ÜBUNG 10

Aufgabe 1

Wie in der Vorlesung dargelegt wurde, werden Turingmaschinen als allgemeines Rechenmodell verstanden.

Geben Sie Turingmaschinen an, die folgende Funktionen berechnen. Dabei wird eine Eingabe $n \in \mathbb{N}$ als $\emptyset^n$ mit $\emptyset \in \Sigma$ dargestellt. Es kann vorausgesetzt werden, dass die Eingabe wohlgeformt auf dem Band vorliegt. Am Ende der Berechnung hält die Turingmaschine in einem Finalzustand und das Band enthält nur das Berechnungsergebnis.

a) Die Turingmaschine $\mathcal{M}_0$ berechnet die Funktion $f : \mathbb{N} \to \mathbb{N}$, $n \mapsto 0$, d. h. das Eingabewort auf dem Band wird gelöscht.

Idee: lösche alle $\emptyset$ der Unärkodierung von n

$\mathcal{M}_0 := (Q, \Sigma, \Gamma, \delta_0, q_0, F)$ mit

$Q := \{q_0, q_F\}$
$\Sigma := \{\emptyset\}$
$\Gamma := \{\emptyset, \_\}$
$F := \{q_F\}$

$\delta(q_0, \emptyset) = (q_0, \_, R)$ "Löschen"
$\delta(q_0, \_) = (q_F, \_, N)$ "Fertig" (auch $f(0) = 0$)

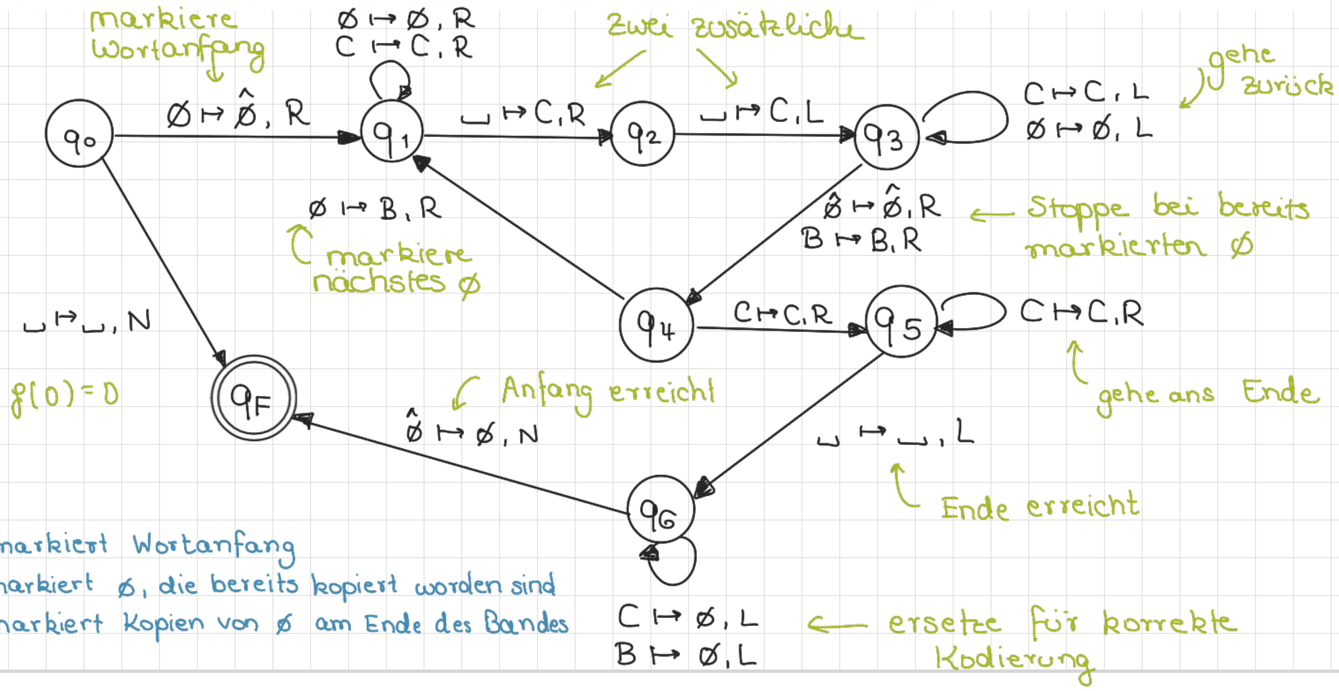
b) Die Turingmaschine $\mathcal{M}_{succ}$ berechnet die Funktion $f : \mathbb{N} \to \mathbb{N}$, $n \mapsto n + 1$.

$\mathcal{M}_{succ} := (\{q_0, q_F\}, \{\emptyset\}, \{\emptyset, \_\}, \delta, q_0, \{q_F\})$

$\delta(q_0, \emptyset) = (q_0, \emptyset, R)$ "Gehe ans Ende"
$\delta(q_0, \_) = (q_F, \emptyset, N)$ "Füge ein $\emptyset$ hinzu"

Beispiel $q_0 \emptyset\emptyset \to A\, q_1 \emptyset \to A \emptyset q_1 \to A \emptyset CC \to A\, q_4 \emptyset CC \to AB\, q_1 CC \to ABCCCC$

c) Die Turingmaschine $\mathcal{M}_{\times 3}$ berechnet die Funktion $f : \mathbb{N} \to \mathbb{N}$, $n \mapsto 3 \cdot n$.



$\hat{\emptyset}$ markiert Wortanfang
B markiert $\emptyset$, die bereits kopiert worden sind
C markiert Kopien von $\emptyset$ am Ende des Bandes

Aufgabe 2

Wir betrachten die Sprache

$$L = \{a^n b^m c^k : n, m, k \geq 1, n = 2m \text{ oder } m = k\}.$$

Zeigen Sie, dass L von Typ 1 ist, indem Sie einen LBA skizzieren, der L entscheidet.

Arbeitsalphabet: $\Gamma = \{a', b', c', b'', b'''\} \cup \Sigma$

$\boxed{\text{Phase 1}}$

testet $n = 2m$

(i) ersetze a durch a', laufe nach rechts zum nächsten b und ersetze durch b'

(ii) erfolgreich $\rightarrow$ zurück zu (i)

(iii) keine b mehr: wie (i) aber b' durch b'' ersetzen

(iv) keine a mehr: zu (v)

v) Prüfe, ob das Band die Form $(a')^+ (b'')^+ (c)^+$ hat

- ja: akzeptiere
- nein: zu Phase 2

$\boxed{\text{Phase 2:}}$

(i) ersetze ein b, b' oder b'' durch b''' (alle äquivalent)
laufe nach rechts zum nächsten c und ersetze durch c'

(ii) wiederhole bis keine b, b', b'' mehr auf dem Band sind

(iii) Prüfe ob das Band ab dem ersten b die Form $(b''')^+ (c')^+$ hat

- ja: akzeptiere
- nein: verwerfe

Aufgabe 3

Im Folgenden bezeichne $\mathcal{M}_w$ eine deterministische Turingmaschine mit einem Band und dem Eingabealphabet $\Sigma = \{0, 1, \#\}$, deren Codewort $\mathsf{enc}(\mathcal{M}_w)$ gleich w ist, falls es ein solches Codewort mit $\mathsf{enc}(\mathcal{M}_w) = w$ gibt (vgl. Vorlesung 19, Folie 27). Andernfalls ist $\mathcal{M}_w = \mathcal{M}_\perp$, eine fest gewählte deterministische Turingmaschine mit dem Eingabealphabet $\Sigma = \{0, 1, \#\}$, die für alle Eingabewörter endlos läuft.

Ist die nachfolgende Sprache entscheidbar?

$$L = \{ w \in \{0,1,\#\}^* \mid \text{es gibt ein Wort } z \in \{0,1,\#\}^* \text{ mit } |z| \leqslant |w|^2, \text{ so dass } \mathcal{M}_w \text{ das Eingabewort } z \text{ in höchstens } |z| \text{ Schritten akzeptiert} \}$$

Konstruktion einer NTM $\mathcal{M}_L$, die L entscheidet:

$\mathcal{M}_L$: Eingabe: w

1. Rate ein Wort $z \in \{0,1,\#\}^*$ mit $|z| \leq |w|^2$
2. Simuliere $\mathcal{M}_w$ auf z für maximal $|z|$ Schritte

   Wenn $\mathcal{M}_w$ z akzeptiert (in $|z|$ Schritten) : akzeptiere
   sonst : verwerfe

$\Rightarrow$ L ist entscheidbar

Aufgabe 4

Sei $\mathcal{M}_w$ wie in Aufgabe 3 und

$$t_{\mathcal{M}_w}(x) := \text{Anzahl der Schritte, die } \mathcal{M}_w \text{ bei Eingabe } x \text{ durchführt.}$$

Ist die Sprache $L = \{w \in \{0,1\}^* \mid t_{\mathcal{M}_w}(w) > 2^{|w|}\}$ entscheidbar? Begründen Sie Ihre Antwort.

Konstruktion einer DTM $\mathcal{M}_L$, die L entscheidet:

$\mathcal{M}_L$: Eingabe: w

1. Prüfe, ob w ein gültiges Encoding einer TM $\mathcal{M}_w$ ist
   - nein: akzeptiere (denn $\mathcal{M}_\perp$ läuft endlos)
   - ja: gehe zu Schritt 2
2. Simuliere $\mathcal{M}_w$ auf w für maximal $2^{|w|}+1$ Schritte

   Wenn $\mathcal{M}_w$ nach $\leq 2^{|w|}$ Schritten hält : verwerfe
   sonst : akzeptiere

Aufgabe 5

(a) Zeigen Sie, dass die Summe der ersten n ungeraden Zahlen n^2 ergibt; also dass

$$\sum_{i=1}^{n}(2i-1) = n^2.$$

(b) Geben Sie eine Grammatik für die Sprache

$$L = \{0^n : n \text{ ist eine Quadratzahl}\}$$

an.

(c) Geben sie eine deterministische Zwei-Band-Turingmaschine an, die L akzeptiert.

a) Induktion nach n: $n=1$: $2 \cdot 1 - 1 = 1 = 1^2$

$n \rightsquigarrow n+1$: $\sum_{i=1}^{n+1}(2i-1) = 2(n+1)-1 + \sum_{i=1}^{n}$

$= 2n+1 + n^2$ (IV)

$= (n+1)^2$ (binomische Formel)

b) Idee: nutze (a) und erzeuge Summe der ersten n ungeraden Zahlen

↳ Erzeuge 0 und N

Anzahl = aktuelle Summe

aktuelle Obergrenze der Summanden

Grammatik "simuliert" eine TM: A ... Anfangsmarker, E ... Endmarker, I ... Lese-/Schreibkopf

Start: I links bei A ⇝ dann entweder akzeptiere (und lösche Nichtterminale)
oder addiere weitere Zahl
↳ aus jedem N wird 0 und N und 00
↳ und zwei N's
↳ in Ausgangszustand zurück

⟶ $G = \langle V, \Sigma, P, S \rangle$ mit $V := \{S, A, E, I, F, N, R\}$, $\Sigma := \{0\}$ und

P:

mit 0^0 oder 0^1 beginnen: $S \to AI0NE \mid \varepsilon$

weitere Zahl addieren: $I0 \to 0I$, $IN \to 0NI$, $IE \to 00NNRE$

in Ausgangszustand zurück: $NR \to RN$, $0R \to R0$, $AR \to AI$

akzeptieren, d.h. Beenden: $AI \to F$
$F0 \to 0F$, $FN \to F$, $0FE \to 0$

Beispiel: $S \to AI0NE \to A0INE \to A00NIE \to A00N00NNRE$ $n=3$ $\Sigma=4$
$\to^* AI00N00NNE$
$\to F00N00NNE \to^* \underbrace{0000}_{2^2=4}$

c) deterministische 2-Band-TM:

Start: Band 1 - Eingabe, Band 2 - leer

(1) q_0 akzeptiert ε oder markiert Anfang von Band 2 mit A und geht zu q_1

$\delta(q_0, \_, \_) = (q_F, \langle \_, N\rangle, \langle \_, N\rangle)$ (leeres Wort)

$\delta(q_0, 0, \_) = (q_1, \langle 0, N\rangle, \langle A, R\rangle)$ (A auf Band 2)

(2) q_1: für jede 0 auf B1, für die es 0 oder A auf B2 gibt - gehe nach rechts

$\delta(q_1, 0, A) = (q_1, \langle 0, R\rangle, \langle A, R\rangle)$ (nach rechts)

$\delta(q_1, 0, 0) = (q_1, \langle 0, R\rangle, \langle 0, R\rangle)$ (zähle Nullen ab)

Ende von beiden Bändern wird gleichzeitig erreicht: akzeptiere

$\delta(q_1, \_, \_) = (q_F, \langle \_, N\rangle, \langle \_, N\rangle)$ (Akzeptanz)

Ende von B2 aber nicht B1: füge 0 zu B2 hinzu, gehe nach rechts und in q_2

$\delta(q_1, 0, \_) = (q_2, \langle 0, N\rangle, \langle 0, R\rangle)$ (plus Null)

(3) q_2: füge 0 zu B2 hinzu und wechsle in q_3

$\delta(q_2, 0, \_) = (q_3, \langle 0, N\rangle, \langle 0, L\rangle)$ (plus Null)

(4) q_3: gehe auf B2 nach links zum Anfang $\leadsto$ weiter mit q_1

$\delta(q_3, 0, 0) = (q_3, \langle 0, N\rangle, \langle 0, L\rangle)$ (nach links auf Band 2)

$\delta(q_3, 0, A) = (q_1, \langle 0, R\rangle, \langle A, R\rangle)$ (Anfang von Band 2)

<u>Bsp.:</u>

$$\underset{q_0}{\begin{pmatrix} 0 & 0 & 0 & 0 \\ \_ & & & \end{pmatrix}} \rightarrow \underset{q_1}{\begin{pmatrix} 0 & 0 & 0 & 0 \\ A & & & \end{pmatrix}} \rightarrow \underset{q_1}{\begin{pmatrix} 0 & 0 & 0 & 0 \\ A & \_ & & \end{pmatrix}} \rightarrow \underset{q_2}{\begin{pmatrix} 0 & 0 & 0 & 0 \\ A & 0 & \_ & \end{pmatrix}} \rightarrow \underset{q_3}{\begin{pmatrix} 0 & 0 & 0 & 0 \\ A & 0 & 0 & \_ \end{pmatrix}}$$

$$\rightarrow \underset{q_1}{\begin{pmatrix} 0 & 0 & 0 & 0 \\ A & 0 & 0 & \_ \end{pmatrix}} \rightarrow \underset{q_1}{\begin{pmatrix} 0 & 0 & 0 & 0 \\ A & 0 & 0 & \_ \end{pmatrix}} \rightarrow \underset{q_F}{\begin{pmatrix} 0 & 0 & 0 & 0 & \_ \\ A & 0 & 0 & \_ & \end{pmatrix}}$$

AKZEPTIERT