

HA-Lösung

TA-Lösung

Einführung in die theoretische Informatik – Aufgabenblatt 2

Beachten Sie: Soweit nicht explizit angegeben, sind Ergebnisse stets zu begründen!

Hausaufgaben: Abgabe bis zum **27.04.2016** (Mittwoch) um 12:00

Aufgabe 2.1

2P+2P+2P

Sei Σ ein Alphabet und $A, B, C \subseteq \Sigma^*$.

Wahr oder falsch? Zeigen Sie die Aussage oder widerlegen sie Sie mithilfe eines geeigneten Gegenbeispiels:

$$(a) A(B \setminus C) = AB \setminus AC \quad (b) (AB)^* = (AB \cup B)^* \quad (c) (A^*B^*)^* = (A \cup B)^*$$

Lösung:

(a) Gegenbeispiel: Sei $A = \{a, aa\}$, $B = \{\epsilon\}$ und $C = \{a\}$.

$$A(B \setminus C) = \{a, aa\} \neq \{a\} = \{a, aa\} \setminus \{aa, aaa\} = AB \setminus AC$$

(b) Gegenbeispiel: Sei $A = \{a\}$ und $B = \{b\}$. Dann gilt $b \in (AB \cup B)^*$ aber $b \notin (AB)^*$.

(c) Wir zeigen die Aussage, indem wir die Teilmengenbeziehung für beide Richtungen zeigen. Beachten Sie, dass Konkatenation und Stern monoton sind, d.h. es gilt $A \subseteq C \Rightarrow AB \subseteq CB$, $B \subseteq C \Rightarrow AB \subseteq AC$ und $A \subseteq B \Rightarrow A^* \subseteq B^*$ für alle Sprachen A, B, C . Somit:

$$\begin{aligned} (A^*B^*)^* &\subseteq ((A \cup B)^*B^*)^* && \text{da } A \subseteq (A \cup B) \\ &\subseteq ((A \cup B)^*(A \cup B)^*)^* && \text{da } B \subseteq (A \cup B) \\ &= ((A \cup B)^*)^* && \text{da } C^* = C^*C^* \text{ für beliebige } C \\ &= (A \cup B)^* && \text{da } C^* = (C^*)^* \text{ für beliebige } C \end{aligned}$$

und umgekehrt:

$$\begin{aligned} (A \cup B)^* &\subseteq ((A \cup B) \cup (A^*B^*))^* && \text{da } (A \cup B) \subseteq (A \cup B) \cup C \text{ für beliebige } C \\ &= (A^*B^*)^* && \text{da } A \cup B \subseteq A^*B^* \end{aligned}$$

Aufgabe 2.2

4P

Konstruieren Sie einen **DFA** mit dem Alphabet $\Sigma = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, ., -, \}$ der nur Zeichenketten, die Dezimalzahlen darstellen, so wie sie von Menschen geschrieben werden, erkennt.

Lösung: Siehe Abbildung 1. Beachten Sie: Der Fangzustand ist nicht nicht eingezeichnet.

*Die folgenden Aufgaben können auch online gelöst werden. Melden Sie sich auf <http://automatatutor.com> an und schreiben Sie sich in den Kurs **90THEO201** mit dem Passwort **3LUPIYNJ** ein. Bitte geben Sie dennoch die Lösungen **handschriftlich** ab. Wir empfehlen die Benutzung von Mozilla Firefox, da mit anderen Browsern die Darstellung teilweise inkorrekt sein kann.*

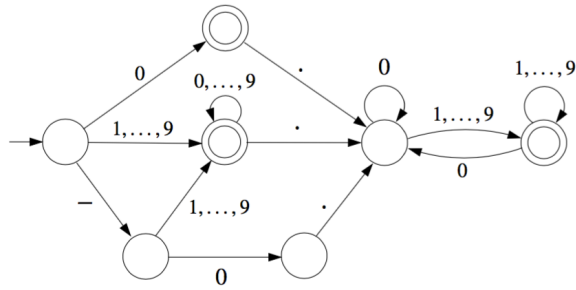


Abbildung 1: DFA für Aufgabe 2.2.

Aufgabe 2.3

1P+1P+1P+1P+1P

Finden Sie zu folgenden Sprachen reguläre Ausdrücke. Verwenden Sie für die ersten drei Aufgaben das Alphabet $\Sigma = \{a, b, c\}$ und für die letzten beiden $\Sigma = \{0, 1\}$.

- (a) Wörter gerader Länge.
- (b) Wörter, die mit einem a beginnen und enden, sowie Wörter, die mit einem b beginnen und enden.
- (c) Wörter, in denen kein a neben einem b steht.
- (d) Zahlen in Binärdarstellung (most-significant-bit-first), die durch 2 teilbar sind.
- (e) Zahlen in Binärdarstellung (most-significant-bit-first), die nicht durch 4 teilbar sind.

Lösung:

- (a) $((a|b|c)(a|b|c))^*$ (b) $a | b | a(a|b|c)^*a | b(a|b|c)^*b$ (c) $((a^*|b^*)cc^*)^*(a^*|b^*)$ (d) $(0|1)^*0$ (e) $(0|1)^*(1|10)$

Tutoraufgaben: Besprechung in KW17

Aufgabe 2.1

Wir erweitern NFAs und ε -NFAs um die Möglichkeit, mehrere Startzustände $I \subseteq Q$ zu besitzen. Ein NFA bzw. ε -NFA $N = (Q, \Sigma, \delta, I, F)$ akzeptiert ein Wort $w \in \Sigma^*$ genau dann, wenn $\emptyset \neq F \cap \bigcup_{q_0 \in I} \delta(q_0, w)$, d.h. wenn es mindestens einen akzeptierenden Ablauf von irgendeinem Startzustand aus gibt.

Zeigen oder widerlegen Sie:

- (a) Aus jedem NFA $N = (Q, \Sigma, \delta, I, F)$ kann ein NFA $N' = (Q', \Sigma, \delta', I', F')$ mit $L(N) = L(N')$ konstruiert werden, so dass
- $\dots |I'| = 1$.
 - $\dots |F'| = 1$.
 - $\dots |I'| = 1$ und $|F'| = 1$.
- (b) Wie (a) nur mit $N = (Q, \Sigma, \delta, I, F)$ ein ε -NFA mit mehreren Startzuständen.
- (c) Wie (a) nur mit einem DFA $M = (Q, \Sigma, \delta, q_0, F)$ (d.h. $|I| = 1$ gilt stets).

Lösung:

- (a) (i) Kurzversion: Ja, z.B. Potenzmengenkonstruktion einfach auf den von I aus erreichbaren Teil anwenden. Resultierender DFA N' akzeptiert dieselbe Sprache wie N .

Lange Version: Man muss erstmal zeigen, dass ein NFA mit mehreren Startzuständen immer noch eine reguläre Sprache akzeptiert. Das ist, siehe Kurzversion, ziemlich offensichtlich, aber beweisen muss man es trotzdem, daher:

Potenzmengenkonstruktion auf N anwenden. **GANZ WICHTIG:** intelligent die Konstruktion anwenden, d.h. Tiefen- oder Breitensuchen ausgehend von I ausführen; **NIE, NIEMALS** und nochmal **NIE** alle 2^Q möglichen Teilmengen konstruieren und dann fröhlich Pfeilchen dazwischen zeichnen.

Sei also $N' = (Q', \delta', I', F')$ mit $Q' \subseteq 2^Q$, $\delta': Q' \rightarrow Q'$, $I' = I$, $F' = \{S \in Q' \mid S \cap F \neq \emptyset\}$, wobei Q' und δ' die kleinsten Mengen (induktive Definition, Erreichbarkeit also DFS/BFS, kleinster Fixpunkt, blah blub) sind, welche folgende Bedingungen erfüllen:

- $I \in Q'$
- Falls $S \in Q'$, dann auch $\bigcup_{q \in S} \delta(q, a) =: S' \in Q'$ und $\delta'(S, a) \in S'$ für jedes $a \in \Sigma$.

Analog zur Vorlesung zeigt man nun, dass $\delta'(I, w) \in F'$ gdw. $\exists q_0 \in Q: \delta(q_0, w) \cap F \neq \emptyset$:

Falls $w \in L(N')$, dann

$$I \xrightarrow{w_1} S_1 \xrightarrow{w_2} S_2 \xrightarrow{w_3} \dots \xrightarrow{w_l} S_l$$

mit $S_l \in F'$, also nach Def. F' : $S_l \cap F \neq \emptyset$. Sei $q_l \in S_l \cap F$. Dann gibt es nach Def. von δ' ein $q_{l-1} \in S_{l-1}$, sodass $q_l \in \delta(q_{l-1}, w_l)$. Induktiv findet so für alle $0 \leq i \leq l$ ein $q_i \in S_i$ mit $q_{i+1} \in \delta(q_i, w_{i+1})$, womit $w \in L(N)$ folgt.

Falls $w \in L(N)$ folgt vollkommen analog, dass auch $w \in L(N')$, einfach die obige Argumentation rückwärts lesen.

(ii) Konstruktion:

Aus dem NFA $N = (Q, \Sigma, \delta, I, F)$ für L konstruieren wir einen NFA $N' = (Q \cup \{q_f\}, \Sigma, \delta', I', \{q_f\})$ mit

- $\delta'(q, a) = \delta(q, a) \cup \{q_f \mid \delta(q, a) \cap F \neq \emptyset\}$ für alle $a \in \Sigma$ und $q \in Q$
- $\delta'(q_f, a) = \emptyset$ für alle $a \in \Sigma$.
- $I' = I \cup \{q_f \mid I \cap F \neq \emptyset\}$

Nach Konstruktion ist der neue Zustand q_f der einzige Endzustand von N' .

Korrektheitsbeweis:

$L(N) \subseteq L(N')$: Falls $\varepsilon \in L(N)$, dann auch $\varepsilon \in L(N')$, da q_f dann nach Konstruktion ein Startzustand ist.

Sei daher $w \in L(N) \setminus \{\varepsilon\}$. Dann existiert eine akzeptierende Berechnung

$$q_0 \xrightarrow{w_1} q_1 \xrightarrow{w_2} q_2 \xrightarrow{w_3} \dots \xrightarrow{w_{l-1}} q_{l-1} \xrightarrow{w_l} q_l$$

mit $q_l \in F$ und $q_{i+1} \in \delta(q_i, w_{i+1})$ und $q_0 \in I$ für $w = w_1 \dots w_l$ mit $w_i \in \Sigma$.

Nach Konstruktion existiert in N' damit die akzeptierende Berechnung

$$q_0 \xrightarrow{w_1} q_1 \xrightarrow{w_2} q_2 \xrightarrow{w_3} \dots \xrightarrow{w_{l-1}} q_{l-1} \xrightarrow{w_l} q_f$$

d.h. $w \in L(N')$.

$L(N') \subseteq L(N)$: Sei nun $\varepsilon \in L(N')$: Damit muss q_f , da einziger Endzustand, auch Startzustand sein, was nach Konstruktion bedeutet, dass $\varepsilon \in L(N)$.

Sei daher $w \in L(N') \setminus \{\varepsilon\}$, d.h. es existiert eine akzeptierende Berechnung

$$q_0 \xrightarrow{w_1} q_1 \xrightarrow{w_2} q_2 \xrightarrow{w_3} \dots \xrightarrow{w_{l-1}} q_{l-1} \xrightarrow{w_l} q_f$$

Nach Konstruktion (1. Schritt) gilt somit $\delta(q_{l-1}, a) \cap F \neq \emptyset$, womit sich die Berechnung auch zu einem akzeptierenden Berechnung in N übersetzen lässt.

(iii) Nein. Sei $L = \{\varepsilon, a\}$. Angenommen es existiert sein NFA $N = (Q, \Sigma, \delta, \{q_0\}, \{q_f\})$ für L . Da $\varepsilon \in L$ und N ein NFA, $q_0 = q_f$. Da $a \in L$, $q_0 = q_f \in \delta(q_0, a)$. Somit gibt es eine a -Schleife am Zustand q_0 und somit auch $aa \in L(N) = L$. Widerspruch.

(b) (i) Ja, (fast) selbes Argument wie in (a), man muss allerdings noch den ε -Abschluss bilden:

- $I \in Q'$
- Falls $S \in Q'$, dann auch $\bigcup_{q \in S} \bigcup_{i \geq 0, j \geq 0} \delta(q, \varepsilon^i a \varepsilon^j) =: S' \in Q'$ und $\delta'(S, a) \in S'$ für jedes $a \in \Sigma$.

(ii) Ja, selbes Argument wie in (a)(ii).

(iii) Ja, da man nun mittels ε -Transitionen von einem neu eingefügten Startzustand zu jedem ursprünglichen Startzustand springen kann.

(c) (i) – (erübrigt sich)

(ii) Nein, analog zu (a)(iii).

(iii) – (erübrigt sich, äquivalent zu (c)(ii))

Aufgabe 2.2

Sei Σ ein endliches Alphabet. Ein *sternfreier* Ausdruck über Σ ist wie folgt induktiv definiert:

• Syntax:

- \emptyset ist ein (atomarer) sternfreier Ausdruck.
- ε ist ein (atomarer) sternfreier Ausdruck.
- a ist ein (atomarer) sternfreier Ausdruck für jedes $a \in \Sigma$.
- Sind α, β sternfreie Ausdrücke, dann auch $(\alpha\beta)$ (Konkatenation), $\bar{\alpha}$ (Komplement) und $(\alpha|\beta)$ (Vereinigung).
- Ansonsten gibt es keine sternfreie Ausdrücke, d.h. jeder sternfreie Ausdruck lässt sich in endlich vielen Schritte mittels Konkatenation, Komplement und Vereinigung aus den atomaren Ausdrücken konstruieren.

• Semantik:

$$L(\emptyset) := \emptyset \quad L(\varepsilon) := \{\varepsilon\} \quad L(a) := \{a\} \quad L((\alpha\beta)) := L(\alpha)L(\beta) \quad L(\bar{\alpha}) := \Sigma^* \setminus L(\alpha) \quad L((\alpha|\beta)) := L(\alpha) \cup L(\beta)$$

(a) Zeigen Sie mittels struktureller Induktion, dass $L(\alpha)$ eine reguläre Sprache über Σ ist, falls α ein sternfreier Ausdruck über Σ ist. Passen Sie hierfür den Beweis aus den Folien, dass jeder reguläre Ausdruck sich in einen NFA übersetzen lässt, entsprechend an. Verwenden Sie hierzu ohne Beweis:

Ist $M = (Q, \Sigma, \delta, q_0, F)$ ein DFA, so akzeptiert der DFA $M' = (Q, \Sigma, \delta, q_0, Q \setminus F)$ gerade die Sprache $L(M') = \Sigma^* \setminus L(M)$.

Machen Sie sich insbesondere klar: Für jeden sternfreien Ausdruck α gibt es einen regulären Ausdruck α' mit $L(\alpha) = L(\alpha')$.

Anmerkung: Die Umkehrung gilt nicht, z.B. ist $L((aa)^*)$ eine reguläre Sprache über $\Sigma = \{a\}$, welche sich nicht durch einen sternfreien Ausdruck beschreiben lässt.

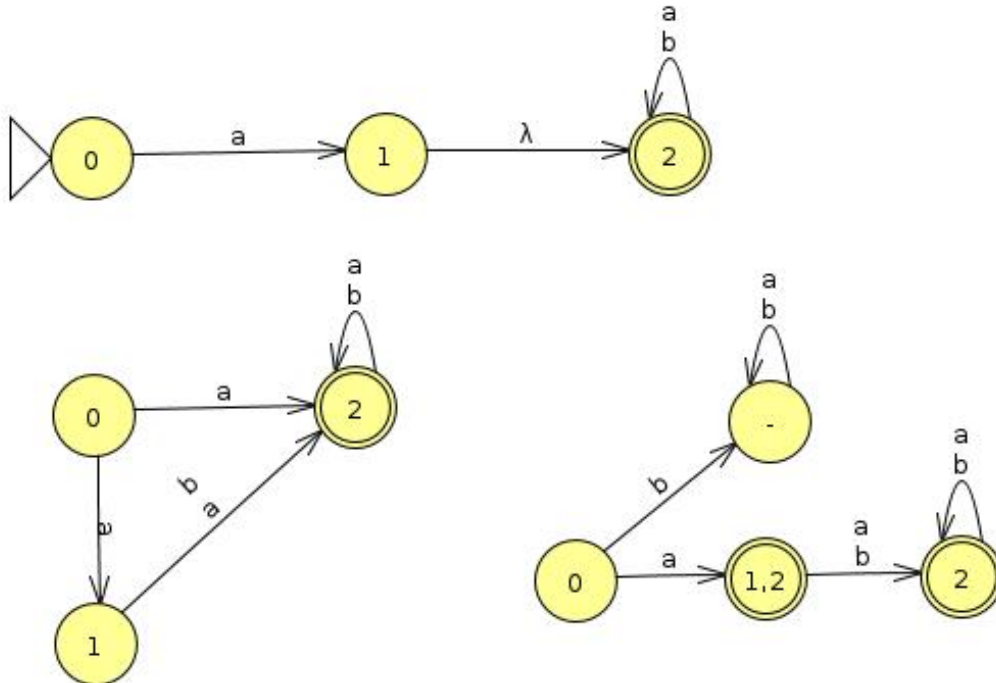
(b) Behauptung: $L(\varepsilon|\bar{a}\bar{b}|\bar{a}\bar{a}\bar{b}|\bar{b}\bar{b}\bar{b}) = L((ab)^*)$ für $\Sigma = \{a, b\}$.

Zeigen Sie die Behauptung, indem sie den sternfreien Ausdruck in einen NFA übersetzen, dann den NFA in einen regulären Ausdruck entsprechend der Vorlesung übersetzen und schließlich den erhaltenen regulären Ausdruck mittels der Äquivalenzen aus der Vorlesung zu $(ab)^*$ vereinfachen.

Für die Übersetzung des finalen NFA in einen regulären Ausdruck vereinfachen Sie den NFA zunächst: Machen Sie sich klar, dass alle Zustände, die nicht einen Endzustand erreichen können, entfernt werden dürfen, ohne dass sich hierdurch die akzeptierte Sprache ändert.

Lösung:

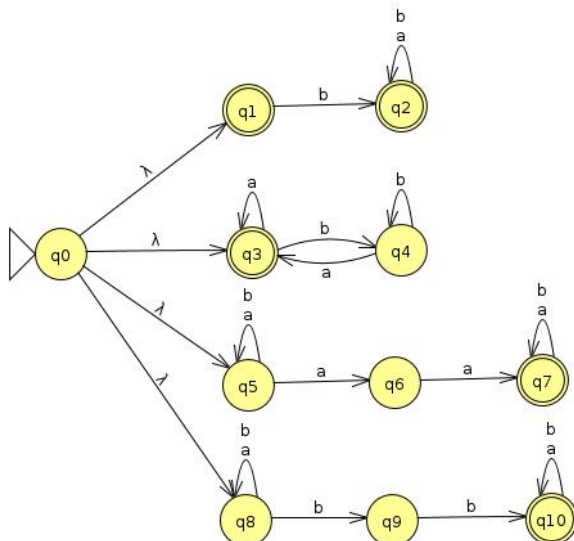
- (a) Konstruktion ist analog zu der aus der Vorlesung. Der einzige Unterschied ist, dass man jetzt nicht nur NFAs verwenden kann, um den Ausdruck in einen Automaten zu "kompilieren"; jetzt muss man beim Komplement eben erst determinisieren, um dann komplementieren zu können. Rest läuft wie bei RE.
- (b) Konstruktion von NFA zu $\overline{a\bar{b}}$:



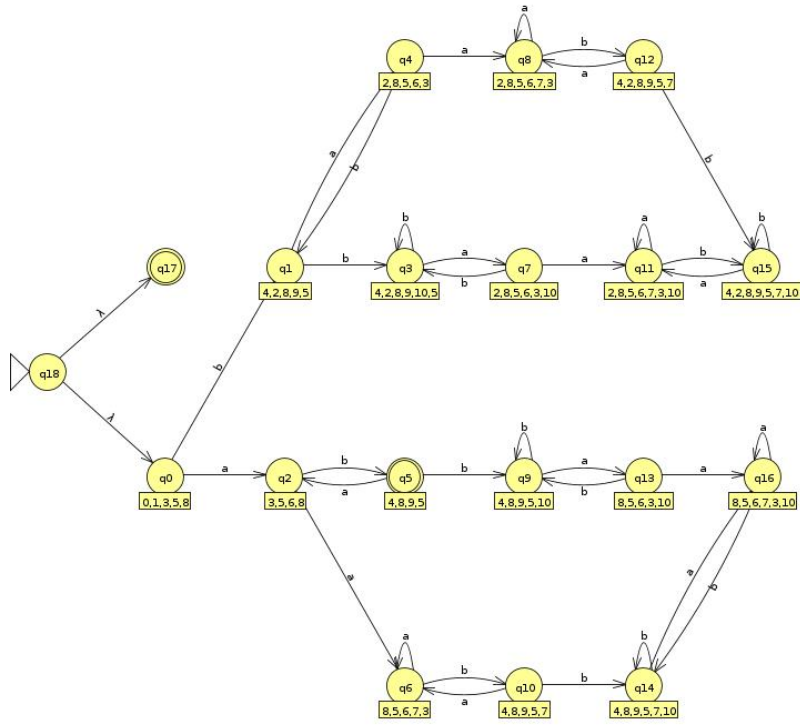
- Oben: ϵ -NFA zu $a\bar{b}$ (JFLAP verwendet λ als Bezeichner für das leere Wort!).
- Unten links: Nach Entfernen der ϵ -Transitionen. Prinzipiell notwendig, da Potenzmengenkonstruktion in VL nur für normale NFA.
- Unten rechts: Nach Potenzmengenkonstruktion.

Bleibt: End- und Nicht-End-Zustände vertauschen, um hier sogar einen DFA zu $\overline{a\bar{b}}$ zu erhalten. Nach dem Vertauschen kann man prinzipiell die Zustände $\{1, 2\}$ und $\{2\}$ wieder wegwerfen, da diese keinen Endzustand erreichen können und somit für eine akzeptierende Berechnung irrelevant sind, womit man einen kleineren NFA erhält.

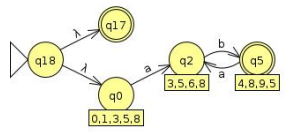
Verfährt man entsprechend für die restlichen Anforderungen und bildet entsprechend der Vorlesung die Vereinigung mit ϵ -Transitionen, so erhält man für $\overline{a\bar{b}}|\overline{b}|a\overline{a\bar{b}}|\overline{b}b\overline{b}$ den ϵ -NFA:



Entfernen von ε -Transition, dann Potenzmengenkonstruktion, dann Flippen der Endzustand und schließlich Hinzufügen von ε führt auf:



Wie man leicht sieht, lassen sich die meisten Zustände entfernen, da sie keinen Endzustand erreichen können, womit sich der Automat reduziert auf:

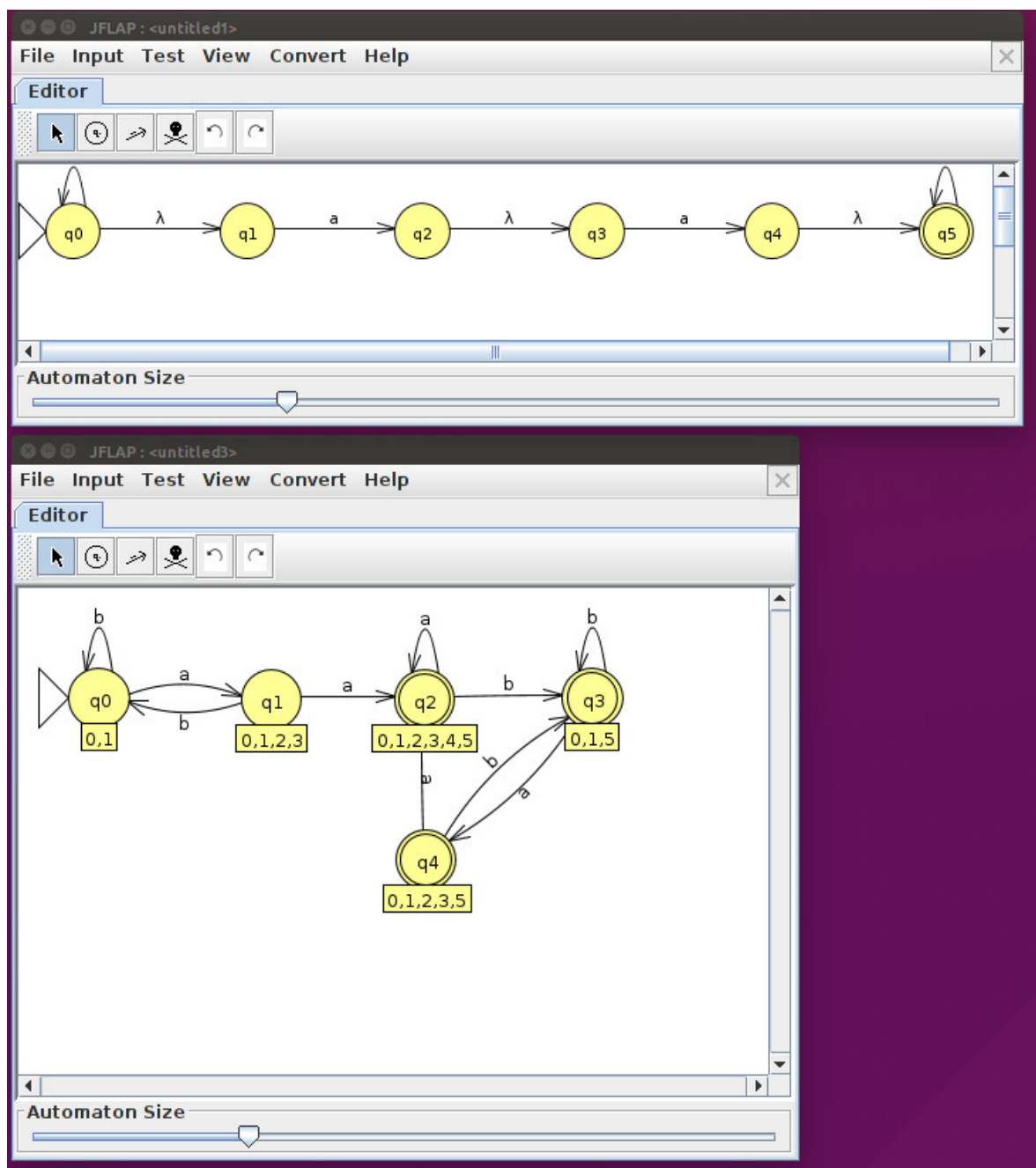


Man kann hier direkt den regulären Ausdruck $\varepsilon|ab(ab)^* \equiv (ab)^*$ ablesen.

Ansonsten noch ε -Transitionen entfernen, dann Potenzmengenkonstruktion, schließlich $R_{i,j}^k$ berechnen.

Zum Üben der Potenzmengenkonstruktion:

Konstruktion von DFA zu $\overline{\emptyset}aa\overline{\emptyset}$:



Oben der NFA zu $\overline{0aa0}$ entsprechend Vorlesung, unten der DFA nach Anwenden der Potenzmengenkonstruktion, wobei dieses Mal direkt der ϵ -Abschluss in der Potenzmengenkonstruktion angewandt wurde, siehe TA2.1 (b)(i). Wichtig: Startzustand ist in diesem Fall die Menge aller durch ϵ von q_0 erreichbaren Zustände.

Knobelaufgaben

Freiwillige Abgabe per E-Mail an theo-uebungsleitung@in.tum.de. Der Preis diese Woche für die ersten drei richtigen Gesamtlösungen ist jeweils eine Club Mate.

Aufgabe 2.1 Reguläre Ausdrücke

Sei $\Sigma = \{a, b\}$. Finden Sie einen möglichst kurzen regulären Ausdruck für die Sprache, in der alle Wörter gleich oft die Zeichenketten ab und ba enthalten.

Aufgabe 2.2 coNFA

Wir betrachten das duale Model zu dem NFA und definieren für diese Aufgabe einen coNFA, der ein Wort w akzeptiert gdw. alle Berechnungen in einem Endzustand enden. Sei $N = (Q, \Sigma, \delta, q_0, F)$ ein coNFA. Dann wird das Wort $w \in \Sigma^*$ genau dann akzeptiert, wenn $\delta(q_0, w) \subseteq F$.

- Kann ein coNFA ohne Endzustände ein Wort akzeptieren?
- Entwerfen Sie eine Übersetzung zu DFAs und beweisen Sie deren Korrektheit.