

## Übungsblatt 1

Vorlesung Theoretische Grundlagen der Informatik im WS 18/19

**Ausgabe** 18. Oktober 2018

**Abgabe** 6. November 2018, 11:00 Uhr (im Kasten im UG von Gebäude 50.34)

### Aufgabe 1

(1 + 1 + 1 + 1 + 1 + 1 = 6 Punkte)

Gegeben seien zwei Sprachen  $L_1, L_2 \subseteq \Sigma^*$  über dem Alphabet  $\Sigma = \{\mathbf{a}, \mathbf{b}\}$ . Dabei sei  $L_1$  die Sprache der Wörter, deren erstes und letztes Zeichen übereinstimmen, und  $L_2$  die Sprache der Wörter, in denen auf jedes  $\mathbf{a}$  ein  $\mathbf{b}$  folgt. Geben Sie reguläre Ausdrücke für folgende Sprachen an.

- (a)  $L_1 \cup L_2$
- (b)  $L_1 \cdot L_2$
- (c)  $L_1 \cap L_2$
- (d)  $L_2 \setminus L_1$
- (e)  $L_1^c$
- (f)  $L_2^c$

### Aufgabe 2

(2 + 2 = 4 Punkte)

In dieser Aufgabe sollen Sie zeigen, dass die Menge der regulären Sprachen unter der Komplementbildung und Schnittbildung abgeschlossen ist. Dazu seien ein beliebiges Alphabet  $\Sigma$  und zwei beliebige reguläre Sprachen  $L_1, L_2$  gegeben. Für  $i \in \{1, 2\}$  sei  $\mathcal{A}_i = (Q_i, \Sigma, \delta_i, s_i, F_i)$  der deterministische endliche Automat, der  $L_i$  erkennt.

- (a) Wandeln Sie  $\mathcal{A}_1$  in einen deterministischen endlichen Automaten  $\bar{\mathcal{A}}$  um, der die Sprache  $L_1^c$  erkennt.
- (b) Geben Sie einen deterministischen endlichen Automaten  $\mathcal{A}_\cap$  an, der die Sprache  $L_1 \cap L_2$  erkennt.

Beweisen Sie jeweils, dass Ihr Automat die gewünschte Sprache erkennt.

### Aufgabe 3

(1 + 1 = 2 Punkte)

Gegeben seien zwei Grammatiken  $G_i = (\Sigma, V, S, R_i)$ ,  $i = 1, 2$  mit  $\Sigma = \{a, b, c\}$ ,  $V = \{S, A, B\}$  und folgenden Produktionsregeln

(a)  $R_1 = \{S \rightarrow ABS \mid \varepsilon, A \rightarrow acA \mid ac, B \rightarrow bcB \mid bc\}$

(b)  $R_2 = \{S \rightarrow AB, A \rightarrow aAbb \mid \varepsilon, B \rightarrow bBc \mid bc\}$

Ist  $L(G_i)$  eine reguläre Sprache? Wenn ja, geben Sie einen regulären Ausdruck an. Wenn nicht, begründen Sie dies kurz (informell) und geben sie eine formale Definition der Sprache in Mengenschreibweise an.

### Aufgabe 4

(2 + 1 + 4 = 7 Punkte)

Gegeben sei das Alphabet  $\Sigma = \{a, b\}$ . Für ein Wort  $v \in \Sigma^*$  sei  $L_v$  die Sprache der Wörter über  $\Sigma$ , die mit  $v$  enden.

- (a) Geben Sie einen deterministischen endlichen Automaten an, der  $L_{bab}$  erkennt. Verwenden Sie genau 4 Zustände.

Sei nun ein beliebiges Wort  $v \in \Sigma^*$  der Länge  $n$  gegeben.

- (b) Geben Sie eine formale Definition von  $L_v$  in Mengenschreibweise an.
- (c) Geben Sie formal einen deterministischen endlichen Automaten  $\mathcal{A}_v$  an, der  $L_v$  erkennt. Verwenden Sie genau  $n + 1$  Zustände. Begründen Sie, warum Ihr Automat deterministisch ist.

### Aufgabe 5

(1 + 2 = 3 Punkte)

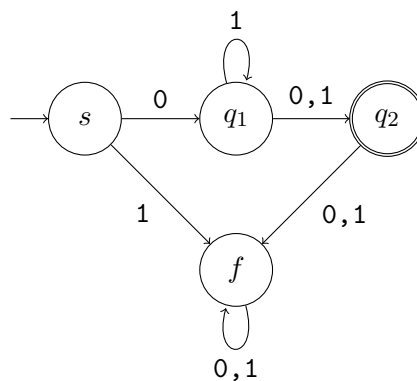
- (a) Offensichtlich lässt sich jeder deterministische endliche Automat (DEA) durch einen nichtdeterministischen endlichen Automaten (NEA) simulieren, indem der Nichtdeterminismus in der Überföhrungsfunktion nicht ausgenutzt wird. Sei  $\mathcal{A} = (Q, \Sigma, \delta, s, F)$  ein NEA, der wie ein DEA arbeitet. Welche Bedingungen müssen für die Überföhrungsfunktion  $\delta: Q \times (\Sigma \cup \{\varepsilon\}) \rightarrow 2^Q$  gelten? Formulieren Sie die Bedingungen als prädikatenlogische Formel.
- (b) Für NEA wurde in der Vorlesung die erweiterte Überföhrungsfunktion  $\bar{\delta}: Q \times \Sigma^* \rightarrow 2^Q$  eingeföhrt, die einen Zustand  $q$  und ein Wort  $w$  auf die Menge der Zustände abbildet, die von  $q$  bei der Abarbeitung von  $w$  erreichbar sind. Mithilfe der Simulation aus dem vorigen Aufgabenteil lässt sich  $\bar{\delta}$  auch bei DEA anwenden. Kreuzen Sie in der folgenden Tabelle an, welche Aussage für *alle* NEA bzw. DEA  $\mathcal{A}$  mit  $L(\mathcal{A}) \neq \emptyset$  und  $L(\mathcal{A}) \neq \Sigma^*$  gelten.

Aussage	NEA	DEA
$\exists w \in L(\mathcal{A}) \forall q \in \bar{\delta}(s, w) : q \in F$		
$\exists w \notin L(\mathcal{A}) \exists q \in \bar{\delta}(s, w) : q \notin F$		
$\forall w \in L(\mathcal{A}) \exists q \in \bar{\delta}(s, w) : q \in F$		
$\forall w \in L(\mathcal{A}) \exists q \in \bar{\delta}(s, w) : q \notin F$		
$\forall w \in L(\mathcal{A}) \forall q \in \bar{\delta}(s, w) : q \in F$		
$\forall w \notin L(\mathcal{A}) \forall q \in \bar{\delta}(s, w) : q \notin F$		

### Aufgabe 6

(1 + 2 = 3 Punkte)

- Geben Sie für die Sprache über dem Alphabet  $\Sigma = \{0, 1\}$ , die von dem unten abgebildeten Automaten erkannt wird, einen regulären Ausdruck an.
- Nutzen Sie die Potenzmengenkonstruktion aus der Vorlesung, um den Automaten in einen deterministischen endlichen Automaten zu überführen.



### Aufgabe 7

(1 + 3 + 2 = 6 Punkte)

Der ebenso geniale, wie auch vergessliche Wissenschaftler und Superbösewicht Doktor Meta ist im Baufieber. Der Hauptzugang zu seinem unterirdischen Geheimlabor soll von einer unüberwindbaren Stahltür geschützt werden, die sich nur durch Eingabe eines gültigen Passworts öffnen lässt. Bei einer falschen Eingabe wird der Eindringling stattdessen durch eine Falltür im Boden den Haien vorgeworfen. Da sich Doktor Meta, aufgrund seiner Vergesslichkeit, nur schlecht an Passwörter erinnern kann, hat er die Tür so eingestellt, dass Sie jedes Wort aus einer zuvor festgelegten formalen Sprache  $L$  über einem endlichen Alphabet  $\Sigma$  akzeptiert. Dieses - sicherheitstechnisch fragwürdige, aber für Doktor Meta sehr praktische - Verfahren soll intern durch einen deterministischen endlichen Automaten realisiert werden. Da Sie als Doktor Metas neuer Sicherheitsexperte eingestellt worden sind, sollen Sie ihm bei einigen letzten Konfigurationen helfen:

- Kann der Öffnungsmechanismus der Falltür für jede formale Sprache  $L$  durch einen deterministischen endlichen Automaten realisiert werden? Warum?
- Sie haben sich mit Doktor Meta auf eine geeignete Sprache  $L$  geeinigt. Leider hat einer ihrer Bauarbeiter, namentlich Ingo N. Kompetenz, das Schloss falsch verbaut. Die Kontrolleinheit

der Tür bekommt das Passwort genau umgedreht zur Überprüfung. Da die Bauarbeiten bereits weit fortgeschritten sind, beauftragt Doktor Meta Sie, sich – natürlich erst nachdem Sie Herrn Kompetenz den Haien vorgeworfen haben – um eine Lösung des Problems zu bemühen. Wie könnten Sie den Automaten mit möglichst wenigen Änderungen umbauen, sodass er dennoch die gleiche Passwortmenge erkennt? Wieso funktioniert Ihr Ansatz? Ist Ihr Automat deterministisch oder nichtdeterministisch? Begründen Sie!

- (c) Doktor Meta möchte einige Mitarbeiter unauffällig verschwinden lassen. Dafür hat er die Passwörter, die sie normalerweise eingeben, studiert und Muster<sup>1</sup> darin gefunden. Er beauftragt sie nun, die Kontrolleinheit so umzubauen, dass gerade die Muster der Leute, die er loswerden möchte von der Kontrolleinheit abgelehnt werden. Der Automat darf also nur noch eine Teilmenge der alten Sprache  $L$  akzeptieren. Können Sie den Automaten für jedes Muster entsprechend umbauen? Begründen Sie Ihre Antwort!

---

<sup>1</sup>Muster sind beliebige Mengen von Wörtern über dem Alphabet  $\Sigma$ .