Informatik III - Tutorium XIII & XV (SR -107)

Tut Nr. 3 – Üb3, Pumping-Lemma, Minimierung

David Münch

Universität Karlsruhe (TH)
Institut für Informatik
ITI Wagner

14. November 2007

Universität Karlsruhe (TH)
Forschungsuniversität • gegründet 1825

Inhaltsverzeichnis

Inhaltsverzeichnis

Inhaltsverzeichnis

Inhaltsverzeichnis

Organisatorisches

Email: muenchdavid@gmail.com

https://www.stud.uni-karlsruhe.de/∼uhbro/

Tutorium 13: Mittwochs 8:00 Uhr - Raum -107
Tutorium 15: Mittwochs 9:45 Uhr - Raum -107

Übungsblattabgabe Donnerstag.

Was wollen wir heute erreichen?

Was wollen wir heute erreichen?

- Pumping-Lemma verstehen und anwenden.

Was wollen wir heute erreichen?

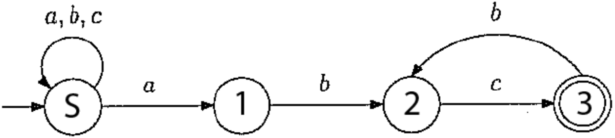
- Pumping-Lemma verstehen und anwenden.
- Minimierungsalgorithmus anwenden.

Was wollen wir heute erreichen?

- Pumping-Lemma verstehen und anwenden.
- Minimierungsalgorithmus anwenden.
- Aufgaben lösen.

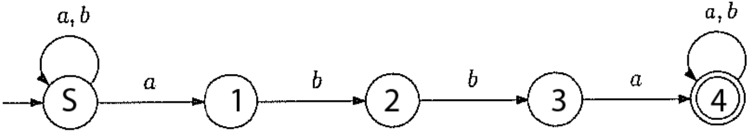
Aufgabe 5a

$$A = (Q, \Sigma, \delta, s, F)$$
$$Q = \{S, 1, 2, 3\}$$
$$\Sigma = \{a, b, c\}$$
$$s = S$$
$$F = \{3\}$$



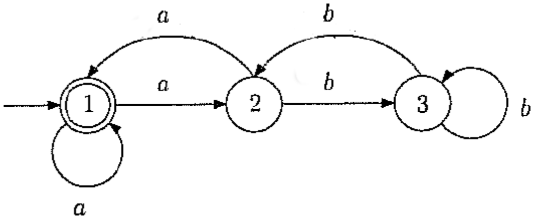
Aufgabe 5b

$$A = (Q, \Sigma, \delta, s, F)$$
$$Q = \{S, 1, 2, 3, 4\}$$
$$\Sigma = \{a, b\}$$
$$s = S$$
$$F = \{3\}$$



Aufgabe 6a

$$A = (Q, \Sigma, \delta, s, F)$$
$$Q = \{1, 2, 3\}$$
$$\Sigma = \{a, b\}$$
$$s = 1$$
$$F = \{1\}$$



Aufgabe 6a

Korrektheitsbeweis: zu zeigen: $L(A) = L(r)$

Sei $w \in L(r)$. Dann gibt es eine Abarbeitung für w, sodass A in einem Endzustand hält. Zerlege dazu w in Teilwörter $w_1, \ldots, w_k$ mit $w_i \in (a \cup ab(b)^*ba)$. Jedes w_i kann so abgearbeitet werden, dass A im Endzustand hält.

Aufgabe 6a

Korrektheitsbeweis: zu zeigen: $L(A) = L(r)$

Sei $w \in L(r)$. Dann gibt es eine Abarbeitung für w, sodass A in einem Endzustand hält. Zerlege dazu w in Teilwörter $w_1, ..., w_k$ mit $w_i \in (a \cup ab(b)^*ba)$. Jedes w_i kann so abgearbeitet werden, dass A im Endzustand hält.

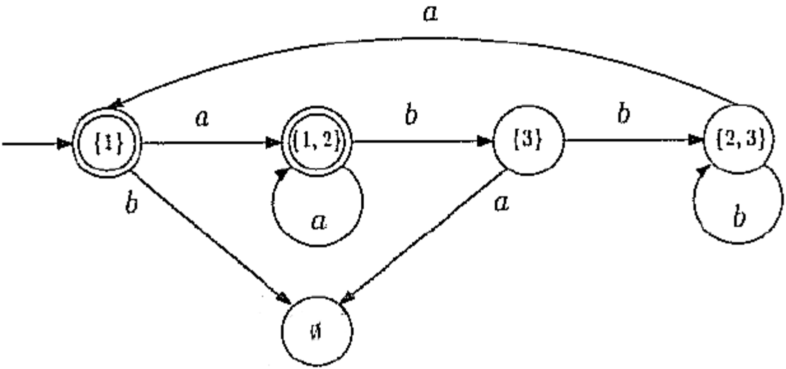
Sei nun $w \in L(A)$, so hält A im Zustand 1. Betrachtet man die Kreise im Graphen, die im Zustand 1 starten und enden, so sind die Kanten der Kreise entweder mit a oder mit $ab(b)^*ba$ beschriftet. Damit ist w auch in $L(r)$. $\square$

Aufgabe 6b

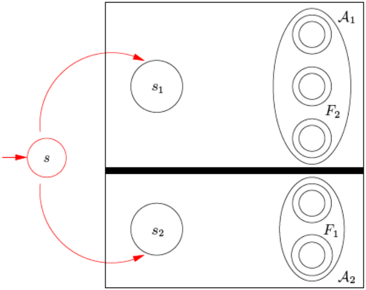
$$\tilde{A} = (\tilde{Q}, \Sigma, \tilde{\delta}, \tilde{s}, \tilde{F})$$
$$\tilde{Q} = \{\{1\}, \{1,2\}, \{3\}, \{2,3\}\}$$
$$\Sigma = \{a, b\}$$
$$\tilde{s} = \{1\}$$
$$\tilde{F} = \{\{1\}, \{1,2\}\}$$

$\tilde{\delta}$	a	b
$\{s\}$	$\{1,2\}$	$\emptyset$
$\{1,2\}$	$\{1,2\}$	$\{3\}$
$\{3\}$	$\emptyset$	$\{2,3\}$
$\{2,3\}$	$\{1\}$	$\{2,3\}$
$\emptyset$	$\emptyset$	$\emptyset$

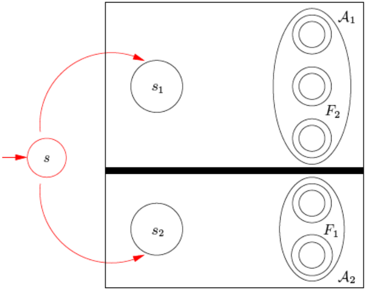
Aufgabe 6b

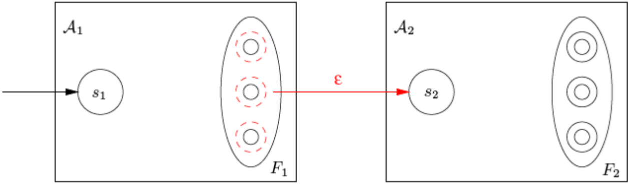


Aufgabe 6c

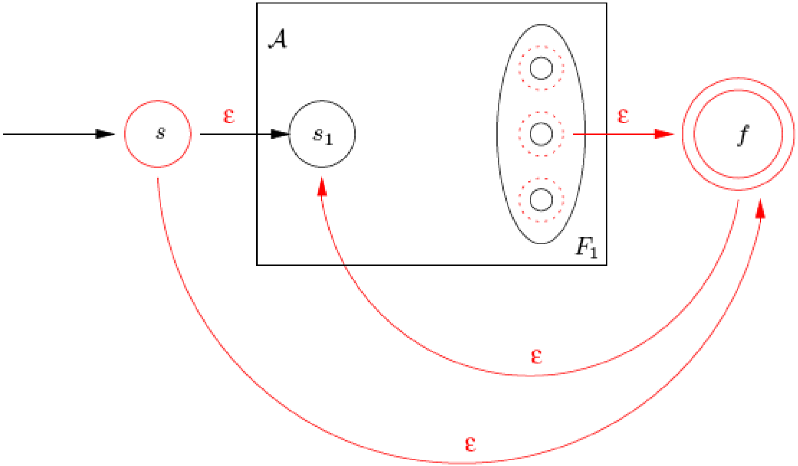


Aufgabe 6c


$\mathcal{A}_1$
s_1
F_2
s
s_2
F_1
$\mathcal{A}_2$


$\mathcal{A}_1$
s_1
F_1
ε
$\mathcal{A}_2$
s_2
F_2

Aufgabe 6c



Aufgabe 7

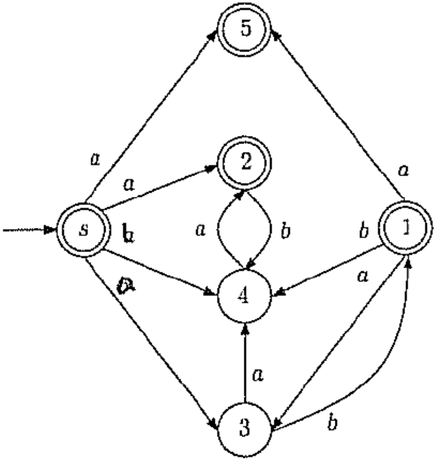
Die vom Automaten akzeptierte Sprache $L(A)$, können wir mit einem regulären Ausdruck beschreiben:
$r = (ab)^*a \cup (ab)^*(\varepsilon \cup aaa)(ba)^*)$
Die Sprache $L(r)$ ist gleich der Sprache $L(A)$.

Aufgabe 7

Die vom Automaten akzeptierte Sprache $L(A)$, können wir mit einem regulären Ausdruck beschreiben:
$r = (ab)^*a \cup (ab)^*(\varepsilon \cup aaa)(ba)^*)$
Die Sprache $L(r)$ ist gleich der Sprache $L(A)$.

ε-Abschluss:
$E(s) = \{s, 1, 2, 5\}$
$E(1) = \{1, 2, 5\}$
$E(2) = \{2, 5\}$
$E(3) = \{3\}$
$E(4) = \{4\}$
$E(5) = \{5\}$

Aufgabe 7



Aufgabe 8

a) Die Sprache aller Suffixe von $L(A)$.

Aufgabe 8

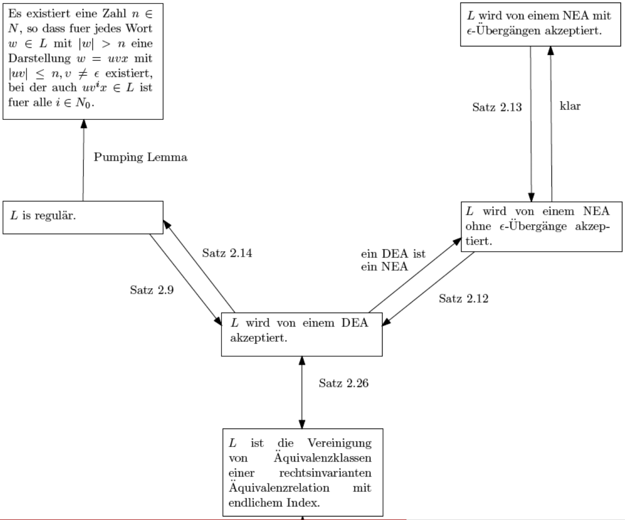
a) Die Sprache aller Suffixe von $L(A)$.

b) Die Sprache aller Präfixe von $L(A)$.

Aufgabe 8

a) Die Sprache aller Suffixe von $L(A)$.
b) Die Sprache aller Präfixe von $L(A)$.
c) Die Sprache aller Teilwörter von $L(A)$.

Überblick

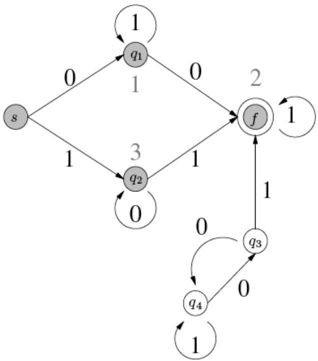

Es existiert eine Zahl $n \in N$, so dass fuer jedes Wort $w \in L$ mit $|w| > n$ eine Darstellung $w = uvx$ mit $|uv| \leq n, v \neq \epsilon$ existiert, bei der auch $uv^ix \in L$ ist fuer alle $i \in N_0$.
Pumping Lemma
L is regulär.
Satz 2.14
Satz 2.9
L wird von einem NEA mit ϵ-Übergängen akzeptiert.
Satz 2.13
klar
L wird von einem NEA ohne ϵ-Übergänge akzeptiert.
ein DEA ist ein NEA
Satz 2.12
L wird von einem DEA akzeptiert.
Satz 2.26
L ist die Vereinigung von Äquivalenzklassen einer rechtsinvarianten Äquivalenzrelation mit endlichem Index.

Definition: überflüssige Zustände

Zustände eines (deterministischen) endlichen Automaten, die vom Anfangszustand aus nicht erreichbar sind, heissen überflüssig.

Definition: überflüssige Zustände

Zustände eines (deterministischen) endlichen Automaten, die vom Anfangszustand aus nicht erreichbar sind, heissen überflüssig.



Definition:

Zwei Zustände p und q eines deterministischen endlichen Automaten heissen äquivalent $(p \equiv q)$, wenn für alle Wörter $w \in \Sigma^*$ gilt:

Definition:

Zwei Zustände p und q eines deterministischen endlichen Automaten heissen äquivalent $(p \equiv q)$, wenn für alle Wörter $w \in \Sigma^*$ gilt:

$$\delta(p, w) \in F \Leftrightarrow \delta(q, w) \in F.$$

Definition:

Zwei Zustände p und q eines deterministischen endlichen Automaten heissen äquivalent $(p \equiv q)$, wenn für alle Wörter $w \in \Sigma^*$ gilt:

$$\delta(p, w) \in F \Leftrightarrow \delta(q, w) \in F.$$

Offensichtlich ist $\equiv$ eine Äquivalenzrelation. Mit $[p]$ bezeichnen wir die Äquivalenzklasse der zu p äquivalenten Zustände.

Defiition: Äquivalenzklassenautomat

Zu einem DEA $A = (Q, \Sigma, \delta, s, F)$ definieren wir den Äquivalenzklassenautomaten $A^{\equiv} = (Q^{\equiv}, \Sigma^{\equiv}, \delta^{\equiv}, s^{\equiv}, F^{\equiv})$ durch:

Defiition: Äquivalenzklassenautomat

Zu einem DEA $A = (Q, \Sigma, \delta, s, F)$ definieren wir den Äquivalenzklassenautomaten $A^{\equiv} = (Q^{\equiv}, \Sigma^{\equiv}, \delta^{\equiv}, s^{\equiv}, F^{\equiv})$ durch:

$$Q^{\equiv} := \{[q] | q \in Q\}$$

Defiition: Äquivalenzklassenautomat

Zu einem DEA $A = (Q, \Sigma, \delta, s, F)$ definieren wir den Äquivalenzklassenautomaten $A^{\equiv} = (Q^{\equiv}, \Sigma^{\equiv}, \delta^{\equiv}, s^{\equiv}, F^{\equiv})$ durch:

$$Q^{\equiv} := \{[q] | q \in Q\}$$
$$\Sigma^{\equiv} := \Sigma$$

Defiition: Äquivalenzklassenautomat

Zu einem DEA $A = (Q, \Sigma, \delta, s, F)$ definieren wir den Äquivalenzklassenautomaten $A^{\equiv} = (Q^{\equiv}, \Sigma^{\equiv}, \delta^{\equiv}, s^{\equiv}, F^{\equiv})$ durch:

$$Q^{\equiv} := \{[q] | q \in Q\}$$
$$\Sigma^{\equiv} := \Sigma$$
$$\delta^{\equiv}([q], a) := [\delta(q, a)]$$

Defiition: Äquivalenzklassenautomat

Zu einem DEA $A = (Q, \Sigma, \delta, s, F)$ definieren wir den Äquivalenzklassenautomaten $A^{\equiv} = (Q^{\equiv}, \Sigma^{\equiv}, \delta^{\equiv}, s^{\equiv}, F^{\equiv})$ durch:

$$Q^{\equiv} := \{[q] | q \in Q\}$$
$$\Sigma^{\equiv} := \Sigma$$
$$\delta^{\equiv}([q], a) := [\delta(q, a)]$$
$$s^{\equiv} := [s]$$

Defiition: Äquivalenzklassenautomat

Zu einem DEA $A = (Q, \Sigma, \delta, s, F)$ definieren wir den Äquivalenzklassenautomaten $A^{\equiv} = (Q^{\equiv}, \Sigma^{\equiv}, \delta^{\equiv}, s^{\equiv}, F^{\equiv})$ durch:

$$Q^{\equiv} := \{[q] | q \in Q\}$$
$$\Sigma^{\equiv} := \Sigma$$
$$\delta^{\equiv}([q], a) := [\delta(q, a)]$$
$$s^{\equiv} := [s]$$
$$F^{\equiv} := \{[f] | f \in F\}$$

Satz

Der Äquivalenzklassenautomat $A^{\equiv}$ zu A akzeptiert dieselbe Sprache wie A.

Satz

Der Äquivalenzklassenautomat $A^{\equiv}$ zu A akzeptiert dieselbe Sprache wie A.

Satz

Der Äquivalenzklassenautomat $A^{\equiv}$ zu einem DEA A ohne überflüssige Zustände ist minimal.

Minimierungsalgorithmus:

1. Entferne alle überflüssigen Zustände.

Minimierungsalgorithmus:

1. Entferne alle überflüssigen Zustände.
2. Stelle eine Tabelle aller Zustandspaare $\{p, q\}$ mit $p \neq q$ auf.

Minimierungsalgorithmus:

1. Entferne alle überflüssigen Zustände.
2. Stelle eine Tabelle aller Zustandspaare $\{p, q\}$ mit $p \neq q$ auf.
3. Markiere alle Paare $\{p, q\}$ mit entweder $p \in F$ oder $q \in F$.

Minimierungsalgorithmus:

1. Entferne alle überflüssigen Zustände.
2. Stelle eine Tabelle aller Zustandspaare $\{p, q\}$ mit $p \neq q$ auf.
3. Markiere alle Paare $\{p, q\}$ mit entweder $p \in F$ oder $q \in F$.
4. Für jedes noch unmarkierte Paar $\{p, q\}$ und jedes $a \in \Sigma$ teste, ob $\{\delta(p, a), \delta(q, a)\}$ bereits markiert ist. Wenn ja, dann markiere auch $\{p, q\}$.

Minimierungsalgorithmus:

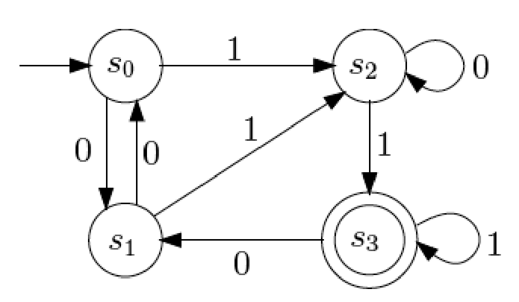
1. Entferne alle überflüssigen Zustände.
2. Stelle eine Tabelle aller Zustandspaare $\{p, q\}$ mit $p \neq q$ auf.
3. Markiere alle Paare $\{p, q\}$ mit entweder $p \in F$ oder $q \in F$.
4. Für jedes noch unmarkierte Paar $\{p, q\}$ und jedes $a \in \Sigma$ teste, ob $\{\delta(p, a), \delta(q, a)\}$ bereits markiert ist. Wenn ja, dann markiere auch $\{p, q\}$.
5. Wiederhole Schritt 4 bis sich keine Änderung in der Tabelle mehr ergibt.

Minimierungsalgorithmus:

1. Entferne alle überflüssigen Zustände.
2. Stelle eine Tabelle aller Zustandspaare $\{p, q\}$ mit $p \neq q$ auf.
3. Markiere alle Paare $\{p, q\}$ mit entweder $p \in F$ oder $q \in F$.
4. Für jedes noch unmarkierte Paar $\{p, q\}$ und jedes $a \in \Sigma$ teste, ob $\{\delta(p, a), \delta(q, a)\}$ bereits markiert ist. Wenn ja, dann markiere auch $\{p, q\}$.
5. Wiederhole Schritt 4 bis sich keine Änderung in der Tabelle mehr ergibt.
6. Verschmelze alle jetzt noch unmarkierten Paare zu jeweils einem Zustand.

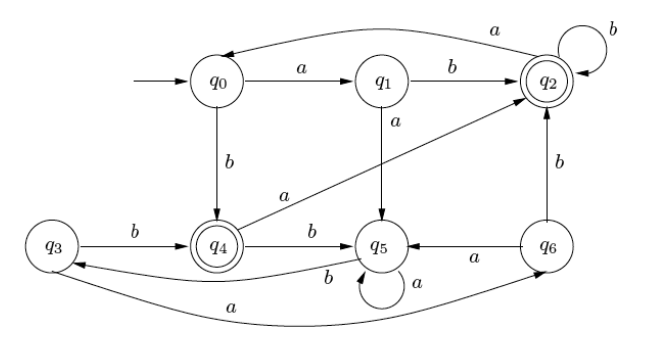
Aufgabe 1

Minimiere den folgenden Automaten mittels Konstruktion des Äquivalenzklassenautomatens. Gib den Minimalautomaten graphisch an.



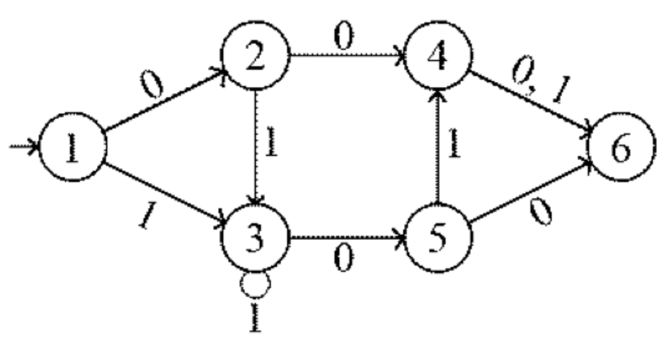
Aufgabe 2

Minimiere den folgenden Automaten mittels Konstruktion des Äquivalenzklassenautomatens. Gib den Minimalautomaten graphisch an.



Aufgabe 3

Minimiere den folgenden Automaten mittels Konstruktion des Äquivalenzklassenautomatens. Gib den Minimalautomaten graphisch an. $F = 6$



Satz: Pumping-Lemma

Sei L eine reguläre Sprache. Dann existiert eine Zahl $n \in \mathbb{N}$, sodass für jedes Wort $w \in L$ mit $|w| > n$ eine Darstellung

$$w = uvx \text{ mit } |uv| \leq n, v \neq \epsilon,$$

existiert, bei der auch $uv^i x \in L$ ist für alle $i \in \mathbb{N}_0$.

Satz: Pumping-Lemma

Sei L eine reguläre Sprache. Dann existiert eine Zahl $n \in \mathbb{N}$, sodass für jedes Wort $w \in L$ mit $|w| > n$ eine Darstellung

$$w = uvx \text{ mit } |uv| \leq n, v \neq \epsilon,$$

existiert, bei der auch $uv^ix \in L$ ist für alle $i \in \mathbb{N}_0$.

Hinweis:
Man verwendet das Pumping-Lemma um zu zeigen, dass eine Sprache nicht regulär ist.

Beispiel:
$L = \{a^n b^n | n > 0, n \in \mathbb{N}\}$

Um zu zeigen, dass L nicht regulär ist, verwenden wir das Pumping-Lemma.

Beispiel:
$L = \{a^n b^n | n > 0, n \in \mathbb{N}\}$

Um zu zeigen, dass L nicht regulär ist, verwenden wir das Pumping-Lemma.

Angenommen L sei regulär. Dann $\exists n$, sodass sich alle Wörter $w \in L$ der Länge $\geq n$ wie im Pumping Lemma beschrieben zerlegen lassen. Betrachten wir speziell das Wort $a^n b^n$ der Länge $2n$. Die entsprechende Zerlegung uvx dieses Wortes ist aufgrund $v \neq \epsilon$ so, dass v nicht leer ist, und aufgrund $|uv| \leq n$ kann v nur aus a's bestehen. Aufgrund von $uv^i x \in L$ wäre dann das Wort $ux = a^{n-|v|}b^n$ in der Sprache, was im Widerspruch zur Definition von L steht. Deshalb ist L nicht regulär.

Aufgabe 4

$L = \{0^m | m \text{ ist Quadratzahl}\}$ Zeige, dass L nicht regulär ist.

Aufgabe 4

$L = \{0^m | m \text{ ist Quadratzahl}\}$ Zeige, dass L nicht regulär ist.

Aufgabe 5

$L = \{0^p | p \text{ ist Primzahl}\}$ Zeige, dass L nicht regulär ist.

Aufgabe 6

$L = \{a^n b a^m b a^{n+m} | n, m \geq 1\}$ Zeige, dass L nicht regulär ist.

Aufgabe 6

$L = \{a^n b a^m b a^{n+m} | n, m \geq 1\}$ Zeige, dass L nicht regulär ist.

Aufgabe 7

Zeige für die Sprache $L = \{01^k 0^l 1 | k, l \geq 0\}$ explizit, dass die im Pumping-Lemma formulierte notwendige Bedingung für Regularität erfüllt ist. (Gib also eine konkrete Belegung für die existenzquantifizierten Ausdrücke an und begründe.)

Aufgabe 8

$L = \emptyset$

Ist für L das Pumping-Lemma anwendbar?

Aufgabe 8

$L = \emptyset$
Ist für L das Pumping-Lemma anwendbar?

Aufgabe 9

$L = \{00, 11\}$
Ist für L das Pumping-Lemma anwendbar?

Tipps zum aktuellen Übungsblatt

Aufgabe 5 Wie ist das Spiegelwort definiert?

Tipps zum aktuellen Übungsblatt

Aufgabe 5 Wie ist das Spiegelwort definiert?

Aufgabe 6 Sehr einfaches Schema. Unbedingt mit dem Schema arbeiten! Vermeintlich richtige Lösungen ohne Lösungsweg werden mit 0 Punkten bewertet.

Tipps zum aktuellen Übungsblatt

Aufgabe 5 Wie ist das Spiegelwort definiert?

Aufgabe 6 Sehr einfaches Schema. Unbedingt mit dem Schema arbeiten! Vermeintlich richtige Lösungen ohne Lösungsweg werden mit 0 Punkten bewertet.

Aufgabe 7 Haben wir heute ausführlich gemacht.

Tipps zum aktuellen Übungsblatt

Aufgabe 5 Wie ist das Spiegelwort definiert?

Aufgabe 6 Sehr einfaches Schema. Unbedingt mit dem Schema arbeiten! Vermeintlich richtige Lösungen ohne Lösungsweg werden mit 0 Punkten bewertet.

Aufgabe 7 Haben wir heute ausführlich gemacht.

Aufgabe 8 Hat diese Aufgabe vielleicht etwas mit dem aktuellen Thema zu tun?

Reflexion

Was haben wir heute gelernt?

Reflexion

Was haben wir heute gelernt?

- Automatenminimierung

Reflexion

Was haben wir heute gelernt?

- Automatenminimierung
- Pumping-Lemma

Noch Fragen?

Vorschau

Vorschau

- Nerode-Relation

Vorschau

- Nerode-Relation
- Turingmaschine

Bis zum nächsten Mal

Unshelved® by Bill Barnes and Gene Ambaum ©2007 Overdue Media LLC unshelved@overduemedia.com www.unshelved.com