Institut für Theoretische Informatik
Lehrstuhl Prof. Dr. D. Wagner

# Übungsblatt 2

Vorlesung Theoretische Grundlagen der Informatik im WS 20/21

**Ausgabe:** 24. November 2020
**Abgabe:** 8. Dezember 2020, 11:30 Uhr (digital im ILIAS)

## Aufgabe 1

(1 + 2 + 4 = 7 Punkte)

(a) Gegeben sei die Sprache $L_1 = \{w \in \{0\}^* \mid |w| \text{ ist durch 3 teilbar}\}$. Geben Sie einen deterministischen endlichen Automaten in Graphdarstellung an, der $L_1$ erkennt.

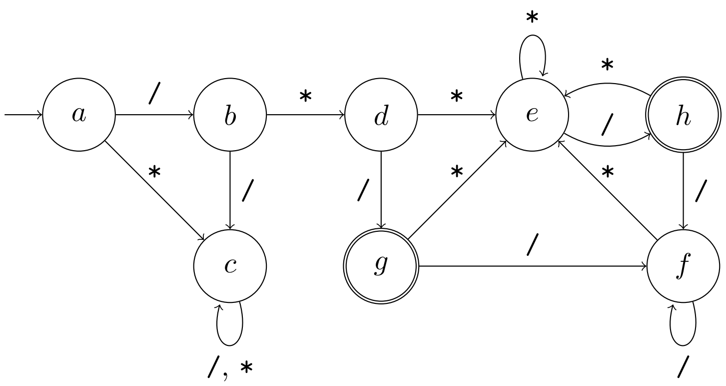
(b) Gegeben sei die Sprache $L_2 = \{w \in \{0,1\}^* \mid \text{bin}(w) \text{ ist durch 3 teilbar}\}$, wobei $\text{bin}(w)$ die natürliche Zahl ist, deren Binärdarstellung $w$ ist. Der Einfachheit halber sei $\text{bin}(\varepsilon) = 0$, also durch $k$ teilbar. Geben Sie einen deterministischen endlichen Automaten in Graphdarstellung an, der $L_2$ erkennt.

(c) Es seien $k, b \in \{1, 2, \dots\}$ und $\Sigma = \{0, 1, \dots, b-1\}$. Gegeben ist die Sprache $L_3 = \{w \in \Sigma^* \mid b\text{-adi}(w) \text{ ist durch } k \text{ teilbar}\}$, wobei $b\text{-adi}(w)$ die natürliche Zahl ist, deren $b$-adische Darstellung $w$ ist. Der Einfachheit halber sei $b\text{-adi}(\varepsilon) = 0$, also durch $k$ teilbar. Geben Sie einen deterministischen endlichen Automaten an, der $L_3$ erkennt. Beweisen Sie die Korrektheit Ihrer Konstruktion.

## Aufgabe 2

(3 Punkte)

Konstruieren Sie zu folgendem Automaten den zugehörigen Äquivalenzklassenautomat. Geben Sie in jedem Schritt alle Äquivalenzklassen an, und durch welche Zeugen sie getrennt werden. Zeichnen Sie den Übergangsgraphen des Äquivalenzklassenautomaten.



## Aufgabe 3

(2 + 3 + 2 + 1 = 8 Punkte)

Ein Kommilitone von Ihnen behauptet, dass er ein alternatives Verfahren zur Konstruktion von Äquivalenzklassenautomaten gefunden hat. Statt nach möglicherweise langen Zeugen suchen zu müssen, betrachtet er immer nur einzelne Zeichen. Zu einem deterministischen endlichen Automaten $\mathcal{A} = (Q, \Sigma, \delta, s, F)$ geht er folgendermaßen vor. Im ersten Schritt partitioniert er die Zustandsmenge in zwei Mengen $Q \setminus F$ und $F$. In jedem weiteren Schritt wählt er zunächst ein Zeichen $a \in \Sigma$. Jede Menge[1] $[q]$ trennt er genau dann weiter auf, wenn für zwei Zustände $q_1, q_2 \in [q]$ nach dem vorherigen Schritt galt, dass $[\delta(q_1, a)] \neq [\delta(q_2, a)]$. Solch einen Schritt wiederholt er, bis sich bei keinem Zeichen $a \in \Sigma$ weitere Trennungen ergeben. Mit den entstandenen Mengen und dem Verfahren aus der Vorlesung konstruiert er dann den Äquivalenzklassenautomaten.

(a) Führen Sie das Verfahren für den Automaten aus Aufgabe 2 durch, indem Sie die unten stehende Tabelle ausfüllen. Finden Sie dieselben Mengen wie die Äquivalenzklassen aus Aufgabe 2?

| Schritt | Zeichen $a$ | Partition nach Trennung durch $a$ |
|---|---|---|
| 1 | | $\{a, b, c, d, e, f\}$ , $\{g, h\}$ |
| 2 | / | $\{a, b, c, f\}$ , $\{d, e\}$ , $\{g, h\}$ |
| 3 | | |
| 4 | | |
| 5 | | |
| 6 | | |
| 7 | | |
| 8 | | |
| 9 | | |

[1] Disjunkte Mengen $\{\ldots, q, \ldots\} = [q]$ können durch einen Repräsentanten $q$ identifiziert werden.

(b) Folgern Sie aus $[q_1] \neq [q_2]$ induktiv die Existenz eines Zeugen $w$, der $q_1$ und $q_2$ trennt.

(c) Zeigen Sie, dass am Ende des Verfahrens $[q_1] = [q_2]$ impliziert, dass kein Zeuge existiert, der $q_1$ und $q_2$ trennt.

(d) Ist das Verfahren Ihres Kommilitonen korrekt? Begründen Sie!

## Aufgabe 4 (1 + 1 + 1 + 2 + 1 = 6 Punkte)

Ist die Aussage des Pumping-Lemmas für folgende Sprachen erfüllt? Welche Sprachen sind regulär? Begründen Sie!

(a) $\Sigma = \{\texttt{A}, \ldots, \texttt{Z}\} \cup \{\texttt{␣}, \texttt{-}\} \cup \{\texttt{0}, \ldots, \texttt{9}\}$, $L_a = \{\text{mögliches Autokennzeichen in Karlsruhe}\}$
Beispielsweise gilt $\texttt{KA-RT␣911} \in L_a$, aber $\texttt{KA-RT-OFFEL} \notin L_a$.

(b) $\Sigma = \{\texttt{a}, \texttt{b}, \texttt{c}\}, L_b = \{w \in \Sigma^* \mid |w|_\texttt{a} + |w|_\texttt{b} = |w|_\texttt{c}\}$. Dabei bezeichnet $|w|_x$ die Anzahl der Vorkommen des Symbols $x$ in $w$.

(c) $\Sigma = \{\texttt{0}, \texttt{1}\}$. Für ein Wort $w \in \Sigma^*$ sei $\text{neg}(w)$ die bitweise Negation von $w$, d.h. $\text{neg}(w)_i = 1 - w_i$, wobei $w_i$ das i-te Zeichen von $w$ beschreibt. $L_c = \{uv \in \Sigma^* \mid v = \text{neg}(u)\}$.

(d) Für eine beliebige natürliche Zahl $k \in \mathbb{N}$ definieren wir $L_k = \{w \in \{1, \ldots, k\}^* \mid w \text{ ist eine monoton aufsteigende Zahlenfolge}\}$. Das Wort $4\,4\,7\,8$ ist beispielsweise in $L_8$ enthalten, das Wort $8\,8\,4\,8$ hingegen nicht. Wir unterscheiden Zahlen aus $\{1, \ldots, k\}$ von Wörtern (Zahlenfolgen) aus $L_k$, indem wir die Zahlen mit einem Leerzeichen trennen. Sie können also die Zahlenfolge $8\,8\,4\,8$ von der Zahl 8848 unterscheiden.

(e) Ist die Menge $L_\infty = \{w \in \mathbb{N}^* \mid w \text{ ist eine monoton aufsteigende Zahlenfolge}\}$ eine reguläre Sprache?

## Aufgabe 5 (2 + 2 + 3 = 7 Punkte)

In dieser Aufgabe betrachten wir Sprachen über dem Alphabet $\Sigma = \{0, 1, 2\} \subseteq \mathbb{N}$.

(a) Geben Sie für $L_1 = \{w = w_1 \ldots w_n \in \Sigma^* \mid \forall i = 2, \ldots, n : w_i \geq w_{i-1}\}$ alle Äquivalenzklassen der Nerode-Relation (in Mengenschreibweise) an. Geben Sie außerdem für jede Äquivalenzklasse $K$ die Menge $S_K$ der gültigen Suffixe an, also die Menge der $z \in \Sigma^*$, sodass für jedes $u \in K$ gilt: $uz \in L_1$.

(b) Geben Sie für $L_1$ einen DEA mit minimaler Anzahl von Zuständen an.

(c) Für ein Wort $w = w_1 \ldots w_n$ definieren wir $q(w) := \sum_{i=1}^{n} w_i$. Wenn wir $w$ als Zahl zur Basis 3 interpretieren, ist $q(w)$ also die Quersumme dieser Zahl. Zeigen Sie mithilfe der Nerode-Relation, dass $L_2 = \{w \in \Sigma^* \mid q(w) = |w|\}$ nicht regulär ist.

**Aufgabe 6** (Bonus: 2 + 2 + 1 + 1 + 2 + 2 + 2 = 12 Punkte)

Wir betrachten typischerweise Sprachen über *endlichen* Alphabeten. Man kann sich aber auch überlegen, was passiert, wenn man Sprachen über *unendlichen* Alphabeten definiert. Genau das werden wir in dieser Aufgabe tun.

Sei also $\Sigma_\infty$ eine (möglicherweise unendliche) Menge von Zeichen. Die Begriffe „Sprache“ und „deterministischer endlicher Automat“ übertragen sich direkt. Sei $L_\infty$ eine Sprache über dem Alphabet $\Sigma_\infty$, die von einem deterministischen endlichen Automaten $\mathcal{A}_\infty = (Q, \Sigma_\infty, \delta_\infty, s, F)$ erkannt wird. Beachten Sie, dass $Q$ weiterhin eine endliche Menge sein muss. Wir definieren nun eine zweistellige Relation $\sim \subseteq \Sigma_\infty \times \Sigma_\infty$ mit folgender Vorschrift:

$$a \sim b \quad :\Longleftrightarrow \quad \forall q \in Q\colon \delta_\infty(q,a) = \delta_\infty(q,b)$$

Vergegenwärtigen Sie sich, dass $\sim$ eine Äquivalenzrelation ist. Zwei Zeichen $a$ und $b$ sind also genau dann bezüglich $\sim$ äquivalent, wenn sie von allen Zustandsübergängen in $\mathcal{A}_\infty$ gleich behandelt werden.

(a) Zeigen Sie, dass $\sim$ endlich viele Äquivalenzklassen hat.

(b) Zeigen Sie, dass ein endliches Alphabet $\Sigma$ und eine Funktion $f : \Sigma_\infty \to \Sigma$ existieren, sodass die Sprache $L = \{f(w_1)f(w_2)\dots f(w_n) \mid w_1w_2\dots w_n \in L_\infty\}$ über dem Alphabet $\Sigma$ von einem deterministischen endlichen Automaten erkannt wird.

Den Begriff einer regulären Sprache passen wir folgendermaßen an. Eine Sprache $L \subseteq \Sigma_\infty^*$ heißt $\infty$-*regulär*, wenn für sie einer der folgenden Punkte gilt: (induktive Definition)

- Verankerung:
  - $L \subseteq \Sigma_\infty$ oder
  - $L = \{\varepsilon\}$
- Induktion: Es gibt $\infty$-reguläre Sprachen $L_1, L_2$ sodass
  - $L = L_1 \cdot L_2$ oder
  - $L = L_1 \cup L_2$ oder
  - $L = L_1^*$

(c) Zeigen Sie, dass diese Definition zu der Definition von regulären Sprachen aus der Vorlesung äquivalent ist, wenn $\Sigma_\infty$ endlich ist.

Sei $L_\infty$ jetzt eine $\infty$-reguläre Sprache. Vergegenwärtigen Sie sich, dass die induktive Definition von $L_\infty$ nur auf endlich viele Verankerungen zurückgreift. Bezeichne diese mit $L_1, L_2, \dots, L_n$. Im Allgemeinen sind diese Mengen nicht paarweise disjunkt.

(d) Zeigen Sie, dass Sie $L_\infty$ auch so definieren können, dass die Verankerungen paarweise disjunkt sind.

(e) Zeigen Sie, dass ein endliches Alphabet $\Sigma$ und eine Funktion $f : \Sigma_\infty \to \Sigma$ existieren, sodass die Sprache $L = \{f(w_1)f(w_2)\dots f(w_n) \mid w_1w_2\dots w_n \in L_\infty\}$ regulär gemäß der Definition aus der Vorlesung ist.

Sie sind nun bereit zu zeigen, dass deterministische endliche Automaten mit möglicherweise unendlichem Eingabealphabet genau die $\infty$-regulären Sprachen erkennen können.

(f) Sei $L_\infty$ eine Sprache, die von einen deterministischen endlichen Automaten $\mathcal{A}_\infty$ mit Eingabealphabet $\Sigma_\infty$ erkannt wird. Zeigen Sie, dass $L_\infty$ $\infty$-regulär ist.

Verwenden Sie das entsprechende Ergebnis für endliche Automaten aus der Vorlesung und das Ergebnis aus Teilaufgabe (b).

(g) Sei $L_\infty$ eine $\infty$-reguläre Sprache über dem Alphabet $\Sigma_\infty$. Zeigen Sie, dass es einen deterministischen endlichen Automaten $\mathcal{A}_\infty$ mit Eingabealphabet $\Sigma_\infty$ gibt, der $L_\infty$ erkennt.

Verwenden Sie das entsprechende Ergebnis für reguläre Sprachen aus der Vorlesung und das Ergebnis aus Teilaufgabe (e).