Übungen zu „Grundbegriffe der Mathematischen Logik“ Blatt 1 (für Mittwoch, 10.03.2010)

Aufgabe 1 – Addition

(a) Das folgende Programm beschreibt eine Registermaschine. „Kompilieren“ Sie es, d.h. schreiben Sie die Registermaschine als ein 9-Tupel $\mathcal{M} = (b_0, b_1, \dots, b_8)$ von Befehlen.

0: $\texttt{IF } \mathbb{R}_1 = 0 \texttt{ THEN GOTO } 4$ $\texttt{ELSE GOTO 1}$
1: $\mathbb{R}_1 := \mathbb{R}_1 \dot{-} 1$
2: $\mathbb{R}_2 := \mathbb{R}_2 + 1$
3: $\texttt{GOTO } 0$ (eigentlich: $\texttt{IF } \mathbb{R}_0 = 0 \texttt{ THEN GOTO 0 ELSE GOTO 0}$)
4: $\texttt{IF } \mathbb{R}_2 = 0 \texttt{ THEN GOTO } 8$ $\texttt{ELSE GOTO 5}$
5: $\mathbb{R}_2 := \mathbb{R}_2 \dot{-} 1$
6: $\mathbb{R}_0 := \mathbb{R}_0 + 1$
7: $\texttt{GOTO } 4$ (eigentlich: $\texttt{IF } \mathbb{R}_0 = 0 \texttt{ THEN GOTO 4 ELSE GOTO 4}$)
8: $\texttt{STOP}$

(b) Die Registermaschine aus (a) wird mit der Anfangskonfiguration $y_0^{2,7} = (0, 0, 2, 7)$ gestartet, d.h. das Programm beginnt in der Zeile 0 mit den Werten $\mathbb{R}_0 = 0$, $\mathbb{R}_1 = 2$ und $\mathbb{R}_2 = 7$. Nach wieviel Schritten erreicht das Programm den Stop-Befehl? Was steht am Ende in den drei Registern?

Schritt	Zeile	$\mathbb{R}_0$	$\mathbb{R}_1$	$\mathbb{R}_2$
		0	2	7
0	0			
1	1		1	
2	2	1		
3	3			
4	0			
5	1		0	
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
?	10	?	?	?

(c) Die Funktion $f\colon \mathbb{N}^2 \to \mathbb{N}$, die durch $f(a, b) = a + b$ gegeben ist, ist maschinenberechenbar.

Aufgabe 2 – Division

(a) Die folgende Registermaschine wird mit der Anfangskonfiguration $y_0^4 = (0, 0, 4)$ gestartet:

0: $\texttt{IF } \mathbb{R}_1 = 0 \texttt{ THEN GOTO } 10$ $\texttt{ELSE GOTO 1}$
1: $\mathbb{R}_1 := \mathbb{R}_1 \dot{-} 1$
2: $\texttt{IF } \mathbb{R}_1 = 0 \texttt{ THEN GOTO } 9$ $\texttt{ELSE GOTO 3}$
3: $\mathbb{R}_1 := \mathbb{R}_1 \dot{-} 1$
4: $\texttt{IF } \mathbb{R}_1 = 0 \texttt{ THEN GOTO } 8$ $\texttt{ELSE GOTO 5}$
5: $\mathbb{R}_1 := \mathbb{R}_1 \dot{-} 1$
6: $\mathbb{R}_0 := \mathbb{R}_0 + 1$
7: $\texttt{GOTO } 0$ (eigentlich: $\texttt{IF } \mathbb{R}_0 = 0 \texttt{ THEN GOTO 0 ELSE GOTO 0}$)
8: $\mathbb{R}_1 := \mathbb{R}_1 + 1$
9: $\mathbb{R}_1 := \mathbb{R}_1 + 1$
10: $\texttt{STOP}$

Was steht im Register $\mathbb{R}_0$, wenn sie den Stopbefehl erreicht? Was steht dann im Register $\mathbb{R}_1$?
(b) Die Funktion $f\colon \mathbb{N} \to \mathbb{N}$ mit $f(a) = [\frac{a}{2}]$ ist maschinenberechenbar. ($[]$ ist hier die Gaußklammer, d.h. $[x]$ ist die größte ganze Zahl $n \leq x$.)
(c) „Dekompilieren“ Sie die Registermaschine

$$\mathcal{M} = \big((1, -1), (1, -1), (1, 6, 3), (1, -1), (0, 1), (0, 0, 0), 0\big)$$

in ein Programm, um sie besser verstehen zu können. Was ist die durch $\mathcal{M}$ berechnete Funktion $f\colon \mathbb{N} \to \mathbb{N}$?

Aufgabe 3 – Multiplikation

(a) Finden Sie eine Registermaschine (ein Programm), die, wenn man sie mit der Anfangskonfiguration $y_0^{a,b,0} = (0, 0, a, b, 0)$ (d.h. $\mathbb{R}_0 = 0$, $\mathbb{R}_1 = a$, $\mathbb{R}_2 = b$, $\mathbb{R}_3 = 0$) startet, mit $\mathbb{R}_0 = 0$, $\mathbb{R}_1 = a$, $\mathbb{R}_2 = b$, $\mathbb{R}_3 = b$ endet.
(b) Bei Start mit der Anfangskonfiguration $y_0^{a,b}$ (d.h. $\mathbb{R}_0 = 0$, $\mathbb{R}_1 = a$, $\mathbb{R}_2 = b$, $\mathbb{R}_3 = 0$) endet das folgende Programm mit $\mathbb{R}_0 = a \cdot b$. Zur Vereinfachung benutzt es den zusätzlichen Befehl $\mathbb{R}_3 := \mathbb{R}_2$, der den Inhalt von Register $\mathbb{R}_2$ in Register $\mathbb{R}_3$ kopiert und dabei alle anderen Register außer $\mathbb{R}_3$ unverändert lässt.

0: `IF` $\mathbb{R}_1 = 0$ `THEN GOTO` 7 `ELSE GOTO 1`
1: $\mathbb{R}_1 := \mathbb{R}_1 \dot{-} 1$
2: $\mathbb{R}_3 := \mathbb{R}_2$
3: `IF` $\mathbb{R}_3 = 0$ `THEN GOTO` 0 `ELSE GOTO 4`
4: $\mathbb{R}_3 := \mathbb{R}_3 \dot{-} 1$
5: $\mathbb{R}_0 := \mathbb{R}_0 + 1$
6: `GOTO` 3 (eigentlich: `IF` $\mathbb{R}_0 = 0$ `THEN GOTO 3 ELSE GOTO 3`)
7: `STOP`

Klappt das auch, wenn man den Befehl in Zeile 4 durch 4: $\mathbb{R}_2 := \mathbb{R}_2 \dot{-} 1$ ersetzt?
(c) Angenommen, man startet das folgende Programm mit der Anfangskonfiguration $y_0^{a,b}$. Was steht am Ende im Register $\mathbb{R}_0$? Verstehen Sie, warum das Programm so seltsam notiert ist, mit zwei aufeinanderfolgenden `GOTO`-Befehlen?

0: `IF` $\mathbb{R}_1 = 0$ `THEN GOTO` 7 `ELSE GOTO 1`
1: $\mathbb{R}_1 := \mathbb{R}_1 \dot{-} 1$
2: `GOTO` 8 (eigentlich: `IF` $\mathbb{R}_0 = 0$ `THEN GOTO 8 ELSE GOTO 8`)
3: `IF` $\mathbb{R}_3 = 0$ `THEN GOTO` 0 `ELSE GOTO 4`
4: $\mathbb{R}_3 := \mathbb{R}_3 \dot{-} 1$
5: $\mathbb{R}_0 := \mathbb{R}_0 + 1$
6: `GOTO` 3 (eigentlich: `IF` $\mathbb{R}_0 = 0$ `THEN GOTO 3 ELSE GOTO 3`)
7: `STOP`
8: `IF` $\mathbb{R}_2 = 0$ `THEN GOTO` 12 `ELSE GOTO 9`
9: $\mathbb{R}_2 := \mathbb{R}_2 \dot{-} 1$
10: $\mathbb{R}_4 := \mathbb{R}_4 + 1$
11: `GOTO` 8 (eigentlich: `IF` $\mathbb{R}_0 = 0$ `THEN GOTO 8 ELSE GOTO 8`)
12: `IF` $\mathbb{R}_4 = 0$ `THEN GOTO` 17 `ELSE GOTO 13`
13: $\mathbb{R}_4 := \mathbb{R}_4 \dot{-} 1$
14: $\mathbb{R}_2 := \mathbb{R}_2 + 1$
15: $\mathbb{R}_3 := \mathbb{R}_3 + 1$
16: `GOTO` 12 (eigentlich: `IF` $\mathbb{R}_0 = 0$ `THEN GOTO 12 ELSE GOTO 12`)
17: `GOTO` 3 (eigentlich: `IF` $\mathbb{R}_0 = 0$ `THEN GOTO 3 ELSE GOTO 3`)

Version 2: Entscheidende Tippfehler in 1a und 3b korrigiert Donnerstag 23:15.

Übungen zu „Grundbegriffe der Mathematischen Logik“ Blatt 1 (für Mittwoch, 17.03.2010)

Aufgabe 4 – Größenvergleich

(a) Welche Funktion $h_1 \colon \mathbb{N}^2 \to \mathbb{N}$ wird hier berechnet?

0: IF $\mathbb{R}_2 = 0$ THEN GOTO 4
1: $\mathbb{R}_2 := \mathbb{R}_2 \dot{-} 1$
2: $\mathbb{R}_1 := \mathbb{R}_1 \dot{-} 1$
3: GOTO 0
4: IF $\mathbb{R}_1 = 0$ THEN GOTO 8
5: $\mathbb{R}_1 := \mathbb{R}_1 \dot{-} 1$
6: $\mathbb{R}_0 := \mathbb{R}_0 + 1$
7: GOTO 4
8: STOP

(b) Schreiben Sie das Programm aus (a) so um, dass es die folgende Funktion $h_2 \colon \mathbb{N}^2 \to \mathbb{N}$ berechnet:

$$h_2(x, y) = \begin{cases} 1 & \text{falls } x > y \\ 0 & \text{falls } x \leq y. \end{cases}$$

Kommen Sie mit 7 Zeilen aus?

Aufgabe 5 – Einsetzen

(a) Der Einfachheit halber arbeiten wir in dieser Aufgabe wieder mit einer erweiterten Programmiersprache, die auch die Befehle $\mathbb{R}_i := \mathbb{R}_j$ enthält. (In Aufgabe 3c haben wir gesehen, wie man ein Vorkommen von $\mathbb{R}_i := \mathbb{R}_j$ durch ein „Unterprogramm“ ersetzen kann. Im Allgemeinen braucht man für jedes Vorkommen ein solches Unterprogramm, selbst wenn die Werte von i und j immer dieselben sind.) Welche Funktion $g_1 \colon \mathbb{N}^3 \to \mathbb{N}$ wird hier berechnet?

0: IF $\mathbb{R}_2 = 0$ THEN GOTO 7
1: $\mathbb{R}_2 := \mathbb{R}_2 \dot{-} 1$
2: $\mathbb{R}_5 := \mathbb{R}_3$
3: IF $\mathbb{R}_5 = 0$ THEN GOTO 0
4: $\mathbb{R}_5 := \mathbb{R}_5 \dot{-} 1$
5: $\mathbb{R}_4 := \mathbb{R}_4 + 1$
6: GOTO 3
7: IF $\mathbb{R}_4 = 0$ THEN GOTO 11
8: $\mathbb{R}_4 := \mathbb{R}_4 \dot{-} 1$
9: $\mathbb{R}_1 := \mathbb{R}_1 + 1$
10: GOTO 7
11: IF $\mathbb{R}_1 = 0$ THEN GOTO 15
12: $\mathbb{R}_1 := \mathbb{R}_1 \dot{-} 1$
13: $\mathbb{R}_0 := \mathbb{R}_0 + 1$
14: GOTO 11
15: STOP

(b) Schreiben Sie das Programm aus (a) so um, dass es die Funktion $g_2 \colon \mathbb{N}^3 \to \mathbb{N}$ berechnet, die durch $g_2(x, y, z) = x \dot{-} (y \cdot z)$ gegeben ist. Wieviele Zeilen müssen Sie dafür verändern?

Aufgabe 6 – μ-Rekursion

(a) Seien $x, y \in \mathbb{N}$ natürliche Zahlen, wobei $y > 0$. Zeigen Sie: $\lceil x/y \rceil$ ist die kleinste Zahl $z \in \mathbb{N}$, so dass $x \mathbin{\dot{-}} (y \cdot z) = 0$ ist.[1]
(b) Zeigen Sie, dass die partielle(!) Funktion $f \colon \mathbb{N}^2 \to \mathbb{N}$, die durch $f(x, y) = \lceil x/y \rceil$ gegeben ist, primitiv rekursiv ist.
(c) Welche partielle Funktion $\mathbb{N}^2 \to \mathbb{N}$ wird von dem folgenden Programm berechnet?

$$
\begin{array}{rl}
0: & \mathbb{R}_{11} := \mathbb{R}_1 \\
1: & \mathbb{R}_{12} := \mathbb{R}_2 \\
2: & \mathbb{R}_{14} := \mathbb{R}_4 \\
\hline
3: & \texttt{IF } \mathbb{R}_2 = 0 \texttt{ THEN GOTO } 10 \\
4: & \mathbb{R}_2 := \mathbb{R}_2 \mathbin{\dot{-}} 1 \\
5: & \mathbb{R}_5 := \mathbb{R}_3 \\
6: & \texttt{IF } \mathbb{R}_5 = 0 \texttt{ THEN GOTO } 3 \\
7: & \mathbb{R}_5 := \mathbb{R}_5 \mathbin{\dot{-}} 1 \\
8: & \mathbb{R}_4 := \mathbb{R}_4 + 1 \\
9: & \texttt{GOTO } 6 \\
10: & \texttt{IF } \mathbb{R}_4 = 0 \texttt{ THEN GOTO } 14 \\
11: & \mathbb{R}_4 := \mathbb{R}_4 \mathbin{\dot{-}} 1 \\
12: & \mathbb{R}_1 := \mathbb{R}_1 \mathbin{\dot{-}} 1 \\
13: & \texttt{GOTO } 10 \\
14: & \texttt{IF } \mathbb{R}_1 = 0 \texttt{ THEN GOTO } 18 \\
15: & \mathbb{R}_1 := \mathbb{R}_1 \mathbin{\dot{-}} 1 \\
16: & \mathbb{R}_0 := \mathbb{R}_0 + 1 \\
17: & \texttt{GOTO } 14 \\
\hline
18: & \texttt{IF } \mathbb{R}_0 = 0 \texttt{ THEN GOTO } 24 \\
19: & \mathbb{R}_1 := \mathbb{R}_{11} \\
20: & \mathbb{R}_2 := \mathbb{R}_{12} \\
21: & \mathbb{R}_3 := \mathbb{R}_3 + 1 \\
22: & \mathbb{R}_4 := \mathbb{R}_{14} \\
23: & \texttt{GOTO } 3 \\
24: & \mathbb{R}_0 := \mathbb{R}_3 \\
25: & \texttt{STOP}
\end{array}
$$

Was passiert bei der Eingabe $y_0^{7,0} = (0, 0, 7, 0)$?

[1] $\lceil x/y \rceil$ ist definiert als die obere Gaußklammer von x/y, d.h. die kleinste ganze Zahl, die mindestens so groß ist wie x/y.

Übungen zu „Grundbegriffe der Mathematischen Logik“
Blatt 3 (für Mittwoch, 24.03.2010)

Aufgabe 7 – primitive Rekursivität

Zeigen Sie, dass die folgenden Funktionen $f\colon \mathbb{N}^2 \to \mathbb{N}$ primitiv rekursiv sind:

$$\begin{aligned} f(x,y) &= x+y \\ f(x,y) &= x \cdot y \\ f(x,y) &= x^y \\ f(x,y) &= x! \end{aligned}$$

Aufgabe 8 – Ackermannfunktion I

Die (vereinfachte zweistellige) Ackermannfunktion $A\colon \mathbb{N}^2 \to \mathbb{N}$ ist definiert durch

$$\begin{aligned} A(0,y) &= y+1 \\ A(x+1,0) &= A(x,1) \\ A(x+1,y+1) &= A(x, A(x+1,y)) \end{aligned}$$

Damit ist $A(x,y)$ tatsächlich für alle $x, y \in \mathbb{N}$ eindeutig definiert, aber das brauchen Sie nicht zu zeigen. Zeigen Sie statt dessen:

$$\begin{aligned} A(0,y) &= y+1 \\ A(1,y) &= y+2 \\ A(2,y) &= 2y+3 \\ A(3,y) &= 2^{y+3}-3 \end{aligned}$$

Aufgabe 9 – Ackermannfunktion II

(a) Stellen Sie die Werte von $A(x,y)$ für $x, y \leq 3$ in einer Tabelle dar. Werte mit mehr als fünf Stellen dürfen Sie auslassen.
(b) Zu jeder primitiv rekursiven Funktion $f\colon \mathbb{N}^k \to \mathbb{N}$ gibt es ein $a \in \mathbb{N}$, so dass für alle $x_1, \ldots, x_n \in \mathbb{N}$ die Ungleichung $f(x_1, \ldots, x_n) < A(a, x_1 + \cdots + x_n)$ gilt. (Das brauchen Sie nicht zu beweisen.) Folgern Sie, dass A nicht primitiv rekursiv ist.
(c) Ist A μ-rekursiv? Sie brauchen Ihre Antwort nicht zu beweisen, sollten aber eine ganz grobe Beweisidee haben. Sie dürfen dabei ausnahmsweise auch Resultate verwenden, die in der Vorlesung zwar bereits erwähnt wurden, aber noch nicht (fertig) bewiesen.

Übungen zu „Grundbegriffe der Mathematischen Logik“ Blatt 1 (für Mittwoch, 14.04.2010)

Aufgabe 10 – etwas primitiv rekursive Zahlentheorie

(a) Die Relation $x|y$ (d.h. x teilt y) ist primitiv rekursiv.
(b) Die Relation „x ist prim“ ist primitiv rekursiv.
(c) Die Funktion $p\colon \mathbb{N} \to \mathbb{N}$, die x auf die $(x+1)$-te Primzahl abbildet, ist primitiv rekursiv.

Aufgabe 11 – Wie man Tupel nicht gödelisiert

Wir schreiben p_k für die $(k+1)$-te Primzahl und betrachten die Funktion $\langle\cdot\rangle\colon \mathbb{N}^* \to \mathbb{N}$, die jedem Tupel $(x_0, \ldots, x_{n-1}) \in \mathbb{N}^*$ die Zahl $\langle x_0, x_1, \ldots, x_{n-2}, x_{n-1}\rangle = p_0^{x_0} p_1^{x_1} \ldots p_{n-2}^{x_{n-2}} p_{n-1}^{x_{n-1}}$ zuordnet.
Beweisen oder widerlegen Sie jede der folgenden Aussagen:
(a) Es gibt eine wohldefinierte zweistellige Funktion $(\cdot)_\cdot\colon \mathbb{N}^2 \to \mathbb{N}$, so dass $(x)_y = x_y$ ist falls man $x = p_0^{x_0} p_1^{x_1} \ldots p_{n-2}^{x_{n-2}} p_{n-1}^{x_{n-1}}$ schreiben kann (mit $y < n$). Diese Funktion ist primitiv rekursiv.
(b) Sei $\lg\colon \mathbb{N} \to \mathbb{N}$ die Funktion, die jedes $x \in \mathbb{N}$ auf dasjenige k abbildet, so dass p_k der größte Primteiler von x ist. Diese Funktion ist wohldefiniert und primitiv rekursiv.
(c) Für alle $n \in \mathbb{N}$ ist die Einschränkung von $\langle\cdot\rangle$ auf $\mathbb{N}^n$ primitiv rekursiv.
(d) $\langle\cdot\rangle$ definiert eine Bijektion zwischen $\mathbb{N}^*$ und $\mathbb{N}$.

Aufgabe 12 – Wie man Tupel gödelisiert

Wir betrachten nun die Funktion $\langle\cdot\rangle\colon \mathbb{N}^* \to \mathbb{N}$, die jedem Tupel $(x_0, \ldots, x_{n-1}) \in \mathbb{N}^*$ die Zahl $\langle x_0, x_1, \ldots, x_{n-2}, x_{n-1}\rangle = p_0^{x_0} p_1^{x_1} \ldots p_{n-2}^{x_{n-2}} p_{n-1}^{x_{n-1}+1} - 1$ zuordnet.
(a) Es gibt eine wohldefinierte zweistellige Funktion $(\cdot)_\cdot\colon \mathbb{N}^2 \to \mathbb{N}$, so dass $(x)_y = x_y$ ist falls man $x = \langle x_1, \ldots, x_{n-1}\rangle$ schreiben kann. Diese Funktion ist primitiv rekursiv.
(b) Es gibt eine wohldefinierte Funktion $\lg\colon \mathbb{N} \to \mathbb{N}$, so dass $\lg x = k$ ist, falls man $x = \langle x_1, \ldots, x_{n-1}\rangle$ schreiben kann. Diese Funktion ist primitiv rekursiv.
(c) Was ändert sich durch die neue Definition von $\langle\cdot\rangle$ in Aufgabe 11?

Version 2: Der Definitionsbereich von $\langle\cdot\rangle$ in Aufgabe 11 und 12 ist $\mathbb{N}^*$, nicht $\mathbb{N}^n$. 14. April 10:30.

Übungen zu „Grundbegriffe der Mathematischen Logik“ Blatt 5 (für Mittwoch, 21.04.2010)

Aufgabe 13 – Implementation von Operationen auf Zahlentupeln

Zeigen Sie im Detail, dass die folgenden Funktionen aus der Vorlesung primitiv rekursiv sind.
(a) $\mathrm{Ers}\colon \mathbb{N}^3 \to \mathbb{N}, (\langle x_0, \dots, x_i, \dots, x_{n-1}\rangle, i, y) \mapsto \langle x_0, \dots, y, \dots, x_{n-1}\rangle$ (partiell).
(b) $\mathrm{Anh}\colon \mathbb{N}^2 \to \mathbb{N}, (\langle x_0, \dots, x_{n-1}\rangle, y) \mapsto \langle x_0, \dots, x_{n-1}, y\rangle$.
(c) $\mathrm{Str}\colon \mathbb{N} \to \mathbb{N}, \langle x_0, \dots, x_{n-2}, x_{n-1}\rangle \mapsto \langle x_0, \dots, x_{n-2}\rangle$.

Aufgabe 14 – Wortfunktionen

Gegeben ein nichtleeres endliches Alphabet A, d.h. einfach eine endliche Menge. Statt für Zahlenfunktionen $f\colon \mathbb{N}^k \to \mathbb{N}$ kann man auch für Wortfunktionen $f\colon (A^*)^k \to A^*$ Maschinendefinierbarkeit definieren. Dafür bieten sich an dieser Stelle zwei Möglichkeiten an.
(a) Eine Wortfunktion $f\colon (A^*)^k \to A^*$ heiße *maschinendefinierbar (1)*, falls sie von einer Wort-Registermaschine berechnet wird. Wort-Registermaschinen sind definiert wie im Skript von Martin Ziegler, d.h. sie unterscheiden sich von den Registermaschinen aus der Vorlesung wie folgt:

- Die Register enthalten Wörter statt Zahlen.
- An Stelle des Befehls, der ein Register um 1 erhöht, gibt es für jedes Symbol des Alphabets einen Befehl, der dieses Symbol hinten an das Wort im Register anhängt.
- An Stelle des Befehls, der ein Register um 1 erniedrigt (bzw. 0 unverändert lässt), gibt es einen Befehl, der das letzte Symbol des Worts entfernt (bzw. das leere Wort unverändert lässt).

Sei $A = \{1\}$.Geben Sie eine Bijektion zwischen den maschinenberechenbaren Zahlenfunktionen $f\colon \mathbb{N}^k \to \mathbb{N}$ und den maschinenberechenbaren (1) Wortfunktionen $f\colon (A^*)^k \to A^*$ an.
(b) Gegeben ein nichtleeres endliches Alphabet $A = \{a_0, \dots, a_\ell\}$. Die Symbole a_i lassen sich durch natürliche Zahlen i codieren. Daher lassen sich auch die Wörter $a_{i_1} \dots a_{i_m}$ über A als Tupel $(i_1, \dots, i_m)$ von natürlichen Zahlen und letztlich selbst als natürliche Zahlen$\langle i_1, \dots, i_m\rangle$ codieren. Wir können daher jede Wortfunktion $f\colon (A^*)^k \to A^*$ mit einer (immer partiellen) Zahlenfunktion $f\colon \mathbb{N}^k \to \mathbb{N}$ identifizieren. Eine Wortfunktion heißt *maschinendefinierbar (2)*, falls die entsprechende Zahlenfunktion maschinenberechenbar ist.
Hängt diese Definition von der Reihenfolge ab, in der wir das Alphabet A aufzählen?

Aufgabe 15 – Mehr zu Wortfunktionen

(a) Sei $c \in A$. Wir schreiben c^m für das Wort $ccc\dots c$, in dem das Symbol c m-mal vorkommt. Überlegen Sie sich grob, dass die folgenden Wortfunktionen maschinenberechenbar (1) sind.

- $\alpha\colon A^* \to A^*, a_{i_1} \dots a_{i_m} \mapsto c^{\langle i_1, \dots, i_m\rangle}$.
- $\beta\colon A^* \to A^*, c^{\langle i_1, \dots, i_m\rangle} \mapsto a_{i_1} \dots a_{i_m}$.

(b) Folgern Sie, dass jede Wortfunktion $f\colon (A^*)^k \to A^*$, die maschinenberechenbar (2) ist, auch maschinenberechenbar (1) ist.
(c) Überlegen Sie sich ganz grob, wie man die Umkehrung zu (b) beweisen könnte.

Übungen zu „Grundbegriffe der Mathematischen Logik“ Blatt 5 (für Mittwoch, 21.04.2010)

Aufgabe 16 – Robustheit der Haltemenge

In der Vorlesung wurde die Haltemenge $H = \{x \in \mathbb{N} \mid U(x,x) \text{ ist definiert}\} \subset \mathbb{N}$ definiert und gezeigt, dass sie rekursiv aufzählbar ist aber nicht rekursiv. Man könnte alternativ auch mit $H' = \{x \in \mathbb{N} \mid U(x,0) \text{ ist definiert}\} \subset \mathbb{N}$ arbeiten, oder mit $H'' = \{(x,y) \in \mathbb{N}^2 \mid U(x,y) \text{ ist definiert}\} \subset \mathbb{N}^2$. Zeigen Sie das am Fall von H':
(a) H' ist rekursiv aufzählbar.
(b) H' ist nicht rekursiv. (Tipp: Nehmen Sie an, Sie hätten eine Registermaschine, die die charakteristische Funktion von H' berechnet und überlegen Sie sich, wie Sie damit eine Registermaschine konstruieren könnten, die die charakteristische Funktion von H berechnet.)

Aufgabe 17 – Boolesche Kombinationen von rek. aufzb. Mengen

Seien $A, B \subseteq \mathbb{N}^k$ rekursiv aufzählbare Mengen. Zeigen Sie für jede der folgenden Mengen – falls möglich –, dass sie rekursiv aufzählbar sein muss. Geben Sie andernfalls ein Gegenbeispiel an.
(a) $A \cap B$.
(b) $A \cup B$.
(c) $A \setminus B$.
(d) $B \setminus A$.

Aufgabe 18 – Mehr Manipulationen von rek. aufzb. Mengen

(a) Was ändert sich in Aufgabe 17, falls A sogar rekursiv ist?
Sei $A \subseteq \mathbb{N}^{k+1}$ eine rekursiv aufzählbare Menge. Dann sind wieder rekursiv aufzählbar:
(b) $\{x_1, \ldots, x_k) \in \mathbb{N}^k \mid \exists y \colon (x_1, \ldots, x_k, y) \in A\} \subseteq \mathbb{N}^k$.
(c) $\{x_1, \ldots, x_k) \in \mathbb{N}^k \mid \forall y < x_1 \colon (x_1, \ldots, x_k, y) \in A\} \subseteq \mathbb{N}^k$.

Übungen zu „Grundbegriffe der Mathematischen Logik“ Blatt 7 (für Mittwoch, 05.05.2010)

Aufgabe 19 – Wahrheitstabellen

Überprüfen Sie mit Hilfe von Wahrheitstabellen, ob die folgenden aussagenlogischen Formeln erfüllbar oder unerfüllbar sind. Welche sind Tautologien?
(a) $(A \to B) \vee (B \to A)$
(b) $\big(A \wedge (B \wedge C)\big) \leftrightarrow \big((A \wedge B) \wedge C\big)$
(c) $\big(A \to (B \to C)\big) \leftrightarrow \big((A \to B) \to C\big)$
(d) $(A \to B) \wedge (B \to \neg A)$.

Aufgabe 20 – Weitere Beispiele aussagenlogischer Formeln

Welche der folgenden aussagenlogischen Formeln sind erfüllbar? Welche sind Tautologien?
(a) $(A \leftrightarrow B) \wedge (B \leftrightarrow C) \wedge (C \leftrightarrow \neg A)$
(b) $(A \vee \neg B) \wedge (B \vee \neg C) \wedge (C \vee \neg D) \wedge (D \vee \neg A) \to (A \leftrightarrow C)$
(c) $(C \to B) \vee \neg(C \to A) \vee (A \wedge \neg B)$
(d) $(A \leftrightarrow B) \leftrightarrow \big((A \leftrightarrow \neg C) \wedge (\neg B \leftrightarrow C)\big)$.

Aufgabe 21 – Disjunktive Normalform

Eine aussagenlogische Formel ist in *disjunktiver Normalform* wenn sie eine Disjunktion von Konjunktionen von Atomen und negierten Atomen ist. In anderen Worten: Eine Formel in disjunktiver Normalform ist von der Gestalt $\varphi_1 \vee \varphi_2 \vee \cdots \vee \varphi_n$. Dabei ist jedes φ_i seinerseits von der Gestalt $\varphi_i = \varphi_i^1 \wedge \varphi_i^2 \wedge \ldots \varphi_i^{m_i}$, wobei jedes φ_i^j entweder ein Atom (z.B. A) oder ein negiertes Atom (z.B. $\neg A$) ist.
Zu jeder aussagenlogischen Formel φ gibt es eine aussagenlogische Formel ψ in disjunktiver Normalform, die zu φ äquivalent ist, d.h. $\varphi \leftrightarrow \psi$ ist eine Tautologie. (Hinweis: Lesen Sie die Wahrheitstabelle für φ zeilenweise. Wie erhalten Sie die Teilformeln ψ_i?)

Übungen zu „Grundbegriffe der Mathematischen Logik“
Blatt 8 (für Mittwoch, 12.05.2010)

Aufgabe 22 – Gruppen als Strukturen

Sei $L_G = \{\underline{e}, \circ, ^{-1}\}$ die Sprache der Gruppen. Wir können jede Gruppe G als L_G-Struktur $\mathfrak{G}$ auffassen: $\underline{e}^{\mathfrak{G}} \in G$ ist das neutrale Element von G, $\circ^{\mathfrak{G}} : G \times G \to G$ ist die Gruppenoperation und $^{-1\mathfrak{G}} : G \to G$ bildet jedes Element auf sein Inverses ab.
Seien nun $\mathfrak{G}$ und $\mathfrak{H}$ L_G-Strukturen. Ein L_G-Homomorphismus $h\colon \mathfrak{G} \to \mathfrak{H}$ ist eine Abbildung $h\colon G \to H$, für die gilt:

1. $h(\underline{e}^{\mathfrak{G}}) = \underline{e}^{\mathfrak{H}}$,
2. $h(x \circ^{\mathfrak{G}} y) = h(x) \circ^{\mathfrak{H}} h(y)$ für alle $x, y \in G$ und
3. $h(x^{-1\mathfrak{G}}) = h(x)^{-1\mathfrak{H}}$ für alle $x \in G$.

(a) Gruppenhomomorphismen sind wie L_G-Homomorphismen definiert, allerdings nur für Gruppen und ohne die dritte Bedingung. Zeigen Sie, dass jeder Gruppenhomomorphismus ein L_G-Homomorphismus ist.
(b) Sei $\mathfrak{G}$ die L_G-Struktur mit Grundmenge $G = \{0, 1\}$, $\underline{e}^{\mathfrak{G}} = 0$, $0 \circ^{\mathfrak{G}} 0 = 1 \circ^{\mathfrak{G}} 1 = 0$, $0 \circ^{\mathfrak{G}} 1 = 1 \circ^{\mathfrak{G}} 0 = 1$ *[korrigiert!]*, $0^{-1\mathfrak{G}} = 0$ und $1^{-1\mathfrak{G}} = 1$. Sei $\mathfrak{H}$ ebenso definiert, außer dass $1^{-1\mathfrak{H}} = 0$. (Damit ist $\mathfrak{G}$ eine Gruppe, nicht aber $\mathfrak{H}$.) Finden Sie eine Abbildung $h\colon G \to H$, die ein L_G-Homomorphismus von $\mathfrak{G}$ nach $\mathfrak{H}$ ist.
(c) Finden Sie eine Abbildung $h\colon G \to H$, die kein L_G-Homomorphismus von $\mathfrak{G}$ nach $\mathfrak{H}$ ist, weil sie nur die ersten beiden Bedingungen erfüllt.

Aufgabe 23 – Interpretation von Termen

Sei $L_N = \{\underline{0}, S, +, \cdot, <\}$ die Sprache der natürlichen Zahlen, und sei $\mathfrak{N} = \big(\mathbb{N}, (Z^{\mathbb{N}})_{Z \in L_N}\big)$ die L_N-Struktur der natürlichen Zahlen, d.h. $\underline{0}^{\mathfrak{N}} = 0 \in \mathbb{N}$, $S^{\mathfrak{N}}\colon \mathbb{N} \to \mathbb{N}$ ist die Nachfolgerfunktion, $+^{\mathfrak{N}}\colon \mathbb{N}^2 \to \mathbb{N}$ ist die Addition, $\cdot^{\mathfrak{N}}\colon \mathbb{N}^2 \to \mathbb{N}$ ist die Multiplikation und $<^{\mathfrak{N}} = \{(x, y) \in \mathbb{N}^2 \mid x < y\} \subseteq \mathbb{N}^2$. (Wir werden $<$ auf diesem Übungsblatt nicht benötigen.)
Die *Interpretation* eines L_N-Terms t ist eine Funktion $t^{\mathfrak{N}}\colon \mathbb{N}^{\mathbb{N}} \to \mathbb{N}$ (also von Folgen natürlicher Zahlen zu Folgen). Interpretationen sind rekursiv definiert wie folgt:

- $t^{\mathfrak{N}}(x_0, x_1, \dots) = x_n$ falls $t = v_n$ die n-te Variable ist;
- $t^{\mathfrak{N}}(x_0, x_1, \dots) = c^{\mathfrak{N}}$ falls $t = c$ eine Konstante ist;
- $t^{\mathfrak{N}}(x_0, x_1, \dots) = f^{\mathfrak{N}}\big(t_1^{\mathfrak{N}}(x_0, x_1, \dots), \dots, t_k^{\mathfrak{N}}(x_0, x_1, \dots)\big)$ falls $t = f t_1 \dots t_k$ aus kleineren Termen zusammengesetzt ist.

Aus der eindeutigen Lesbarkeit von Termen (Lemma 1.1) folgt, dass die Interpretation eines L_N-Terms wohldefiniert ist.
(a) Für jeden L_N-Term t gibt es ein k, so dass $t^{\mathfrak{N}}(x_0, x_1, \dots)$ nur von $x_0, \dots, x_k$ abhängt.[1]

Aufgabe 24 – Terminterpretierbare und primitiv rekursive Funktionen

(Fortsetzung von Aufgabe 23.) Eine Funktion $f\colon \mathbb{N}^k \to \mathbb{N}$ heiße *terminterpretierbar*[2], falls es einen L_N-Term t gibt, so dass $f(x_0, \dots, x_k) = t^{\mathfrak{N}}(x_0, \dots, x_k, x_{k+1}, \dots)$ ist für alle $x_0, x_1, \dots \in \mathbb{N}$.
(b) Jede terminterpretierbare Funktion $f\colon \mathbb{N}^k \to \mathbb{N}$ ist primitiv rekursiv.
(c) Jede terminterpretierbare Funktion $f\colon \mathbb{N}^k \to \mathbb{N}$ ist entweder konstant, oder es gibt ein n mit $1 \le n \le k$, so dass $f(x_1, \dots, x_k) \ge x_n$ ist für alle $x_1, \dots, x_k \in \mathbb{N} \setminus \{0\}$.
(d) Die Funktion $f(x) = x \dot{-} 1$ ist nicht terminterpretierbar.

[1] Die Aufgabe endet hier, nicht weil das so schwer zu beweisen wäre, sondern weil auch das Verstehen einer langen Aufgabenstellung Arbeit macht. In der Vorlesung werden Interpretationen übrigens etwas anders definiert.
[2] Das ist keine übliche Definition.

Übungen zu „Grundbegriffe der Mathematischen Logik“ Blatt 9 (für Mittwoch, 19.05.2010)

Aufgabe 25 – Formeln und natürliche Zahlen

Sei $L_N = \{\underline{0}, S, +, \cdot, <\}$ die Sprache der natürlichen Zahlen, und sei $\mathfrak{N} = \big(\mathbb{N}, (Z^{\mathbb{N}})_{Z \in L_N}\big)$ wie in Aufgabe 23 die L_N-Struktur der natürlichen Zahlen.
(a) Für welche Belegungen $\beta\colon \{v_0, v_1, v_2, \dots\} \to \mathbb{N}$ gilt $\mathcal{N} \models \exists v_2(v_0 + v_2 = v_1)[\beta]$?
(b) Für welche Belegungen β gilt $\mathcal{N} \models (\exists v_2(v_0 + v_2 = v_1) \leftrightarrow v_0 < v_1)[\beta]$?
(c) Finden Sie eine L_N-Formel φ, in der das Symbol $<$ nicht vorkommt und die die Eigenschaft hat, dass für jede Belegung β gilt: $\mathcal{N} \models (\varphi \leftrightarrow v_0 < v_1)[\beta]$.
(d) Für welche Belegungen β gilt Folgendes?

$$\begin{aligned}\mathcal{N} \models \forall v_1 \forall v_2 (\exists v_3 (v_3 \cdot v_0 = v_1 \cdot v_2) \to \exists v_3 (v_3 \cdot v_0 = v_1 \vee v_3 \cdot v_0 = v_2))&\\ \leftrightarrow \quad \forall v_1 \forall v_2 (v_1 \cdot v_2 = v_0 \to (v_1 = v_0 \vee v_2 = v_0))[\beta].&\end{aligned}$$

Aufgabe 26 – Ein Graph

Sei $L_g = \{R\}$ die Sprache der Graphen, d.h. R ist ein zweistelliges Relationssymbol. Wir fassen Graphen als L_g-Strukturen $\mathcal{G}$ auf, indem wir als zu Grunde liegende Menge G die Punkte des Graphen nehmen und $(x, y) \in R^{\mathcal{G}}$ setzen genau dann, wenn x und y durch eine Kante verbunden sind.
(a) Charakterisieren Sie diejenigen L_g-Strukturen $\mathcal{G}$, so dass für alle $\beta\colon \{v_0, v_1, v_2, \dots\} \to G$ gilt:

$$\mathcal{G} \models \forall v_0 \forall v_1 (R v_0 v_1 \to v_0 \neq v_1)[\beta]$$
$$\mathcal{G} \models \forall v_0 \forall v_1 (R v_0 v_1 \to R v_1 v_0)[\beta]$$
$$\begin{aligned}\mathcal{G} \models &\exists v_0 \exists v_1 \exists v_2 \exists v_3 \exists v_4 (v_0 \neq v_1 \wedge v_0 \neq v_2 \wedge v_0 \neq v_3 \wedge v_0 \neq v_4 \wedge v_1 \neq v_2 \wedge v_1 \neq v_3\\ &\wedge v_1 \neq v_4 \wedge v_2 \neq v_3 \wedge v_2 \neq v_4 \wedge v_3 \neq v_4)[\beta]\end{aligned}$$
$$\begin{aligned}\mathcal{G} \models &\neg \exists v_0 \exists v_1 \exists v_2 \exists v_3 \exists v_4 \exists v_5 (v_0 \neq v_1 \wedge v_0 \neq v_2 \wedge v_0 \neq v_3 \wedge v_0 \neq v_4 \wedge v_0 \neq v_5 \wedge v_1 \neq v_2\\ &\wedge v_1 \neq v_3 \wedge v_1 \neq v_4 \wedge v_1 \neq v_5 \wedge v_2 \neq v_3 \wedge v_2 \neq v_4 \wedge v_2 \neq v_5 \wedge v_3 \neq v_4 \wedge v_3 \neq v_5 \wedge v_4 \neq v_5)[\beta].\end{aligned}$$

(b) Finden Sie eine L_g-Struktur wie in (a), so dass außerdem eine Belegung $\beta\colon \{v_0, v_1, v_2, \dots\} \to G$ existiert, für die gilt:

$$\mathcal{G} \models R v_0 v_1 \wedge R v_1 v_2 \wedge R v_2 v_3 \wedge R v_3 v_4 \wedge R v_4 v_5 \wedge R v_5 v_6 \wedge R v_6 v_7 \wedge R v_7 v_8 [\beta]$$
$$\mathcal{G} \models \neg(v_i = v_j \wedge v_{i+1} = v_{j+1}) \wedge \neg(v_i = v_{j+1} \wedge v_{i+1} = v_j)[\beta] \quad \textit{(für alle } 0 \leq i < j \leq 8\textit{)}$$
$$\begin{aligned}\mathcal{G} \models &\forall v_9 \forall v_{10} \big(R v_9 v_{10} \to\\ &(v_9 = v_0 \wedge v_{10} = v_1) \vee (v_9 = v_1 \wedge v_{10} = v_2) \vee (v_9 = v_2 \wedge v_{10} = v_3) \vee (v_9 = v_3 \wedge v_{10} = v_4)\\ &\vee (v_9 = v_4 \wedge v_{10} = v_5) \vee (v_9 = v_5 \wedge v_{10} = v_6) \vee (v_9 = v_6 \wedge v_{10} = v_7) \vee (v_9 = v_7 \wedge v_{10} = v_8)\\ &\vee (v_9 = v_1 \wedge v_{10} = v_0) \vee (v_9 = v_2 \wedge v_{10} = v_1) \vee (v_9 = v_3 \wedge v_{10} = v_2) \vee (v_9 = v_4 \wedge v_{10} = v_3)\\ &\vee (v_9 = v_5 \wedge v_{10} = v_4) \vee (v_9 = v_6 \wedge v_{10} = v_5) \vee (v_9 = v_7 \wedge v_{10} = v_6) \vee (v_9 = v_8 \wedge v_{10} = v_7))\big)[\beta].\end{aligned}$$

(c) Erfüllt die von Ihnen gefundene Belegung aus (b) die Bedingung $\mathcal{G} \models v_0 = v_8[\beta]$?
(d) Zeichnen Sie den Graphen $\mathcal{G}$. Wie viele Worte hat das Gedicht, mit dem man ihn im deutschsprachigen Raum in Verbindung bringt?

Aufgabe 27 – Allgemeingültigkeit

Finden Sie eine L_g-Formel φ, eine L_g-Struktur $\mathcal{G}$ und eine Belegung $\beta\colon \{v_0, v_1, v_2, \dots\} \to G$, so dass Folgendes *nicht* gilt:

$$\mathcal{G} \models \exists v_1 (R v_0 v_1) \wedge \forall v_1 \forall v_2 ((R v_0 v_1 \wedge R v_0 v_2) \to v_1 = v_2) \quad \to \big(\exists v_1 (R v_0 v_1 \wedge \varphi) \leftrightarrow \forall v_1 (R v_0 v_1 \to \varphi)\big).$$

Falls das nicht gehen sollte, beweisen Sie, dass es nicht geht.

Übungen zu „Grundbegriffe der Mathematischen Logik“
Blatt 10 (für Mittwoch, 26.05.2010)

Aufgabe 28 – Semantisches Folgern

Seien φ und ψ L-Aussagen. $\models \varphi$ bedeutet, dass für jede L-Struktur M gilt: $M \models \varphi$. $\varphi \models \psi$ bedeutet, dass für alle L-Strukturen M gilt: Falls $M \models \varphi$, so auch $M \models \psi$.
(a) Zeigen Sie: $\varphi \models \psi$ gilt genau dann, wenn $\models \varphi \to \psi$ gilt, d.h. $\models \psi \vee \neg\varphi$.
Seien außerdem Φ und Ψ Mengen von L-Aussagen. $M \models \Phi$ bedeutet, dass für alle $\varphi \in \Phi$ gilt: $M \models \varphi$. $\Phi \models \Psi$ bedeutet, dass für alle M gilt: Falls $M \models \Phi$, so auch $M \models \Psi$.
(b) Zeigen Sie: $\Phi \models \Psi$ gilt genau dann, wenn für alle L-Strukturen M und alle $\psi \in \Psi$ gilt: Falls $M \models \Phi$, so auch $M \models \psi$.
(c) Zeigen Sie: $\{\varphi_1, \varphi_2, \dots, \varphi_n\} \models \psi$ gilt genau dann, wenn $\models \varphi_1 \wedge \varphi_2 \wedge \dots \wedge \varphi_n \to \psi$ gilt.
In den folgenden Beispielen arbeiten wir mit einer Signatur, die ein Konstantensymbol c, ein zweistelliges Funktionssymbol f und ein zweistelliges Relationssymbol $<$ enthält.
(d) Gilt $\models \forall x \forall y \forall z (x < y \wedge y < z \to x < z)$?
(e) Gilt $\forall x \exists y (f(x,y) < x) \models \forall x \exists y (y < x)$?
(f) Gilt $\forall x \exists y (f(x,y) = c) \models \exists y \forall x (f(x,y) = c)$?
(g) Gilt $\exists y \forall x (f(x,y) = c) \models \forall x \exists y (f(x,y) = c)$?

Aufgabe 29 – Eine Art Mengenlehre

$\mathcal{P}(X)$ bezeichnet im Folgenden die Potenzmenge (Menge aller Teilmengen) von X. Sei $U_0 = \mathbb{N}$, $U_1 = U_0 \cup \mathcal{P}(U_0)$, $U_2 = U_1 \cup \mathcal{P}(U_1)$ usw. und schließlich $U = \bigcup_{n \in \mathbb{N}} U_n$. Sei $\mathfrak{U} = (U, \in)$ die Struktur, deren zugrunde liegende Menge U ist und in der das zweistellige Symbol $\in$ als die natürliche Relation „ist Element von“ interpretiert wird. Beweisen oder widerlegen Sie jede der folgenden Behauptungen:
(a) $\forall x \forall y \big(\forall z (z \in x \leftrightarrow z \in y) \to x = y\big)$.
(b) $\forall x \forall y \exists z \forall u (u \in z \leftrightarrow u = x \vee u = y)$.
(c) $\exists x \Big(\exists y (y \in x) \wedge \forall y \Big(y \in x \to \exists z \big(z \in x \wedge \forall u (u \in z \leftrightarrow u \in y \vee u = y)\big)\Big)\Big)$.[1]

Aufgabe 30 – Axiomatisierung in Logik der 1. Stufe

Es folgt eine Liste von Eigenschaften, die eine Struktur in der jeweils offensichtlich gemeinten Sprache haben kann oder auch nicht. Axiomatisieren Sie 5 davon in der Logik 1. Stufe, d.h. geben Sie jeweils eine Menge von L-Aussagen („Axiomen“) für das entsprechende L an, so dass eine L-Struktur genau dann die genannte Eigenschaft hat, wenn sie alle Axiome erfüllt. Versuchen Sie, soweit möglich, mit einem einzigen Axiom auszukommen.
(a) $<$ stellt eine dichte lineare Ordnung ohne Endpunkte dar. „Dicht“ bedeutet, dass zwischen je zwei Punkten immer noch ein weiterer liegt.
(b) R ist die Kantenrelation[2] eines zusammenhängenden Graphen.
(c) R ist die Kantenrelation eines Graphen, in dem jedes Paar von Punkten durch einen Weg der Länge höchstens 3 verbunden ist.
(d) R ist die Kantenrelation eines Walds, d.h. eines Graphen ohne Zykel (kreisförmige Wege).
(e) R ist die Kantenrelation eines Baums, d.h. eines zusammenhängenden Graphen ohne Zykel.
(f) (Leere Sprache[3]) Es gibt unendlich viele Elemente.
(g) (Leere Sprache) Es gibt nur endlich viele Elemente.
(h) $(G, \cdot)$ ist eine Gruppe.
(i) $(G, \cdot)$ ist eine zyklische Gruppe, d.h. es gibt ein („erzeugendes“) Element x, so dass jedes Element von G ist von der Form x^k für ein $k \in \mathbb{Z}$.

---

[1] Aufgabe 29c korrigiert 25. Mai 10:20.
[2] D.h. wie in Aufgabe 26.
[3] D.h. es gibt keine Konstanten, Funktionssymbole oder Relationssymbole.

Übungen zu „Grundbegriffe der Mathematischen Logik“ Blatt 11 (für Mittwoch, 04.06.2010)

Aufgabe 31 – Zählen

In dieser Aufgabe betrachten wir $L = \{f\}$, wobei f ein einstelliges Funktionssymbol ist.
(a) Finden Sie eine L-Aussage φ_{inj}, so dass eine L-Struktur M genau dann $M \models \varphi_{\text{inj}}$ erfüllt, wenn die Abbildung $f^M : M \to M$ injektiv ist. Finden Sie eine L-Aussage φ_{surj}, so dass eine L-Struktur M genau dann $M \models \varphi_{\text{surj}}$ erfüllt, wenn die Abbildung f^M surjektiv ist.
(b) Finden Sie eine erfüllbare L-Aussage φ_*, so dass jede L-Struktur M, für die $M \models \varphi_*$ gilt, unendlich ist. (Hinweis: Benutzen Sie (a).)
(c) Für drei der folgenden vier Eigenschaften von L-Strukturen gibt es jeweils eine Menge Φ von L-Aussagen, so dass eine L-Struktur M die jeweilige Eigenschaft genau dann hat, wenn $M \models \Phi$ gilt. Finden Sie die drei Eigenschaften, für die es eine solche Menge Φ gibt, und geben Sie passende Mengen Φ an.

- M hat höchstens 4 Elemente.
- M hat mindestens 4 Elemente.
- M hat nur endlich viele Elemente.
- M hat unendlich viele Elemente.

Aufgabe 32 – Zusammenhang von Graphen

In dieser Aufgabe betrachten wir $L = \{R\}$, wobei R ein zweistelliges Relationssymbol ist. Wir sehen L-Strukturen als Graphen an. Diese Graphen sind im Allgemeinen gerichtet und können Self-Loops (Kanten von einem Punkt zu dem Punkt selbst) haben. (Sie können endlich oder unendlich sein.)
(a) Finden Sie eine L-Aussage φ_Δ, so dass eine L-Struktur G genau dann $G \models \varphi_\Delta$ erfüllt, wenn es einen (gerichteten) Zykel der Länge 3 gibt, d.h. einen Pfad der Länge 3 mit Anfangspunkt = Endpunkt. Finden Sie eine Menge Φ von L-Aussagen, so dass eine L-Struktur G genau dann $G \models \Phi$ erfüllt, wenn es in G gar keine (gerichteten) Zykel gibt.
(b) Für drei der folgenden vier Eigenschaften von L-Strukturen gibt es jeweils eine Menge Φ von L-Aussagen, so dass eine L-Struktur G die jeweilige Eigenschaft genau dann hat, wenn $G \models \Phi$ gilt. Finden Sie die drei Eigenschaften, für die es eine solche Menge Φ gibt, und geben Sie passende Mengen Φ an.

- Je zwei Elemente von G sind durch einen (gerichteten) Pfad der Länge höchstens 3 verbunden.
- Es gibt zwei Elemente von G, deren Abstand mindestens 3 ist.
- G ist zusammenhängend, d.h. zwischen je zwei Elementen gibt es immer einen (gerichteten) Pfad.
- G ist nicht zusammenhängend.

Aufgabe 33 – Körpercharakteristik

In dieser Aufgabe betrachten wir $L = \{0, +, -, 1, \cdot\}$, die Signatur der Ringe und Körper.
(a) Finden Sie eine Menge Φ_{Kp} von L-Aussagen, so dass eine L-Struktur K genau dann $K \models \Phi_{\mathrm{Kp}}$ erfüllt, wenn K ein Körper ist.
(b) Ein Körper hat die Charakteristik p, falls p die kleinste Zahl ≥ 1 ist, so dass $1 + 1 + \cdots + 1 = 0$ ist, wobei die Summe p mal die 1 enthält. Falls es ein solches p gibt, dann ist es eine Primzahl. Andernfalls setzt man $p = 0$ (obwohl $p = \infty$ eigentlich sinnvoller wäre).
Für drei der folgenden vier Eigenschaften von L-Strukturen gibt es jeweils eine Menge Φ von L-Aussagen, so dass eine L-Struktur K die jeweilige Eigenschaft genau dann hat, wenn $K \models \Phi$ gilt. Finden Sie die drei Eigenschaften, für die es eine solche Menge Φ gibt, und geben Sie passende Mengen Φ an.

- K ist ein Körper der Charakteristik 2, 3, 5 oder 7.
- K ist ein Körper der Charakteristik 7 oder größer oder der Charakteristik 0 [letzteres am Sonntag nachträglich hinzugefügt, damit die Aufgabe lösbar ist].
- K ist ein Körper mit einer Charakteristik $\neq 0$.
- K ist ein Körper der Charakteristik 0.

Übungen zu „Grundbegriffe der Mathematischen Logik“
Blatt 12 (für Mittwoch, 09.06.2010)

Aufgabe 34 – Syntax und Semantik auseinanderhalten

In dieser Aufgabe betrachten wir $L = \{f, a\}$, wobei f ein zweistelliges Funktionssymbol ist und a ein Konstantensymbol.
(a) Finden Sie eine Menge Φ von erfüllbaren L-Aussagen, so dass Φ nicht erfüllbar ist.
(b) Finden Sie eine unendliche L-Struktur M, so dass $M \models \neg\exists x\forall y(f(x,y) = a)$ gilt.
(c) Finden Sie für jede natürliche Zahl n eine L-Struktur mit genau n Elementen, so dass $M \models \neg\exists x\forall y(f(x,y) = a)$ gilt.

Aufgabe 35 – Syntax und Semantik auseinanderhalten

In dieser Aufgabe betrachten wir $L = \{+, -, 0, <, f\}$, wobei $+$ und $-$ zweistellige Funktionssymbole sind, f ein einstelliges Funktionssymbol, 0 ein Konstantensymbol und $<$ ein Relationssymbol. Wir schreiben $+$, $-$ und $<$ in Infixnotation, also bspw. $x + y$ statt $+(x, y)$. Sei φ die folgende L-Aussage:

$$\forall x\forall u\Big(0 < u \to \exists v\Big(0 < v \wedge \big((x - z < v \wedge z - x < v) \to (f(x) - f(z) < u \wedge f(z) - f(x) < u)\big)\Big)\Big).$$

(a) Finden Sie eine L-Struktur M, deren Grundmenge die Menge der reellen Zahlen ist, für die $f^M : \mathbb{R} \to \mathbb{R}$ eine stetige Funktion ist, und so dass $M \models \varphi$ gilt.
(b) Finden Sie eine L-Struktur M, deren Grundmenge die Menge der reellen Zahlen ist, für die $M \models \varphi$ gilt, aber $f^M(x) = 1$ falls x rational ist und $f^M(x) = 0$ [korrigiert von vorher 1] sonst.

Aufgabe 36 – Rätselhaftes

In dieser Aufgabe betrachten wir $L_{\mathrm{sdk}} = \{N_1, N_2, \ldots, N_9, Z_1, Z_2, \ldots, Z_9, S_1, S_2, \ldots, S_9\}$, wobei alle 27 Symbole einstellige Relationssymbole sind. Sei Γ_{sdk} die Menge bestehend aus den folgenden L-Aussagen:[1]

$$\begin{gathered}
\forall x(Z_1(x) \vee Z_2(x) \vee \ldots \vee Z_9(x)),\\
\forall x(S_1(x) \vee S_2(x) \vee \ldots \vee S_9(x)),\\
\forall x(N_1(x) \vee N_2(x) \vee \ldots \vee N_9(x)),\\
\neg\exists x\big((N_i(x) \wedge N_j(x)) \vee (Z_i(x) \wedge Z_j(x)) \vee (S_i(x) \wedge S_j(x))\big) \quad \textit{(für alle } i, j \in \{1, \ldots, 9\} \textit{ mit } i \neq j\textit{)},\\
\forall x\forall y\big((Z_i(x) \wedge Z_i(y) \wedge S_j(x) \wedge S_j(y)) \to x = y\big) \quad \textit{(für alle } i, j \in \{1, \ldots, 9\}\textit{)},\\
\forall x\forall y\big((x \neq y \wedge Z_i(x) \wedge Z_i(y)) \to \neg(N_j(x) \wedge N_j(y))\big) \quad \textit{(für alle } i, j \in \{1, \ldots, 9\}\textit{)},\\
\forall x\forall y\big((x \neq y \wedge S_i(x) \wedge S_i(y)) \to \neg(N_j(x) \wedge N_j(y))\big) \quad \textit{(für alle } i, j \in \{1, \ldots, 9\}\textit{)},\\
\begin{aligned}
\forall x\forall y\big(\big(x \neq y \wedge (Z_{3i+1}(x) \vee Z_{3i+2}(x) \vee Z_{3i+3}(x)) \wedge (S_{3j+1}(x) \vee S_{3j+2}(x) \vee S_{3j+3}(x))&\\
\wedge(Z_{3i+1}(y) \vee Z_{3i+2}(y) \vee Z_{3i+3}(y)) \wedge (S_{3j+1}(y) \vee S_{3j+2}(y) \vee S_{3j+3}(y))\big)&\\
\to \neg(N_k(x) \wedge N_k(y))\big) \quad \textit{(für alle } i, j \in \{0, 1, 2\} \textit{ und } k \in \{0, \ldots, 9\}\textit{)}&
\end{aligned}\\
\exists x_1\exists x_2 \ldots \exists x_{81}(x_1 \neq x_2 \wedge x_1 \neq x_3 \wedge \ldots \wedge x_{80} \neq x_{81}).
\end{gathered}$$

Hinweis: Es ist sehr hilfreich, an eine geeignete 9×9-Matrix zu denken. Die N_i geben die Einträge in der Matrix an.
(a) Sei $M \models \Gamma_{\mathrm{sdk}}$. Zeigen Sie, dass es für jedes Paar $i, j \in \{1, \ldots, 9\}$ genau ein Element $a_{ij} \in M$ gibt, so dass $a_{ij} \in Z_i^M \cap S_j^M$ ist.
(b) Sei $M \models \Gamma_{\mathrm{sdk}}$. Zeigen Sie, dass M genau 81 Elemente hat.
(c) Für $i, j, k \in \{1, \ldots, 9\}$ sei φ_{ijk} die L_{sdk}-Aussage $\exists x(Z_i(x) \wedge S_j(x) \wedge N_k(x))$. Finden Sie die (bis auf Isomorphie eindeutig bestimmte) L_{sdk}-Struktur M, die

$$\begin{aligned}
M \models \Gamma_{\mathrm{sdk}} \cup \{&\varphi_{123}, \varphi_{241}, \varphi_{259}, \varphi_{265}, \varphi_{329}, \varphi_{338}, \varphi_{386}, \varphi_{418}, \varphi_{456}, \varphi_{514}, \varphi_{563}, \varphi_{591}, \varphi_{652},\\
&\varphi_{726}, \varphi_{772}, \varphi_{788}, \varphi_{844}, \varphi_{851}, \varphi_{869}, \varphi_{895}, \varphi_{987}\}
\end{aligned}$$

erfüllt.

[1] Vorletzte Formel (zweimal!) korrigiert am 8. Juni.

Übungen zu „Grundbegriffe der Mathematischen Logik“ Hausaufgabenteil der Prüfung – mit Lösungen

Teil I – Rekursionstheorie

Aufgabe 1. Welche (totale) Funktion $f\colon \mathbb{N}^2 \to \mathbb{N}$ wird von dieser Registermaschine berechnet?

0: `IF` $\mathbb{R}_2 = 0$ `THEN GOTO` 1 `ELSE GOTO` 2	4: $\mathbb{R}_0 := \mathbb{R}_0 + 1$
1: `IF` $\mathbb{R}_1 = 0$ `THEN GOTO` 6	5: `GOTO` 0
2: $\mathbb{R}_1 := \mathbb{R}_1 \dot{-} 1$	6: `STOP`
3: $\mathbb{R}_2 := \mathbb{R}_2 \dot{-} 1$	

Lösung: $f(x, y) = \max(x, y)$.

Aufgabe 2. Der Programmierer hat sich bei der Eingabe des obigen Programms in Zeile 0 vertippt. Das Resultat ist die folgende Registermaschine $\mathcal{M}$:

0: `IF` $\mathbb{R}_2 = 0$ `THEN GOTO` 1 `ELSE GOTO` 4	4: $\mathbb{R}_0 := \mathbb{R}_0 + 1$
1: `IF` $\mathbb{R}_1 = 0$ `THEN GOTO` 6	5: `GOTO` 0
2: $\mathbb{R}_1 := \mathbb{R}_1 \dot{-} 1$	6: `STOP`
3: $\mathbb{R}_2 := \mathbb{R}_2 \dot{-} 1$	

wahr (W) oder falsch (F)?	Antwort
$\mathcal{M}$ stoppt genau dann, wenn am Anfang $\mathbb{R}_2 = 0$ ist.	**W**
$\mathcal{M}$ berechnet die Identität $\mathrm{id}\colon \mathbb{N} \to \mathbb{N}$, d.h. $\mathrm{id}(n) = n$.	**W**
$\mathcal{M}$ berechnet eine totale Funktion von $\mathbb{N}^2$ nach $\mathbb{N}$.	**F**
$\mathcal{M}$ berechnet eine partielle Funktion g von $\mathbb{N}^3$ nach $\mathbb{N}$, so dass $g(a, b, c) = \max(a, b)$ wann immer $g(a, b, c)$ definiert ist.	**W**

Kommentar: Wenn am Anfang $\mathbb{R}_2 = 0$ ist, dann bleibt das auch so, und die fehlerhafte Stelle wird nie erreicht. Andernfalls gerät das Programm in eine Schleife, in der $\mathbb{R}_2$ nicht verändert wird. Das Programm ist „offensichtlich“ für eine zweistellige Funktion geschrieben, aber nach den formalen Definitionen kann man es auch als k-stellig auffassen für beliebiges k. Wie immer wird dann am Anfang $\mathbb{R}_i = 0$ gesetzt für alle $i > k$. Es gilt $\max(n, 0) = n = \mathrm{id}(n)$.

Aufgabe 3. Hier sollten Sie insgesamt 12 Antworten geben.

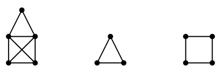
wahr (W), falsch (F) oder sinnlos (S)?	prim. rek.	rek.	rek. aufzb.
Jedes Polynom $p\colon \mathbb{N}^k \to \mathbb{N}$ ist …	**W**	**W**	**S**
Wenn $g\colon \mathbb{N}^3 \to \mathbb{N}$ primitiv rekursiv ist, $f(x, 0) = 2x$ und $f(x, y+1) = g(y, f(x, y), x)$, dann ist f …	**W**	**W**	**S**
Die Ackermannfunktion $A\colon \mathbb{N}^2 \to \mathbb{N}$ ist …	**F**	**W**	**S**
Die Menge $\{(x, y, z) \in \mathbb{N}^3 \mid y \neq 2^x\}$ ist …	**W**	**W**	**W**

„Sinnlos“ bedeutet, dass die Behauptung „nicht einmal falsch“ ist. Beispiel: „Jede reelle Zahl ist gleichschenklig.“
Kommentar: Eine Menge ist laut Definition in der Vorlesung primitiv rekursiv, wenn ihre charakteristische Funktion es ist.

Teil II – Prädikatenlogik

Sei $L = \{R\}$ die Sprache der Graphen, wobei R ein zweistelliges Relationssymbol ist. Wir fassen (gerichtete) Graphen (allenfalls mit Schleifen, allenfalls unendlich) als L-Strukturen $\mathcal{G} = (G, R^{\mathcal{G}})$ auf. Hierbei ist G die Menge der Knoten und $(a, b) \in R^{\mathcal{G}}$ gilt genau dann, wenn es eine Kante von a nach b gibt.

Aufgabe 4. Wir betrachten die folgenden vier Graphen:
$\mathcal{G}_1 = (\mathbb{N}, R^{\mathcal{G}})$ mit $R^{\mathcal{G}} = \{(a, b) \in \mathbb{N}^2 \mid b = a + 1\}$.



$\mathcal{G}_2$ $\quad$ $\mathcal{G}_3$ $\quad$ $\mathcal{G}_4$

($\mathcal{G}_1$ ist ein gerichteter Graph. Die gezeichneten Kanten in den Abbildungen sind ungerichtet, d.h. jede von ihnen steht für zwei entgegengesetzt gerichtete Kanten.)

wahr (W) oder falsch (F)?	$i = 1$	$i = 2$	$i = 3$	$i = 4$
$\mathcal{G}_i \models \forall xyz(R(x, y) \land R(y, z) \to R(x, z))$ gilt für ...	**F**	**F**	**F**	**F**
$\mathcal{G}_i \models \exists wxyz(R(w, x) \land R(x, y) \land R(y, z) \land R(z, w))$ gilt für ...	**F**	**W**	**W**	**W**
$\mathcal{G}_i \models \exists wxyz(w \neq x \land w \neq y \land w \neq z \land x \neq y \land x \neq z \land y \neq z)$ gilt für ...	**W**	**W**	**F**	**W**

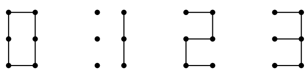
Kommentar: Jeder der Graphen hat eine Kante aber keine Schleifen. Daher ist die erste Formel (Transitivität) für keinen der Graphen erfüllt. Für die zweite Formel genügt schon die Existenz einer Kante, weil dann sogar $\exists wxyz(R(w, x) \land R(x, y) \land R(y, z) \land R(z, w) \land w = y \land x = z)$ erfüllt ist.

Aufgabe 5. Zeichnen Sie zwei nicht isomorphe Graphen, die beide die folgende L-Aussage erfüllen:

$$\mathcal{G}_i \models \forall x(\neg R(x, x)) \quad \land \quad \forall xy(R(x, y) \to R(y, x)) \quad \land$$
$$\exists wxyz\big(R(w, x) \land R(x, y) \land R(y, z) \land R(z, w) \land \forall u(u = w \lor u = x \lor u = y \lor u = z)\big).$$

Kommentar: Gerichtet, ohne Schleifen, mit einem „Weg“ der Länge 4, der auch Überschneidungen erlaubt, und keinen Knoten von außerhalb des „Wegs“. Die ersten drei Lösungen enthalten jeweils nur die Kanten des „Wegs“. Die Liste ist vollständig.

Aufgabe 6. Geben Sie für jeden Graphen $\mathcal{G}_i$ eine Formel φ_i an, so dass $\mathcal{G}_i \models \varphi_j \iff i = j$.



$\mathcal{G}_0$ $\quad$ $\mathcal{G}_1$ $\quad$ $\mathcal{G}_2$ $\quad$ $\mathcal{G}_3$

$$\begin{aligned}
\varphi_0 &\equiv \forall x \exists yz(y \neq z \land R(x, y) \land R(x, z)) \\
\varphi_1 &\equiv \exists x \forall y(\neg R(x, y)) \\
\varphi_2 &\equiv \exists x_1 \ldots x_6(x_1 \neq x_2 \land x_1 \neq x_3 \land \ldots \land x_5 \neq x_6 \land R(x_1, x_2) \land R(x_2, x_3) \land \ldots \land R(x_5, x_6)) \\
\varphi_3 &\equiv \exists uxyz(x \neq y \land x \neq z \land y \neq z \land R(u, x) \land R(u, y) \land R(u, z))
\end{aligned}$$

Kommentar: Die Formeln sind natürlich nur (möglichst einfache) Beispiele. φ_0: Jeder Knoten hat mindestens Grad 2. φ_1: Es gibt mindestens einen isolierten Knoten. φ_2: Es gibt mindestens einen (überschneidungsfreien!) Pfad der Länge 5. φ_3: Es gibt mindestens einen Knoten vom Grad 3. Formeln der Art $\varphi_0 \equiv \neg\varphi_1 \land \neg\varphi_2 \land \neg\varphi_3$ sind eine Art Joker, den man für den schwersten Fall einsetzen kann.

Übungen zu „Grundbegriffe der Mathematischen Logik“ Ungerade Seriennummern, mit Lösungen

Teil I – Rekursionstheorie

Aufgabe 1. Welche (totale) Funktion $f\colon \mathbb{N}^2 \to \mathbb{N}$ wird von dieser Registermaschine berechnet?

0: `IF` $\mathbb{R}_1 = 0$ `THEN GOTO` 6	4: $\mathbb{R}_0 := \mathbb{R}_0 + 1$
1: `IF` $\mathbb{R}_2 = 0$ `THEN GOTO` 6	5: `GOTO` 0
2: $\mathbb{R}_1 := \mathbb{R}_1 \dot{-} 1$	6: `STOP`
3: $\mathbb{R}_2 := \mathbb{R}_2 \dot{-} 1$	

Lösung: $f(x, y) = \min(x, y)$.

Aufgabe 2. Das obige Programm wurde aus einem alten Buch eingescannt. Dabei kam es zu einem Fehler in Zeile 0. Das Resultat ist die folgende Registermaschine $\mathcal{M}$:

0: `IF` $\mathbb{R}_1 = 0$ `THEN GOTO` 0	4: $\mathbb{R}_0 := \mathbb{R}_0 + 1$
1: `IF` $\mathbb{R}_2 = 0$ `THEN GOTO` 6	5: `GOTO` 0
2: $\mathbb{R}_1 := \mathbb{R}_1 \dot{-} 1$	6: `STOP`
3: $\mathbb{R}_2 := \mathbb{R}_2 \dot{-} 1$	

wahr (W) oder falsch (F)?	Antwort
$\mathcal{M}$ berechnet eine totale Funktion von $\mathbb{N}^2$ nach $\mathbb{N}$.	**F**
$\mathcal{M}$ stoppt genau dann, wenn am Anfang $\mathbb{R}_1 = 0$ ist.	**F**
$\mathcal{M}$ berechnet eine partielle Funktion h von $\mathbb{N}^3$ nach $\mathbb{N}$, so dass $h(a, b, c)$ genau dann definiert ist, wenn $a > b$.	**W**
$\mathcal{M}$ berechnet eine partielle Funktion g von $\mathbb{N}^2$ nach $\mathbb{N}$, so dass $g(a, b) = \min(a, b)$ wann immer $g(a, b)$ definiert ist.	**W**

Kommentar: Das Programm stoppt genau dann, wenn der Fehler (sofortige Endlosschleife) nicht erreicht wird. Der Fehler wird bspw. erreicht wenn am Anfang $\mathbb{R}_1 = 0$ ist, aber auch, wenn $\mathbb{R}_1$ mindestens so früh wie $\mathbb{R}_2$ Null wird, d.h. falls am Anfang $\mathbb{R}_1 \leq \mathbb{R}_2$ gilt. Sonst wie Hausaufgabe.

Aufgabe 3. Hier sollten Sie insgesamt 12 Antworten geben.

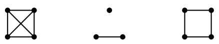
wahr (W), falsch (F) oder sinnlos (S)?	prim. rek.	rek.	rek. aufzb.
Jede stetig differenzierbare Funktion $f\colon \mathbb{R}^k \to \mathbb{R}$ ist ...	**S**	**S**	**S**
Wenn $f\colon \mathbb{N}^3 \to \mathbb{N}$ rekursiv ist, $g(0, y) = f(y, 0, y)$ und $g(x+1, y) = f(x, y, g(x, y))$, dann ist g immer ...	**F**	**W**	**S**
Die totale Funktion $h\colon \mathbb{N}^2 \to \mathbb{N}$, definiert durch $h(0, y) = y! + y + 7$, $h(x+1, 0) = h(x, 1)$, $h(x+1, y+1) = h(x, h(x+1, y))$ ist ...	**F**	**W**	**S**
Jede Menge der Form $A \setminus B$ mit A primitiv rekursiv und B rekursiv aufzählbar ist ...	**F**	**F**	**F**

Kommentar: Zeile 1 ist etwas verunglückt, weil Jakob Kellner in der Vorlesung das Problem der Erweiterung der Definitionen auf reelle Funktionen kurz angesprochen hatte; es gibt aber keine allgemein bekannten Standarddefinitionen. In Zeile 2 steht eine primitive Rekursion, wie in der Hausaufgabe. Die rekursiven Funktionen sind unter primitiver Rekursion abgeschlossen, aber wenn man mit einer Funktion anfängt, die selbst nicht einmal primitiv rekursiv ist, kann man natürlich im Allgemeinen keine primitive Rekursivität erwarten. Die Funktion in Zeile 3 ist wie die Ackermannfunktion definiert, nur mit größeren Startwerten. Insbesonders ist das keine primitive Rekursion! Nach der Church-Turing-These muss die Funktion natürlich rekursiv sein. Aber es gibt keinen Grund anzunehmen, dass sie primitiv rekursiv ist, und weil sie schneller wächst als die Ackermannfunktion ist sie das auch nicht. Zeile 4: Wenn B rekursiv aufzählbar aber nicht rekursiv ist, dann ist nicht einmal $\mathbb{N} \setminus B$ rekursiv aufzählbar.

Teil II – Prädikatenlogik

Sei $L = \{R\}$ die Sprache der Graphen, wobei R ein zweistelliges Relationssymbol ist. Wir fassen (gerichtete) Graphen (allenfalls mit Schleifen, allenfalls unendlich) als L-Strukturen $\mathcal{G} = (G, R^{\mathcal{G}})$ auf. Hierbei ist G die Menge der Knoten und $(a, b) \in R^{\mathcal{G}}$ gilt genau dann, wenn es eine Kante von a nach b gibt.

Aufgabe 4. Wir betrachten die folgenden vier Graphen:
$\mathcal{G}_1 = (\mathbb{N}, R^{\mathcal{G}})$ mit $R^{\mathcal{G}} = \{(a, b) \in \mathbb{N}^2 \mid a \text{ ist Teiler von } b\}$.



$\mathcal{G}_2$ $\mathcal{G}_3$ $\mathcal{G}_4$

($\mathcal{G}_1$ ist ein gerichteter Graph. Die gezeichneten Kanten in den Abbildungen sind ungerichtet, d.h. jede von ihnen steht für zwei entgegengesetzt gerichtete Kanten.)

wahr (W) oder falsch (F)?	$i = 1$	$i = 2$	$i = 3$	$i = 4$
$\mathcal{G}_i \models \forall xyz(R(x,y) \wedge R(y,z) \to R(x,z))$ gilt für ...	**W**	**F**	**F**	**F**
$\mathcal{G}_i \models \exists wxyz(R(w,x) \wedge R(x,y) \wedge R(y,z) \wedge R(z,w))$ gilt für ...	**W**	**W**	**W**	**W**
$\mathcal{G}_i \models \exists wxyz(w \neq x \wedge w \neq y \wedge w \neq z \wedge x \neq y \wedge x \neq z \wedge y \neq z)$ gilt für ...	**W**	**W**	**F**	**W**

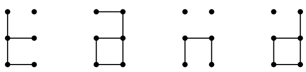
Kommentar: R in $\mathcal{G}_1$ ist eine partielle Ordnung, also transitiv. R in $\mathcal{G}_1$ ist außerdem reflexiv, so dass dort auch die Formel in der zweiten Zeile gilt. Ansonsten alles *genau* wie in der Hausaufgabe.

Aufgabe 5. Zeichnen Sie zwei nicht isomorphe Graphen, die beide die folgende L-Aussage erfüllen:

$$\begin{aligned}\mathcal{G}_i \models \forall x(\neg R(x,x)) \quad &\wedge \quad \forall xy(R(x,y) \to R(y,x)) \quad \wedge \quad \exists x(x = x)\\ \wedge \quad \forall w \exists xyz\Big(& x \neq y \wedge x \neq z \wedge y \neq z \wedge R(w,x) \wedge R(w,y) \wedge R(w,z)\\ &\wedge \forall u(R(w,u) \to u = x \vee u = y \vee u = z)\Big)\end{aligned}$$

Lösung: Hier gab es unendlich viele mögliche Lösungen. Die meisten Teilnehmer haben gemerkt, dass $\mathcal{G}_2$ aus Aufgabe 4 eine Lösung ist. Die einfachste Variation bestand darin, den unzusammenhängenden Graphen zu nehmen, der aus zwei Kopien von $\mathcal{G}_2$ besteht. Es gibt aber noch viele weitere Variationsmöglichkeiten wie beispielsweise ein $2n$-Eck mit Diagonalen oder ein Würfel.

Aufgabe 6. Geben Sie für jeden Graphen $\mathcal{G}_i$ eine Formel φ_i an, so dass $\mathcal{G}_i \models \varphi_j \iff i = j$.



$\mathcal{G}_0$ $\mathcal{G}_1$ $\mathcal{G}_2$ $\mathcal{G}_3$

$$\begin{aligned}
\varphi_0 &\equiv \exists uxyz(x \neq y \wedge x \neq z \wedge y \neq z \wedge R(u,x) \wedge R(u,y) \wedge R(u,z))\\
&\qquad \wedge \neg\exists uxyz(u \neq y \wedge x \neq z \wedge R(u,x) \wedge R(x,y) \wedge R(y,z) \wedge R(z,u))\\
\varphi_1 &\equiv \exists x_1 \ldots x_6(x_1 \neq x_2 \wedge x_1 \neq x_3 \wedge \ldots \wedge x_5 \neq x_6 \wedge R(x_1,x_2) \wedge R(x_2,x_3) \wedge \ldots \wedge R(x_5,x_6))\\
\varphi_2 &\equiv \exists xy(x \neq y \wedge \forall z \neg R(x,z) \wedge \forall z \neg R(y,z))\\
\varphi_3 &\equiv \exists x \forall y \neg R(x,y) \wedge \exists wxyz(w \neq x \wedge w \neq y \wedge w \neq z \wedge x \neq y \wedge x \neq z \wedge y \neq z\\
&\qquad \wedge R(w,x) \wedge R(x,y) \wedge R(y,z) \wedge R(z,w))
\end{aligned}$$

Kommentar: φ_0: Es gibt einen Knoten vom Grad 3, aber keinen Zykel der Länge 4. (Letzteres abgekürzt. Die Abkürzung nutzt aus, dass alle vier Graphen keine Schleifen haben.) φ_1: Es gibt einen (überschneidungsfreien!) Weg der Länge 5. (Wie φ_2 der Hausaufgabe.) φ_2: Es gibt zwei (verschiedene!) isolierte Punkte. φ_3: Es gibt einen isolierten Punkt und einen Zykel der Länge 4.

Übungen zu „Grundbegriffe der Mathematischen Logik“ Prüfung am Mittwoch, 16.06.2010

Teil I – Rekursionstheorie

Aufgabe 1. Welche (totale) Funktion $f\colon \mathbb{N}^2 \to \mathbb{N}$ wird von dieser Registermaschine berechnet?

0: IF $\mathbb{R}_2 = 0$ THEN GOTO 6	4: $\mathbb{R}_0 := \mathbb{R}_0 + 1$
1: IF $\mathbb{R}_1 = 0$ THEN GOTO 6	5: GOTO 0
2: $\mathbb{R}_1 := \mathbb{R}_1 \dot{-} 1$	6: STOP
3: $\mathbb{R}_2 := \mathbb{R}_2 \dot{-} 1$	

Lösung: $f(x, y) = \min(x, y)$.

Aufgabe 2. Das obige Programm wurde aus einem alten Buch eingescannt. Dabei kam es zu einem Fehler in Zeile 0. Das Resultat ist die folgende Registermaschine $\mathcal{M}$:

0: IF $\mathbb{R}_2 = 0$ THEN GOTO 0	4: $\mathbb{R}_0 := \mathbb{R}_0 + 1$
1: IF $\mathbb{R}_1 = 0$ THEN GOTO 6	5: GOTO 0
2: $\mathbb{R}_1 := \mathbb{R}_1 \dot{-} 1$	6: STOP
3: $\mathbb{R}_2 := \mathbb{R}_2 \dot{-} 1$	

wahr (W) oder falsch (F)?	Antwort
$\mathcal{M}$ berechnet eine totale Funktion von $\mathbb{N}^2$ nach $\mathbb{N}$.	**F**
$\mathcal{M}$ stoppt genau dann, wenn am Anfang $\mathbb{R}_2 = 0$ ist.	**F**
$\mathcal{M}$ berechnet eine partielle Funktion h von $\mathbb{N}^2$ nach $\mathbb{N}$, so dass $h(a, b) = \min(a, b)$ wann immer $h(a, b)$ definiert ist.	**W**
$\mathcal{M}$ berechnet eine partielle Funktion g von $\mathbb{N}^3$ nach $\mathbb{N}$, so dass $g(a, b.c)$ genau dann definiert ist, wenn $a < b$.	**W**

Kommentar: Das Programm stoppt genau dann, wenn der Fehler (sofortige Endlosschleife) nicht erreicht wird. Der Fehler wird bspw. erreicht wenn am Anfang $\mathbb{R}_2 = 0$ ist, aber auch, wenn $\mathbb{R}_2$ mindestens so früh wie $\mathbb{R}_1$ Null wird, d.h. falls am Anfang $\mathbb{R}_1 \geq \mathbb{R}_2$ gilt. Sonst wie Hausaufgabe.

Aufgabe 3. Hier sollten Sie insgesamt 12 Antworten geben.

wahr (W), falsch (F) oder sinnlos (S)?	prim. rek.	rek.	rek. aufzb.
Jede stetig differenzierbare Funktion $f\colon \mathbb{R}^k \to \mathbb{R}$ ist ...	**S**	**S**	**S**
Wenn $f\colon \mathbb{N}^3 \to \mathbb{N}$ rekursiv ist, $g(0, y) = f(0, y, y)$ und $g(x + 1, y) = f(x, y, g(x, y))$, dann ist g immer ...	**F**	**W**	**S**
Die totale Funktion $h\colon \mathbb{N}^2 \to \mathbb{N}$, definiert durch $h(0, y) = 3y + y + 2$, $h(x + 1, 0) = h(x, 1)$, $h(x + 1, y + 1) = h(x, h(x + 1, y))$ ist ...	**F**	**W**	**S**
Jede Menge der Form $A \setminus B$ mit A primitiv rekursiv und B rekursiv aufzählbar ist ...	**F**	**F**	**F**

Kommentar: Zeile 1 ist etwas verunglückt, weil Jakob Kellner in der Vorlesung das Problem der Erweiterung der Definitionen auf reelle Funktionen kurz angesprochen hatte; es gibt aber keine allgemein bekannten Standarddefinitionen. In Zeile 2 steht eine primitive Rekursion, wie in der Hausaufgabe. Die rekursiven Funktionen sind unter primitiver Rekursion abgeschlossen, aber wenn man mit einer Funktion anfängt, die selbst nicht einmal primitiv rekursiv ist, kann man natürlich im Allgemeinen keine primitive Rekursivität erwarten. Die Funktion in Zeile 3 ist wie die Ackermannfunktion definiert, nur mit größeren Startwerten. Insbesonders ist das keine primitive Rekursion! Nach der Church-Turing-These muss die Funktion natürlich rekursiv sein. Aber es gibt keinen Grund anzunehmen, dass sie primitiv rekursiv ist, und weil sie schneller wächst als die Ackermannfunktion ist sie das auch nicht. Zeile 4: Wenn B rekursiv aufzählbar aber nicht rekursiv ist, dann ist nicht einmal $\mathbb{N} \setminus B$ rekursiv aufzählbar.

Teil II – Prädikatenlogik

Sei $L = \{R\}$ die Sprache der Graphen, wobei R ein zweistelliges Relationssymbol ist. Wir fassen (gerichtete) Graphen (allenfalls mit Schleifen, allenfalls unendlich) als L-Strukturen $\mathcal{G} = (G, R^{\mathcal{G}})$ auf. Hierbei ist G die Menge der Knoten und $(a,b) \in R^{\mathcal{G}}$ gilt genau dann, wenn es eine Kante von a nach b gibt.

Aufgabe 4. Wir betrachten die folgenden vier Graphen:
$\mathcal{G}_1 = (\mathbb{N}, R^{\mathcal{G}})$ mit $R^{\mathcal{G}} = \{(a,b) \in \mathbb{N}^2 \mid a \leq b\}$.

$\mathcal{G}_2$ $\mathcal{G}_3$ $\mathcal{G}_4$

($\mathcal{G}_1$ ist ein gerichteter Graph. Die gezeichneten Kanten in den Abbildungen sind ungerichtet, d.h. jede von ihnen steht für zwei entgegengesetzt gerichtete Kanten.)

wahr (W) oder falsch (F)?	$i=1$	$i=2$	$i=3$	$i=4$
$\mathcal{G}_i \models \forall xyz(R(x,y) \land R(y,z) \to R(x,z))$ gilt für ...	**W**	**F**	**F**	**F**
$\mathcal{G}_i \models \exists wxyz(R(w,x) \land R(x,y) \land R(y,z) \land R(z,w))$ gilt für ...	**W**	**W**	**W**	**W**
$\mathcal{G}_i \models \exists wxyz(w \neq x \land w \neq y \land w \neq z \land x \neq y \land x \neq z \land y \neq z)$ gilt für ...	**W**	**W**	**F**	**W**

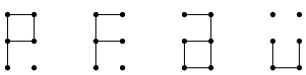
Kommentar: R in $\mathcal{G}_1$ ist eine partielle Ordnung, also transitiv. R in $\mathcal{G}_1$ ist außerdem reflexiv, so dass dort auch die Formel in der zweiten Zeile gilt. Ansonsten alles *genau* wie in der Hausaufgabe.

Aufgabe 5. Zeichnen Sie zwei nicht isomorphe Graphen, die beide die folgende L-Aussage erfüllen:

$$\begin{aligned}\mathcal{G}_i \models \quad & \exists x(x = x) \quad \land \quad \forall x(\neg R(x,x)) \quad \land \quad \forall xy(R(x,y) \to R(y,x)) \\ & \land \quad \forall w \exists xyz\Big(R(w,x) \land R(w,y) \land R(w,z) \land x \neq y \land x \neq z \land y \neq z \\ & \qquad \land \forall u(R(w,u) \to u = x \lor u = y \lor u = z)\Big)\end{aligned}$$

Lösung: Hier gab es unendlich viele mögliche Lösungen. Die meisten Teilnehmer haben gemerkt, dass $\mathcal{G}_2$ aus Aufgabe 4 eine Lösung ist. Die einfachste Variation bestand darin, den unzusammenhängenden Graphen zu nehmen, der aus zwei Kopien von $\mathcal{G}_2$ besteht. Es gibt aber noch viele weitere Variationsmöglichkeiten wie beispielsweise ein $2n$-Eck mit Diagonalen oder ein Würfel.

Aufgabe 6. Geben Sie für jeden Graphen $\mathcal{G}_i$ eine Formel φ_i an, so dass $\mathcal{G}_i \models \varphi_j \iff i = j$.



$\mathcal{G}_0$ $\mathcal{G}_1$ $\mathcal{G}_2$ $\mathcal{G}_3$

$$\begin{aligned}\varphi_0 \quad &\equiv \quad \exists x \forall y \neg R(x,y) \land \exists wxyz(w \neq x \land w \neq y \land w \neq z \land x \neq y \land x \neq z \land y \neq z \\ & \qquad \land R(w,x) \land R(x,y) \land R(y,z) \land R(z,w)) \\ \varphi_1 \quad &\equiv \quad \exists uxyz(x \neq y \land x \neq z \land y \neq z \land R(u,x) \land R(u,y) \land R(u,z)) \\ & \qquad \land \neg \exists uxyz(u \neq y \land x \neq z \land R(u,x) \land R(x,y) \land R(y,z) \land R(z,u)) \\ \varphi_2 \quad &\equiv \quad \exists x_1 \ldots x_6(x_1 \neq x_2 \land x_1 \neq x_3 \land \ldots \land x_5 \neq x_6 \land R(x_1,x_2) \land R(x_2,x_3) \land \ldots \land R(x_5,x_6)) \\ \varphi_3 \quad &\equiv \quad \exists xy(x \neq y \land \forall z \neg R(x,z) \land \forall z \neg R(y,z))\end{aligned}$$

Kommentar: φ_0: Es gibt einen isolierten Punkt und einen Zykel der Länge 4. φ_1: Es gibt einen Knoten vom Grad 3, aber keinen Zykel der Länge 4. (Letzteres abgekürzt. Die Abkürzung nutzt aus, dass alle vier Graphen keine Schleifen haben.) φ_2: Es gibt einen (überschneidungsfreien!) Weg der Länge 5. (Wie φ_2 der Hausaufgabe.) φ_3: Es gibt zwei (verschiedene!) isolierte Punkte.