

- 0. Grundbegriffe
- 1. Endliche Automaten**
- 2. Formale Sprachen
- 3. Berechenbarkeitstheorie
- 4. Komplexitätstheorie

- 1.1. Grundlagen
- 1.2. Minimierungsalgorithmus
- 1.3. Grenzen endlicher Automaten**

► Gibt es Sprachen, die nicht Automaten Sprachen sind?

- Endliche Automaten bilden ein sehr einfaches Beschreibungsmittel und Berechnungsmodell.
Falls es Sprachen gibt, die keine Automaten Sprachen sind, benötigt man komplexere (oder zumindest andere) Beschreibungsmittel bzw. Berechnungsmodelle.
- Wir wollen in diesem Abschnitt zeigen, dass es tatsächlich Sprachen gibt, die keine Automaten Sprachen sind.
- Dabei werden wir allgemeine Eigenschaften von Automaten Sprachen identifizieren.
- Gleichzeitig werden wir einen neuen Weg zum Minimalautomaten kennenlernen.

► Ausblick

- Wir betrachten zwei brauchbare Ansätze, um ggf. nachzuweisen, dass es zu einer gegebenen Sprache L keinen endlichen Automaten A mit $L(A) = L$ gibt:
 - Man analysiert die Eigenschaften eines aus L konstruierten „Automaten“.
 - Man wendet das sogenannte Pumping-Lemma an.
- ... und werden ihre Vor- und Nachteile diskutieren.

- ▶ Die Fortsetzungsmengen-Abbildung einer Sprache über Σ

Vgl. Algebra:
 $(ab)^{-1}(abcb) = cb$

Für jede Sprache $L \subseteq \Sigma^*$ und jedes Wort $u \in \Sigma^*$ sei $u^{-1}(L) := \{v \in \Sigma^* \mid u \circ v \in L\}$.

Weg von L nach $u^{-1}(L)$:
Alle Wörter **wegwerfen**, die **nicht** mit u beginnen. Vom Rest die Anfangs- u **löschen**.

... das, was in L
„nach u noch kommen kann“,
die **Restsprache** von L nach u

- $a^{-1}(\emptyset) = \emptyset$
- $a^{-1}(\{\epsilon\}) = \emptyset$
- $a^{-1}(\{ba\}) = \emptyset$
- $a^{-1}(\{a, abc, bad\}) = \{\epsilon, bc\}$

► Der Restsprachenautomat $A(L)$ einer Sprache L

Es sei $L \subseteq \Sigma^*$ eine formale Sprache über dem Alphabet Σ .

Der (nicht notwendig endliche) **Restsprachenautomat** von L ist $A(L) = [Z, \Sigma, z_0, F, \delta]$ mit

- $Z = \{ u^{-1}(L) \mid u \in \Sigma^* \},$
- $z_0 = L,$
- $F = \{ z \in Z \mid \varepsilon \in z \},$
- Für alle $z \in Z$ und $a \in \Sigma$:
 $\delta(z, a) = a^{-1}(z) = \{ a^{-1}(w) \mid w \in z \}.$

neuer Begriff: ein **evtl. unendlicher Automat**

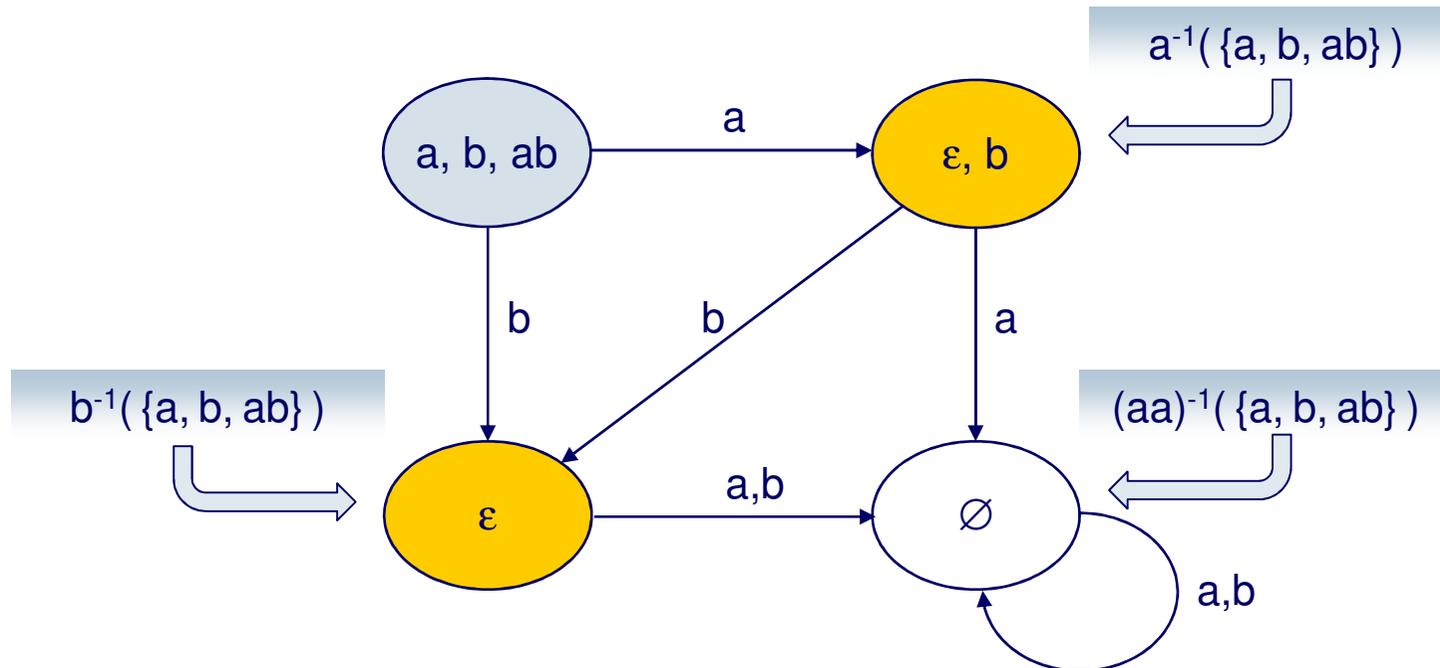
Gewöhnungsbedürftig: Jeder Zustand ist hier eine komplette Sprache!

$A(L)$ ist nicht unbedingt gut handhabbar, denn:

Wie gehen wir dabei z.B. mit unendlich vielen unendlichen Sprachen um?

► Der Restsprachenautomat $A(L)$ einer Sprache L (Beispiel 1)

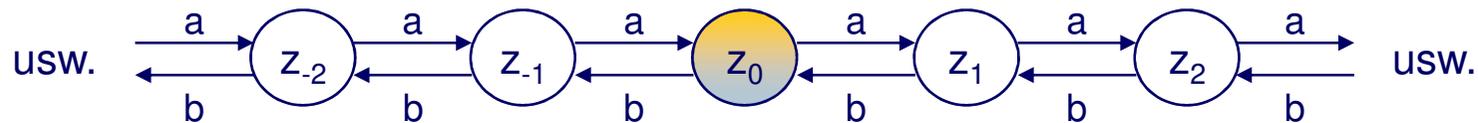
- Sei $\Sigma = \{a, b\}$ und $L_1 = \{a, b, ab\}$



- Beispiel 2 $L_2 = \{w \in \{a, b\}^* \mid \#(a, w) = \#(b, w)\}$ → $A(L_2) = ?$ 

► Der Restsprachenautomat $A(L)$ einer Sprache L (Beispiel 2)

- Sei $\Sigma = \{a, b\}$ und $L_2 = \{ w \in \{a, b\}^* \mid \#(a, w) = \#(b, w) \}$.
- Dann sieht $A(L_2)$ wie folgt aus:



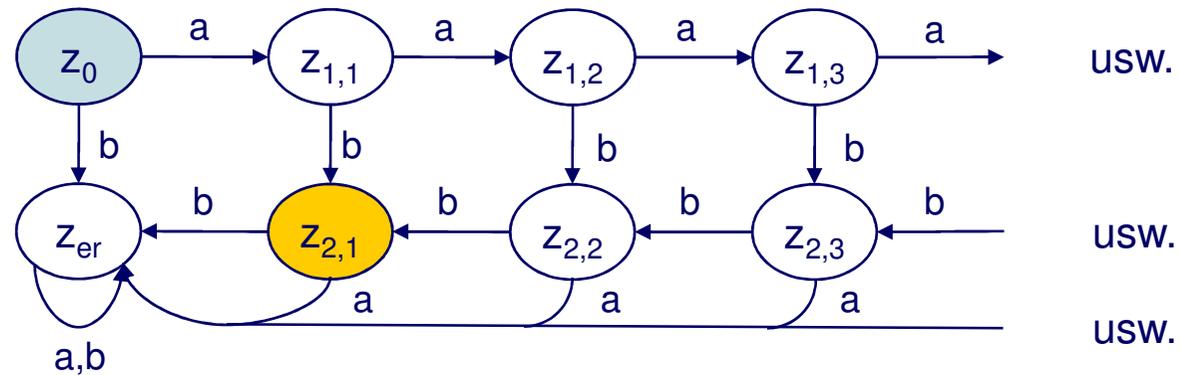
$$z_n = \{ w \in \{a, b\}^* \mid \#(a, w) - \#(b, w) = n \}.$$

Es gibt in $A(L_2)$ keinen Müllzustand – warum?)

- Beispiel 3 $L_3 = \{ a^n b^n \mid n \in \mathbb{N} \} = \{ a^n b^n \mid n \geq 1 \} \rightarrow A(L_3) = ?$ 

► Der Restsprachenautomat $A(L)$ einer Sprache L (Beispiel 3)

- Sei $L_3 = \{ a^n b^n \mid n > 0 \}$.
- Dann sieht $A(L_3)$ wie folgt aus:



$$z_0 = L_3$$

$$z_{er} = \emptyset$$

$$z_{1,k} = \{ a^n b^{n+k} \mid n \geq 0 \}$$

$$z_{2,k} = \{ b^{k-1} \}$$

- ▶ Der Restsprachenautomat $A(L)$ einer Sprache L
... hat tolle Eigenschaften (Myhill/Nerode):

Fakt:

Genau dann ist der **Restsprachenautomat** $A(L)$ **endlich**, wenn L die Sprache $L(A)$ eines **endlichen Automaten** A ist.

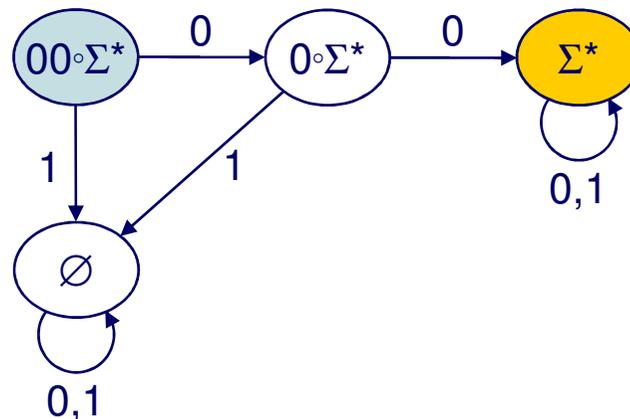
Fakt:

Für jeden endlichen Automaten A ist $A(L(A))$ ein **Minimalautomat** zu A , d.h. der Restsprachenautomat ist minimal.

*Da $A(L)$ für jede Sprache existiert und von fundamentaler Bedeutung ist, nennt man ihn auch den **kanonischen Automaten** von L .*

► Der Restsprachenautomat $A(L)$ einer Sprache L (Beispiel 4)

- Sei $\Sigma = \{ 0,1 \}$ und $L = \{ 00 \cdot w \mid w \in \Sigma^* \}$.



► Konsequenzen

Um für eine gegebene Sprache L zu überprüfen, ob es einen endlichen Automaten A mit $L(A) = L$ gibt, kann man wie folgt vorgehen:

- Stelle fest, ob L endlich oder unendlich viele Restsprachen hat.
 - Falls es endlich viele sind, gibt es einen endlichen Automaten A mit $L(A) = L$.
 - Falls es unendlich viele sind, gibt es keinen endlichen Automaten A mit $L(A) = L$.

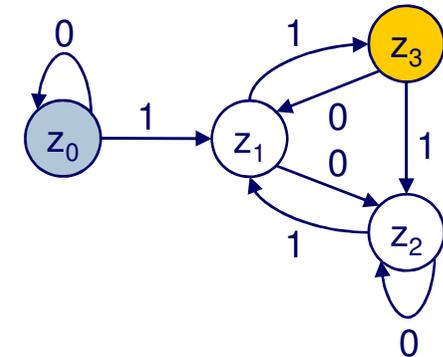
*Hinweis: Dieser Ansatz greift **im Prinzip** immer.
Es ist nur manchmal nicht ganz leicht festzustellen,
wie viele Restsprachen es gibt.
Vor allem: Wir haben dafür **keinen Algorithmus!***

▶ weiteres Vorgehen

- Wir schauen uns jetzt eine weitere Eigenschaft der Sprachen an, die mit endlichen Automaten beschrieben werden können.
- Dabei konzentrieren wir uns auf unendliche Sprachen und fragen uns, wie ein endlicher deterministischer Automat Wörter verarbeitet, die mindestens so viele Zeichen haben, wie der Automat Zustände hat.

► Beispiel

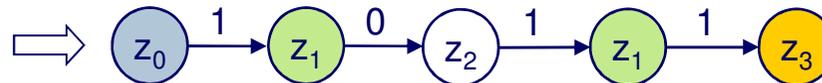
- Wir betrachten das folgende Wort $w \in L(A)$:
 - $w = a_1 a_2 a_3 a_4 = 1011$
- A verarbeitet das Wort $w \in L(A)$ wie folgt:
 - Start in z_0 ; $\delta(z_0, a_1) = z_1$, $\delta(z_1, a_2) = z_2$, $\delta(z_2, a_3) = z_1$ und $\delta(z_1, a_4) = z_3$
- **Konsequenz:**
 - bei der Verarbeitung des Wortes w werden fünf Zustände besucht: der Startzustand und die vier Zustände, die besucht werden, wenn a_1 , a_2 , a_3 bzw. a_4 verarbeitet wurden.
 - Da A nur vier Zustände hat, muss deshalb mindestens ein Zustand mehrfach besucht werden



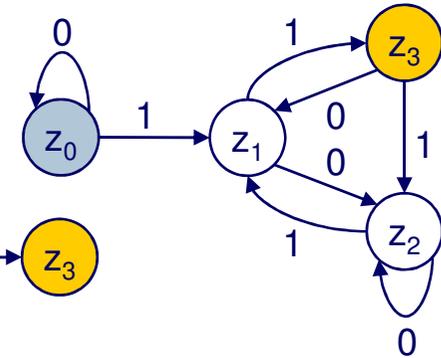
Vergleiche: Schleifen-Lemma für Wege!

► Beispiel (weiter)

- A verarbeitet das Wort $1011 \in L(A)$ wie folgt:



$$\Rightarrow 1 \cdot 01 \cdot 1 = u \cdot v \cdot w = u \cdot v^1 \cdot w \in L(A)$$



- Da der Zustand z_1 zweimal besucht wird, können wir das Wort w in die Bestandteile $u=1$, $v=01$ und $w=1$ zerlegen mit:

- $\delta^*(z_0, u) = z_1$, $\delta^*(z_1, v) = z_1$, $\delta^*(z_1, w) \in F$, so dass
- für alle $k \in \mathbb{N}_0$ das Wort $u \cdot v^k \cdot w$ ebenfalls zur Sprache $L(A)$ gehört, z.B.



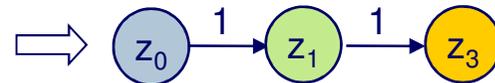
($k=2$)



$$\Rightarrow 1 \cdot 0101 \cdot 1 = u \cdot v v \cdot w = u \cdot v^2 \cdot w \in L(A)$$



($k=0$)



$$\Rightarrow 1 \cdot \varepsilon \cdot 1 = u \cdot v^0 \cdot w \in L(A)$$

► Hilfsbegriff (Aufpumpbarkeit)

- Es sei $s \in L \subseteq \Sigma^*$.
- Es sei $n \in \mathbb{N}$.

s heißt **n-aufpumpbar in L** , falls es Wörter $u, v, w \in \Sigma^*$ gibt, für die gilt:

- $s = u \circ v \circ w$, $|u \circ v| \leq n$ und $|v| \geq 1$,
- $u \circ v^k \circ w \in L$ für alle $k \in \mathbb{N}_0$.

$k=0$ ist eher „Luft ’rauslassen“ als aufpumpen.

► ... und die „Kontraposition“ (Nicht- Aufpumpbarkeit)

s ist nicht n-aufpumpbar in L gdw. für alle Wörter $u, v, w \in \Sigma^*$ gilt:

- Wenn die Bedingungen $s = u \circ v \circ w$, $|u \circ v| \leq n$ und $|v| \geq 1$ erfüllt sind, so gilt $u \circ v^k \circ w \notin L$ für mindestens ein $k \in \mathbb{N}_0$.

► Beispiele

- Es sei $L = \{ 111 \} \cup \{ 1 \cdot 0^t \mid t > 0 \}$

Dann gilt:

- $s = 100$ gehört zu L und ist 2-aufpumpbar in L :
 - Wähle $u = 1, v = 0, w = 0$
- $s = 111$ gehört zu L und ist nicht 2-aufpumpbar in L :
 - Egal, wie $u, v,$ und w gewählt werden gilt, $u \cdot v^2 \cdot w \notin L$.
- Jedes Wort $s \in L$ mit $|s| \geq 4$ ist 2-aufpumpbar in L :
 - $s = 1 \cdot 0 \cdot s'$. Wähle $u = 1, v = 0, w = s'$.

► Pumping-Lemma für Automatensprachen

Beweis wie beim Schleifen-Lemma für Wege!

- Ist L eine unendliche Sprache, dann gilt:

Wenn es einen endlichen Automaten A mit $L(A)=L$ gibt, dann gibt es eine Zahl $n \in \mathbb{N}$, so dass jedes Wort $w \in L$ mit $|w| \geq n$ ein n -aufpumpbares Wort ist (die sog. **Pumping-Eigenschaft**).

► Logisch äquivalente Formulierung (Kontraposition)

- Ist L eine unendliche Sprache, dann gilt:

und bei **endlichen** Sprachen? ► S.20

Wenn es zu jeder Zahl $n \in \mathbb{N}$ ein Wort $s_n \in L$ mit $|s_n| \geq n$ gibt, das nicht n -aufpumpbar ist (**No-Pumping-Eigenschaft**) so gibt es keinen endlichen Automaten A mit $L(A)$.



*... leider **nicht** „Genau dann, wenn“, denn manche Nicht-Automatensprachen haben auch die Pumping-Eigenschaft.*

► Verwendung

- Um zu zeigen, dass es für eine gegebene Sprache L keinen endlichen Automaten A mit $L(A) = L$ gibt, benutzt man die Kontraposition des Pumping-Lemmas und geht wie folgt vor:
 - Man zeigt, dass es zu jedem $n \in \mathbb{N}$ ein Wort $s_n \in L$ mit $|s_n| \geq n$ gibt, das nicht n -aufpumpbar ist.

Für jedes n bezeugt das Wort s_n , dass es keinen endlichen Automat A mit n Zuständen gibt, für den $L = L(A)$ gilt.

- Man will also zeigen, dass es zu jedem $n \in \mathbb{N}$ ein Wort $s_n \in L$ mit $|s_n| \geq n$ gibt, so dass für alle Wörter $u, v, w \in \Sigma^*$ gilt:
 - Wenn die Bedingungen $s_n = u \circ v \circ w$, $|u \circ v| \leq n$ und $|v| \geq 1$ erfüllt sind, so gilt für mindestens ein $k \in \mathbb{N}$ $u \circ v^k \circ w \notin L$.

► Beispiel für das „No-Pumping-Lemma“ (1)

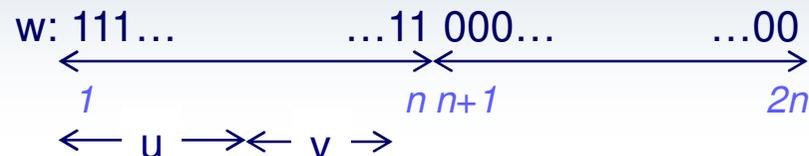
- Sei $\Sigma = \{ 0,1 \}$ und $L = \{ w \in \Sigma^* \mid w \text{ hat genauso viele Nullen wie Einsen} \}$
- Wir wollen zeigen, dass es keinen endlichen Automaten A mit $L(A) = L$ gibt.
- Es sei nun $n \in \mathbb{N}$
- Wir wählen das Wort $s_n = 1^n \cdot 0^n$ (Offenbar gilt $s_n \in L$ und $|s_n| \geq n$.)
- Es seien u, v und w so gewählt, dass $s_n = u \cdot v \cdot w$, $|u \cdot v| \leq n$ und $|v| \geq 1$ gilt
- Also gilt:
 - $u = 1^{|u|}$, $v = 1^{|v|}$ und $w = 1^{n-|u \cdot v|} 0^n$
- Daher gilt für das Wort $u \cdot v^2 \cdot w$ (Wir nehmen also $k = 2$):
 - $u \cdot v^2 \cdot w = 1^{|u|} \cdot 1^{2|v|} \cdot 1^{n-|u \cdot v|} 0^n = 1^{n+|v|} 0^n$, und $n+|v| > n$,
 - also $u \cdot v^2 \cdot w \notin L$, da dieses Wort mehr Einsen als Nullen hat.

► Beispiel für das „No-Pumping-Lemma“ (2)

- $L = \{ w \in \Sigma^* \mid w \text{ hat genauso viele Nullen wie Einsen} \}$
- Wir wählen das Wort $s_n = 1^n 0^n$ (Offenbar gilt $s_n \in L$ und $|s_n| \geq n$.)
- Es seien u, v und w so gewählt, dass $s_n = u \circ v \circ w$, $|u \circ v| \leq n$ und $|v| \geq 1$ gilt.

Anschaulich und weniger mathematisch:

Das Aufpumpen (0-/2-/3-...fach v), geschähe rein im Bereich der Einsen und würde die Anzahlen der Einsen und Nullen ungleich machen. Das geht nicht innerhalb von L .



Übrigens: Bei v wird nicht nur „aufgepumpt“, sondern als Spezialfall (v^0) auch mal „die Luft herausgelassen“.

► Anmerkungen

- Mit der Kontraposition des Pumping-Lemmas kann man offenbar nur für unendliche Sprachen L zeigen, dass es keinen endlichen Automaten A mit $L(A) = L$ gibt. Man braucht ja unendlich viele Wörter.
- Das ist aber keine Einschränkung, weil man sich leicht überlegen kann, dass es für jede endliche Sprache L einen endlichen Automaten A mit $L(A) = L$ gibt

... was wir auf mindestens zwei Arten sehen können:

- *Es gibt eine triviale Baum-Konstruktion für einen passenden Automaten.*
- *Es gibt nur endlich viele Restklassensprachen.*
- Endliche Sprachen L haben sowieso trivialerweise die Pumping-Eigenschaft. *Begründung:* Die maximale Wortlänge in L sei m . Dann kann man jedes Wort $w \in L$ der Länge $|w| \geq n = m+1$ innerhalb der Sprache aufpumpen. Wie? Warum? ... weil es solche w einfach nicht gibt! (Vgl. „Alle meine Rolls-Royce sind lila.“ – wahr, da es keine gibt.)

► weitere Anmerkungen

- Unschön ist, dass es (natürlich unendliche) Sprachen L gibt, die
 - einerseits keine Automaten Sprachen sind, d.h. es gibt **keinen** endlichen Automaten A mit $L(A) = L$,und die
 - andererseits **trotzdem** die Pumping-Eigenschaft haben, d.h. es gibt ein $n \in \mathbb{N}$, so dass jedes Wort $w \in L$ mit $|w| \geq n$ n -aufpumpbar ist.
- Beispielsweise gilt mit
 - $\Sigma = \{0,1\}$ und $L = \{0^n 1^m 0^m \mid n, m \in \mathbb{N}\} \cup \{1^n 0^m \mid n, m \in \mathbb{N}_0\}$
 - L ist keine Automaten Sprache.
 - Jedes Wort $w \in L$ mit $|w| \geq 1$ ist ein 1-aufpumpbares Wort.

Wie sieht man das ein ?



► weitere Anmerkungen, Antwort auf vorige Frage

- Beispielsweise gilt mit $\Sigma = \{0,1\}$ und $L = \{0^n 1^m 0^m \mid n, m \in \mathbb{N}\} \cup \{1^n 0^m \mid n, m \in \mathbb{N}_0\}$
- L ist keine Automatensprache.
- Jedes Wort $w \in L$ mit $|w| \geq 1$ ist ein 1-aufpumpbares Wort.

Wie sieht man das ein ?

- Etwa so:
- L hat unendlich viele Restsprachen (vgl. $a^n b^n$ -Beispiel), bzw. ein Automat für L bräuchte ein unendlich großes Gedächtnis:
Falls 0 vorne, wie viele 1en kamen bisher?
 - W hat $n > 1$ führende Nullen vorne →
führende Null streich- und vervielfachbar zu Wort aus erster Menge.
Einsame 0 vorne streichbar zu Wort aus zweiter Menge, vervielfachbar zu Wort erster Menge.
Führende 1 vorne streichbar und vervielfachbar zu Wort aus zweiter Menge.

► Klausurschwerpunkte

- wichtige Fähigkeiten/Fertigkeiten
 - endliche Automaten für einfache Sprachen angeben,
 - endliche Automaten minimieren,
 - begründen, weshalb es für eine gegebene Sprache keinen endlichen Automaten gibt, der sie akzeptiert.
- relevante Begriffe und Zusammenhänge
 - die von einem endlichen Automaten per $K_A(z)$ induzierte Klasseneinteilung auf Σ^* und die per $L_A(z)$ induzierte Äquivalenzrelation auf der Menge der Zustände,
 - der zu einem endlichen Automaten gehörende Quotientenautomat als Minimalautomat,
 - der Restsprachenautomat einer Sprache als Minimalautomat,
 - Pumping- und No-Pumping-Eigenschaften, Pumping-Lemma und Nicht-Automaten-Sprachen.