Universität Siegen
Lehrstuhl Theoretische Informatik
Markus Lohrey



# Übungsblatt 4

**Aufgabe 1**
Sind folgende Aussagen wahr oder falsch? Begründen Sie.

(a) Es gibt unendlich viele Funktionen $f$, so dass $\mu f$ primitiv rekursiv ist.

(b) Es gibt eine WHILE-berechenbare Funktion $f$, die total ist, aber für die $\mu f$ nicht LOOP-berechenbar ist.

(c) Für jede nicht totale Funktion $f$ ist $\mu f$ ebenfalls nicht total.

(d) Sei $f : \mathbb{N}^{k+1} \to \mathbb{N}$ ($k \geq 1$) eine (Turing-)berechenbare Funktion. Für jedes $k$-Tupel $\bar{x} = (x_1, \ldots, x_k) \in \mathbb{N}^k$ sei $f_{\bar{x}} : \mathbb{N} \to \mathbb{N}$ eine totale, surjektive Funktion, wobei $f_{\bar{x}}$ definiert ist durch $f_{\bar{x}}(n) = f(n, x_1, \ldots, x_k)$. Dann ist $\mu f$ total.

(e) Seien $f, g : \mathbb{N} \to \mathbb{N}$ berechenbar. Dann ist $f \circ g$ berechenbar.

(f) Die Rückrichtung in (e) gilt.

(g) Wenn eine Sprache $L$ entscheidbar ist, dann ist auch $\overline{L}$ entscheidbar.

(h) Jede reguläre Sprache ist entscheidbar.

(i) Sei $f : \mathbb{N} \to \mathbb{N}$ primitiv-rekursiv. Dann ist die Menge $\{n \in \mathbb{N} \mid n \geq f(n)\}$ entscheidbar.

**Lösung**

(a) Ja. Sei $f_i(n, x) = n - i \cdot x$, $i \in \mathbb{N}$. Dann ist $f_i(0, x) = 0 - i \cdot x = 0$ für alle $x \in \mathbb{N}$, also ist $n = 0$ die kleinste Nullstelle. Wir erhalten $(\mu f_i)(x) = 0$.

(b) Ja: Zum Beispiel ist die Funktion definiert durch $f(n, x) = x + n + 1$ LOOP-berechenbar und damit insbesondere WHILE-berechenbar. Aber: $(\mu f)(x)$ ist überall undefiniert, also nicht LOOP-berechenbar.

(c) Falsch. Die Funktion

$$f(n, x) = \begin{cases} 0 & \text{falls } n = 0, \\ \text{undefiniert} & \text{sonst.} \end{cases}$$

ist nicht total, aber es gilt $(\mu f)(x) = 0$.

(d) Wahr: Da $f_{\bar{x}}$ surjektiv ist, gibt es eine Zahl $n \in \mathbb{N}$ mit $f_{\bar{x}}(n) = 0$. Außerdem sind alle Werte $f_{\bar{x}}(i)$ für $i < n$ definiert, weil $f_{\bar{x}}$ total ist. Somit ist für alle $\bar{x} \in \mathbb{N}^k$

$$\{n \in \mathbb{N} \mid f(n, \bar{x}) = 0 \text{ und } \forall m < n : f(m, \bar{x}) \text{ ist definiert}\}$$

nicht-leer, sprich $(\mu f)(\bar{x})$ ist definiert, also ist $\mu f$ total.

(e) Wahr. Wenn man $f$ durch $T_1$ und $g$ durch $T_2$ berechnet, kann man $f \circ g$ berechnen, indem man erst $T_1$ laufen lässt und dann den Output als neuen Input auf $T_2$ laufen lässt. Beachte: Wenn $g(n)$ nicht definiert ist, so läuft die Turingmaschine für $g$ in eine Endlosschleife und $f(g(n))$ ist ebenfalls undefiniert.

(f) Falsch. Man nehme z.B. eine totale, aber nicht berechenbare Funktion $g$ (z.B. Busy-Beaver) und $f$ definiert durch $f(n) = 0$. Dann ist $f \circ g$ ebenfalls die Nullfunktion und somit berechenbar. Alternative zu Busy-Beaver: $\chi_L$ für $L$ unentscheidbar (z.B. Hilberts 10. Problem, siehe Aufgabe 3). Allerdings muss $\chi_L$ dafür in eine Funktion $\chi_L^* : \mathbb{N} \to \mathbb{N}$ umgewandelt werden. Das geht jedoch, da $\Sigma^*$ abzählbar ist.

(g) Wahr. Die charakteristische Funktion $\chi_L$ kann in $\chi_{\overline{L}}$ umgewandelt werden, indem 1 und 0 getauscht werden. Bei einer Turingmaschine für $L$ muss man also für jeden Input $w$ einfach den Output 1 und 0 vertauschen.

(h) Wahr. Für reguläre Sprachen kann das Wortproblem einfach durch einen DFA entschieden werden.

(i) Wahr. Primitiv-rekursive Funktionen sind insbesondere (Turing-)berechenbar. Außerdem sind sie total. Also kann man bei Eingabe $n$ einfach $f(n)$ berechnen und dann $n \geq f(n)$ testen.

**Aufgabe 2**

Zeigen Sie, dass die Menge der Primzahlen $\mathcal{P}$ und die Menge der Quadratzahlen $\mathcal{Q}$ entscheidbar sind. Geben Sie zudem ein WHILE-Programm an, welches die halbe charakteristische Funktion $\chi'_{\mathcal{Q}}$ berechnet.

**Lösung**

Der erste Teil lässt sich lösen, indem man beispielsweise ein LOOP-Programm angibt, welches die Ausgabe 1 oder 0 erzeugt, je nachdem, ob die Eingabe (also $x_1$) eine Primzahl (beziehungsweise eine Quadratzahl) ist.

Quadratzahl:

$$
\begin{array}{l}
x_2 := x_1; \\
x_1 := 0; \\
x_3 := 0; \\
\textsc{Loop}\ x_2\ \textsc{Do} \\
\quad x_4 := x_3 \cdot x_3; \\
\quad x_3 := x_3 + 1; \\
\quad x_5 := x_4 - x_2; \\
\quad x_6 := x_2 - x_4; \\
\quad x_7 := x_5 + x_6; \\
\quad \textsc{If}\ x_7 = 0\ \textsc{Then}\ x_1 := 1\ \textsc{End} \\
\textsc{End}
\end{array}
$$

Hierbei sind $x_5, x_6, x_7$ Hilfsvariablen, um $x_4 = x_2$ zu simulieren.
Primzahl:

$$
\begin{array}{l}
x_2 := x_1; \\
x_3 := x_1 - 2; \\
x_4 := 1; \\
x_1 := 1; \\
\textsc{Loop}\ x_3\ \textsc{Do} \\
\quad x_4 := x_4 + 1; \\
\quad x_5 := x_2 \mod x_4; \\
\quad \textsc{If}\ x_5 = 0\ \textsc{Then}\ x_1 := 0\ \textsc{End} \\
\textsc{End}; \\
x_6 := x_2 - 1; \\
\textsc{If}\ x_6 = 0\ \textsc{Then}\ x_1 := 0\ \textsc{End}
\end{array}
$$

Die Funktion mod haben wir bereits auf Blatt 2 simuliert. Das Programm testet, ob der Input $x_1$ durch $2, \ldots, x_1 - 1$ teilbar ist. Für die Sonderfälle $x_1 = 0$ und $x_1 = 1$ wurde noch eine weitere Abfrage eingefügt.
Das While-Programm für die halbe charakteristische Funktion $\chi'_Q$ bekommen wir beispielsweise aus dem 1. Teil wie folgt: Sei $Q$ das Loop-Programm für $\chi_Q$. Dann erhalten wir $\chi'_Q$ durch

$$
\begin{array}{l}
Q; \\
x_8 := 0; \\
\textsc{If}\ x_1 = 0\ \textsc{Then}\ x_8 := 1\ \textsc{End}; \\
\textsc{While}\ x_8 \neq 0\ \textsc{Do}\ x_8 := 1\ \textsc{End}
\end{array}
$$

**Aufgabe 3**

Ein bekanntes Problem aus der Mathematik ist Hilberts zehntes Problem: Gegeben ein Polynom $p(x_1, \dots, x_n)$ mit ganzzahligen Koeffizienten in $n$ Variablen ($n \geq 1$ beliebig), existieren $x_1, \dots, x_n \in \mathbb{Z}$ mit $p(x_1, \dots, x_n) = 0$? Erst 1970 wurde bewiesen, dass dieses Problem unentscheidbar ist.

(a) Ist Hilberts zehntes Problem semi-entscheidbar?

(b) Ist das Komplement semi-entscheidbar?

**Lösung**

(a) Wahr. Die Menge der möglichen Belegungen für ein Polynom über $\mathbb{Z}$ mit $n$ Variablen ist $\mathbb{Z}^n$, welches eine abzählbar unendliche Menge darstellt.[1] Das bedeutet, dass man alle möglichen Belegungen sukzessive durchprobieren kann. Das Verfahren terminiert, wenn es eine Belegung gibt, die das Polynom zu Null auswertet. Ansonsten läuft das Programm für immer weiter. Dies beschreibt also einen Semi-Entscheidungs-Algorithmus.

(b) Falsch. Aus a) und der Unentscheidbarkeit des Problems folgt, dass b) nicht semi-entscheidbar ist (Satz 20, Folie 115).

**Aufgabe 4**

Wir haben in FSA gesehen, dass Typ-0-Sprachen, also semi-entscheidbare Sprachen, unter Homomorphismenbildung abgeschlossen sind. Zur Erinnerung: Ein (Monoid-)Homomorphismus ist eine Abbildung $h : \Sigma^* \to \Gamma^*$ mit $h(uv) = h(u)h(v)$ und $h(\varepsilon) = \varepsilon$. Zeigen Sie, dass auch entscheidbare Sprachen unter Homomorphismenbildung abgeschlossen sind, d.h. wenn $L$ entscheidbar ist, dann auch $h(L) = \{h(w) \mid w \in L\}$.

**Lösung**

Sei $L$ eine entscheidbare Sprache und $M$ eine Turingmaschine, die $L$ entscheidet (also die charakteristische Funktion $\chi_L$ berechnet). Dann können wir wie folgt eine nichtdeterministische Turingmaschine $M'$ für $h(L)$ bzw. $\chi_{h(L)}$ erhalten:
Sei $w \in \Gamma^*$ ein Eingabewort. Die NTM $M'$ rät eine Zerlegung $w = w_1 \cdots w_n$ von $w$ mit $w_i = h(a_i)$, wobei $a_i \in \Sigma$ ist. Das Wort $v = a_1 \cdots a_n$ wird jetzt auf ein weiteres Band kopiert (es geht natürlich auch auf einem Band, aber so ist es einfacher). Man beachte also, dass das Bandalphabet von $M'$ mindestens die Zeichen aus $\Sigma$ zusätzlich enthalten muss. Jetzt geht der Leseschreibkopf auf das erste Zeichen von $v$. Auf dem Hilfsband wird $v$ genauso behandelt wie $M$ es tun würde, d.h. $M$ wird auf dem Hilfsband simuliert. Das bedeutet, wir haben es so definiert, dass $M'$ genau dann terminiert, wenn $M$ terminiert und $w$ wird genau dann akzeptiert, wenn $v$ von $M$ akzeptiert wird. Sonderfall: $h^{-1}(w) = \varnothing$, dann kann nichts aufs Hilfsband kopiert werden und wir erhalten $\chi_{h(L)}(w) = 0$. Somit berechnet $M'$ tatsächlich $\chi_{h(L)}$ und wir haben gezeigt, dass $h(L)$ entscheidbar ist.

[1] Der Beweis ist analog zur Konstruktion der Funktion $\langle n_1, n_2, \dots, n_k \rangle$ auf Folie 62, welches eine Bijektion von $\mathbb{N}^k$ nach $\mathbb{N}$ ist.