

# Übungsblatt 1

## Aufgabe 1

Wahr oder falsch?

- (a) Es gibt überabzählbar unendlich viele Wörter über einem endlichen Alphabet.
- (b) Es gibt abzählbar unendlich viele berechenbare Funktionen  $f : \mathbb{N}^k \rightarrow \mathbb{N}$ .
- (c) Es gibt überabzählbar unendlich viele Funktionen  $f : \mathbb{N}^k \rightarrow \mathbb{N}$ .

## Lösung

- (a) falsch

Es gibt abzählbar unendlich viele Wörter über einem endlichen Alphabet. Eine mögliche Abzählung aller Wörter ergäbe sich z.B. aus der längen-lexikographischen Ordnung, bei der zuerst nach Länge und bei gleicher Länge alphabetisch sortiert wird.

- (b) wahr

Siehe Folien 7 / 8 der Vorlesung.

Es gibt höchstens abzählbar viele Maschinen / Programme, die eine Funktion der Form  $f : \mathbb{N} \rightarrow \mathbb{N}$  berechnen.

Dies lässt sich auf Funktionen der Form  $f : \mathbb{N}^k \rightarrow \mathbb{N}$  verallgemeinern.

- (c) wahr

Siehe Folie 8 der Vorlesung: Schon die Menge  $\{f \mid f : \mathbb{N} \rightarrow \mathbb{N}\}$  ist überabzählbar.

Für Funktionen der Form  $f : \mathbb{N}^k \rightarrow \mathbb{N}$  gilt dies also erst recht.

Mit der gleichen Technik wie in der Vorlesung können wir auch einen formalen Beweis dafür führen:

Angenommen  $\mathcal{F} = \{f \mid f : \mathbb{N}^k \rightarrow \mathbb{N}\}$ , die Menge alle Funktionen der Form  $f : \mathbb{N}^k \rightarrow \mathbb{N}$ , ist abzählbar.

Dann gibt es eine bijektive Abbildung  $F : \mathbb{N} \rightarrow \mathcal{F}$ .

Wir konstruieren eine Funktion  $g : \mathbb{N}^k \rightarrow \mathbb{N}$  mit

$$g(x_1, \dots, x_k) = f_{x_1}(x_1, \dots, x_k) + 1$$

wobei  $f_{x_1} = F(x_1)$ .

Da  $F$  bijektiv (und somit surjektiv) ist, muss es ein  $i \in \mathbb{N}$  geben mit  $F(i) = g$ .

Für dieses  $i$  gilt also  $g(i, x_2, \dots, x_k) = f_i(i, x_2, \dots, x_k)$ .

Dies ist aber ein Widerspruch zur Definition von  $g$  mit

$$g(i, x_2, \dots, x_k) = f_i(i, x_2, \dots, x_k) + 1.$$

## Aufgabe 2

Gegeben sei die Turingmaschine  $M = (Q, \Sigma, \Sigma \cup \{\square\}, \delta, z_0, \square, \{z_2\})$  mit  $Q = \{z_0, z_1, z_2\}$ ,  $\Sigma = \{0, 1\}$  und folgenden Transitionen:

$$\delta(z_0, 0) = (z_0, 0, R)$$

$$\delta(z_0, 1) = (z_0, 1, R)$$

$$\delta(z_0, \square) = (z_1, 0, L)$$

$$\delta(z_1, 0) = (z_1, 0, L)$$

$$\delta(z_1, 1) = (z_1, 1, L)$$

$$\delta(z_1, \square) = (z_2, \square, R)$$

- (a) Untersuchen Sie, wie sich die Turingmaschine auf den Eingaben 10, 11 und 110 verhält. Was tut sie allgemein bei Eingaben  $w \in 1\{0, 1\}^* \cup \{0\}$ ?
- (b) Wie verändert sich das Verhalten, wenn man die Transition  $\delta(z_0, \square) = (z_1, 0, L)$  durch  $\delta(z_0, \square) = (z_1, 1, L)$  ersetzt?
- (c) Ändern Sie die Turingmaschine  $M$  so ab, dass sie die Funktion  $f(n) = 4n + 1$  berechnet.

**Lösung**(a) Bei beliebigen Eingaben  $w \in \Sigma^*$  läuft die Turingmaschine bis zum Ende der Eingabe, ersetzt das erste Blanksymbol durch eine 0 und läuft dann wieder zum linken Ende der Eingabe zurück.

Aus der Eingabe 10 wird so 100, aus 11 wird 110 und aus 110 wird 1100.

Eingaben der Form  $w \in 1\{0, 1\}^* \cup \{0\}$  können als binär kodierte Zahlen verstanden werden.

Kodiert die Eingabe die Zahl  $n$ , so steht nach dem Durchlauf der Turingmaschine die Binärcodierung von  $2n$  auf dem Band, da das Anfügen einer 0 an eine Zahl im Binärsystem der Multiplikation der Zahl mit 2 entspricht.

- (b) Mit der Änderung wird am Bandende eine 1 statt einer 0 angefügt.

Interpretieren wir die Eingabe wieder als eine binär kodierte Zahl  $n$ , so steht nach dem Durchlauf die Binärcodierung von  $2n + 1$  auf dem Band.

- (c) Die Turingmaschine muss 01 an die Binärdarstellung des Eingabewortes anhängen. Somit modifizieren wir  $M$  wie folgt:

$M' = (Q', \Sigma, \Sigma \cup \{\square\}, \delta, z_0, \square, \{z_2\})$  mit  $Q' = \{z_0, z_1, z_2, z_3\}$ ,  $\Sigma = \{0, 1\}$  und folgende Transitionen:

$$\delta(z_0, 0) = (z_0, 0, R)$$

$$\delta(z_0, 1) = (z_0, 1, R)$$

$$\delta(z_0, \square) = (z_1, 0, R)$$

$$\delta(z_1, \square) = (z_3, 1, L)$$

$$\delta(z_3, 0) = (z_3, 0, L)$$

$$\delta(z_3, 1) = (z_3, 1, L)$$

$$\delta(z_3, \square) = (z_2, \square, R)$$

Für eine deterministische Maschine müssten formal noch  $\delta(z_1, a) = (z_1, a, N)$  ( $a \in \{0, 1\}$ ) hinzugefügt werden. Zur Vereinfachung lassen wir das weg.

### Aufgabe 3

Geben Sie eine 2-Band-Turingmaschine an, die bei Eingabe  $w \in \{a, b\}^*$  das Wort  $ww$  auf das erste Band schreibt, den Lesekopf auf das erste Zeichen von  $ww$  bewegt und in einen Endzustand übergeht.

### Lösung

Im Gegensatz zu einer 1-Band Konstruktion, ist eine 2-Band TM deutlich weniger aufwendig (Übung: Was ist die Idee bei einer 1-Band TM?). Die Idee ist, dass wir  $w$  von Band 1 auf Band 2 kopieren, mit dem Lesekopf auf Band 1 stehen bleiben, während wir auf Band 2 wieder an den Anfang fahren und zum Schluss die Kopie von  $w$  von Band 2 auf Band 1 hinter die Eingabe anhängen. Wir können ab diesem Zeitpunkt das zweite Band ignorieren und auf Band 1 an den Start zurücklaufen.

Formal sähe die eben beschriebene 2-Band TM wie folgt aus:

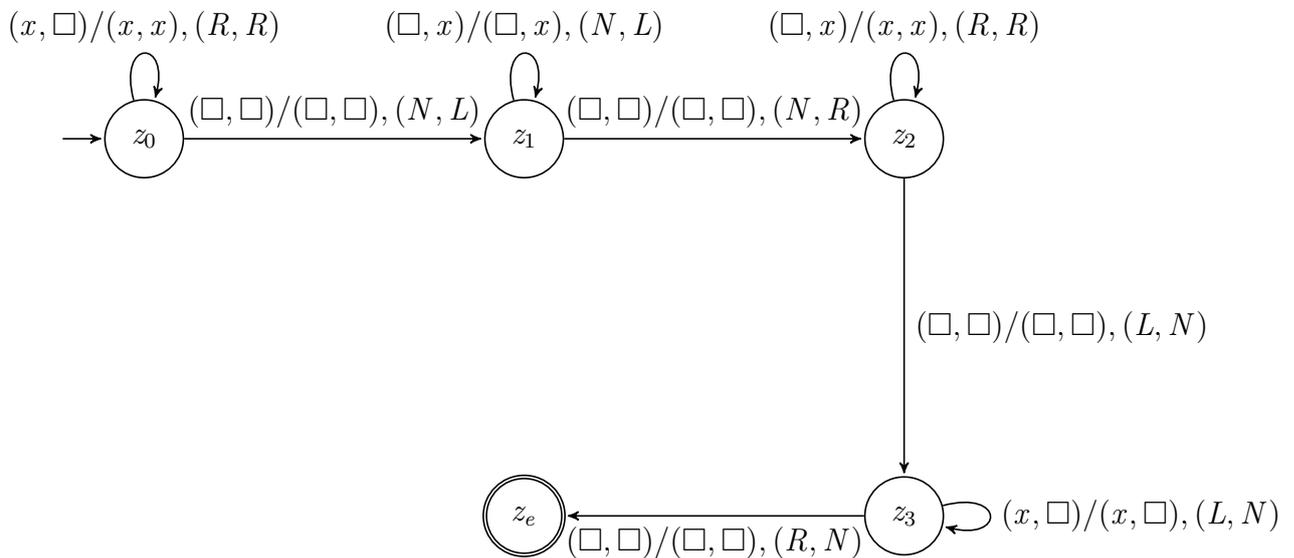
- $M = (Z, \Sigma, \Gamma, \delta, z_0, \square, \{z_e\})$
- $Z = \{z_0, z_1, z_2, z_3, z_e\}$
- $\Sigma = \{a, b\}$
- $\Gamma = \Sigma \cup \{\square\}$

Hierbei ist  $\delta$  gegeben durch

- $\delta(z_0, (x, \square)) = (z_0, (x, x), (R, R))$  für alle  $x \in \{a, b\}$
- $\delta(z_0, (\square, \square)) = (z_1, (\square, \square), (N, L))$
- $\delta(z_1, (\square, x)) = (z_1, (\square, x), (N, L))$  für alle  $x \in \{a, b\}$
- $\delta(z_1, (\square, \square)) = (z_2, (\square, \square), (N, R))$
- $\delta(z_2, (\square, x)) = (z_2, (x, x), (R, R))$  für alle  $x \in \{a, b\}$
- $\delta(z_2, (\square, \square)) = (z_3, (\square, \square), (L, N))$
- $\delta(z_3, (x, \square)) = (z_3, (x, \square), (L, N))$  für alle  $x \in \{a, b\}$
- $\delta(z_3, (\square, \square)) = (z_e, (\square, \square), (R, N))$

Auch hier müssten für nicht erreichbare Paare  $(z, \gamma) \in Z \times \Gamma^2$  eigentlich auch Übergänge definiert werden, wenn wir eine deterministische TM erhalten wollen.

Wie auch in FSA dürfen Sie die Diagrammrepräsentation für Turingmaschinen benutzen. Der Unterschied ist bei  $k$ -Band Turingmaschinen, dass im Diagramm Vektoren der Länge  $k$  verwendet werden.



Hierbei ist  $x \in \{a, b\}$ .

#### Aufgabe 4

Geben Sie (formal) Turingmaschinen  $M_1$  bzw.  $M_2$  an, die die Funktionen  $f_i : \mathbb{N}^2 \rightarrow \mathbb{N}$  mit

$$f_i(n_1, n_2) = n_i \quad (i = 1, 2)$$

berechnen.

## Lösung

Zu Beginn steht  $\text{bin}(n_1)\#\text{bin}(n_2)\#$  auf dem Band. Nach einem Durchlauf der TM für  $f_i, i \in \{1, 2\}$  soll nur noch  $\text{bin}(n_i)$  auf dem Band stehen.

Beachte: Der Lesekopf der TM muss nach dem Durchlauf auf dem ersten Symbol von  $\text{bin}(n_1)$  bzw.  $\text{bin}(n_2)$  stehen.

### TM für $f_1$

- $M_1 = (Z_1, \Sigma_1, \Gamma_1, \delta_1, z_0, \square, \{z_e\})$
- $Z_1 = \{z_0, z_1, z_2, z_3, z_e\}$
- $\Sigma_1 = \{0, 1, \#\}$
- $\Gamma_1 = \Sigma \cup \{\square\}$

Dabei ist  $\delta_1$  wie folgt definiert:

#### $z_0$ : Überspringen von $\text{bin}(n_1)$

- $\delta_1(z_0, x) = (z_0, x, R)$  für alle  $x \in \{0, 1\}$
- $\delta_1(z_0, \#) = (z_1, \#, R)$

Wir durchlaufen das Band von links nach rechts, bis wir auf ein  $\#$  treffen. Dann gehen wir in  $z_1$  über, um den restlichen Bandinhalt zu löschen.

#### $z_1$ : Löschen des restlichen Bandinhaltes

- $\delta_1(z_1, x) = (z_1, \square, R)$  für alle  $x \in \{0, 1, \#\}$
- $\delta_1(z_1, \square) = (z_2, \square, L)$

Alle Symbole von  $\text{bin}(n_2)\#$  werden mit  $\square$  überschrieben. Wenn das rechte Bandende erreicht ist, gehen wir in  $z_2$  über, um zurück zum linken Bandende zu laufen.

#### $z_2, z_3$ : Zurück zum linken Bandende

- $\delta_1(z_2, \square) = (z_2, \square, L)$
- $\delta_1(z_2, \#) = (z_3, \square, L)$
- $\delta_1(z_3, x) = (z_3, x, L)$  für alle  $x \in \{0, 1\}$
- $\delta_1(z_3, \square) = (z_e, \square, R)$

Hier brauchen wir zwei Zustände, da die  $\square$ 's rechts und links von  $\text{bin}(n_1)$  unterschiedlich behandelt werden.

Das mittlere  $\#$  haben wir zu Beginn mit  $z_0$  stehen gelassen um diesen Übergang zu markieren.

**TM für  $f_2$**

- $M_2 = (Z_2, \Sigma_2, \Gamma_2, \delta_2, z_0, \square, \{z_e\})$
- $Z_2 = \{z_0, z_1, z_2, z_e\}$
- $\Sigma_2 = \{0, 1, \#\}$
- $\Gamma_2 = \Sigma_2 \cup \{\square\}$

Dabei ist  $\delta_2$  wie folgt definiert:

$z_0$ : Löschen von  $\text{bin}(n_1)$

- $\delta_2(z_0, x) = (z_0, \square, R)$  für alle  $x \in \{0, 1\}$
- $\delta_2(z_0, \#) = (z_1, \square, R)$

$z_1$ : Löschen des letzten  $\#$

- $\delta_2(z_1, x) = (z_1, x, R)$  für alle  $x \in \{0, 1\}$
- $\delta_2(z_1, \#) = (z_2, \square, L)$

$z_2$ : Zurück zum Anfang von  $\text{bin}(n_2)$

- $\delta_2(z_2, x) = (z_2, x, L)$  für alle  $x \in \{0, 1\}$
- $\delta_2(z_2, \square) = (z_e, \square, R)$