M1 - Kellerautomaten

a) Geben Sie die formale Definition eines *nichtdeterministischen Kellerautomaten* an. Vervollständigen Sie hierfür den nachfolgenden Text:

   Ein nichtdeterministischer Kellerautomat (PDA) ist ein Sechs-Tupel $\mathcal{M} = \langle \quad , \quad , \quad , \quad , \quad \quad , \quad \rangle$ mit den folgenden Bestandteilen: . . .

b) Welcher andere Akzeptanzbegriff für Kellerautomaten ist laut Anmerkung in der Vorlesung auch möglich?

c) Benennen Sie formal die Unterschiede zwischen deterministischen und nicht-deterministischen Kellerautomaten.

d) Welcher Typ formaler Sprachen wird durch deterministische und welcher durch nichtdeterministische Kellerautomaten charakterisiert? Benennen Sie jeweils eine Sprache genau diesen Typs.

M1 - Kellerautomaten

a) Geben Sie die formale Definition eines *nichtdeterministischen Kellerautomaten* an. Vervollständigen Sie hierfür den nachfolgenden Text:

Ein nichtdeterministischer Kellerautomat (PDA) ist ein Sechs-Tupel
$\mathcal{M} = \langle \quad , \quad , \quad , \quad , \quad , \quad \rangle$ mit den folgenden Bestandteilen: . . .
$\mathcal{M} = \langle Q, \Sigma, \Gamma, \delta, Q_0, F \rangle$ mit den folgenden Bestandteilen:

Q: endliche Menge von Zuständen

Σ: Eingabealphabet

Γ: Kelleralphabet

δ: Übergangsfunktion, eine totale Funktion: $Q \times \Sigma_\varepsilon \times \Gamma_\varepsilon \to 2^{Q \times \Gamma_\varepsilon}$,

Q_0: Menge möglicher Startzustände $Q_0 \subseteq Q$

F: Menge von Endzuständen $F \subseteq Q$

M1 - Kellerautomaten

b) Welcher andere Akzeptanzbegriff für Kellerautomaten ist laut Anmerkung in der Vorlesung auch möglich?

Neben der Akzeptanz über Endzustände in $F \subseteq Q$ gibt es die (äquivalente) Akzeptanz über leeren Keller.

c) Benennen Sie formal die Unterschiede zwischen deterministischen und nicht-deterministischen Kellerautomaten.

Ein deterministischer Kellerautomat (DPDA) ist ein Sechs-Tupel $\mathcal{M} = \langle Q, \Sigma, \Gamma, \delta, q_0, F \rangle$ mit den folgenden Bestandteilen:

> δ: Übergangsfunktion, eine partielle Funktion $Q \times \Sigma_\varepsilon \times \Gamma_\varepsilon \to Q \times \Gamma_\varepsilon$, so dass für alle $q \in Q$, $a \in \Sigma$ und $A \in \Gamma$ jeweils nur eines der folgenden definiert ist:
>
> $$\delta(q, a, A) \qquad \delta(q, a, \varepsilon) \qquad \delta(q, \varepsilon, A) \qquad \delta(q, \varepsilon, \varepsilon)$$
>
> q_0: ein Startstand $q_0 \in Q$

M1 - Kellerautomaten

d) Welcher Typ formaler Sprachen wird durch deterministische und welcher durch nichtdeterministische Kellerautomaten charakterisiert? Benennen Sie jeweils eine Sprache genau diesen Typs.

Automat	**Sprachklasse (bzw. Typ)**		**Beispielsprache**
PDA	kontextfrei	Typ 2	$L = \{a^i b^j c^k \mid i \neq j \text{ oder } j \neq k\}$
DPDA	det. kontextfrei	det. Typ 2	$L = \{w \in \Sigma^* \mid \lvert w\rvert_a + \lvert w\rvert_b = \lvert w\rvert_c\}$

Hierbei bezeichnet $|w|_x$ die Anzahl der $x \in \Sigma$ in $w \in \Sigma^*$.

M2 - Pumping-Lemma

a) Formulieren Sie formal präzise das *Pumping-Lemma für kontextfreie Sprachen*. Vervollständigen Sie hierfür den nachfolgenden Text:

Für jede kontextfreie Sprache L gibt es eine natürliche Zahl $n \geqslant 0$, so dass gilt:

$\ldots$

b) Zeigen Sie mithilfe des *Pumping-Lemmas für reguläre Sprachen*, dass die Sprache $L = \{0^p \mid p \text{ ist eine Primzahl}\}$ nicht regulär ist.

M2 - Pumping-Lemma

a) Formulieren Sie formal präzise das *Pumping-Lemma für kontextfreie Sprachen*. Vervollständigen Sie hierfür den nachfolgenden Text:

Für jede kontextfreie Sprache L gibt es eine natürliche Zahl $n \geqslant 0$, so dass gilt: für jedes Wort $z \in L$ mit $|z| \geqslant n$ gibt es eine Zerlegung $z = uvwxy$ mit $|vx| \geqslant 1$ und $|vwx| \leqslant n$, so dass für jede Zahl $k \geqslant 0$ gilt: $uv^kwx^ky \in L$.

---

b) Zeigen Sie mithilfe des *Pumping-Lemmas für reguläre Sprachen*, dass die Sprache $L = \{0^p \mid p \text{ ist eine Primzahl}\}$ nicht regulär ist.

Annahme: L erfüllt das Pumping-Lemma. D.h. es gibt $n \geqslant 0$, so dass für jedes $z \in L$ mit $|z| \geqslant n$ eine Aufteilung $z = uvw$ existiert mit:

(1) $|v| \geqslant 1$ (2) $|uv| \leqslant n$ (3) $uv^kw \in L$ für jedes $k \geqslant 0$.

Wähle eine Primzahl $\ell > n + 2$. Laut Pump-Eigenschaft finden wir eine Zerlegung von $0^\ell = uvw$, für die insbesondere gilt: $uv^kw \in L$ für $k = |uw|$. Aber $uv^{|uw|}w = 0^{(|v|+1)|uw|} \notin L$. Widerspruch. L ist daher nicht regulär.

M3 - Grammatiken

Gegeben sei die Grammatik $G = (V, \Sigma, P, S)$ mit

$$V = \{S, A, B\},\ \Sigma = \{a, b\} \text{ und}$$

$$P = \{S \to ASB,\ S \to AB,\ AB \to BA,\ A \to a,\ B \to b\}.$$

a) Von welchem maximalen Typ ist G? Begründen Sie Ihre Antwort.

b) Geben Sie vier Wörter $w_1, w_2, w_3, w_4 \in L(G)$ mit $|w_1| = |w_2| = |w_3| = |w_4| = 4$ an.

c) Beschreiben Sie die durch G erzeugte Sprache $L(G)$ in einer geeigneten Notation.

---

a) G ist vom Typ 1 (kontextsensitiv) und nicht vom Typ 2 (kontextfrei).

b) $w_1 = aabb$, $w_2 = abab$, $w_3 = abba$, $w_4 = baba$

c) $L(G) = \{w \in \{a, b\}^+ \mid |w|_a = |w|_b\}$

M4 - Grammatiken: CNF/CYK

Gegeben sei das Wort $w = abac$ und die Grammatik $G = (V, \Sigma, P, S)$ mit

$$V = \{S, A, B, C, D\},\ \Sigma = \{a, b, c\} \text{ und}$$

$$P = \{S \to AB,\ A \to BA,\ A \to a, B \to AC,\ B \to BB,\ B \to b, C \to c,\ D \to AB\}.$$

a) Ist die Grammatik G in *Chomsky*-Normalform? Begründen Sie Ihre Antwort.

b) Entscheiden Sie mithilfe des Cocke-Younger-Kasami-Algorithmus, ob $w \in L(G)$ gilt. Transformieren Sie, falls notwendig, G in *Chomsky*-Normalform.

c) Entfernen Sie in G, sofern vorhanden, nichtterminierende und nichterreichbare Symbole. Begründen Sie Ihr Vorgehen.

---

a) ja. Jede Regel ist von der Form $A \to BC$ oder $A \to x$ $(A, B, C \in V, x \in \Sigma)$.

b) ...

c) D ist nicht erreichbar, $G' = (V', \Sigma, P', S)$ mit $V' = \{S, A, B, C\}$ und
$P' = \{S \to AB, A \to BA, A \to a, B \to AC, B \to BB, B \to b, C \to c\}$
hat weder nichtterminierende noch nichterreichbare Symbole.

M4 - Grammatiken: CNF/CYK

Gegeben sei das Wort $w = abac$ und die Grammatik $G = (V, \Sigma, P, S)$ mit

$$V = \{S, A, B, C, D\},\ \Sigma = \{a, b, c\} \text{ und}$$

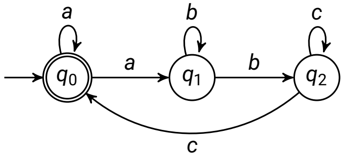
$$P = \{S \to AB,\ A \to BA,\ A \to a, B \to AC,\ B \to BB,\ B \to b, C \to c,\ D \to AB\}.$$

a	$\{A\}$	$\{D, S\}$	$\varnothing$	$\{D, S\}$
b		$\{B\}$	$\{A\}$	$\{B\}$
a			$\{A\}$	$\{B\}$
c				$\{C\}$
	a	b	a	c

Damit ist $w = abac \in L$.

M5 - NFA/Reguläre Ausdrücke/DFA

Gegeben sei der NFA $\mathcal{M} = (\{q_0, q_1, q_2\}, \{a, b, c\}, \delta, \{q_0\}, \{q_0\})$ mit δ:

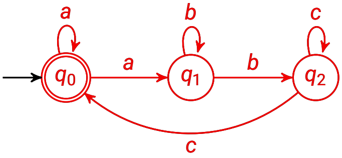


a) Berechnen Sie mithilfe des Arden-Lemmas einen regulären Ausdruck α mit $L(\alpha) = L(\mathcal{M})$.

b) Konstruieren Sie einen zu $\mathcal{M}$ äquivalenten DFA $\mathcal{M}'$. Verwenden Sie dazu die Potenzmengenkonstruktion aus der Vorlesung. Stellen Sie dabei sicher, dass der konstruierte Automat keine unerreichbaren Zustände enthält.

M5 - NFA/Reguläre Ausdrücke/DFA

Gegeben sei der NFA $\mathcal{M} = (\{q_0, q_1, q_2\}, \{a, b, c\}, \delta, \{q_0\}, \{q_0\})$ mit δ:

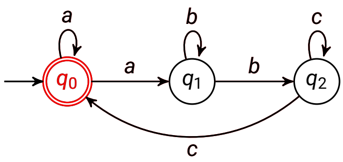


a) Berechnen Sie mithilfe des Arden-Lemmas einen regulären Ausdruck α mit $L(\alpha) = L(\mathcal{M})$.

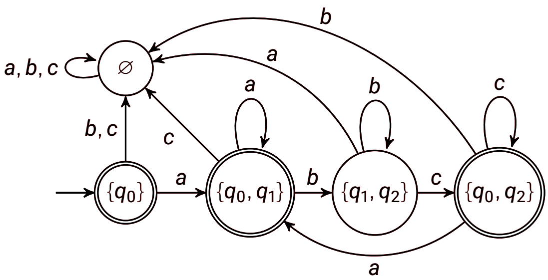
$$
\begin{aligned}
\alpha_0 &\equiv a\alpha_0 \mid a\alpha_1 \mid \varepsilon &&\equiv a\alpha_0 \mid ab^+c^+\alpha_0 \mid \varepsilon &&\equiv (a \mid ab^+c^+)^* \\
\alpha_1 &\equiv b\alpha_1 \mid b\alpha_2 &&\equiv b^*b\alpha_2 \equiv b^+\alpha_2 \\
\alpha_2 &\equiv c\alpha_2 \mid c\alpha_0 &&\equiv c^*c\alpha_0 \equiv c^+\alpha_0
\end{aligned}
$$

M5 - NFA/Reguläre Ausdrücke/DFA

Gegeben sei der NFA $\mathcal{M} = (\{q_0, q_1, q_2\}, \{a, b, c\}, \delta, \{q_0\}, \{q_0\})$ mit δ:



b) DFA $\mathcal{M}'$:



M6 - Nerode/Minimalautomat

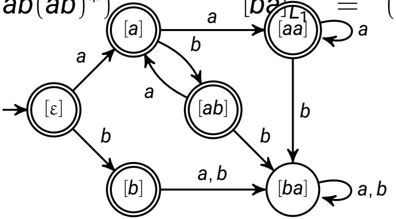
Geben Sie die Nerode-Äquivalenzklassen für die nachfolgenden Sprachen an und geben Sie den Minimalautomaten für L_1 an.

$$L_1 = L((ab)^*a^* \mid b) \quad \text{und} \quad L_2 = \{w \in \{a,b\}^* \mid w = w^R\}$$

Für L_1 lauten die Äquivalenzklassen:

$$\begin{aligned}
[\varepsilon]_{L_1} &= L(\varepsilon) & [a]_{L_1} &= L((ab)^*a) \\
[b]_{L_1} &= L(b) & [aa]_{L_1} &= L((ab)^*aaa^*) \\
[ab]_{L_1} &= L(ab(ab)^*) & [ba]_{L_1} &= (a,b)^* \setminus L_1
\end{aligned}$$

Minimalautomat:



Für L_2 lauten die Äquivalenzklassen: $[w]_{L_2} = \{w\}$ mit $w \in \{a,b\}^*$.

D.h. jedes Wort aus $\{a,b\}^*$ bildet eine eigene Äquivalenzklasse.

Damit ist der Nerode-Index von L_2 unendlich, d.h. die Sprache L_2 ist nicht regulär.

Aufgabe 2 ($2 + 4 + 2 = 8$ Punkte)

Welche der folgenden Probleme sind entscheidbar? Begründen Sie Ihre Antwort.

1. Gegeben eine Turing-Maschine $\mathcal{M}$ über dem Eingabealphabet $\{0, 1, \ldots, 9\}$ und eine Zahl n, hält $\mathcal{M}$ nach höchstens n Schritten bei Eingabe 42?
2. Gegeben eine Turing-Maschine $\mathcal{M}$, ist $L(\mathcal{M})$ unendlich?
3. Gegeben eine Turing-Maschine $\mathcal{M}$ über einem einelementigen Eingabealphabet, erkennt $\mathcal{M}$ nur Palindrome?

Aufgabe 2a (2 Punkte)

Welche der folgenden Probleme sind entscheidbar? Begründen Sie Ihre Antwort.

1. Gegeben eine Turing-Maschine $\mathcal{M}$ über dem Eingabealphabet $\{0, 1, \ldots, 9\}$ und eine Zahl n, hält $\mathcal{M}$ nach höchstens n Schritten bei Eingabe 42?

Aufgabe 2a (2 Punkte)

Welche der folgenden Probleme sind entscheidbar? Begründen Sie Ihre Antwort.

1. Gegeben eine Turing-Maschine $\mathcal{M}$ über dem Eingabealphabet $\{0, 1, \ldots, 9\}$ und eine Zahl n, hält $\mathcal{M}$ nach höchstens n Schritten bei Eingabe 42?

Dieses Problem ist entscheidbar.

Man kann die TM $\mathcal{M}$ auf Eingabe 42 für n Schritte simulieren. Wenn $\mathcal{M}$ hält, akzeptieren wir, sonst nicht.

Aufgabe 2b (4 Punkte)

Welche der folgenden Probleme sind entscheidbar? Begründen Sie Ihre Antwort.

1. Gegeben eine Turing-Maschine $\mathcal{M}$ über dem Eingabealphabet $\{0, 1, \dots, 9\}$ und eine Zahl n, hält $\mathcal{M}$ nach höchstens n Schritten bei Eingabe 42? ✓
2. Gegeben eine Turing-Maschine $\mathcal{M}$, ist $L(\mathcal{M})$ unendlich?

Aufgabe 2b (4 Punkte)

Welche der folgenden Probleme sind entscheidbar? Begründen Sie Ihre Antwort.

1. Gegeben eine Turing-Maschine $\mathcal{M}$ über dem Eingabealphabet $\{0, 1, \dots, 9\}$ und eine Zahl n, hält $\mathcal{M}$ nach höchstens n Schritten bei Eingabe 42? ✓
2. Gegeben eine Turing-Maschine $\mathcal{M}$, ist $L(\mathcal{M})$ unendlich?

Dieses Problem ist unentscheidbar.

Es handelt sich um das Wortproblem der Sprache

$$\{enc(\mathcal{M}) \mid L(\mathcal{M}) \text{ist unendlich}\}$$

Da aber Unendlichkeit eine nicht-triviale Eigenschaft formaler Sprachen ist (Σ^* hat die Eigenschaft, $\emptyset$ nicht), ist diese Menge nach dem Satz von Rice unentscheidbar.

Aufgabe 2 c (2 Punkte)

Welche der folgenden Probleme sind entscheidbar? Begründen Sie Ihre Antwort.

1. Gegeben eine Turing-Maschine $\mathcal{M}$ über dem Eingabealphabet $\{0, 1, \ldots, 9\}$ und eine Zahl n, hält $\mathcal{M}$ nach höchstens n Schritten bei Eingabe 42? ✓
2. Gegeben eine Turing-Maschine $\mathcal{M}$, ist $L(\mathcal{M})$ unendlich? ✗
3. Gegeben eine Turing-Maschine $\mathcal{M}$ über einem einelementigen Eingabealphabet, erkennt $\mathcal{M}$ nur Palindrome?

Aufgabe 2 c (2 Punkte)

Welche der folgenden Probleme sind entscheidbar? Begründen Sie Ihre Antwort.

1. Gegeben eine Turing-Maschine $\mathcal{M}$ über dem Eingabealphabet $\{0, 1, \ldots, 9\}$ und eine Zahl n, hält $\mathcal{M}$ nach höchstens n Schritten bei Eingabe 42? ✓
2. Gegeben eine Turing-Maschine $\mathcal{M}$, ist $L(\mathcal{M})$ unendlich? ✗
3. Gegeben eine Turing-Maschine $\mathcal{M}$ über einem einelementigen Eingabealphabet, erkennt $\mathcal{M}$ nur Palindrome?

Dieses Problem ist entscheidbar.

Mit einem einelementigen Alphabet lassen sich nur Palindrome bilden. Es handelt sich also um eine triviale Eigenschaft der Sprache.

Aufgabe 2 ($2 + 4 + 2 = 8$ Punkte)

Welche der folgenden Probleme sind entscheidbar? Begründen Sie Ihre Antwort.

1. Gegeben eine Turing-Maschine $\mathcal{M}$ über dem Eingabealphabet $\{0, 1, \ldots, 9\}$ und eine Zahl n, hält $\mathcal{M}$ nach höchstens n Schritten bei Eingabe 42? ✓
2. Gegeben eine Turing-Maschine $\mathcal{M}$, ist $L(\mathcal{M})$ unendlich? ✗
3. Gegeben eine Turing-Maschine $\mathcal{M}$ über einem einelementigen Eingabealphabet, erkennt $\mathcal{M}$ nur Palindrome? ✓

Aufgabe 3 (8 Punkte)

Sei $f\colon \mathbb{N} \to \mathbb{N}$ mit $f(x) = \left\lfloor \sqrt{3x} \right\rfloor$. Hierbei bezeichnet die Funktion $\lfloor \cdot \rfloor$ die Abrundefunktion auf natürliche Zahlen, also $\lfloor x \rfloor := \max \{n \in \mathbb{N} \mid n \leq x\}$.

Geben Sie ein LOOP-Programm an, welches f berechnet.

Aufgabe 3 (8 Punkte)

Sei $f\colon \mathbb{N} \to \mathbb{N}$ mit $f(x) = \left\lfloor \sqrt{3x} \right\rfloor$. Hierbei bezeichnet die Funktion $\lfloor \cdot \rfloor$ die Abrundefunktion auf natürliche Zahlen, also $\lfloor x \rfloor := \max \{ n \in \mathbb{N} \mid n \leq x \}$.

Geben Sie ein LOOP-Programm an, welches f berechnet.

Aufgabe 3 (8 Punkte)

Sei $f\colon \mathbb{N} \to \mathbb{N}$ mit $f(x) = \left\lfloor \sqrt{3x} \right\rfloor$. Hierbei bezeichnet die Funktion $\lfloor \cdot \rfloor$ die Abrundefunktion auf natürliche Zahlen, also $\lfloor x \rfloor := \max\{n \in \mathbb{N} \mid n \leq x\}$.

Geben Sie ein LOOP-Programm an, welches f berechnet.

Das LOOP-Programm berechnet zuerst $x \mapsto 3x$ und danach $y \mapsto \lfloor \sqrt{v} \rfloor$:

$x_2 := x_1 + x_1$

Aufgabe 3 (8 Punkte)

Sei $f\colon \mathbb{N} \to \mathbb{N}$ mit $f(x) = \left\lfloor \sqrt{3x} \right\rfloor$. Hierbei bezeichnet die Funktion $\lfloor \cdot \rfloor$ die Abrundefunktion auf natürliche Zahlen, also $\lfloor x \rfloor := \max\{n \in \mathbb{N} \mid n \leq x\}$.

Geben Sie ein LOOP-Programm an, welches f berechnet.

Das LOOP-Programm berechnet zuerst $x \mapsto 3x$ und danach $y \mapsto \lfloor \sqrt{v} \rfloor$:

```
x2 := x1 + x1
x2 := x2 + x1 ;; x2 ist jetzt 3x1
```

Aufgabe 3 (8 Punkte)

Sei $f\colon \mathbb{N} \to \mathbb{N}$ mit $f(x) = \left\lfloor \sqrt{3x} \right\rfloor$. Hierbei bezeichnet die Funktion $\lfloor \cdot \rfloor$ die Abrundefunktion auf natürliche Zahlen, also $\lfloor x \rfloor := \max\{n \in \mathbb{N} \mid n \leq x\}$.

Geben Sie ein LOOP-Programm an, welches f berechnet.

Das LOOP-Programm berechnet zuerst $x \mapsto 3x$ und danach $y \mapsto \lfloor \sqrt{y} \rfloor$:

```
x2 := x1 + x1
x2 := x2 + x1 ;; x2 ist jetzt 3x1
x3 := 0 ;; Initialisierung (nicht unbedingt erforderlich)
```

Aufgabe 3 (8 Punkte)

Sei $f\colon \mathbb{N} \to \mathbb{N}$ mit $f(x) = \left\lfloor \sqrt{3x} \right\rfloor$. Hierbei bezeichnet die Funktion $\lfloor \cdot \rfloor$ die Abrundefunktion auf natürliche Zahlen, also $\lfloor x \rfloor := \max\{n \in \mathbb{N} \mid n \leq x\}$.

Geben Sie ein LOOP-Programm an, welches f berechnet.

Das LOOP-Programm berechnet zuerst $x \mapsto 3x$ und danach $y \mapsto \lfloor \sqrt{y} \rfloor$:

```
x2 := x1 + x1
x2 := x2 + x1 ;; x2 ist jetzt 3x1
x3 := 0 ;; Initialisierung (nicht unbedingt erforderlich)
LOOP x2 DO
```

Aufgabe 3 (8 Punkte)

Sei $f\colon \mathbb{N} \to \mathbb{N}$ mit $f(x) = \left\lfloor \sqrt{3x} \right\rfloor$. Hierbei bezeichnet die Funktion $\lfloor \cdot \rfloor$ die Abrundefunktion auf natürliche Zahlen, also $\lfloor x \rfloor := \max\{n \in \mathbb{N} \mid n \leq x\}$.

Geben Sie ein LOOP-Programm an, welches f berechnet.

Das LOOP-Programm berechnet zuerst $x \mapsto 3x$ und danach $y \mapsto \left\lfloor \sqrt{y} \right\rfloor$:

```
x2 := x1 + x1
x2 := x2 + x1 ;; x2 ist jetzt 3x1
x3 := 0 ;; Initialisierung (nicht unbedingt erforderlich)
LOOP x2 DO
 x4 := x3 * x3
```

Aufgabe 3 (8 Punkte)

Sei $f\colon \mathbb{N} \to \mathbb{N}$ mit $f(x) = \left\lfloor \sqrt{3x} \right\rfloor$. Hierbei bezeichnet die Funktion $\lfloor \cdot \rfloor$ die Abrundefunktion auf natürliche Zahlen, also $\lfloor x \rfloor := \max\{n \in \mathbb{N} \mid n \leq x\}$.

Geben Sie ein LOOP-Programm an, welches f berechnet.

Das LOOP-Programm berechnet zuerst $x \mapsto 3x$ und danach $y \mapsto \lfloor \sqrt{v} \rfloor$:

```
x2 := x1 + x1
x2 := x2 + x1 ;; x2 ist jetzt 3x1
x3 := 0 ;; Initialisierung (nicht unbedingt erforderlich)
LOOP x2 DO
 x4 := x3 * x3
 x5 := x2 + 1
```

Aufgabe 3 (8 Punkte)

Sei $f\colon \mathbb{N} \to \mathbb{N}$ mit $f(x) = \left\lfloor \sqrt{3x} \right\rfloor$. Hierbei bezeichnet die Funktion $\lfloor \cdot \rfloor$ die Abrundefunktion auf natürliche Zahlen, also $\lfloor x \rfloor := \max\{n \in \mathbb{N} \mid n \leq x\}$.

Geben Sie ein LOOP-Programm an, welches f berechnet.

Das LOOP-Programm berechnet zuerst $x \mapsto 3x$ und danach $y \mapsto \lfloor \sqrt{y} \rfloor$:

```
x2 := x1 + x1
x2 := x2 + x1 ;; x2 ist jetzt 3x1
x3 := 0 ;; Initialisierung (nicht unbedingt erforderlich)
LOOP x2 DO
  x4 := x3 * x3
  x5 := x2 + 1
  x5 := x5 - x4
```

Aufgabe 3 (8 Punkte)

Sei $f \colon \mathbb{N} \to \mathbb{N}$ mit $f(x) = \left\lfloor \sqrt{3x} \right\rfloor$. Hierbei bezeichnet die Funktion $\lfloor \cdot \rfloor$ die Abrundefunktion auf natürliche Zahlen, also $\lfloor x \rfloor := \max\{n \in \mathbb{N} \mid n \leq x\}$.

Geben Sie ein LOOP-Programm an, welches f berechnet.

Das LOOP-Programm berechnet zuerst $x \mapsto 3x$ und danach $y \mapsto \lfloor \sqrt{y} \rfloor$:

```
x2 := x1 + x1
x2 := x2 + x1 ;; x2 ist jetzt 3x1
x3 := 0 ;; Initialisierung (nicht unbedingt erforderlich)
LOOP x2 DO
 x4 := x3 * x3
 x5 := x2 + 1
 x5 := x5 - x4
 IF x5!=0 THEN  ;; x5!=0 genau dann, wenn x3^2 ≤ x2
  x0 := x3
 END
```

Aufgabe 3 (8 Punkte)

Sei $f \colon \mathbb{N} \to \mathbb{N}$ mit $f(x) = \left\lfloor \sqrt{3x} \right\rfloor$. Hierbei bezeichnet die Funktion $\lfloor \cdot \rfloor$ die Abrundefunktion auf natürliche Zahlen, also $\lfloor x \rfloor := \max\{n \in \mathbb{N} \mid n \leq x\}$.

Geben Sie ein LOOP-Programm an, welches f berechnet.

Das LOOP-Programm berechnet zuerst $x \mapsto 3x$ und danach $y \mapsto \lfloor \sqrt{v} \rfloor$:

```
x2 := x1 + x1
x2 := x2 + x1 ;; x2 ist jetzt 3x1
x3 := 0 ;; Initialisierung (nicht unbedingt erforderlich)
LOOP x2 DO
 x4 := x3 * x3
 x5 := x2 + 1
 x5 := x5 - x4
 IF x5!=0 THEN  ;; x5!=0 genau dann, wenn x3^2 ≤ x2
  x0 := x3
 END
 x3 := x3 + 1
```

Aufgabe 3 (8 Punkte)

Sei $f : \mathbb{N} \to \mathbb{N}$ mit $f(x) = \left\lfloor \sqrt{3x} \right\rfloor$. Hierbei bezeichnet die Funktion $\lfloor \cdot \rfloor$ die Abrundefunktion auf natürliche Zahlen, also $\lfloor x \rfloor := \max\{n \in \mathbb{N} \mid n \leq x\}$.

Geben Sie ein LOOP-Programm an, welches f berechnet.

Das LOOP-Programm berechnet zuerst $x \mapsto 3x$ und danach $y \mapsto \lfloor \sqrt{y} \rfloor$:

```
x2 := x1 + x1
x2 := x2 + x1 ;; x2 ist jetzt 3x1
x3 := 0 ;; Initialisierung (nicht unbedingt erforderlich)
LOOP x2 DO
 x4 := x3 * x3
 x5 := x2 + 1
 x5 := x5 - x4
 IF x5!=0 THEN  ;; x5!=0 genau dann, wenn x3^2 ≤ x2
  x0 := x3
 END
 x3 := x3 + 1
END
```

Aufgabe 8: Interleaving und Shuffle

Zu einem gegebenen Alphabet Σ seien zwei nichtdeterministische endliche Automaten (NFA) $\mathcal{M}_i = \langle Q_i, \Sigma, \delta_i, Q_0^i, F_i \rangle$ $(i = 1, 2)$ mit disjunkten Zustandsmengen (d.h. $Q_1 \cap Q_2 = \emptyset$) gegeben.

Wir definieren den Automaten $\mathcal{M}_1 \parallel \mathcal{M}_2$ durch $\langle Q_1 \times Q_2, \Sigma, \delta, Q_0^1 \times Q_0^2, F_1 \times F_2 \rangle$ mit der Übergangsfunktion δ, sodass für alle $(q_1, q_2) \in Q_1 \times Q_2$ und $a \in \Sigma$ gilt:

$$\delta((q_1, q_2), a) := \{(q_{1'}, q_2) \mid q_{1'} \in \delta_1(q_1, a)\} \cup \{(q_1, q_{2'}) \mid q_{2'} \in \delta_2(q_2, a)\}$$

Wir nennen $\mathcal{M}_1 \parallel \mathcal{M}_2$ das Interleaving der Automaten $\mathcal{M}_1$ und $\mathcal{M}_2$.

Aufgabe 8: Interleaving und Shuffle

a) Welchen Sprachoperator implementiert $\parallel$? Gegeben zwei (reguläre) Sprachen L_1, L_2 mit entsprechenden NFAs $\mathcal{M}_1, \mathcal{M}_2$ (d.h. $L(\mathcal{M}_1) = L_1$ und $L(\mathcal{M}_2) = L_2$). Beschreiben Sie die Sprache $L(\mathcal{M}_1 \parallel \mathcal{M}_2)$ in Abhängigkeit von L_1 und L_2.

Wir definieren zunächst $w_1 \parallel w_2$ für zwei Wörter Σ^* induktiv:

Anfang Ist $w_i = \varepsilon$, so gilt $w_1 \parallel w_2 = w_j$ ($i, j \in \{1, 2\}$ und $i \neq j$).

Schritt Gilt $w_1 = av_1$ und $w_2 = bv_2$ für $a, b \in \Sigma$ und $v_1, v_2 \in \Sigma^*$,

$$w_1 \parallel w_2 := (\{a\} \circ (v_1 \parallel w_2)) \cup (\{b\} \circ (w_1 \parallel v_2))$$

Aufgabe 8: Interleaving und Shuffle

b) Der soeben eingeführte Sprachoperator heißt auch *Shuffle* ($L_1 \parallel L_2$ ist der Shuffle von L_1 und L_2). Sind reguläre Sprachen unter Shuffle abgeschlossen? Begründen Sie Ihre Antwort kurz.[1]

Ja, reguläre Sprachen sind unter Shuffle abgeschlossen. Das Interleaving zweier endlicher Automaten ergibt einen endlichen Automaten. Sind also L_1 und L_2 regulär, so existieren zwei endliche Automaten $\mathcal{M}_1$ und $\mathcal{M}_2$ mit $L(\mathcal{M}_1) = L_1$ und $L(\mathcal{M}_2) = L_2$ und $L_1 \parallel L_2 = L(\mathcal{M}_1 \parallel \mathcal{M}_2)$.

[1] Sie dürfen davon ausgehen, dass Ihre Lösung von a) korrekt ist.

Aufgabe 8: Interleaving und Shuffle

c) Sind die entscheidbaren Sprachen ebenfalls unter Shuffle abgeschlossen? Begründen Sie Ihre Antwort.

Ja, auch die entscheidbaren Sprachen sind unter $\parallel$ abgeschlossen. Seien $\mathcal{M}_1$ und $\mathcal{M}_2$ zwei Entscheider für L_1 und L_2.

Eine 3-Band-Turingmaschine $\mathcal{M}$ teilt zunächst das Eingabewort nichtdeterministisch $w \in \Sigma^*$ auf die Bänder 2 und 3 in $w_1, w_2 \in \Sigma^*$ auf, sodass $w \in w_1 \parallel w_2$. Im nächsten Schritt simuliert $\mathcal{M}$ die Entscheider $\mathcal{M}_i$ auf w_i $(i \in \{1, 2\})$ und akzeptiert genau dann, wenn beide Entscheider akzeptieren. Sonst lehnt $\mathcal{M}$ ab.

Gibt es zwei Wörter $w_1 \in L_1$ und $w_2 \in L_2$ mit $w \in w_1 \parallel w_2$, so findet $\mathcal{M}$ die passende Zerlegung auf einem Berechnungspfad und akzeptiert die Eingabe damit.

Die Zerlegungen können auch deterministisch systematisch durchgegangen werden, benötigt aber mehr Platz zum Schreiben.