Theoretische Informatik und Logik

2. Übungsblatt

Sommersemester 2018

Die folgenden Aufgaben werden nicht in den Übungen besprochen und dienen der Selbstkontrolle.

Aufgabe C

Zeigen Sie, dass $\{1\}^*$ unentscheidbare Teilmengen besitzt.

Aufgabe D

Welche der folgenden Aussagen sind wahr? Begründen Sie Ihre Antwort.

a) Jedes LOOP-Programm terminiert.

b) Zu jedem WHILE-Programm gibt es ein äquivalentes LOOP-Programm.

c) Die Anzahl der Ausführungen von P in der LOOP-Schleife

$$\texttt{LOOP}\ x_i\ \texttt{DO}\ P\ \texttt{END}$$

kann beeinflusst werden, indem x_i in P entsprechend modifiziert wird.

d) Die Ackermannfunktion ist total und damit LOOP-berechenbar.

Aufgabe 1

Zeigen Sie, dass folgende Funktionen $f\colon \mathbb{N}^2 \to \mathbb{N}$ LOOP-berechenbar sind:

a) $f(x, y) := \mathsf{max}(x - y, 0)$

b) $f(x, y) := x \cdot y$

c) $f(x, y) := \mathsf{max}(x, y)$

d) $f(x, y) := \mathsf{ggT}(x, y)$, wobei $\mathsf{ggT}(x, y)$ den größten gemeinsamen Teiler von x und y bezeichnet.

Implementieren Sie einen Interpreter für LOOP-Programme, der in der Lage ist beliebige LOOP-Programme auszuführen. Verwenden Sie Ihren Interpreter um Ihre LOOP-Programme aus den Aufgabenteilen a–d zu testen. Achten Sie hierbei erneut auf mögliche Randfälle.

Aufgabe 2

Mit $\mathsf{kgV}(x_1, x_2)$ bezeichnen wir das kleinste gemeinsame Vielfache zweier natürlicher Zahlen x_1 und x_2.

a) Geben Sie ein WHILE-Programm an, das die Funktion $f\colon \mathbb{N}^2 \to \mathbb{N}, (x_1, x_2) \mapsto \mathsf{kgV}(x_1, x_2)$ berechnet und erklären Sie seine Arbeitsweise.

b) Erweitern Sie Ihren Interpreter für LOOP-Programme aus Aufgabe 1 um WHILE-Schleifen und testen Ihr WHILE-Programm für kgV mithilfe Ihres Interpreters. Achten Sie erneut auf mögliche Randfälle.

Aufgabe 3

Es sei Σ ein fest gewähltes Alphabet mit mindestens zwei Elementen. Wir betrachten eine Programmiersprache L über Σ, die in der Lage ist, Turing-Maschinen zu simulieren. Für ein Wort $w \in \Sigma^*$ ist die *Kolmogorov-Komplexität* $K_L(w)$ die Länge des kürzesten Programms in L, welches bei leerer Eingabe das Wort w als Ausgabe produziert.

Zeigen Sie folgende Aussagen:

a) Es gibt für jede natürliche Zahl $n \in \mathbb{N}$ ein Wort $w \in \Sigma^*$ der Länge $|w| = n$, so dass $K_L(w) \geq n$.

b) Es gibt eine Konstante $c \in \mathbb{N}$, so dass gilt: Ist w das Ergebnis der Berechnung einer Turing-Maschine M mit Eingabe x, dann

$$K_L(w) \leq |\mathsf{enc}(M)\#\#\,\mathsf{enc}(x)| + c,$$

wobei $\mathsf{enc}(M)\#\#\,\mathsf{enc}(x)$ eine (effektive) Kodierung der Maschine M und der Eingabe x als ein Wort über Σ ist.

c) Die Abbildung $w \mapsto K_L(w)$ ist nicht berechenbar.

Damit ist insbesondere gezeigt, dass es niemals einen Compiler geben kann, der ein gegebenes Programm in ein kleinstmögliches übersetzt.