX}^{\mathbf{K}}(x,y) & = x \\
\mathrm{projY}^{\mathbf{K}}: & (\mathbb{Z} \times \mathbb{Z})^{\perp} \to \mathbb{Z}^{\perp} \\
\mathrm{projY}^{\mathbf{K}}(x,y) & = y \\
\mathrm{moveto}^{\mathbf{K}}: & (\mathbb{Z} \times \mathbb{Z})^{\perp} \times \mathbb{Z}^{\perp} \times \mathbb{Z}^{\perp} \to (\mathbb{Z} \times \mathbb{Z})^{\perp} \\
\mathrm{moveto}^{\mathbf{K}}((x,y),dx,dy) & = ((x+dx),(y+dy))
\end{array}$$

b) Eine Implementierung der oben spezifizierten Sorten und Funktionen könnte lauten:

```
data Point = Make Int Int

projX :: Point -> Int
projX (Make x y) = x

projY :: Point -> Int
projY (Make x y) = y

moveto :: Point -> Int -> Int -> Point
moveto (Make x y) dx dy = Make (x + dx) (y + dy)

p1 = Make 3 4
p2 = moveto p1 2 1
```