Übungen zur Wiederholung – quer durch den Stoff
Mit Lösungsbeispielen

Vollständigkeit wird nicht garantiert, und
einige sind klausuruntypisch umfangreich.
Erläuterungen (zusätzlich zur Lösung) stehen in BLAU.

1. Aufgabe (Automaten und Grammatiken)

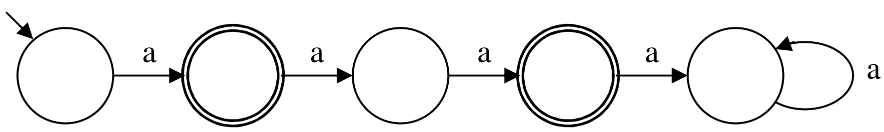
Definieren Sie die Sprache $\{a, aaa\}$ mithilfe je eines der folgenden Beschreibungsmittel:

a) deterministischer endlicher Automat,
b) nichtdeterministischer endlicher Automat, der kein deterministischer endlicher Automat ist,
c) reguläre Grammatik,
d) kontextfreie aber nicht reguläre Grammatik,
e) kontextsensitive (nicht verkürzende) aber nicht kontextfreie Grammatik,
f) nicht kontextsensitive (also irgendwo verkürzende) Chomsky-Grammatik.

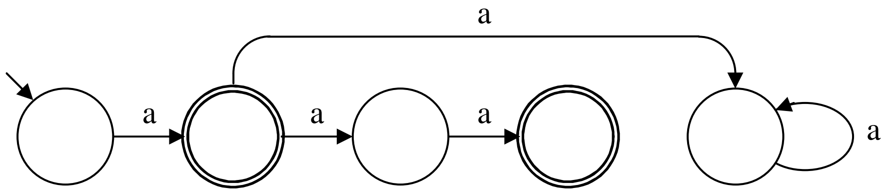
Bei Grammatiken genügen die Ableitungsregeln.

Lösungsbeispiele

a) (Zustandsnamen weggelassen, da irrelevant. Hier übrigens Minimalautomat)



b) (Zustandsnamen weggelassen, da irrelevant.
Geht z.B. auch ohne den Zustand ganz rechts.)



c) S → aA | a , A → aB , B → a

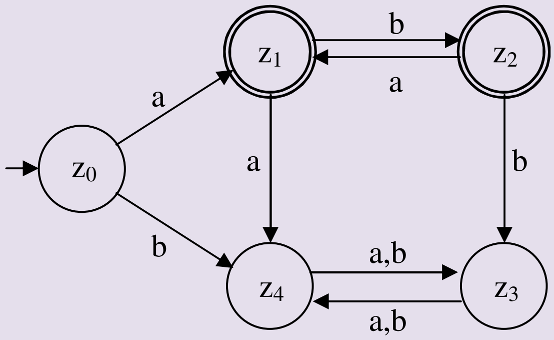
d) S → Aa | a , A → aa

e) S → a | aaA, aA → aa

f) S → aaa , aaa → a

2. Aufgabe (Minimalautomat)

Es sei der folgende endliche deterministische Automat *A* gegeben:



a) Zeigen oder widerlegen Sie, dass die Sprache *L*(*A*) unendlich viele Wörter enthält.

b) Bestimmen Sie mit Hilfe des Minimierungsalgorithmus einen minimalen Automaten *B* mit *L*(*B*) = *L*(*A*), und zeichnen Sie den Automaten *B*.

c) Geben Sie eine reguläre Grammatik *G* mit *L*(*G*) = *L*(*A*) an.

Lösungsbeispiele

a) Der Automat akzeptiert u.a. (nämlich in z_1) alle $a(ba)^n$, n = 0,1,2,... , also unendlich viele Wörter.

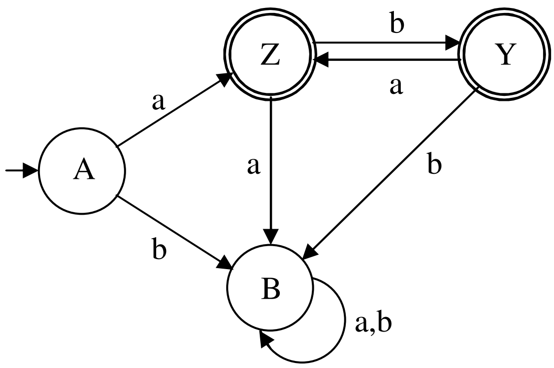
b) Minimierungsalgorithmus

	a	b
z0	z1	z4
z1	z4	z2
z2	z1	z3
z3	z4	z4
z4	z3	z3

	a	b
A	Z	A
Z	A	Z
Z	Z	A
A	A	A
A	A	A

	a	b
A	Z	B
Z	B	Y
Y	Z	B
B	B	B
B	B	B

Abgekürzt: Danach keine weitere Aufspaltung der Klassen, da in jeder Klasse alle Elemente die gleiche „Zukunft“ haben.



c) Startsymbol A

A → aZ | a | *bB*

Z → *aB* | bY | b

Y → aZ | a | *bB*

B → *aB* | *bB*

oder kürzer:

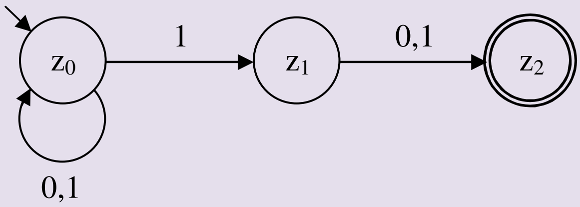
A → a | aZ

Z → b | bS

Die erste Grammatik ist nach Rezept gestrickt. Die 5 kursiven Regeln (die der „Müllbehandlung“) sind verzichtbar, da ein B am Ende immer bleibt, also daraus kein a-b-Wort mehr entstehen kann. Die zweite Grammatik entsteht durch Entfernung der verzichtbaren Regeln und die Beobachtung, dass A und Y „dasselbe tun“.
Zusatzübung: Leiten Sie aus der zweiten [einen ND-Automat, einen det. Automat und] einen Minimalautomat ab, den Sie mit dem in (b) vergleichen. Es sollte (bis auf evtl. die Zustandsnamen) der gleiche sein.

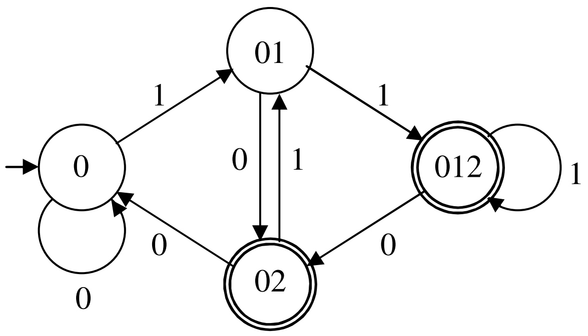
3. Aufgabe (Nichtdeterministischer Automat)

Konstruieren Sie einen deterministischen endlichen Automaten, der die gleiche Sprache akzeptiert wie der folgende NDA:



Ist der konstruierte Automat minimal?

Lösungsbeispiel



Wir wenden das Zustandsmengenrezept an. Zustand xy steht für $\{z_x,z_y\}$. Hier wurde die Konstruktion der erreichbaren Zustandsmengen gleich graphisch dargestellt. (Das geht auch in einer Tabelle. Und es geht auch mit allen Zustandsmengen, einschließlich der unerreichbaren, wäre aber unnötige Schreib-/Malarbeit.) Begonnen wird wie im NDA, hier in z_0. Akzeptiert wird überall, wo der NDA akzeptieren könnte, also wo z_2 vorkommt.

Gezeichneter Automat minimal?

Minimierung beginnen:

		0	1
0	A	A	A
01	A	Z	Z
02	Z	A	A
012	Z	Z	Z

→

A
B
Z
Y

usw.

– ja, da die Minimierung einen Automaten mit 4 Zuständen ergibt, was wir ab „usw.“ wissen, denn A, B, Y, Z können nicht weiter zerlegt werden.

4. Aufgabe (Reguläre Sprachen)

Es seien $\Sigma = \{a,b\}$ und $L_1 = \Sigma^* \setminus \{ab\}$

a) Geben Sie einen deterministischen endlichen Automaten A mit $L(A) = L_1$ an.

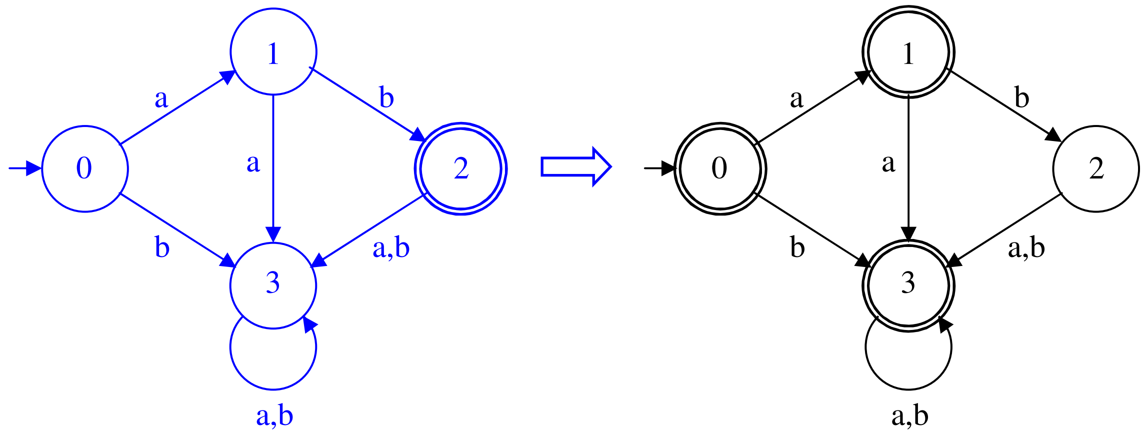
b) Geben Sie einen regulären Ausdruck R über Σ mit $L(R) = L_1$ an.

c) Geben Sie eine Sprache L_2 über Σ an, für die $L_1 \cdot L_2$ nicht regulär ist.
(ohne Begründung, aber erkennbar so, dass die Prüfung auf Zugehörigkeit zur Sprache unbeschränkten Speicher erfordert)

d) Geben Sie eine Sprache L_3 über Σ an, für die $L_1 \cap L_3$ nicht regulär ist.
(mit kurzer Begründung)

Lösungsbeispiele

a) Lösungsweg: Wir zeichnen den Automaten für {ab} und verwandeln ihn dann nach Rezept in den Automaten für das Komplement.



b) $\varepsilon + a + b + (a+b)\,(a+b)\,(a+b)\,(a+b)^* + aa + ba + bb$
(d.h. alle Wörter der Längen 0, 1 oder >2, sowie die drei der Länge 2 ohne ab, oder auch anders, z.B.)
$\varepsilon + a + b(a+b)^* + a(a+b(a+b))(a+b)^*$
(d.h. nach einem Anfangs-a kann alles kommen außer nur ein b)

c) zum Beispiel $\{ a^n b^n \mid n>0\}$
(Ein Algorithmus für das Wortproblem muss sich immer merken, wie viele a`s hintereinander zuletzt gelesen wurden; das erfordert unbegrenztes Gedächtnis.)

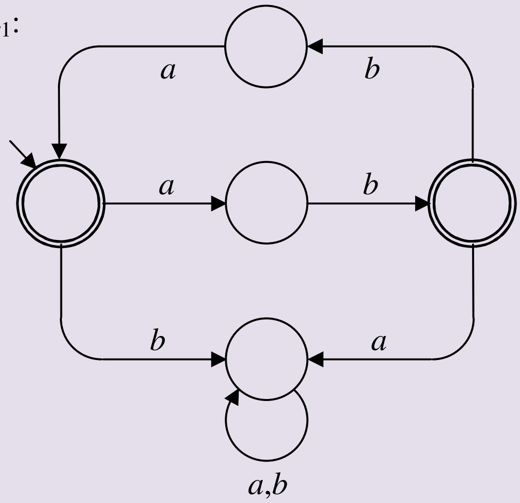
d) $L_3 := \{w \in \Sigma^* \mid \#_a(w) \neq \#_b(w)\}$. Wäre L_3 regulär, dann das Komplement $\{w \in \Sigma^* \mid \#_a(w) = \#_b(w)\}$ auch; das ist es aber nicht (erfordert unbegrenztes Gedächtnis). Also ist L_3 nicht regulär. Und $L_1 \cap L_3 = L_3 \setminus \{ab\} = L_3$.
alternativ ähnlich z.B. mit
$L_3 := \{b^n a^n \mid n \in \mathbb{N}\}$ (begründen … !)

5. Aufgabe (Reguläre Sprachen)

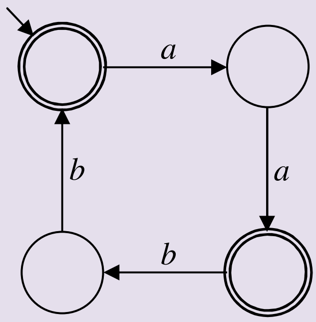
Im Folgenden sind 9 reguläre Sprachen definiert, und zwar durch

- Automaten (deterministische und nichtdeterministische), die sie akzeptieren:

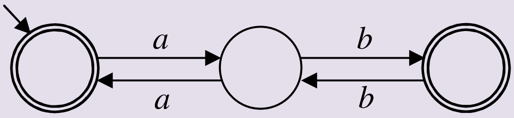
L_1:



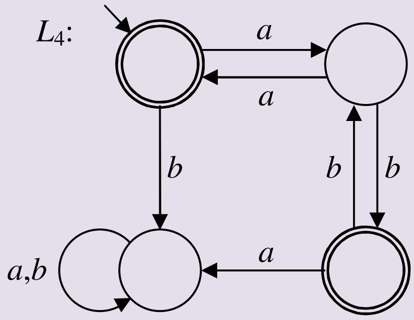
L_2:



L_3:



L_4:



- reguläre Ausdrücke:

  L_5: $(aabb)^*(aa+\varepsilon)$

  L_6: $(abba)^*(ab+\varepsilon)$

- Chomsky-Grammatiken über $\{a,b\}$ und mit Startsymbol S, die sie erzeugen:

L_7:
$$\begin{aligned} S &\to \varepsilon \mid T \\ T &\to aA \\ A &\to a \mid aT \mid b \mid bB \\ B &\to bA \end{aligned}$$

L_8:
$$\begin{aligned} S &\to \varepsilon \mid T \\ T &\to aA \\ A &\to a \mid aB \\ B &\to bC \\ C &\to b \mid bT \end{aligned}$$

L_9:
$$\begin{aligned} S &\to \varepsilon \mid T \\ T &\to aA \\ A &\to b \mid bB \\ B &\to bC \\ C &\to a \mid aT \end{aligned}$$

Schreiben Sie jede Sprachbezeichnung (L_2, …, L_9) in genau eine der unteren Boxen, und zwar so, dass jeweils identische Sprachen in der gleichen Box und unterschiedliche Sprachen in verschiedenen Boxen stehen. Evtl. bleiben Boxen leer.

Tipp: Welche drei bis vier kürzesten Wörter werden jeweils akzeptiert bzw. produziert?

L_1, L_6, L_9	L_2, L_5, L_8	L_3, L_4, L_7	
ε, ab, abab, …	ε, aa, aabb, …	ε, aa, ab, …	

6. Aufgabe (Restsprachenautomat)

Sei L die Sprache aller Wörter über $\Sigma = \{ab\}$, die nirgendwo zwei aufeinanderfolgende a enthalten, d.h. $L = \{w \in \{a,b\}^* \mid$ es gibt keine $u,v \in \{a,b\}^*$ mit $w = u$¶aa¶$v\}$.

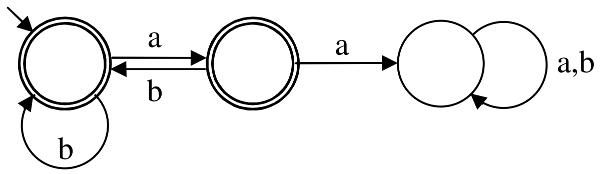
a) Zeichnen Sie einen endlichen Automaten für L mit möglichst wenigen Zuständen.

b) Geben Sie einen regulären Ausdruck für L an.

c) Tragen Sie reguläre Ausdrücke so in den Automaten aus (a) ein, dass Sie den Restsprachenautomat von L erhalten.

Lösungsbeispiel

a)



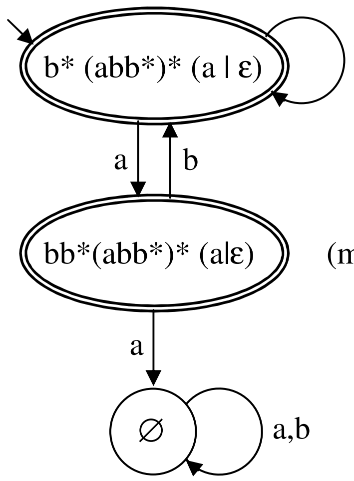
b) b* (abb*)* (a | ε)

nach den letzten *b*'s evtl. noch ein einzelnes *a*

null, ein- oder mehrmals: ein *a*, gefolgt von mind. einem *b*

evtl. *b*'s vor dem ersten *a*

c)



(mind. ein *b* vor dem ersten *a*, sonst wie oben)

Restsprachen-Schritte, z.B. vom obersten zum mittleren Zustand:

- Welche Wörter beginnen mit a?
  (abb*)* (a | ε) (außer ε)
- Welche bleiben nach Entfernung des Anfangs-a?
  Wenn ein abb* vorkam: Das bb* vom ersten abb*, dann immer noch beliebig viele abb* und dann (a | ε), also bb*(abb*)* (a|ε).
- Wenn kein abb* vorkam (also das einsame a hinten gestrichen wurde):
  ε, und das wird durch bb*(abb*)* (a|ε) (alle * null-mal wählen) mit abgedeckt.