Nachklausur zur Vorlesung „Grundlagen der Theoretischen Informatik“

WS 2019/20 / 24. Februar 2020

Vorname: ______________________________

Nachname: ______________________________

Matrikelnummer: ______________________________

Aufgabe	Punktzahl	Erreicht
1	20	
2	9	
3	6	
4	6	
5	6	
6	6	
7	6	
8	6	
9	6	
10	4	
11	3	
12	6	
13	6	
14	4	
15	6	
16	0	
Σ	100	

Generelle Hinweise:

- Prüfungsdauer: **180 Minuten**
- Wenn Sie in der Klausur **50 Punkte** erreichen, haben Sie mit Sicherheit bestanden.
- Hilfsmittel: Ein beidseitig beschriebenes DIN-A4-Blatt.
- Benutzen Sie ein **dokumentenechtes Schreibgerät**.
- Überprüfen Sie die Ihnen ausgehändigte Klausur auf Vollständigkeit (**16 Aufgaben** auf 16 Seiten).
- Tragen Sie **auf jedes Blatt** Ihren **Namen** und Ihre **Matr.-Nr.** in die entsprechenden Felder ein.
- Schreiben Sie Ihre Lösungen in die dafür vorgesehenen Felder. Reicht der Platz in einem Feld nicht aus, so benutzen Sie die Rückseite des entsprechenden Blattes und vermerken Sie dies auf der Vorderseite. Reicht der Platz dennoch nicht aus, können Sie die Aufsicht nach zusätzlichen Blättern fragen.
- Schreiben Sie bitte **deutlich**. Unleserliche Lösungen sind ungültig.
- Ein **Täuschungsversuch** führt umgehend zum Ausschluss und **Nichtbestehen**. Es erfolgt keine Vorwarnung.
- Alle mitgeführten **elektronischen Geräte** sind vor der Klausur bzw. spätestens jetzt auszuschalten.

Inhaltliche Hinweise:

- Endliche Automaten können wahlweise grafisch oder tabellarisch angegeben werden.
- In den WHILE- und LOOP-Programmen dürfen Sie die Addition, Multiplikation, die in der Vorlesung definierte Subtraktion und für eine Variable x die Bedingung IF $x = 0$ THEN P END als WHILE- bzw. LOOP-berechenbar voraussetzen und in Ihren Programmen verwenden. Geben Sie an, in welcher Variable der Ausgabewert am Ende steht.
- Sie dürfen annehmen, dass die Addition und Multiplikation zweier natürlicher Zahlen primitiv-rekursiv sind.
- Für die Konstruktion von μ- bzw. primitiv-rekursiven Funktionen und für WHILE- und LOOP-Programme darf die Schreibweise aus den Übungen benutzt werden.

Aufgabe 1. (20 Punkte) Beantworten Sie die folgenden zehn Fragen. Für jede Teilaufgabe gibt es zwei Punkte, wenn Sie die Aufgabe vollständig und korrekt beantwortet haben.

(1) Sei Σ ein Alphabet mit $|\Sigma| \geq 2$. Beweisen oder widerlegen Sie: Für alle $u, v \in \Sigma^*$ gilt $uv = vu$.

> **Lösung:** Falsch. Gegenbeispiel: $\Sigma = \{a, b\}$, $u = a$, $v = b$. Dann ist $uv = ab \neq ba = vu$.

(2) Gegeben sei der NFA $M = (Q, \Sigma, \delta, S, F)$. Definieren Sie formal den Potenzmengenautomat von M.

> **Lösung:** Potenzmengenkonstruktion: $M' = (2^Q, \Sigma, \delta', S, F')$, wobei
> $\delta'(A, a) = \bigcup_{q \in A} \delta(q, a)$ für $A \subseteq Q, a \in \Sigma$ und
> $F' = \{A \subseteq Q \mid A \cap F \neq \emptyset\}$.

(3) Nennen Sie alle Formalismen für die Beschreibung regulärer Sprachen, welche Sie in der Vorlesung kennengelernt haben.

> **Lösung:** NFAs, DFAs, reguläre Ausdrücke, reguläre Grammatiken

(4) Die Sprache $L = \{a^n b^n \mid n \in \mathbb{N}\}$ ist nicht regulär. Geben Sie reguläre Sprachen A, B an, sodass $A \subseteq L \subseteq B$.

> **Lösung:** $A = \emptyset$, $B = \{a, b\}^*$

(5) Sei M ein DFA. Sei ferner Q die Zustandsmenge und Σ das Alphabet. Definieren Sie die Übergangsfunktion von M formal (d.h. Definitions- und Wertebereich).

> **Lösung:** $\delta : Q \times \Sigma \to Q$

(6) Gegeben sei die Grammatik $G = (V, \Sigma, P, S)$ mit $V = \{S\}$, $\Sigma = \{a, b, c\}$ und $P = \{S \to aSa \mid bSb \mid c\}$. Beschreiben Sie die Sprache $L(G)$ möglichst einfach.

Lösung: $L(G) = \{wcw^R \mid w \in \{a, b\}^*\}$

(7) Welche Laufzeit hat der CYK-Algorithmus?

Lösung: $\mathcal{O}(n^3)$

(8) Gegeben seien die vier Operatoren $\cup$, $\cap$, $^-$ und $\circ$ (Vereinigung, Schnitt, Komplement und Konkatenation). Unter welchen dieser Operatoren sind kontextfreie Sprachen abgeschlossen?

Lösung: Abschluss nur unter Vereinigung und Konkatenation.

(9) Geben Sie eine entscheidbare Sprache und eine unentscheidbare Sprache an.

Lösung: Sei $\Sigma = \{0, 1\}$.
Σ^* ist entscheidbar,
$\{w \in \Sigma^* \mid$ Die Turingmaschine M_w hält angesetzt auf ein leeres Band$\}$ ist unentscheidbar.

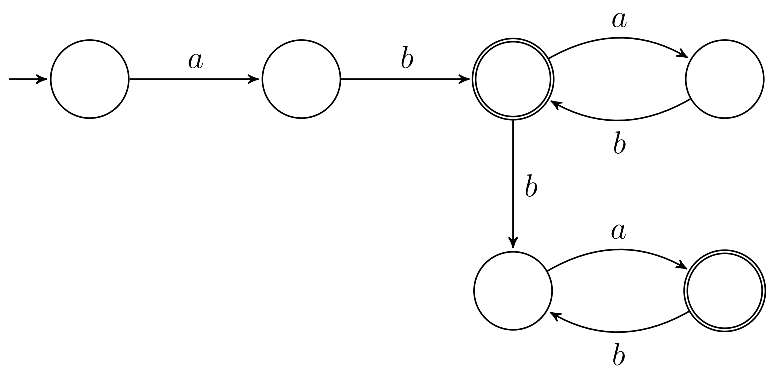
(10) Begründen Sie, dass es kein Programm geben kann, das entscheidet, ob ein WHILE-Programm auf einer beliebigen Eingabe terminiert.

Lösung: Jedes WHILE-Programm kann in eine Turingmaschine umgewandelt werden, die genau dann hält, wenn das WHILE-Programm terminiert. Das Halteproblem für Turingmaschinen ist aber unentscheidbar.

Aufgabe 2. (9 Punkte) Sei $\Sigma = \{a, b\}$. Geben Sie für jede der folgenden Sprachen einen endlichen Automaten und einen regulären Ausdruck an.

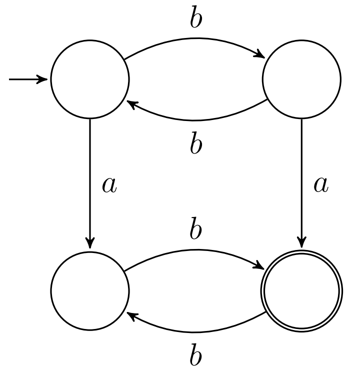
(a) $L_1 = \{(ab)^n(ba)^m \mid n \geq 1, m \geq 0\}$

Lösung: $(ab)(ab)^*(ba)^*$



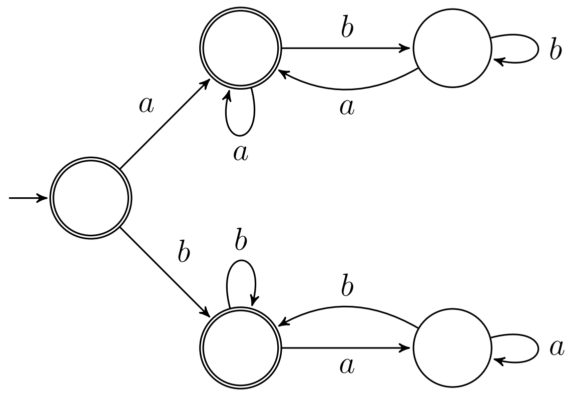
(b) $L_2 = \{w \in \Sigma^* \mid w$ enthält genau ein a und $|w|$ ist gerade$\}$

Lösung: $(bb)^*(ba|ab)(bb)^*$



(c) $L_3 = \{w \in \Sigma^* \mid w$ beginnt und endet mit dem gleichen Buchstaben$\}$

Lösung: $\varepsilon|a|b|(a(a|b)^*a)|(b(a|b)^*b)$

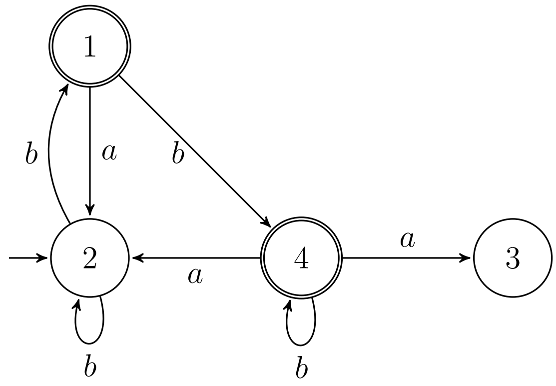


Aufgabe 3. (6 Punkte) Gegeben sei der NFA $M = (\{1,2,3,4\}, \{a,b\}, \delta, \{2\}, \{1,4\})$, wobei δ gegeben ist durch:

δ	a	b
1	$\{2\}$	$\{4\}$
2	$\emptyset$	$\{1,2\}$
3	$\emptyset$	$\emptyset$
4	$\{2,3\}$	$\{4\}$

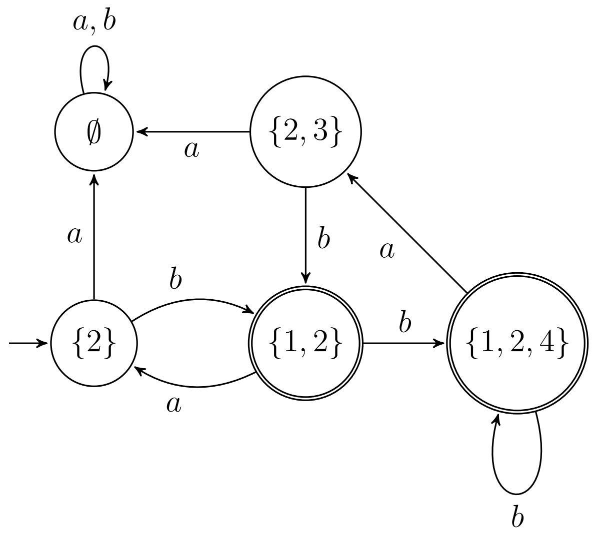
(a) Zeichnen Sie das zu M gehörige Automatendiagramm.

Lösung:

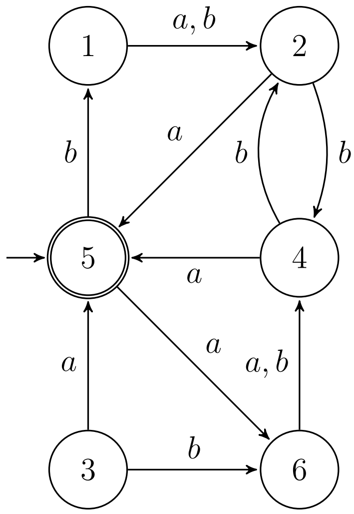


(b) Geben Sie mittels Potenzmengenkonstruktion einen zu M äquivalenten DFA an. Es genügt, den vom Startzustand erreichbaren Teil anzugeben.

Lösung:



Aufgabe 4. (6 Punkte) Minimieren Sie den folgenden DFA mit dem Algorithmus aus der Vorlesung. **Geben Sie an, welche Zustandspaare in welcher Reihenfolge markiert werden, und zeichnen Sie den erhaltenen minimalen DFA**.

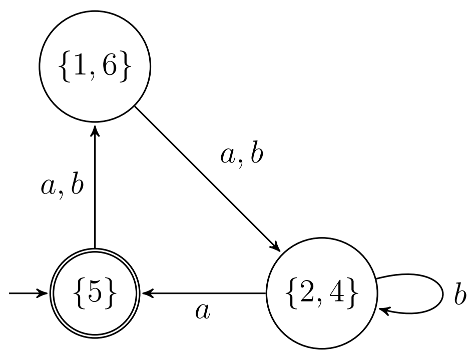


Lösung: Zustand 3 wird entfernt, weil nicht erreichbar.

2	2	-	-	-
4	3		-	-
5	1	1	1	-
6		4	5	1
	1	2	4	5

1. Trenne End- und Nichtendzustände.
2. Markiere $(1, 2)$ wegen $\delta(1, a) = 2$ und $\delta(2, a) = 5$ und $(2, 5)$ markiert.
3. Markiere $(1, 4)$ wegen $\delta(1, a) = 2$ und $\delta(4, a) = 5$ und $(2, 5)$ markiert.
4. Markiere $(2, 6)$ wegen $\delta(2, a) = 5$ und $\delta(6, a) = 4$ und $(4, 5)$ markiert.
5. Markiere $(4, 6)$ wegen $\delta(4, a) = 5$ und $\delta(6, a) = 4$ und $(4, 5)$ markiert.

1 und 6 sowie 2 und 4 sind erkennungsäquivalent.



Aufgabe 5. (6 Punkte) Für $w = a_1 a_2 \cdots a_n$ mit $a_i \in \{0,1\}$ für alle $i \in \{1,\ldots,n\}$ definieren wir $\bar{w} = b_1 b_2 \cdots b_n$, so dass $b_i = 1 - a_i$ für alle $i \in \{1,\ldots,n\}$ gilt. In anderen Worten, $\bar{w}$ entsteht aus w durch invertieren aller Bits in w (z.B. $w = 110$, $\bar{w} = 001$).

Gegeben ist nun folgende Sprache $L \subseteq \{0,1\}^*$:

$$L = \{w\bar{w} \mid w \in \{0,1\}^+\}.$$

Zeigen Sie, dass L nicht regulär ist.

Lösung: Sei $n \in \mathbb{N}$. Wähle $w = 0^n 1^n$. Es gilt $|w| \geq n$ und $w \in L$, weil $\bar{0^n} = 1^n$. Sei $w = xyz$ mit $|xy| \leq n$ und $y \neq \varepsilon$ eine Zerlegung von w. Dann gibt es $\ell, k \in \mathbb{N}$ mit $x = 0^\ell$, $y = 0^k$ mit $k \geq 1$ und $z = 0^{n-k-\ell}1^n$. Sei $w' := xy^0z = 0^\ell 0^{n-\ell-k}1^n = 0^{n-k}1^n$. Weil $k \geq 1$, gilt $w' \notin L$, denn es gibt kein $u \in \{0,1\}^*$ mit $w' = u\bar{u}$. Nach dem Pumping-Lemma ist L nicht regulär.

Aufgabe 6. (6 Punkte) Gegeben sei der folgende nichtdeterministische Kellerautomat $M = (Z, \Sigma, \Gamma, \delta, z_0, \#)$, wobei $Z = \{z_0\}$, $\Sigma = \{a, b\}$, $\Gamma = \{\#, A, B\}$ und

(1) $\delta(z_0, a, \#) = \{(z_0, A\#)\}$

(2) $\delta(z_0, b, \#) = \{(z_0, B\#)\}$

(3) $\delta(z_0, a, A) = \{(z_0, AA)\}$

(4) $\delta(z_0, a, B) = \{(z_0, \varepsilon)\}$

(5) $\delta(z_0, b, A) = \{(z_0, \varepsilon)\}$

(6) $\delta(z_0, b, B) = \{(z_0, BB)\}$.

Die Modifikationen aus Teilaufgabe (b) und (c) werden für jede weitere Teilaufgabe beibehalten.

(a) Welche Sprache wird von dem Kellerautomaten erkannt?

Lösung: $L(M) = \emptyset$

(b) Füge die Transition (7) $\delta(z_0, \varepsilon, \#) = \{(z_0, \varepsilon)\}$ hinzu. Welche Sprache erkennt der Kellerautomat nach dieser Modifikation?

Lösung: $L(M) = \{w \in \Sigma^* \mid \#_a(w) = \#_b(w)\}$

(c) Ändere Transition (4) in $\delta(z_0, a, B) = \{(z_0, B)\}$ und Transition (5) in $\delta(z_0, b, A) = \{(z_0, A)\}$. Welche Sprache erkennt der Automat nun?

Lösung: $L(M) = \{\varepsilon\}$

(d) Wir möchten nun, dass $L(M) = \Sigma^*$ gilt. Welche kleine Änderung muss dafür bei Transition (1) und (2) gemacht werden?

Lösung: A und B werden nicht auf den Keller geschrieben, d.h. wir erhalten mit den Transitionen (1) $\delta(z_0, a, \#) = \{(z_0, \#)\}$ und (2) $\delta(z_0, b, \#) = \{(z_0, \#)\}$ genau $L(M) = \Sigma^*$.

Aufgabe 7. (6 Punkte) Gegeben sei die Grammatik $G = (\{S, A, B, C\}, \{a, b\}, S, P)$ mit

$$\begin{aligned} P : S &\to b \mid AB \\ A &\to CS \\ B &\to b \\ C &\to a. \end{aligned}$$

Testen Sie mit dem CYK-Algorithmus, ob das Wort $aabbb$ in $L(G)$ enthalten ist.

Lösung: Der CYK-Algorithmus liefert folgende Tabelle:

Länge	a	a	b	b	b
1	C	C	S, B	S, B	S, B
2	-	A	-	-	-
3	-	S	-	-	-
4	A	-	-	-	-
5	S	-	-	-	-

Also ist $aabbb \in L(G)$.

Aufgabe 8. (6 Punkte) Geben Sie für

$$L = \{w \in \{a,b\}^* \mid w \text{ enthält mindestens so viele } a\text{'s wie } b\text{'s}\}$$

eine kontextfreie Grammatik an.

Lösung: $S \to \varepsilon \mid a \mid SS \mid aSb \mid bSa$

Aufgabe 9. (6 Punkte) Sei $M = (\{z_0, z_1, z_2, z_f\}, \{a, b\}, \{a, b, \Box\}, \delta, z_0, \Box, \{z_f\})$ eine deterministische Turingmaschine mit

$$
\begin{aligned}
\delta(z_0, a) &= (z_1, a, R) \\
\delta(z_0, b) &= (z_0, b, N) \\
\delta(z_0, \Box) &= (z_0, \Box, N) \\
\delta(z_1, a) &= (z_1, a, R) \\
\delta(z_1, b) &= (z_1, b, R) \\
\delta(z_1, \Box) &= (z_2, \Box, L) \\
\delta(z_2, a) &= (z_2, a, N) \\
\delta(z_2, b) &= (z_2, b, R) \\
\delta(z_2, \Box) &= (z_f, \Box, N)
\end{aligned}
$$

(a) Akzeptiert M das Wort $w_1 = ababa$? ◯ ja $\surd$ **nein**

(b) Akzeptiert M das Wort $w_2 = aabbaabb$? $\surd$ **ja** ◯ nein

(c) Welche Sprache akzeptiert M?

Lösung: $L(M) = \{a\}\Sigma^*\{b\}$

Aufgabe 10. (4 Punkte) Geben Sie ein LOOP-Programm an, das die folgende Funktion $f\colon \mathbb{N}^3 \to \mathbb{N}$ berechnet:

$$f(x,y,z) = \begin{cases} z^{x+y}, & \text{falls } z > 0, \\ \min(x,y), & \text{sonst.} \end{cases}$$

Lösung:

$x_4 := x_1 - x_2$;
$x_5 := x_1$;
LOOP x_4 DO
  $x_5 := x_2$
END;
LOOP x_3 DO
  $x_4 := x_1 + x_2$;
  $x_5 := 1$;
  $x_6 := x_3$;
  LOOP x_4 DO
    $x_5 := x_5 \cdot x_6$
  END
END;
$x_1 := x_5$

Aufgabe 11. (3 Punkte) Geben Sie an, welche partielle Funktion $f\colon \mathbb{N}^2 \to \mathbb{N}$ das folgende WHILE-Programm berechnet (Eingabevariablen sind x_1 und x_2, die Ausgabevariable ist x_1):

$x_3 := x_1 - x_2$;
$x_4 := x_2 - x_1$;
WHILE $x_3 \neq 0$ DO END;
WHILE $x_4 \neq 0$ DO END;
$x_1 := x_3 + x_4$

Lösung:

$$f(x_1,x_2) = \begin{cases} 0, & \text{wenn } x_1 = x_2 \\ \text{undefiniert}, & \text{sonst.} \end{cases}$$

Aufgabe 12. (6 Punkte) Geben Sie für die folgenden partiellen Funktionen jeweils ein WHILE-Programm an, das sie berechnet.

(a) $f\colon \mathbb{N}^2 \to \mathbb{N}$ mit $f(a_1, a_2) = \begin{cases} \text{undefiniert}, & \text{falls } a_1 < a_2, \\ 1 & \text{sonst.} \end{cases}$

Lösung:

WHILE $x_1 \neq 0$ DO
  $x_1 := x_1 - 1$;
  $x_2 := x_2 - 1$
END;
WHILE $x_2 \neq 0$ DO
  $x_2 := 987$
END;
$x_1 := 1$

Ergebnis in x_1.

(b) $g\colon \mathbb{N} \to \mathbb{N}$ mit $g(n) = \begin{cases} n, & \text{wenn } n \text{ gerade ist,} \\ n+1, & \text{wenn } n \text{ ungerade ist.} \end{cases}$

Lösung:

$(x_2 := 0)$
WHILE $x_1 \neq 0$ DO
  $x_2 := x_2 + 1$;
  $x_1 := x_1 - 2$
END;
$x_1 := x_2 + x_2$

Ergebnis in x_1.

Aufgabe 13. Zeigen Sie, dass die folgenden Funktionen primitiv-rekursiv sind.

Hinweis: Sie sollten bei Teilaufgabe (b) und (c) die Ergebnisse aus den vorigen Teilaufgaben verwenden!

(a) (2 Punkte) $h\colon \mathbb{N} \to \mathbb{N}$ mit $h(z) = z!$

Lösung: Es gilt

$$\begin{aligned} h(0) &= 1 \\ h(z+1) &= h(z) \cdot s(z), \end{aligned}$$

daher ist h primitiv-rekursiv.

(b) (2 Punkte) $g\colon \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ mit $g(y,z) = (z!)^y$

Lösung:

$$\begin{aligned} g(0,z) &= 1 \\ g(y+1,z) &= (z!)^{y+1} = (z!)^y \cdot (z!) = g(y,z) \cdot h(z) \end{aligned}$$

(c) (2 Punkte) $f\colon \mathbb{N} \times \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ mit $f(x,y,z) = \sum_{i=1}^{x} i(z!)^y$

Lösung:

$$\begin{aligned} f(0,y,z) &= 0 \\ f(x+1,y,z) &= f(x,y,z) + s(x) \cdot g(y,z) \end{aligned}$$

Aufgabe 14. (4 Punkte) Geben Sie an, welche Funktionen von μf und μg berechnet werden, wobei f und g wie folgt definiert sind.

(a) $f\colon \mathbb{N}^3 \to \mathbb{N}$ mit $f(n,x,y) = 5x - 2ny$

Lösung: $\mu f(x,y) = \begin{cases} \text{undefiniert}, & \text{falls } x > 0 \text{ und } y = 0 \\ 0, & \text{falls } x = y = 0 \\ \left\lceil \frac{5x}{2y} \right\rceil & \text{sonst.} \end{cases}$

(b) $g\colon \mathbb{N}^4 \to \mathbb{N}$ mit $g(n,x,y,z) = n(x-1)(y+1) + z$

Lösung:
$\mu g(x,y,z) = \begin{cases} 0, & \text{falls } z = 0, \\ \text{undefiniert} & \text{sonst.} \end{cases}$

Aufgabe 15. (6 Punkte) Gegeben sei die Funktion $f\colon \mathbb{N} \to \mathbb{N}$ mit

$$f(n) = (n \bmod 4)$$

Geben Sie eine Turing-Maschine an, die f berechnet.

Hinweis: Beachten Sie, dass die Eingabezahl n als Binärzahl auf dem Band steht (z.B. kodiert 101 die Eingabezahl 5). Denken Sie außerdem daran, dass der Lesekopf am Ende auf dem ersten Symbol der Ausgabe stehen muss. Machen Sie sich an genügend (nicht zu komplizierten) Beispielen klar, wie die Turing-Maschine arbeiten muss!

Lösung: Sei $M = (\{s, z_1, z_2, z_3, z_4, f\}, \{0,1\}, \{0,1,\Box\}, \delta, s, \Box, \{f\})$ mit

$$\begin{aligned}
\delta(s,a) &= \{(s,a,R) \mid a \in \{0,1\}\}\\
\delta(s,\Box) &= \{(z_1,\Box,L)\}\\
\delta(z_1,a) &= \{(z_2,a,L) \mid a \in \{0,1\}\}\\
\delta(z_2,a) &= \{(z_3,a,L) \mid a \in \{0,1\}\}\\
\delta(z_2,\Box) &= \{(f,\Box,R)\}\\
\delta(z_3,a) &= \{(z_3,\Box,L) \mid a \in \{0,1\}\}\\
\delta(z_3,\Box) &= \{(z_4,\Box,R)\}\\
\delta(z_4,\Box) &= \{(z_4,\Box,R)\}\\
\delta(z_4,1) &= \{(f,1,N)\}\\
\delta(z_4,0) &= \{(f,\Box,R)\}
\end{aligned}$$

Idee: Da n in Binärdarstellung auf dem Band gegeben ist, ist die Berechnung modulo 4 sehr einfach. Angenommen $\text{bin}(n) = a_1a_2\cdots a_{k-1}a_k$ mit $a_i \in \{0,1\}$. Wir erhalten eine eindeutige Darstellung $n = 4l + r$ mit $0 \le r \le 3$ und $l \ge 0$, also ist $\text{bin}(l) = a_1a_2\cdots a_{k-2}$ und $\text{bin}(r) = 1a_k$ ($a_{k-1} = 1$) bzw. $\text{bin}(r) = a_k$ ($a_{k-1} = 0$), sofern $k \ge 3$ ist. Der Fall $k \le 2$ bedeutet $n \le 3$, also ist $(n \bmod 4) = n$. Dies lässt sich leicht auf einer Turingmaschine simulieren:

Der Leseschreibkopf läuft ans rechte Ende von $\text{bin}(n)$. Die Turingmaschine löscht anschließend alle Zeichen ab dem dritten Bit von rechts. Danach läuft der Leseschreibkopf zu den verbleibenden 2 Bits zurück. Sonderfälle wie $n < 3$ werden dabei beachtet.

Aufgabe 16. (5 Punkte (Bonus)) Sei $G = (\{S, A, B\}, \{a, b\}, P, S)$ die Grammatik mit den folgenden Produktionen:

$$
\begin{aligned}
S &\to \varepsilon \mid ASB \\
AB &\to BA \\
A &\to a \\
B &\to b
\end{aligned}
$$

Beschreiben Sie die Sprache $L(G)$!

Lösung: $L(G) = \{w \in \{a, b\}^* \mid w$ enthält so viele a's wie b's$\}$.