

# Theoretische Informatik 2

## Übungsblatt 1

Thomas Haas  
Prof. Dr. Roland Meyer

TU Braunschweig  
Sommersemester 2021

Ausgabe: 27.04.2021

Abgabe: 07.05.2021, 18:00

Geben Sie Ihre Lösungen bis Freitag, 07.05.2021 18:00 Uhr, ab. Fertigen Sie dazu ihre Hausaufgaben direkt in .pdf Form an oder scannen ihre handschriftlichen Hausaufgaben ein und laden diese in ihrer Gruppe auf stud.ip hoch.

### Aufgabe 1: Turing-Maschinen [5 Punkte]

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

1. [2 Punkte] Geben Sie eine Beschreibung (Idee) der Arbeitsweise der TM an.
2. [2 Punkte] Geben Sie eine formale Beschreibung der NTM als Tupel an.
3. [1 Punkte] Geben Sie die Konfigurationsabfolge der Berechnung Ihrer NTM für die Eingabe 1110 an.

### Aufgabe 2: TMs im Vergleich zu NFAs [8 Punkte]

Die in Aufgabe 1 betrachtete Sprache ist regulär. Sie sollen nun zeigen, 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) [1 Punkt] 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) [3 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  $n$ ). 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.