TU Ilmenau, Fachgebiet Automaten und Logik
Prof. Dr. Dietrich Kuske, M.Sc. Christian Schwarz
Sommersemester 2023

Berechenbarkeit und Komplexität – Übung 2

Abgabe bis zum 27. April um 13:00 Uhr vor der Vorlesung bzw. im Briefkasten.

Aufgabe 1* — 3 Punkte

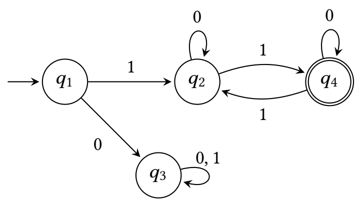
Sei $a : \mathbb{N}^2 \longrightarrow \{0,1\}$ die Funktion mit $a(k,n) \equiv ack(k,n) \bmod 2$, d.h., $a(k,n)$ ist die Parität des Wertes $ack(k,n)$. Zeigen oder widerlegen Sie: Die Funktion a ist LOOP-berechenbar.

Aufgabe 2* — 4 Punkte

Sei $(f_k)_{k\in\mathbb{N}}$ eine Folge von LOOP-berechenbaren Funktionen $f_k : \mathbb{N} \longrightarrow \mathbb{N}$ so, dass jedes f_k durch ein LOOP-Programm mit k LOOP-Schleifen berechnet werden kann, nicht aber durch ein Programm mit $k-1$ LOOP-Schleifen. Zeigen oder widerlegen Sie: Die Funktion g mit $g(k,n) = f_k(n)$ ist LOOP-berechenbar.

Aufgabe 3* — 4 Punkte

Wir betrachten den folgenden DFA M über dem Alphabet $\Sigma = \{0,1\}$:



Für ein Wort $w = a_1 \dots a_n \in \Sigma^*$ mit $a_1, \dots, a_n \in \Sigma$ sei $\mathrm{code}(w) \in \mathbb{N}$ diejenige natürliche Zahl, deren Binärdarstellung $1w = 1a_1 \dots a_n$ entspricht. Zum Beispiel ist $\mathrm{code}(10) = \langle 110\rangle_2 = 6$, $\mathrm{code}(0) = \langle 10\rangle_2 = 2$ und $\mathrm{code}(\varepsilon) = \langle 1\rangle_2 = 1$.
Geben Sie ein LOOP-Programm P an, welches die folgende Funktion f berechnet:

$$f : \mathbb{N} \longrightarrow \mathbb{N}, f(n) = \begin{cases} 1 & \text{, falls } \exists w \in \Sigma^*\colon\ \mathrm{code}(w) = n \text{ und } w \in L(M) \\ 0 & \text{, falls } \exists w \in \Sigma^*\colon\ \mathrm{code}(w) = n \text{ und } w \notin L(M) \text{ oder } n = 0\,. \end{cases}$$

Sie dürfen dabei alle in der Vorlesung definierten Programme als Unterprogramme verwenden.

Aufgabe 4* — 4 Punkte

Wir erweitern in dieser Aufgabe die in der Vorlesung vorgestellten LOOP-Programme (vgl. Folien 1.18ff) um einen Stack, auf den mit dem Befehl $\mathtt{push}(x_i)$ der Wert von x_i gelegt werden kann und von dem mit dem Befehl $x_i := \mathtt{pop}()$ der oberste Wert ausgelesen werden kann, welcher dann gleichzeitig vom Stack entfernt wird. Falls der Stack leer ist, gibt `pop` den Wert 0 zurück.
Zeigen Sie, dass eine Funktion $f : \mathbb{N}^k \longrightarrow \mathbb{N}$ genau dann LOOP-berechenbar ist, wenn es ein LOOP-Programm mit Stack P gibt, welches höchstens die Variablen $x_1, \dots, x_l$ mit $l \geq k$ verwendet und für welches

$$f(\overline{n}) = \pi_1^l(\llbracket P \rrbracket_l(\overline{n}, 0, \dots, 0))$$

für alle $\overline{n} \in \mathbb{N}^k$ gilt.

Aufgabe 5

Sei $ack : \mathbb{N}^2 \longrightarrow \mathbb{N}$ die Ackermann-Funktion (vgl. Folien 3.4f). Implementieren Sie in PROLOG[1] ein Prädikat $\mathtt{ack}(M,N,X)$, für welches gilt: werden M, N mit natürlichen Zahlen m, n instantiiert, so ist $\mathtt{?-ack}(M,N,X)\,\mathtt{.}$ genau dann erfüllt, wenn $X = ack(m,n)$ ist.

Zusatz. Verwenden Sie Listen, um natürliche Zahlen binär zu codieren und 'rechnen' Sie anschließend mit solchen Listen, d.h. verwenden Sie den Operator `is/2` in Ihrer Implementierung nicht.

[1] https://www.swi-prolog.org/Download.html