

Theoretische Informatik 2

Übungsblatt 1

René Maseli
Prof. Dr. Roland Meyer

TU Braunschweig
Sommersemester 2024

Ausgabe: 15.04.2024

Abgabe: 25.04.2024, 23:59

Geben Sie Ihre Lösungen bis Donnerstag, den 25.04.2024 um 23:59 Uhr, ab. Laden Sie sie dazu als PDF oder Scan im Vips im Stud.IP hoch. Geben Sie als 4er-Gruppe ab. Achten Sie darauf, dass **Studiengang, Name Vorname und Matrikelnummer** jedes Gruppenmitglieds lesbar vorne auf Ihrer Abgabe zu finden sind.

Hausaufgabe 1: TM-Konstruktion [5 Punkte]

Konstruieren Sie eine NTM, die die Sprache $L_1 := \{a^n b^m c^n d^m \mid n, m \in \mathbb{N}\} \subseteq \{a, b, c, d\}^*$ akzeptiert. Beachten Sie, dass n und m auch 0 sein dürfen. Gehen Sie wie folgt vor.

a) [1 Punkt] Geben Sie eine Beschreibung (Idee) der Arbeitsweise der TM an.

b) [4 Punkte] Zeichnen Sie den Zustandsgraphen Ihrer TM M_1 mit $\mathcal{L}(M_1) = L_1$.

Hausaufgabe 2: Binärkodierung [5 Punkte]

Betrachten Sie nun die Sprache

$$L_2 := \{\text{bin}(n)\#1^n \mid n \in \mathbb{N}\} \subseteq \{0, 1, \#\}^* .$$

Wörter dieser Sprache sind Paare aus Binär- und Unärdarstellungen der gleichen Zahl n . Dazwischen soll ein Trennsymbol $\#$ liegen. Für jedes $n \in \mathbb{N}$ ist $\text{bin}(n) \in \{0, 1\}^*$ die kleinste Binärdarstellung von n , startend mit dem *most significant bit*. Gültige Wörter sind damit z.B. $\#, 1\#1, 10\#11, 11\#111, 100\#1111 \in L_2$.

a) [1 Punkt] Geben Sie in ein bis zwei Sätzen eine Idee für die Arbeitsweise einer TM an.

b) [4 Punkte] Zeichnen Sie den Zustandsgraphen Ihrer TM M_2 mit $\mathcal{L}(M_2) = L_2$.

Die folgenden Aufgaben werden nicht zum Bestehen der Studienleistung benötigt.

Übungsaufgabe 3:

Konstruieren Sie eine NTM, welche eine Eingabe $w \in \{0, 1\}^+$ akzeptiert, falls die Binärzahl w (most significant bit first) gerade ist.

Übungsaufgabe 4:

Konstruieren Sie eine kontext-sensitiv Grammatik für die Sprache

$$L_{<} := \{v\#w \mid v \in 0^* \cdot \text{bin}(n) \text{ UND } w \in 0^* \cdot \text{bin}(m) \text{ UND } |v| = |w| \text{ UND } n < m\} \subseteq \{0, 1, \#\}^* .$$

Geben Sie Ihre Grammatik formal als Tupel an. Erläutern Sie kurz, wie Ihre Grammatik die Sprache beschreibt. Geben Sie eine Ableitung Ihrer Grammatik für das Wort $001\#010 \in L_{<}$ an.

Übungsaufgabe 5:

Zeigen Sie, dass alle regulären Sprachen auch per TM erkannt werden können.

Übungsaufgabe 6:

Bisher haben wir Turing-Maschinen, wie vorherige Automatenmodelle, nur als Entscheider betrachtet, d.h. als eine Maschine, die nur Ja/Nein Antworten liefert. Allgemeiner können wir Turing-Maschinen aber auch benutzen um partielle Funktionen zu berechnen. Dazu betrachten wir den Bandinhalt beim Akzeptieren als die Ausgabe der Turing-Maschine. Falls die Maschine gar nicht akzeptiert, dann gibt es keine Ausgabe (daher partielle Funktion).

Konstruieren Sie eine Turing-Maschine M^- , die als Eingabe eine Binärzahl (most significant bit first) $w \in \{0, 1\}^+$ bekommt und diese um 1 verringert. D.h. die Maschine soll die Binärzahl um 1 verringern und dann akzeptieren. Sie können davon ausgehen, dass die eingegebene Zahl immer positiv ist und um 1 verringert werden kann. Geben Sie die TM formal als Tupel an und beschreiben Sie ihre Arbeitsweise.

Übungsaufgabe 7:

Konstruieren Sie nun eine Turing-Maschine M^{copy} , die die Funktion $w \mapsto w\#w$ ($w \in \{0, 1\}^*$) berechnet, d.h. als Eingabe ein Wort w bekommt und als Ausgabe $w\#w$ produzieren soll. Geben Sie die TM formal als Tupel an und beschreiben Sie ihre Arbeitsweise.

Übungsaufgabe 8:

Turing-Maschinen können als Subroutinen für kompliziertere Turing-Maschinen benutzt werden. Dies soll in der folgenden Aufgabe demonstriert werden.

Wir erweitern die Transitionsfunktion von Turing-Maschinen zu $\delta : Q \times \Gamma \rightarrow (Q \times \Gamma \times \{L, R, N\}) \cup (Q \times \{M_1, M_2, \dots, M_k\})$, wobei M_1 bis M_k Turing-Maschinen sind, die als Subroutine verwendet werden dürfen. Bei einer Transition $\delta(q, a) = (q', M_i)$ passiert folgendes:

Der Inhalt des Bandes unter und rechts vom Lese/Schreibkopf wird als Eingabe in die Turing-Maschine M_i gesteckt, und dann durch die Ausgabe von M_i ersetzt. Dabei bewegt sich der Lese/Schreibkopf nicht, aber die Turing-Maschine ändert ihren Zustand zu q' . Dies alles passiert in einem Berechnungsschritt. Sie können davon ausgehen, dass M_i eine totale Funktion darstellt, also immer eine Ausgabe hat.

Konstruieren Sie nun formal eine Turing-Maschine, die folgende Funktion berechnet. Zu einer Zahl n (binär-kodiert) soll die absteigende Sequenz $bin(n)\#bin(n-1)\#\dots\#bin(1)\#bin(0)$ berechnet werden ($bin(n)$ ist die binäre Kodierung der Zahl n , wie in Aufgabe 1). Sie dürfen dabei die Turing-Maschinen aus Aufgabe 3 als Subroutinen benutzen, aber auch eigene Subroutinen definieren (diese müssen Sie aber formal als TM definieren!). Erklären Sie die Arbeitsweise ihrer TM und geben Sie eine formale Beschreibung als Tupel an.