Matthias Krause
Alexander Moch

Mannheim, den 06.03.2020

CS 406 Theoretische Informatik, Frühjahrssemester 2020

Übungsblatt 2

AUFGABE 2.1:

Man zeige, dass das Halteproblem, das spezielle Halteproblem und die Diagonalsprache alle rekursiv aufzählbar sind.

AUFGABE 2.2:

Man zeige: Ist für eine rekursiv aufzählbare Sprache L auch ihr Komplement $\bar{L}$ rekursiv aufzählbar, so ist L sogar rekursiv.

AUFGABE 2.3:

Wir haben gezeigt, dass es keinen globalen Algorithmus gibt, der für ein gegebenes Computerprogramm P und eine gegebene Eingabe entscheidet, ob P auf dieser hält. Andererseits ist es anhand eines konkreten Programms oft einfach festzustellen, ob dieses Programm auf einer gegebenen Eingabe hält oder nicht. Man kann sich also die Frage stellen, ob eventuell für jedes Programm P ein lokales Verifikationsprogramm V_P existiert, das feststellt, ob P auf einer gegebenen Eingabe hält.

Man zeige nun, dass dies nicht der Fall ist. Das heißt, man gebe eine konkrete Turingmaschine $\tilde{M}$ an, für die die folgende Sprache $H^{\tilde{M}}$ nicht entscheidbar ist: $H^{\tilde{M}}$ bezeichnet die Menge all jener Eingaben, auf denen die TM $\tilde{M}$ anhält.

AUFGABE 2.4:

Beweise: Eine Sprache L ist genau dann rekursiv, wenn es eine Turingmaschine gibt, die die Wörter $w \in L$ in kanonischer Reihenfolge auf ein Ausgabeband schreibt. (Unter *kanonischer Reihenfolge* verstehen wir hier, dass kürzere Worte vor längeren Worten stehen und bei gleicher Wortlänge die lexikographische Ordnung ausschlaggebend ist.)

AUFGABE 2.5:

Beweise: Eine Sprache L ist genau dann rekursiv aufzählbar, wenn es eine Turingmaschine gibt, die die Wörter $w \in L$ (in beliebiger Reihenfolge) auf ein Ausgabeband schreibt.

AUFGABE 2.6:

Sei $n \in \mathbb{N}$, $n \geq 1$, und $x \in \{0,1\}^n$. Man gebe eine Turingmaschine M_x über dem Arbeitsalphabet $\Sigma = \{0, 1, \#\}$ an, die startend auf dem leeren Band das Wort $x = (x_1, \ldots, x_n)$ auf das Band schreibt und mit dem Kopf auf x_1 stoppt.

AUFGABE 2.7:

Man gebe eine Turingmaschine über dem Arbeitsalphabet $\Sigma = \{0, 1, \#\}$ an, die nur einen Zustand besitzt und auf keiner Eingabe stoppt.

AUFGABE 2.8:

Die Funktion $p : \{0,1\}^* \longrightarrow \{0,1\}$ sei wie folgt definiert:

$$p(x) = 1 \;\Leftrightarrow\; x \text{ ist eine Primzahl in Binärdarstellung.}$$

Ist für Programme (allgemein) entscheidbar, ob diese p berechnen?