



## Übungen zu Woche 10: Turingmaschinen

Das Übungsblatt enthält alle empfohlenen Lernaktivitäten für die aktuelle Woche.

- **Heimarbeit bis Montag 17:00.**
  - Schau die Videos an und lies die Buchkapitel.
  - Bearbeite die 🌱-Aufgabe in [Moodle](#). (Feste Abgabefrist!)
  - Lese den Aufgabentext aller Übungsaufgaben.
- **Heimarbeit.** Bearbeite die Übungsaufgaben soweit möglich. Probier zumindest alle mal!
- **Dienstag/Donnerstag.**
  - **8:00–8:15.** Besprechung im Hörsaal.
  - **8:15–9:15.** Bearbeite jetzt die Übungen, die du noch nicht lösen konntest. Sprich mit anderen Studis! Frag das Vorlesungsteam um Hilfe!
  - **9:15–9:45.** Lösungsspaziergang zu den Aufgaben für heute.
- **Heimarbeit bis Freitag, den 14.01., 17:00.** Gib deine Lösungen zu der ★-Aufgabe von diesem Übungsblatt in [Moodle](#) ab. (Feste Abgabefrist!)

## Dienstag

### Aufgabe 10.1 (Turingmaschinen I).

- a) Gegeben ist die Turingmaschine  $M_1 = (Q, \Sigma, \Gamma, \delta, q_0, \text{acc}, \text{rej}, \square)$  mit  $Q = \{q_0, q_1, q_2, q_3, q_4, \text{acc}, \text{rej}\}$ ,  $\Sigma = \{0, 1\}$  und  $\Gamma = \Sigma \cup \{\emptyset, \lambda, \square\}$ . Die Überföhrungsfunktion  $\delta$  ist gegeben durch:

$$\begin{array}{ll} \delta(q_0, 0) = (q_1, \emptyset, +1) & \delta(q_3, \lambda) = (q_3, \lambda, -1) \\ \delta(q_0, \square) = (\text{acc}, \square, +1) & \delta(q_3, 0) = (q_3, 0, -1) \\ \delta(q_0, \lambda) = (q_4, \lambda, +1) & \delta(q_3, \emptyset) = (q_0, \emptyset, +1) \\ \\ \delta(q_1, 0) = (q_1, 0, +1) & \delta(q_4, \lambda) = (q_4, \lambda, +1) \\ \delta(q_1, \lambda) = (q_1, \lambda, +1) & \delta(q_4, \square) = (\text{acc}, \square, -1) \\ \delta(q_1, 1) = (q_2, \lambda, +1) & \\ \\ \delta(q_2, 1) = (q_3, \lambda, -1) & \delta(q, a) = (\text{rej}, a, +1) \text{ sonst} \end{array}$$

Föhre  $M_1$  auf den beiden Eingaben 001111 und 00011111 aus. Gib jeweils die Konfiguration nach Ende der Berechnung, also bei Erreichen von **acc** oder **rej**, an.

Welche Sprache entscheidet  $M_1$ ?

- b) Gegeben ist die Turingmaschine  $M_2 = (Q, \Sigma, \Gamma, \delta, q_0, \text{halt}, \square)$  mit  $Q = \{q_0, q_1, q_2, q_3, q_4, q_5, q_6, \text{halt}\}$ ,  $\Sigma = \{0, 1\}$  und  $\Gamma = \Sigma \cup \{\emptyset, \lambda, \underline{0}, \underline{1}, \underline{\emptyset}, \underline{\lambda}, \square\}$ . Die Überföhrungsfunktion  $\delta$  ist gegeben durch:

$$\begin{aligned}\delta(q_0, 0) &= (q_1, \emptyset, +1) \\ \delta(q_0, 1) &= (q_2, \underline{1}, +1) \\ \delta(q_0, \square) &= (\text{halt}, \square, +1)\end{aligned}$$

$$\begin{aligned}\delta(q_4, 0) &= (q_1, \emptyset, +1) \\ \delta(q_4, 1) &= (q_2, \underline{1}, +1) \\ \delta(q_4, \underline{0}) &= (q_5, \underline{0}, -1) \\ \delta(q_4, \underline{1}) &= (q_5, \underline{1}, -1)\end{aligned}$$

$$\begin{aligned}\delta(q_1, x) &= (q_1, x, +1) \text{ f\"ur } x \in \{0, 1, \underline{0}, \underline{1}\} \\ \delta(q_1, \square) &= (q_3, \underline{0}, -1)\end{aligned}$$

$$\begin{aligned}\delta(q_5, z) &= (q_5, z, -1) \text{ f\"ur } z \in \{\emptyset, \underline{1}\} \\ \delta(q_5, \emptyset) &= (q_6, 0, +1) \\ \delta(q_5, \underline{1}) &= (q_6, 1, +1)\end{aligned}$$

$$\begin{aligned}\delta(q_2, x) &= (q_2, x, +1) \text{ f\"ur } x \in \{0, 1, \underline{0}, \underline{1}\} \\ \delta(q_2, \square) &= (q_3, \underline{1}, -1)\end{aligned}$$

$$\begin{aligned}\delta(q_6, \emptyset) &= (q_6, 0, +1) \\ \delta(q_6, \underline{1}) &= (q_6, 1, +1) \\ \delta(q_6, \underline{0}) &= (q_6, 0, +1) \\ \delta(q_6, \underline{1}) &= (q_6, 1, +1) \\ \delta(q_6, \square) &= (\text{halt}, \square, -1)\end{aligned}$$

$$\begin{aligned}\delta(q_3, x) &= (q_3, x, -1) \text{ f\"ur } x \in \{0, 1, \underline{0}, \underline{1}\} \\ \delta(q_3, y) &= (q_4, y, +1) \text{ f\"ur } y \in \{\emptyset, \underline{1}, \underline{0}, \underline{1}\}\end{aligned}$$

$$\delta(q, a) = (\text{halt}, \square, +1) \text{ sonst}$$

F\"uhre  $M_2$  auf den beiden Eingaben 0001 und 1010 aus. Gib jeweils die Konfiguration nach Ende der Berechnung, also bei Erreichen von **halt**, an.

Welche Funktion berechnet  $M_2$ ?

**Aufgabe 10.2 (Turingmaschinen II).** In [Abschnitt 6.3 aus Ericksons „Models of Computation“](#) ist eine Turingmaschine f\"ur die Sprache  $L = \{0^n 1^n 0^n \mid n \geq 0\}$  angegeben. Diese enth\"alt einige Fehler. Warum entscheidet die Turingmaschine nicht die Sprache  $L$  und wie lassen sich die Fehler beheben?

**Aufgabe 10.3 (Turingmaschinen III).** Sei  $L = \{0^n 1^m 0^{n+m} \mid n, m \in \mathbb{N}\}$ .

- Beschreibe auf informelle Art und Weise eine Turingmaschine, die die Sprache  $L$  entscheidet.
- Gib die beschriebene Turingmaschine formal an. Insbesondere ist also auch die \"Uberf\"uhrungsfunktion  $\delta$  der Maschine anzugeben.

## Donnerstag

**Aufgabe 10.4 (Turingmaschinen IV).** Beschreibe Turingmaschinen, die die folgenden Sprachen entscheiden, bzw. die folgenden Funktionen berechnen. Die Maschine muss nicht formal angegeben werden.

- $\{ww \mid w \in \{0, 1\}^*\}$
- $1^n 0 1^m \mapsto 1^{nm}$  (falls die Eingabe nicht von der Form  $1^n 0 1^m$  f\"ur gewisse  $n$  und  $m$  ist, darf die Maschine sich beliebig verhalten)
- $1^n \mapsto$  Bin\"ardarstellung von  $n$

**Aufgabe 10.5 (Rekursive Aufzählbarkeit).** Eine Sprache  $L \subseteq \Sigma^*$  heißt *rekursiv aufzählbar*, wenn  $L = \emptyset$  oder es eine berechenbare, totale Funktion  $f: \{0, 1\}^* \rightarrow \Sigma^*$  gibt, sodass  $f(\{0, 1\}^+) = L$ , also  $L = \{f(0), f(1), f(00), f(01), f(10), f(11), f(000), \dots\}$  gilt.

- a) (*schwer*) Zeige, dass jede rekursiv aufzählbare Sprache semi-entscheidbar ist.
- b) (*sehr schwer*) Zeige, dass jede semi-entscheidbare Sprache rekursiv aufzählbar ist.  
*Hinweis:* Du kannst verwenden, dass Turingmaschinen Paare von Zahlen aus einzelnen Binärwörtern dekodieren können und sich dabei jedes Paar aus einer gewissen Zahl ergibt. (Dies ist mit der Cantor-Abzählung möglich.)

### Aufgabe 10.6 (Turingmaschinen V).

- a) Bei einer Zwei-Band-Maschine gibt es zwei Bänder und jedes Band hat einen eigenen Lese-/Schreibkopf. Eine *Zwei-Stapel-Maschine* ist eine Turingmaschine mit zwei Bändern und dem folgendem eingeschränkten Verhalten:  
Zu jeder Zeit ist auf jedem Band jede Zelle rechts vom Kopf das Blanksymbol und jede Zelle links vom Kopf ist kein Blanksymbol. Deshalb kann sich der Kopf nur nach rechts bewegen, indem ein Symbol, das kein Blank ist, in eine Zelle mit Blank geschrieben wird. Der Kopf kann sich nur nach links bewegen, wenn die rechteste Zelle, die kein Blank ist, zu einem Blank wird. Jedes Band verhält sich also wie ein Stapel (*stack*) und die beiden Köpfe sind immer ganz oben auf dem Stapel. Um ein Überlaufen des Kopfes am Anfang des Bands zu vermeiden, gibt es dort ein spezielles Symbol, das nicht überschrieben werden kann. Zu Beginn enthält das erste Band den Input, wobei sich der Kopf über dem letzten Symbol der Eingabe befindet. Das andere Band ist, bis auf das spezielle Symbol am Anfang des Bandes, leer.  
Zeige, dass jede normale Turingmaschine durch eine Zwei-Stapel-Maschine simuliert werden kann. Gegeben ist also eine normale Turingmaschine  $M$ , und du sollst eine Zwei-Stapel-Maschine  $M'$  beschreiben, die genau die gleichen Eingaben akzeptiert und verwirft wie  $M$ .
- b) (*schwer*) Erkläre, warum nichtdeterministische Turingmaschinen genauso mächtig sind wie gewöhnliche Turingmaschinen. Erkläre also, warum die folgende Aussage gilt:

Für alle Sprachen  $L$  gibt es eine nichtdeterministische Turingmaschine, die  $L$  entscheidet, genau dann, wenn es eine gewöhnliche Turingmaschine gibt, die  $L$  entscheidet.

## Sternaufgabe

**Aufgabe 10.7 (★: Turingmaschine simulieren (Coderunner)).** Implementiere ein Programm, das eine gewöhnliche Turingmaschine als Eingabe erhält, diese simuliert, und die Ausgabe der Turingmaschine berechnet. Die genaue Ein- und Ausgabespezifikation des geforderten Programms findest du in Moodle.