

# Theoretische Informatik 2

## Übungsblatt 1

René Maseli  
Prof. Dr. Roland Meyer

TU Braunschweig  
Sommersemester 2022

Ausgabe: 26.04.2022

Abgabe: 06.05.2022, 17:00

Geben Sie Ihre Lösungen bis Freitag, den 06.05.2022 um 17:00 Uhr, ab. Hinterlassen Sie sie dazu im passenden Fach des Briefkastens vor IZ 343 oder laden Sie sie in den passenden Ordner auf Stud.IP hoch. Achten Sie darauf, dass Studiengang, Name, Vorname und Matrikelnummer jedes Gruppenmitglieds lesbar vorne auf Ihrer Abgabe zu finden sind.

### Aufgabe 1: Kontext-sensitive Sprachen [5 Punkte]

Konstruieren Sie eine kontext-sensitive Grammatik für die Sprache

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

Dabei ist  $bin: \mathbb{N} \rightarrow \{0, 1\}^*$  die Binärdarstellung natürlicher Zahlen, startend mit dem most-significant bit, mit z.B.  $msb(0) = \varepsilon$  und  $msb(4) = 100$ . Beachten Sie, dass  $L_{<}$  vor den Zahlen padding benutzt.

1. [2 Punkte] Geben Sie Ihre Grammatik formal als Tupel  $\langle N, \Sigma, P, S \rangle$  an.
2. [1 Punkt] Erläutern Sie kurz, wie Ihre Grammatik die Sprache beschreibt.
3. [2 Punkte] Geben Sie eine Ableitung Ihrer Grammatik für das Wort  $001\#010 \in L_{<}$  an.

### Aufgabe 2: TMs im Vergleich zu NFAs [10 Punkte]

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

1. a) [1 Punkt] Geben Sie einen NFA  $A$  für die Sprache  $L = a^*. (b + bb). cc$  an.  
b) [2 Punkte] Konstruieren Sie nun formal eine TM  $M$ , die wie  $A$  die obige Sprache akzeptiert.  
c) [2 Punkte] Geben Sie die Konfigurationsabfolge der Berechnung Ihrer TM für die Eingabe  $aabbcc$  an.
2. Sei  $A = (Q, \Sigma, q_0, \rightarrow, Q_F)$  nun ein beliebiger NFA.  
a) [4 Punkte] Konstruieren Sie nun formal eine TM  $M$ , die die gleiche Sprache wie  $A$  akzeptiert. Erklären Sie die Arbeitsweise der TM.  
b) [1 Punkt] Wie viel Speicherplatz benötigt ihre TM. Handelt es sich um einen LBA?

### Aufgabe 3: Turing-Maschinen als partielle Funktionen [8 Punkte]

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).

1. [4 Punkte] 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.

2. [4 Punkte] 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.

### Aufgabe 4: Subroutinen für Turing-Maschinen [7 Punkte]

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  $bin(n)\#bin(n-1)\#bin(n-2)\#bin(n-3)\#\dots\#bin(1)\#bin(0)$  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.