Institut für Informatik
M. Mutz

Übungsblatt 11

Theoretische Informatik

Bearbeitungszeit: 24.06.2024 bis 30.06.2024, 16:00 Uhr
Besprechung: 01.07.2024, 10:30 Uhr in Hörsaal 5E

Abgabe: als PDF über das ILIAS
Gruppenabgaben möglich und erwünscht

Aufgabe 1 (Partiell rekursive Funktionen) 14P

(a) Sei die Funktion $eq : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ gegeben mit

$$eq(x, y) = \begin{cases} 0, & \text{falls } x = y \\ \text{undefiniert}, & \text{sonst} \end{cases}$$

Berechnen Sie $\mu\, eq$.

(b) Gegeben seien die partiell rekursiven Funktionen $lt : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ und $gt : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$, die wie folgt definiert sind:

$$lt(x, y) = \begin{cases} x, & \text{falls } x < y \\ y, & \text{falls } x > y \\ \text{undefiniert}, & \text{falls } x = y \end{cases} \qquad gt(x, y) = \begin{cases} y, & \text{falls } x < y \\ x, & \text{falls } x > y \\ \text{undefiniert}, & \text{falls } x = y \end{cases}$$

Welche Funktionen berechnen $\mu\, lt$ und $\mu\, gt$? Bitte geben Sie eine formale mathematische Beschreibung für $\mu\, lt$ und $\mu\, gt$ an.

(c) Zeigen Sie, dass die Funktion $l : \mathbb{N} \to \mathbb{N}$ mit

$$l(m) = \begin{cases} log_2(m), & \text{falls } log_2(m) \in \mathbb{N} \\ \text{undefiniert}, & \text{falls } log_2(m) \notin \mathbb{N} \text{ oder } m = 0 \end{cases}$$

partiell rekursiv ist. Dafür dürfen Sie alle (aus der Vorlesung und den Übungen inklusive dieses Übungsblattes) bekannten primitiv rekursiven Funktionen direkt verwenden. Dass deren Komposition selbst primitiv rekursiv ist, müssen Sie jedoch mit dem Normalschema zeigen.

Lösungsvorschlag:

(a) Bei Anwendung des μ-Operators auf die Funktion eq entsteht eine Funktion $f : \mathbb{N} \to \mathbb{N}$, die wie folgt definiert ist:

$$f(y) = \min\{x \in \mathbb{N} \mid eq(x, y) = 0 \land \forall z < x \text{ ist } eq(z, y) \text{ definiert}\}.$$

Im Falle, dass $y > 0$ und $eq(x, y) = 0$ (insbesondere gilt $x = y$) gilt, können wir ein $z \in \mathbb{N}$ finden (zum Beispiel $z = x - 1$), das kleiner als x und y ist, so dass $eq(z, y)$ nicht definiert ist (da $z \neq y$). Daraus folgern wir also, dass $f(y)$ für alle $y > 0$ nicht definiert ist.
Andernfalls, wenn y gleich Null ist, ist $eq(x, y) = 0$ genau dann, wenn x auch gleich Null ist. Und weil es keine kleinere natürliche Zahl als 0 gibt, können wir kein z finden, so dass $eq(z, 0)$ nicht definiert ist. Darum folgt $f(0) = \min\{0\} = 0$.
Also ist die Funktion $f = \mu\, eq$ wie folgt definiert:

$$f(y) = \begin{cases} 0, & \text{falls } y = 0 \\ \text{undefiniert}, & \text{falls } y \neq 0 \end{cases}$$

(b) Betrachte zunächst lt. Ist dort $y \geq 1$, dann ist das kleinste x mit $lt(x, y) = 0$ die Null selbst, also $x = 0$. Da $0 < y$ gilt dann auch $\mu\, lt(x, y) = 0$.

Sei nun $y = 0$. Dann ist $lt(x, 0)$ an $x = 0$ bereits nicht definiert. Folglich ist auch $\mu\, lt(x, y)$ dann auch undefiniert.

$$\mu\, lt(x, y) = \begin{cases} 0, & \text{falls } y \geq 1 \\ \text{undefiniert}, & \text{falls } y = 0 \end{cases}$$

Betrachte nun gt. $gt(x, y)$ wählt die größere der beiden Zahlen aus oder ist undefiniert, wenn beide 0 sind. Folglich ist $\mu\, gt(x, y)$ auch undefiniert, da $gt(x, y)$ niemals die 0 annimmt.

$$\mu\, gt(x, y) = \text{undefiniert}$$

(c) Um zu zeigen, dass $l : \mathbb{N} \to \mathbb{N}$ partiell rekursiv ist, müssen wir eine geeignete primitiv rekursive Funktion $f : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ finden, so dass $\mu f = l$ gilt. Die Funktion l ist genau dann definiert, wenn $log_2(m)$ eine natürliche Zahl ist und das ist erfüllt, wenn es ein $n \in \mathbb{N}$ gibt, so dass $m = 2^n$.
In der Vorlesung und in den Übungen haben wir gezeigt, dass die Exponentialfunktion exp (siehe Bsp. 9.2), die Additionsfunktion, die modifizierte Differenz und die modifizierte Differenz mit vertauschten Argumenten, primitiv rekursiv sind. Außerdem gilt für die Abstandsfunktion $A(x, y) = add(md(x, y), dm(x, y))$

Deshalb können wir zum Beispiel mit dem Normalschema zeigen, dass die Funktion $f : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ mit

$$f(n, m) = |2^n - m|$$

ebenfalls primitiv rekursiv ist. Die Funktion lässt sich mit dem Normalschema wie folgt beschreiben:

$f(n, m) = A(e(n, m), id_2^2(n, m)) \cong |2^n - m|$, wobei gilt:

$e(n, m) = exp(id_1^2(n, m), c\_2(n, m)) \cong 2^n$ (Substitution)

$c\_2(n, m) = 2(id_1^2(n, m)) \cong 2$ (konstante Funktion)

Da f primitiv rekursiv ist und die Klasse der partiell rekursiven Funktionen $\mathbb{P}$ unter dem μ-Operator abgeschlossen ist, können wir $l \in \mathbb{P}$ zeigen, indem wir nachweisen, dass $l(m) = \mu n[f(n, m)]$ gilt. Nach Anwendung des μ-Operators erhalten wir folgende Gleichungen:

$$\begin{aligned}
\mu n[f(n,m)] &= \begin{cases} \text{das kleinste } n \in \mathbb{N} \text{ mit } f(n,m) = 0, & \text{falls es existiert} \\ \text{undefiniert,} & \text{sonst} \end{cases} \\
&= \begin{cases} \text{das kleinste } n \in \mathbb{N} \text{ mit } |2^n - m| = 0, & \text{falls es existiert} \\ \text{undefiniert,} & \text{sonst} \end{cases} \\
&= \begin{cases} \text{das kleinste } n \in \mathbb{N} \text{ mit } 2^n = m, & \text{falls es existiert} \\ \text{undefiniert,} & \text{sonst} \end{cases} \\
&= \begin{cases} \text{das kleinste } n \in \mathbb{N} \text{ mit } n = log_2(m), & \text{falls es existiert} \\ \text{undefiniert,} & \text{sonst} \end{cases} \\
&= \begin{cases} log_2(m), & \text{falls } log_2(m) \in \mathbb{N} \\ \text{undefiniert,} & \text{sonst} \end{cases} \\
&= l(m)
\end{aligned}$$

Hiermit zeigten wir, dass $l : \mathbb{N} \to \mathbb{N}$ partiell rekursiv ist.

Aufgabe 2 (Gödelisierung von Typ-1-Grammatiken) 8P

(a) Geben Sie zu folgendem Gödelwort einer Typ-1-Grammatik die passende Grammatik G an:

$$ba^5bba^3bba^5bba^6bba^4bba^5bba^3bba^2bba^2bba^5bba^4bba^5bba^3bba^2bba^2bba^4b$$
$$ba^6bba^3bbabba^4bba^6bba^3bbabba^6b$$

(b) Ist die obige Gödelisierung von G in $L(G)$ enthalten?

Lösungsvorschlag:

(a) $G = (\{a, b\}, \{X_0, X_1\}, X_0, P)$ mit

$$P = \{X_0 \to X_0X_1 \mid bbX_0 \mid bb, \\ X_1 \to a \mid aX_1\}.$$

(b) $w_G \notin L(G)$, da alle Wörter in $L(G)$ mit bb anfangen.

Aufgabe 3 (Rekursive Aufzählbarkeit) 9P

Zeigen Sie den Satz aus der Vorlesung, dass RE abgeschlossen ist unter Vereinigung.

Lösungsvorschlag:
Seien $A \in \text{RE}$ und $B \in \text{RE}$. Zu zeigen ist, dass $A \cup B \in \text{RE}$.
Sind $A = \emptyset$ oder $B = \emptyset$, dann ist $A \cup B = A$ oder $A \cup B = B$, also $A \cup B \in \text{RE}$.
Seien also $A \neq \emptyset$ und $B \neq \emptyset$. Dann existieren Aufzählfunktionen $f, g \in \mathbb{R}$ für A und B, sodass $A = \mathrm{W}_f$ und $B = \mathrm{W}_g$.
Wir geben nun einen Algorithmus an, der $\chi'_{A \cup B}$ berechnen und zeigen somit, dass $A \cup B$ semi-entscheidbar sind. Mit Satz 10.15 folgt dann, dass $A \cup B \in \text{RE}$.

Algorithmus für $A \cup B$ mit Eingabe x:

- Berechne $f(0), g(0), f(1), g(1), f(2), g(2), \ldots$ solange, bis $x = f(i)$ oder $x = g(i)$ für ein i gilt.
- Falls ein i gefunden wurde, gib 1 aus.

Falls $x \in A \cup B$, also $x \in A$ oder $x \in B$, berechnet der Algorithmus i in endlicher Zeit (selbst wenn entweder $x \notin A$ oder $x \notin B$ gilt, werden f und g nur für die Werte 0 bis i berechnet) und gibt 1 aus. Falls $x \notin A \cup B$, also $x \notin A$ und $x \notin B$, hält der Algorithmus nie, da für jedes i gilt, dass $x \neq f(i)$ und $x \neq g(i)$.

Aufgabe 4 (Typ-1-Wortproblem) 9P

Gegeben sei die kontextsensitive Grammatik $G = (\Sigma, N, S, P)$ mit $\Sigma = \{a, b, c\}$, $N = \{S, A, B, C\}$ und

$$\begin{aligned} P \ = \ \{ & S \to SABC \mid ABC, \\ & BA \to AB, \quad CB \to BC, \quad CA \to AC, \\ & AA \to aa, \quad BB \to bb, \quad CC \to cc\}. \end{aligned}$$

Der Algorithmus aus der Vorlesung für das Typ-1-Wortproblem wird nun für G und das Wort $w_1 = aabbcc$ angewendet.

(a) Geben Sie die Mengen T und T_1 jedem der ersten vier Schleifendurchläufe an.

(b) Wie oft wird die Schleife durchlaufen und was ist dann die Ausgabe des Algorithmus?

(c) Geben Sie $\chi_{L(G)}(w_1)$ an.

Lösungsvorschlag:

(a)

	T	T_1
0	$\{S\}$	$\emptyset$
1	$\{S, SABC, ABC\}$	$\{S\}$
2	$\{S, SABC, ABC, ABCABC\}$	$\{S, SABC, ABC\}$
3	$\{S, SABC, ABC, ABCABC, ABACBC\}$	$\{S, SABC, ABC, ABCABC\}$
4	$\{S, SABC, ABC, ABCABC, ABACBC, AABCBC, ABABCC\}$	$\{S, SABC, ABC, ABCABC, ABACBC\}$

(b) Es sind acht Schleifendurchläufe. Danach sind alle Wörter über $\Sigma \cup N$ in T, die sich mit acht Ableitungsschritten bilden lassen, zu denen auch w_1 gehört. Die Ausgabe lautet folglich "$w_1 \in L(G)$".

(c) Da w_1 in der Menge $L(G)$ ist, gilt $\chi_{L(G)}(w_1) = 1$.