TIL Tutorium

Themenbereiche des 1. und 2. Übungsblatts

Dominik Apel, Rupert Ettrich

E192 Institut of Logic and Computation

Table of contents

Entscheidbarkeit

Entscheidbarkeit - Rekursiv Aufzählbar

- Akzeptierte Sprache: Die von der deterministischen Turingmaschine M akzeptierte Sprache $L(M)$ besteht aus genau all jenen Wörtern, bei deren Anyalyse M einen Endzustand erreicht.

Entscheidbarkeit - Rekursiv Aufzählbar

- Akzeptierte Sprache: Die von der deterministischen Turingmaschine M akzeptierte Sprache $L(M)$ besteht aus genau all jenen Wörtern, bei deren Anyalyse M einen Endzustand erreicht.
- Rekursiv Aufzählbar: Eine Sprache $L(M)$ heißt rekursiv aufzählbar, wenn sie von einer deterministischen Turingmaschine M akzeptiert wird.

Entscheidbarkeit - Rekursiv Aufzählbar

- Akzeptierte Sprache: Die von der deterministischen Turingmaschine M akzeptierte Sprache $L(M)$ besteht aus genau all jenen Wörtern, bei deren Anyalyse M einen Endzustand erreicht.
- Rekursiv Aufzählbar: Eine Sprache $L(M)$ heißt rekursiv aufzählbar, wenn sie von einer deterministischen Turingmaschine M akzeptiert wird. Das bedeutet, dass die Turingmaschine M jedenfalls hält, wenn sie auf ein Wort w der Sprache $L(M)$ gestartet wird.

Entscheidbarkeit - Rekursiv Aufzählbar

- Akzeptierte Sprache: Die von der deterministischen Turingmaschine M akzeptierte Sprache $L(M)$ besteht aus genau all jenen Wörtern, bei deren Anyalyse M einen Endzustand erreicht.
- Rekursiv Aufzählbar: Eine Sprache $L(M)$ heißt rekursiv aufzählbar, wenn sie von einer deterministischen Turingmaschine M akzeptiert wird. Das bedeutet, dass die Turingmaschine M jedenfalls hält, wenn sie auf ein Wort w der Sprache $L(M)$ gestartet wird. Es bedeutet aber nicht notwendigerweise, dass die Turingmaschine hält, wenn sie auf ein Wort w', welches nicht in $L(M)$ liegt, gestartet wird

Entscheidbarkeit - Rekursive Sprachen

- Wenn eine Turingmaschine M immer hält, egal ob das Eingabewort Teil der Sprache $L(M)$ ist oder nicht, dann nennt man $L(M)$ rekursiv oder entscheidbar

Entscheidbarkeit - Rekursive Sprachen

- Wenn eine Turingmaschine M immer hält, egal ob das Eingabewort Teil der Sprache $L(M)$ ist oder nicht, dann nennt man $L(M)$ rekursiv oder entscheidbar
- Die Turingmaschine kann also immer entscheiden, ob ein Wort Teil der Sprache ist oder nicht.

Entscheidbarkeit - Rekursive Sprachen

- Wenn eine Turingmaschine M immer hält, egal ob das Eingabewort Teil der Sprache $L(M)$ ist oder nicht, dann nennt man $L(M)$ rekursiv oder entscheidbar
- Die Turingmaschine kann also immer entscheiden, ob ein Wort Teil der Sprache ist oder nicht.
- Zur Erinnerung: Eine Turingmaschine verwirft die Eingabe, wenn sie in einem Zustand q hält, welcher kein Endzustand ist.

Entscheidbarkeit - Beispiel 1

Sei $A \leq B$ und B rekursiv aufzählbar, so ist auch das Komplement von A rekursiv aufzählbar

Entscheidbarkeit - Beispiel 1

Sei $A \leq B$ und B rekursiv aufzählbar, so ist auch das Komplement von A rekursiv aufzählbar

Falsch

Entscheidbarkeit - Beispiel 1

Sei $A \leq B$ und B rekursiv aufzählbar, so ist auch das Komplement von A rekursiv aufzählbar

Falsch

Aus der Reduktion folgt, dass A rekursiv aufzählbar ist. Das Komplement einer rekursiv aufzählbaren Sprache ist nicht notwendigerweise rekursiv aufzählbar.

Entscheidbarkeit - Beispiel 2

Ist eine Sprache L entscheidbar, so ist auch jede Teilmenge von L zumindest rekursiv aufzählbar.

Entscheidbarkeit - Beispiel 2

Ist eine Sprache L entscheidbar, so ist auch jede Teilmenge von L zumindest rekursiv aufzählbar.

Falsch

Entscheidbarkeit - Beispiel 2

Ist eine Sprache L entscheidbar, so ist auch jede Teilmenge von L zumindest rekursiv aufzählbar.

Falsch

Sei $L = \{\underline{0}, \underline{1}\}^*$. L ist entscheidbar. Das Halteproblem L_u ist eine Teilmenge von L, genauso wie das Inverse des Halteproblems, $\overline{L_u}$. Wir wissen, dass das Halteproblem rekursiv aufzählbar ist, aber nicht rekursiv ist. Daher kann das Inverse des Halteproblems nicht rekursiv aufzählbar sein.

Satz von Rice

Satz von Rice - Definition

Jede nicht triviale Eigenschaft der rekursiv aufzählbaren Sprachen ist unentscheidbar.

Satz von Rice - Eigenschaft

- Eine Eigenschaft P beschreibt eine Teilmenge der rekursiv aufzählbaren Sprachen.
- Liegt eine Sprache L in P, so ist P eine Eigenschaft von L.
- P ist eine triviale Eigenschaft, wenn sie auf alle oder keine rekursiv aufzählbare Sprache zutrifft.

Satz von Rice - Beispiel 1

Wird die von einer Turingmaschine akzeptierten Sprache L auch von einer Turingmaschine mit Kellerautomatenbedingung akzeptiert?

Satz von Rice - Beispiel 1

Wird die von einer Turingmaschine akzeptierten Sprache L auch von einer Turingmaschine mit Kellerautomatenbedingung akzeptiert?

Nicht entscheidbar

Satz von Rice - Beispiel 1

Wird die von einer Turingmaschine akzeptierten Sprache L auch von einer Turingmaschine mit Kellerautomatenbedingung akzeptiert?

Nicht entscheidbar

Vorgriff auf Kapitel Grammatiken: Eine allgemeine Turingmaschine ist aussagekräftiger als ein Kellerautomat. Das heißt es gibt rekursiv aufzählbare Sprachen, die nicht durch einen Kellerautomat dargestellt werden können. Daher ist der Satz von Rice anwendbar:

Satz von Rice - Beispiel 1

Wird die von einer Turingmaschine akzeptierten Sprache L auch von einer Turingmaschine mit Kellerautomatenbedingung akzeptiert?

Nicht entscheidbar

Vorgriff auf Kapitel Grammatiken: Eine allgemeine Turingmaschine ist aussagekräftiger als ein Kellerautomat. Das heißt es gibt rekursiv aufzählbare Sprachen, die nicht durch einen Kellerautomat dargestellt werden können. Daher ist der Satz von Rice anwendbar:

- Beispiel: $L_1 = \{\underline{0}, \underline{1}\}^*$. Es gibt einen Kellerautomaten, der diese rekursiv aufzählbare Sprache akzeptiert.

Satz von Rice - Beispiel 1

Wird die von einer Turingmaschine akzeptierten Sprache L auch von einer Turingmaschine mit Kellerautomatenbedingung akzeptiert?

Nicht entscheidbar

Vorgriff auf Kapitel Grammatiken: Eine allgemeine Turingmaschine ist aussagekräftiger als ein Kellerautomat. Das heißt es gibt rekursiv aufzählbare Sprachen, die nicht durch einen Kellerautomat dargestellt werden können. Daher ist der Satz von Rice anwendbar:

- Beispiel: $L_1 = \{\underline{0}, \underline{1}\}^*$. Es gibt einen Kellerautomaten, der diese rekursiv aufzählbare Sprache akzeptiert.
- Gegenbeispiel: $L_2 = \{\underline{0}^n \underline{1}^n \underline{0}^n \mid n \in \mathbb{N}\}$. Diese Sprache ist nicht kontextfrei, es kann also kein Kellerautomat angegeben werden, welcher diese Sprache akzeptiert.

Einschub - Übungsbeispiel 1.1

Geben sie eine Turingmaschine nach der Definition von Folie 72 an, welche die Kellerautomatenbedingung erfüllt und die Sprache $L = \{w \in \{\underline{0}, \underline{1}\}^* \mid |w|_{\underline{0}} = |w|_{\underline{1}}\}$ akzeptiert!

Einschub - Übungsbeispiel 1.1

Geben sie eine Turingmaschine nach der Definition von Folie 72 an, welche die Kellerautomatenbedingung erfüllt und die Sprache $L = \{w \in \{\underline{0}, \underline{1}\}^* \mid |w|_{\underline{0}} = |w|_{\underline{1}}\}$ akzeptiert!

Wiederholung: Diese Turingmaschine soll zwei Bänder haben, ein Arbeitsband sowie ein Eingabeband. Das Eingabewort darf nur von links nach rechts gelesen werden. Das Arbeitsband muss dem LIFO-Prinzip gehorchen.

Einschub - Übungsbeispiel 1.1

Geben sie eine Turingmaschine nach der Definition von Folie 72 an, welche die Kellerautomatenbedingung erfüllt und die Sprache $L = \{w \in \{\underline{0}, \underline{1}\}^* \mid |w|_{\underline{0}} = |w|_{\underline{1}}\}$ akzeptiert!

Wiederholung: Diese Turingmaschine soll zwei Bänder haben, ein Arbeitsband sowie ein Eingabeband. Das Eingabewort darf nur von links nach rechts gelesen werden. Das Arbeitsband muss dem LIFO-Prinzip gehorchen.

Idee: Zwei Zustände:

Einschub - Übungsbeispiel 1.1

Geben sie eine Turingmaschine nach der Definition von Folie 72 an, welche die Kellerautomatenbedingung erfüllt und die Sprache $L = \{w \in \{\underline{0}, \underline{1}\}^* \mid |w|_{\underline{0}} = |w|_{\underline{1}}\}$ akzeptiert!

Wiederholung: Diese Turingmaschine soll zwei Bänder haben, ein Arbeitsband sowie ein Eingabeband. Das Eingabewort darf nur von links nach rechts gelesen werden. Das Arbeitsband muss dem LIFO-Prinzip gehorchen.

Idee: Zwei Zustände:

- Wenn das erste gelesene Zeichen $\underline{0}$ ist, dann schreibe jedes gelesene Zeichen $\underline{0}$ auf das Arbeitsband, und lösche für jedes gelesene Zeichen $\underline{1}$ eine $\underline{0}$ weg.

Einschub - Übungsbeispiel 1.1

Geben sie eine Turingmaschine nach der Definition von Folie 72 an, welche die Kellerautomatenbedingung erfüllt und die Sprache $L = \{w \in \{\underline{0}, \underline{1}\}^* \mid |w|_{\underline{0}} = |w|_{\underline{1}}\}$ akzeptiert!

Wiederholung: Diese Turingmaschine soll zwei Bänder haben, ein Arbeitsband sowie ein Eingabeband. Das Eingabewort darf nur von links nach rechts gelesen werden. Das Arbeitsband muss dem LIFO-Prinzip gehorchen.

Idee: Zwei Zustände:

- Wenn das erste gelesene Zeichen $\underline{0}$ ist, dann schreibe jedes gelesene Zeichen $\underline{0}$ auf das Arbeitsband, und lösche für jedes gelesene Zeichen $\underline{1}$ eine $\underline{0}$ weg.
- Wenn das erste gelesene Zeichen $\underline{1}$ ist, dann schreibe jedes gelesene Zeichen $\underline{1}$ auf das Arbeitsband, und lösche für jedes gelesene Zeichen $\underline{0}$ eine $\underline{1}$ weg.

Einschub - Übungsbeispiel 1.1

Geben sie eine Turingmaschine nach der Definition von Folie 72 an, welche die Kellerautomatenbedingung erfüllt und die Sprache $L = \{w \in \{\underline{0}, \underline{1}\}^* \mid |w|_{\underline{0}} = |w|_{\underline{1}}\}$ akzeptiert!

Wenn:

- Arbeitsband leer und der Lesekopf am Ende des Eingabewortes: **Akzeptieren**. Es wurden gleich viele $\underline{0}$ wie $\underline{1}$ gelesen

Einschub - Übungsbeispiel 1.1

Geben sie eine Turingmaschine nach der Definition von Folie 72 an, welche die Kellerautomatenbedingung erfüllt und die Sprache $L = \{w \in \{\underline{0}, \underline{1}\}^* \mid |w|_{\underline{0}} = |w|_{\underline{1}}\}$ akzeptiert!

Wenn:

- Arbeitsband leer und der Lesekopf am Ende des Eingabewortes: **Akzeptieren**. Es wurden gleich viele $\underline{0}$ wie $\underline{1}$ gelesen
- Arbeitsband nicht leer und Lesekopf am Ende des Eingabewortes: **Ablehnen**. Es wurden nicht gleich viele $\underline{0}$ wie $\underline{1}$ gelesen

Einschub - Übungsbeispiel 1.1

Geben sie eine Turingmaschine nach der Definition von Folie 72 an, welche die Kellerautomatenbedingung erfüllt und die Sprache $L = \{w \in \{\underline{0}, \underline{1}\}^* \mid |w|_{\underline{0}} = |w|_{\underline{1}}\}$ akzeptiert!

Wenn:

- Arbeitsband leer und der Lesekopf am Ende des Eingabewortes: **Akzeptieren**. Es wurden gleich viele $\underline{0}$ wie $\underline{1}$ gelesen
- Arbeitsband nicht leer und Lesekopf am Ende des Eingabewortes: **Ablehnen**. Es wurden nicht gleich viele $\underline{0}$ wie $\underline{1}$ gelesen
- Arbeitsband leer und Lesekopf nicht am Ende des Eingabewortes: **Beginne von vorn**.

Satz von Rice - Beispiel 2

Gilt für die von einer Turingmaschine akzeptierte Sprache L, dass $\overline{L} = \Sigma^*$?

Satz von Rice - Beispiel 2

Gilt für die von einer Turingmaschine akzeptierte Sprache L, dass $\overline{L} = \Sigma^*$?

Nicht entscheidbar

Satz von Rice - Beispiel 2

Gilt für die von einer Turingmaschine akzeptierte Sprache L, dass $\overline{L} = \Sigma^*$?

Nicht entscheidbar

Die Eigenschaft ist $P = \{L \mid \overline{L} = \Sigma^*\}$. Das Komplement der Allsprache ist in jedem Fall die Leersprache, unabhängig vom Alphabet. Insofern kommt diese Eigenschaft nur einer Sprache $L = \{\}$ zu. Keine andere rekursiv aufzählbare Sprache ist in P, dementsprechend ist P nicht trivial und damit nach dem Satz von Rice unentscheidbar.

Pumping Lemma für reguläre Sprachen

Pumping Lemma - Intuition

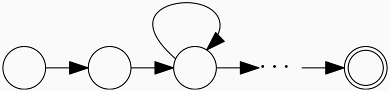
Sei L eine unendliche reguläre Sprache. Wie wir wissen gibt es für jede reguläre Sprache einen deterministischen endlichen Automaten (DEA), der diese Sprache akzeptiert.

Pumping Lemma - Intuition

Sei L eine unendliche reguläre Sprache. Wie wir wissen gibt es für jede reguläre Sprache einen deterministischen endlichen Automaten (DEA), der diese Sprache akzeptiert. Da L unendlich ist, muss es für jede beliebige Schranke $n \in \mathbb{N}$ ein Wort in L geben, das mindestens n Symbole hat. Ein Automat hat aber nur eine endliche Anzahl an Zuständen. Daher muss es Wörter in L geben, die einen Zustand q im Automaten erreichen, indem es einen "loop" gibt.

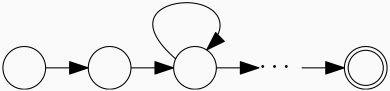
Pumping Lemma - Intuition

Sei L eine unendliche reguläre Sprache. Wie wir wissen gibt es für jede reguläre Sprache einen deterministischen endlichen Automaten (DEA), der diese Sprache akzeptiert. Da L unendlich ist, muss es für jede beliebige Schranke $n \in \mathbb{N}$ ein Wort in L geben, das mindestens n Symbole hat. Ein Automat hat aber nur eine endliche Anzahl an Zuständen. Daher muss es Wörter in L geben, die einen Zustand q im Automaten erreichen, indem es einen "loop" gibt.



Pumping Lemma - Intuition

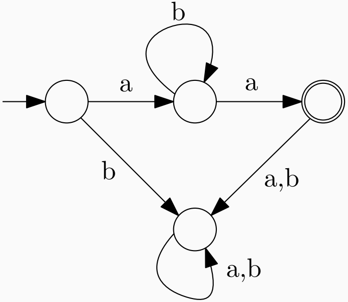
Sei L eine unendliche reguläre Sprache. Wie wir wissen gibt es für jede reguläre Sprache einen deterministischen endlichen Automaten (DEA), der diese Sprache akzeptiert. Da L unendlich ist, muss es für jede beliebige Schranke $n \in \mathbb{N}$ ein Wort in L geben, das mindestens n Symbole hat. Ein Automat hat aber nur eine endliche Anzahl an Zuständen. Daher muss es Wörter in L geben, die einen Zustand q im Automaten erreichen, indem es einen "loop" gibt. Für den DEA macht es keinen Unterschied, ob in diesem Zustand q 1, 10, 100, 1000, Mal "geloopt" wird. Wenn ein Wort, das in L ist den Zustand q erreicht, so müssen auch alle Wörter in L sein, die in diesem Zustand 1, 10, 100, 1000 Mal "loopen".



Pumping Lemma - Intuition

Der folgende Automat hat 4 Zustände - Wörter die deutlich mehr als 4 Zeichen haben müssen also irgendwo "loopen".

$$L = \{ab^n a \mid n \geq 0\} = \{aa, aba, abba, abbba, \ldots\}$$



Pumping Lemma - Definition:

Pumping Lemma für reguläre Sprachen:

Sei L eine unendliche reguläre Sprache. Dann gibt es eine (nur von L abhängige) Schranke $m > 0$ so, dass für jedes Wort $w \in L$ mit $|w| \geq m$ Wörter x, y, z so existieren, dass

$$w = xyz \qquad \text{mit } |xy| \leq m \text{ und } |y| > 0$$

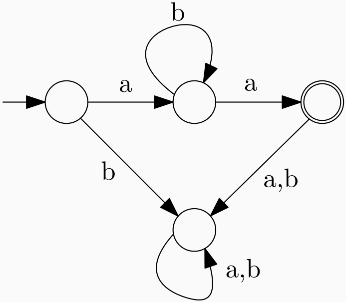
sowie

$$w_i = xy^i z \in L \qquad \text{für alle } i \geq 0.$$

Pumping Lemma - Intuition

Für jedes Wort w das mehr als $m = 3$ Symbole hat, gilt:

$$L = \{ab^n a \mid n \geq 0\} = \{aa, aba, abba, abbba, \ldots\}$$

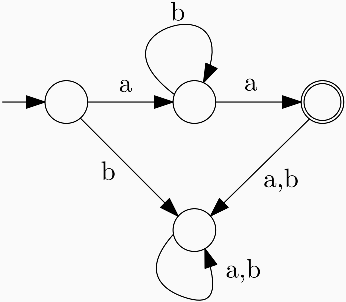


Pumping Lemma - Intuition

Für jedes Wort w das mehr als $m = 3$ Symbole hat, gilt:

- Wir können es in $w = xyz$ aufteilen, so dass $|xy| \leq m$ und $|y| > 0$

$$L = \{ab^n a \mid n \geq 0\} = \{aa, aba, abba, abbba, \ldots\}$$

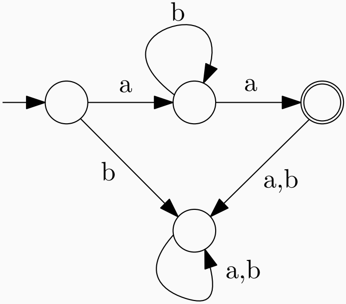


Pumping Lemma - Intuition

Für jedes Wort w das mehr als $m = 3$ Symbole hat, gilt:

- Wir können es in $w = xyz$ aufteilen, so dass $|xy| \leq m$ und $|y| > 0$
- Unter den ersten $m = 3$ Symbolen jedes Wortes in L ist immer ein b: daher können wir xy so wählen, dass y nur Symbole b enthält

$$L = \{ab^n a \mid n \geq 0\} = \{aa, aba, abba, abbba, \ldots\}$$

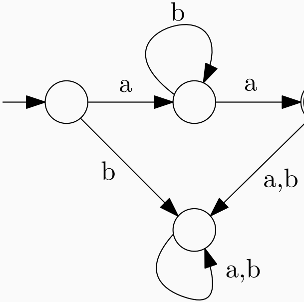


Pumping Lemma - Intuition

Für jedes Wort w das mehr als $m = 3$ Symbole hat, gilt:

- Wir können es in $w = xyz$ aufteilen, so dass $|xy| \leq m$ und $|y| > 0$
- Unter den ersten $m = 3$ Symbolen jedes Wortes in L ist immer ein b: daher können wir xy so wählen, dass y nur Symbole b enthält
- Wir können also y beliebig "aufpumpen", und das Wort $w_i = xy^i z$ wird weiterhin in L sein für alle $i \geq 0$

$$L = \{ab^n a \mid n \geq 0\} = \{aa, aba, abba, abbba, \ldots\}$$



Pumping Lemma - Anwendung

Das Pumping Lemma wird aber dafür genutzt, zu zeigen, dass eine Sprache nicht regulär ist. Das geschieht durch einen Beweis durch Widerspruch: Wir nehmen an, dass eine Sprache regulär ist (obwohl wir davon ausgehen, dass sie es nicht ist), und zeigen dass, egal wie wir das Wort $w = xyz$ einteilen (sodass $|w| \geq m$, $|xy| \leq m, |y| > 0$), das Wort $w_i = xy^iz$ wird nicht für alle $i \geq 0$ in L sein.

Pumping Lemma - Beispiel 1

Gegeben ist die Sprache $L = \{\underline{a}^i \underline{b}^j \mid i, j \geq 0, i - j = 5\}$.

Pumping Lemma - Beispiel 1

Gegeben ist die Sprache $L = \{\underline{a}^i \underline{b}^j \mid i, j \geq 0, i - j = 5\}$.

Ist die Sprache unendlich?

Pumping Lemma - Beispiel 1

Gegeben ist die Sprache $L = \{\underline{a}^i \underline{b}^j \mid i, j \geq 0, i - j = 5\}$.

Ist die Sprache unendlich? Ja, es gibt unendlich viele Lösungen für die Gleichung $i - j = 5$ für $i, j \geq 0$

Pumping Lemma - Beispiel 1

Gegeben ist die Sprache $L = \{\underline{a}^i \underline{b}^j \mid i, j \geq 0, i - j = 5\}$.

Ist die Sprache unendlich? Ja, es gibt unendlich viele Lösungen für die Gleichung $i - j = 5$ für $i, j \geq 0$

$\underline{aaaaa}, \underline{aaaaaab}, \underline{aaaaaaabb}, \underline{aaaaaaaabbb}, \underline{aaaaaaaaabbbb} \cdots \in L$

Pumping Lemma - Beispiel 1

$L = \{\underline{a}^i \underline{b}^j \mid i, j \geq 0, i - j = 5\}$ ist **nicht regulär.**

Pumping Lemma - Beispiel 1

$L = \{\underline{a}^i \underline{b}^j \mid i, j \geq 0, i - j = 5\}$ ist **nicht regulär.**

Beweis: Angenommen, L ist regulär. Für beliebiges m (Konstante aus dem Pumping Lemma) wähle z.B. das Wort:

$$w = \underline{a}^{m+5} \underline{b}^m.$$

Pumping Lemma - Beispiel 1

$L = \{\underline{a}^i \underline{b}^j \mid i, j \geq 0, i - j = 5\}$ ist **nicht regulär.**

Beweis: Angenommen, L ist regulär. Für beliebiges m (Konstante aus dem Pumping Lemma) wähle z.B. das Wort:

$$w = \underline{a}^{m+5} \underline{b}^m.$$

Dann gilt $w \in L$ und $|w| = 2m + 5 > m$.

Pumping Lemma - Beispiel 1

$L = \{\underline{a}^i \underline{b}^j \mid i, j \geq 0, i - j = 5\}$ ist **nicht regulär.**

Beweis: Angenommen, L ist regulär. Für beliebiges m (Konstante aus dem Pumping Lemma) wähle z.B. das Wort:

$$w = \underline{a}^{m+5} \underline{b}^m.$$

Dann gilt $w \in L$ und $|w| = 2m + 5 > m$. Wir teilen nun w in xyz so auf, dass $|xy| \leq m$ und $|y| > 0$. Somit kann y nur aus Symbolen $\underline{a}$ bestehen.

Pumping Lemma - Beispiel 1

$L = \{\underline{a}^i \underline{b}^j \mid i, j \geq 0, i - j = 5\}$ ist **nicht regulär.**

Beweis: Angenommen, L ist regulär. Für beliebiges m (Konstante aus dem Pumping Lemma) wähle z.B. das Wort:

$$w = \underline{a}^{m+5}\underline{b}^m.$$

Dann gilt $w \in L$ und $|w| = 2m + 5 > m$. Wir teilen nun w in xyz so auf, dass $|xy| \leq m$ und $|y| > 0$. Somit kann y nur aus Symbolen $\underline{a}$ bestehen. Nach dem Pumping Lemma muss $xy^iz \in L$ für alle $i \geq 0$ gelten.

Pumping Lemma - Beispiel 1

$L = \{\underline{a}^i \underline{b}^j \mid i, j \geq 0, i - j = 5\}$ ist **nicht regulär.**

Beweis: Angenommen, L ist regulär. Für beliebiges m (Konstante aus dem Pumping Lemma) wähle z.B. das Wort:

$$w = \underline{a}^{m+5}\underline{b}^m.$$

Dann gilt $w \in L$ und $|w| = 2m + 5 > m$. Wir teilen nun w in xyz so auf, dass $|xy| \leq m$ und $|y| > 0$. Somit kann y nur aus Symbolen $\underline{a}$ bestehen. Nach dem Pumping Lemma muss $xy^iz \in L$ für alle $i \geq 0$ gelten. Wählen wir aber z.B. $i = 0$, bekommen wir ein Wort, bei dem nicht mehr genau 5 Symbole $\underline{a}$ mehr als Symbole $\underline{b}$ vorkommen

Pumping Lemma - Beispiel 2

Gegeben ist die Sprache $L = \{\underline{a}^i \underline{b}^j \mid i, j \geq 0, i + j = 5\}$.

Pumping Lemma - Beispiel 2

Gegeben ist die Sprache $L = \{\underline{a}^i \underline{b}^j \mid i, j \geq 0, i + j = 5\}$.

Ist die Sprache unendlich?

Pumping Lemma - Beispiel 2

Gegeben ist die Sprache $L = \{\underline{a}^i \underline{b}^j \mid i, j \geq 0, i + j = 5\}$.

Ist die Sprache unendlich? Nein, es gibt endlich viele Lösungen für die Gleichung $i + j = 5$ für $i, j \geq 0$. Das Pumping Lemma kann daher nicht angewandt werden

Pumping Lemma - Beispiel 2

Gegeben ist die Sprache $L = \{\underline{a}^i \underline{b}^j \mid i, j \geq 0, i + j = 5\}$.

Ist die Sprache unendlich? Nein, es gibt endlich viele Lösungen für die Gleichung $i + j = 5$ für $i, j \geq 0$. Das Pumping Lemma kann daher nicht angewandt werden

i	**j**
0	5
1	4
2	3
3	2
4	1
5	0

Pumping Lemma - Beispiel 2

Gegeben ist die Sprache $L = \{\underline{a}^i \underline{b}^j \mid i, j \geq 0, i + j = 5\}$.

Ist die Sprache unendlich? Nein, es gibt endlich viele Lösungen für die Gleichung $i + j = 5$ für $i, j \geq 0$. Das Pumping Lemma kann daher nicht angewandt werden

Jede endliche Sprache ist regulär.

Pumping Lemma - Beispiel 2

Gegeben ist die Sprache $L = \{\underline{a}^i \underline{b}^j \mid i, j \geq 0, i + j = 5\}$.

Ist die Sprache unendlich? Nein, es gibt endlich viele Lösungen für die Gleichung $i + j = 5$ für $i, j \geq 0$. Das Pumping Lemma kann daher nicht angewandt werden

Jede endliche Sprache ist regulär. Warum? Für eine endliche Menge an Wörtern kann immer ein endlicher Automat konstruiert werden, der die so erstellte Sprache akzeptiert.

Pumping Lemma - Beispiel 3

Ist die Sprache $L = \{uvu^r \mid u, v \in \{\underline{a}, \underline{b}, \underline{c}\}^*\}$ regulär oder nicht?

Pumping Lemma - Beispiel 3

Ist die Sprache $L = \{uvu^r \mid u, v \in \{\underline{a}, \underline{b}, \underline{c}\}^*\}$ regulär oder nicht?

Behauptung: Diese Sprache ist nicht regulär

Pumping Lemma - Beispiel 3

Ist die Sprache $L = \{uvu^r \mid u, v \in \{\underline{a}, \underline{b}, \underline{c}\}^*\}$ regulär oder nicht?

Behauptung: Diese Sprache ist nicht regulär

Beweis: Angenommen, L sei regulär. Für beliebiges m (Konstante aus dem Pumping Lemma) wähle z.B. das Wort:

$$w = \underline{a}^m \underline{ba}^m$$

Pumping Lemma - Beispiel 3

Ist die Sprache $L = \{uvu^r \mid u, v \in \{\underline{a}, \underline{b}, \underline{c}\}^*\}$ regulär oder nicht?

Behauptung: Diese Sprache ist nicht regulär

Beweis: Angenommen, L sei regulär. Für beliebiges m (Konstante aus dem Pumping Lemma) wähle z.B. das Wort:

$$w = \underline{a}^m \underline{ba}^m$$

Dann gilt $w \in L$ und $|w| = 2m + 1 > m$.

Pumping Lemma - Beispiel 3

Ist die Sprache $L = \{uvu^r \mid u, v \in \{\underline{a}, \underline{b}, \underline{c}\}^*\}$ regulär oder nicht?

Behauptung: Diese Sprache ist nicht regulär

Beweis: Angenommen, L sei regulär. Für beliebiges m (Konstante aus dem Pumping Lemma) wähle z.B. das Wort:

$$w = \underline{a}^m \underline{ba}^m$$

Dann gilt $w \in L$ und $|w| = 2m + 1 > m$. Wir teilen nun w in xyz so auf, dass $|xy| \leq m$ und $|y| > 0$. Somit kann y nur aus Symbolen $\underline{a}$ bestehen.

Pumping Lemma - Beispiel 3

Ist die Sprache $L = \{uvu^r \mid u, v \in \{\underline{a}, \underline{b}, \underline{c}\}^*\}$ regulär oder nicht?
Behauptung: Diese Sprache ist nicht regulär
Beweis: Angenommen, L sei regulär. Für beliebiges m (Konstante aus dem Pumping Lemma) wähle z.B. das Wort:

$$w = \underline{a}^m \underline{ba}^m$$

Dann gilt $w \in L$ und $|w| = 2m + 1 > m$. Wir teilen nun w in xyz so auf, dass $|xy| \leq m$ und $|y| > 0$. Somit kann y nur aus Symbolen $\underline{a}$ bestehen. Nach dem Pumping Lemma muss $xy^iz \in L$ für alle $i \geq 0$ gelten.

Pumping Lemma - Beispiel 3

Ist die Sprache $L = \{uvu^r \mid u, v \in \{\underline{a}, \underline{b}, \underline{c}\}^*\}$ regulär oder nicht?

Behauptung: Diese Sprache ist nicht regulär

Beweis: Angenommen, L sei regulär. Für beliebiges m (Konstante aus dem Pumping Lemma) wähle z.B. das Wort:

$$w = \underline{a}^m \underline{ba}^m$$

Dann gilt $w \in L$ und $|w| = 2m + 1 > m$. Wir teilen nun w in xyz so auf, dass $|xy| \leq m$ und $|y| > 0$. Somit kann y nur aus Symbolen $\underline{a}$ bestehen.
Nach dem Pumping Lemma muss $xy^iz \in L$ für alle $i \geq 0$ gelten.
Wählen wir nun z.B. $i = 2$, bekommen wir das Wort $w_2 = \underline{a}^{m+|y|}\underline{ba}^m$.

Pumping Lemma - Beispiel 3

Ist die Sprache $L = \{uvu^r \mid u, v \in \{\underline{a}, \underline{b}, \underline{c}\}^*\}$ regulär oder nicht?
Behauptung: Diese Sprache ist nicht regulär
Beweis: Angenommen, L sei regulär. Für beliebiges m (Konstante aus dem Pumping Lemma) wähle z.B. das Wort:

$$w = \underline{a}^m \underline{ba}^m$$

Dann gilt $w \in L$ und $|w| = 2m + 1 > m$. Wir teilen nun w in xyz so auf, dass $|xy| \leq m$ und $|y| > 0$. Somit kann y nur aus Symbolen $\underline{a}$ bestehen.
Nach dem Pumping Lemma muss $xy^iz \in L$ für alle $i \geq 0$ gelten.
Wählen wir nun z.B. $i = 2$, bekommen wir das Wort $w_2 = \underline{a}^{m+|y|}\underline{ba}^m$.
Aber: Dieses Wort liegt nach wie vor in der Sprache, es gilt $w_2 \in L$ weil $w_2 = \underline{a}^m \underline{a}^{|y|} \underline{ba}^m$. Wie man sich leicht überlegen kann, liegt das Wort xy^iz für alle $i \geq 0$ in L.

Pumping Lemma - Beispiel 3

Ist die Sprache $L = \{uvu^r \mid u, v \in \{\underline{a}, \underline{b}, \underline{c}\}^*\}$ regulär oder nicht?
Behauptung: Diese Sprache ist nicht regulär
Beweis: Angenommen, L sei regulär. Für beliebiges m (Konstante aus dem Pumping Lemma) wähle z.B. das Wort:

$$w = \underline{a}^m \underline{ba}^m$$

Dann gilt $w \in L$ und $|w| = 2m + 1 > m$. Wir teilen nun w in xyz so auf, dass $|xy| \leq m$ und $|y| > 0$. Somit kann y nur aus Symbolen $\underline{a}$ bestehen.
Nach dem Pumping Lemma muss $xy^iz \in L$ für alle $i \geq 0$ gelten.
Wählen wir nun z.B. $i = 2$, bekommen wir das Wort $w_2 = \underline{a}^{m+|y|}\underline{ba}^m$.
Aber: Dieses Wort liegt nach wie vor in der Sprache, es gilt $w_2 \in L$ weil $w_2 = \underline{a}^m\underline{a}^{|y|}\underline{ba}^m$. Wie man sich leicht überlegen kann, liegt das Wort xy^iz für alle $i \geq 0$ in L.
Achtung: Der Schluss, dass deshalb die Sprache regulär sein muss, ist nicht zulässig - mit dem Pumping Lemma lassen sich nur Widerspruchsbeweise führen.

Pumping Lemma - Beispiel 3

Ist die Sprache $L = \{uvu^r \mid u, v \in \{\underline{a}, \underline{b}, \underline{c}\}^*\}$ regulär oder nicht?

Pumping Lemma - Beispiel 3

Ist die Sprache $L = \{uvu^r \mid u, v \in \{\underline{a}, \underline{b}, \underline{c}\}^*\}$ regulär oder nicht?

Behauptung: Diese Sprache ist regulär

Pumping Lemma - Beispiel 3

Ist die Sprache $L = \{uvu^r \mid u, v \in \{\underline{a}, \underline{b}, \underline{c}\}^*\}$ regulär oder nicht?

Behauptung: Diese Sprache ist regulär

Beweis: Wählt man u gleich ϵ, so bleibt nur mehr $\{v \mid v \in \{\underline{a}, \underline{b}, c\}^*$ übrig.

Pumping Lemma - Beispiel 3

Ist die Sprache $L = \{uvu^r \mid u, v \in \{\underline{a}, \underline{b}, \underline{c}\}^*\}$ regulär oder nicht?

Behauptung: Diese Sprache ist regulär

Beweis: Wählt man u gleich ϵ, so bleibt nur mehr $\{v \mid v \in \{\underline{a}, \underline{b}, c\}^*$ übrig. Es gilt also $L = \{\underline{a}, \underline{b}, \underline{c}\}^*$ und damit ist L regulär.

Grammatiken

Grammatik - Definition

Eine Grammatik $G = \langle N, T, P, S \rangle$ ist ein 4-Tupel, wobei:

$N \dots$ endliche Menge von Nonterminalen
$T \dots$ endliche Menge von Terminalsymbolen ($N \cap T = \{\}$)
$P \subseteq (N \cup T)^+ \times (N \cup T)^* \dots$ Produktionen
$S \in N \dots$ Startsymbol

Notation:
$\alpha \to \beta$ statt $(\alpha, \beta) \in P$
$\alpha \to \beta_1 \mid \cdots \mid \beta_n$ statt $\alpha \to \beta_1, \dots, \alpha \to \beta_n$

Grammatiken: Beispiele

$L = \{a^n \mid n \geq 0\}$

Grammatiken: Beispiele

$L = \{a^n \mid n \geq 0\}$

$G = (\{S\}, \{a\}, \{S \to \epsilon, S \to aS\}, S)$, $\mathcal{L}(G) = L$

Grammatiken: Beispiele

$L = \{a^n \mid n \geq 0\}$

$G = (\{S\}, \{a\}, \{S \to \epsilon, S \to aS\}, S)$, $\mathcal{L}(G) = L$

Ableitung für das Wort *aaa*:
$S \implies aS \implies aaS \implies aaaS \implies aaa$.

Grammatiken: Reguläre Grammatiken

Eine Grammatik heißt regulär, wenn alle Produktionen von der Form

$$A \rightarrow aB \qquad\qquad \text{oder} \qquad\qquad A \rightarrow \epsilon$$

sind ($A, B \in N, a \in T$). Die Sprache, die von einer regulären Grammatik erzeugt wird, ist regulär. Umgekehrt gibt es zu jeder reguären Sprache eine reguläre Grammatik, die sie erzeugt.

Grammatiken: Kontextfreie Grammatiken

Eine Grammatik heißt kontextfrei, wenn alle Produktionen von der Form

$$A \to \beta$$

sind ($A \in N, \beta \in (N \cup T)^*$).

In einer kontextfreien Grammatik haben alle Produktionen auf der "linken Seite" nur ein einzelnes Nonterminalsymbol.

Die Sprache, die von einer kontextfreien Grammatik erzeugt wird ist kontextfrei. Umgekehrt gibt es zu jeder kontextfreien Sprache eine kontextfreie Grammatik, die sie erzeugt.

Grammatiken: Beispiel 1

Gegeben ist $L = \{\underline{a}^{2n}\underline{b}^{3k-2}\underline{a}^{2n} \mid n, k \geq 1\}$

Grammatiken: Beispiel 1

Gegeben ist $L = \{\underline{a}^{2n}\underline{b}^{3k-2}\underline{a}^{2n} \mid n, k \geq 1\}$

Gesucht: Eine kontextfreie Grammatik G, sodass $\mathcal{L}(G) = L$

Grammatiken: Beispiel 1

Gegeben ist $L = \{\underline{a}^{2n}\underline{b}^{3k-2}\underline{a}^{2n} \mid n, k \geq 1\}$

Gesucht: Eine kontextfreie Grammatik G, sodass $\mathcal{L}(G) = L$

Für die äußeren Symbole $\underline{a}$ brauchen wir eine Produktion der Art
$S \to \underline{aa}S\underline{aa}$

Grammatiken: Beispiel 1

Gegeben ist $L = \{\underline{a}^{2n}\underline{b}^{3k-2}\underline{a}^{2n} \mid n, k \geq 1\}$

Gesucht: Eine kontextfreie Grammatik G, sodass $\mathcal{L}(G) = L$

Für die äußeren Symbole $\underline{a}$ brauchen wir eine Produktion der Art
$S \rightarrow \underline{aa}S\underline{aa}$

Für die inneren Symbole $\underline{b}$ brauchen wir eine Produktion, die 1,4,7,10,... Symbole $\underline{b}$ erzeugen kann. Also z.B. $A \rightarrow \underline{b} \mid \underline{bbb}A$

Grammatiken: Beispiel 1

Gegeben ist $L = \{\underline{a}^{2n}\underline{b}^{3k-2}\underline{a}^{2n} \mid n, k \geq 1\}$

Gesucht: Eine kontextfreie Grammatik G, sodass $\mathcal{L}(G) = L$

Für die äußeren Symbole $\underline{a}$ brauchen wir eine Produktion der Art $S \to \underline{aa}S\underline{aa}$

Für die inneren Symbole $\underline{b}$ brauchen wir eine Produktion, die 1,4,7,10,... Symbole $\underline{b}$ erzeugen kann. Also z.B. $A \to \underline{b} \mid \underline{bbb}A$

Schließlich benötigen wir noch einen Verbindung der Nonterminale S und A: $S \to \underline{aa}A\underline{aa}$

Grammatiken: Beispiel 1

Gegeben ist $L = \{\underline{a}^{2n}\underline{b}^{3k-2}\underline{a}^{2n} \mid n, k \geq 1\}$

Gesucht: Eine kontextfreie Grammatik G, sodass $\mathcal{L}(G) = L$

Für die äußeren Symbole $\underline{a}$ brauchen wir eine Produktion der Art $S \rightarrow \underline{aa}S\underline{aa}$

Für die inneren Symbole $\underline{b}$ brauchen wir eine Produktion, die 1,4,7,10,... Symbole $\underline{b}$ erzeugen kann. Also z.B. $A \rightarrow \underline{b} \mid \underline{bbb}A$

Schließlich benötigen wir noch einen Verbindung der Nonterminale S und A: $S \rightarrow \underline{aa}A\underline{aa}$

$G = (\{S, A\}, \{a, b\}, \{S \rightarrow \underline{aa}S\underline{aa} \mid \underline{aa}A\underline{aa}, A \rightarrow \underline{b} \mid \underline{bbb}A\}, S)$

Grammatiken: Pitfalls

Gegeben ist $L = \{\underline{a}^{2n}\underline{b}^{3k-2}\underline{a}^{2n} \mid n, k \geq 1\}$

Grammatiken: Pitfalls

Gegeben ist $L = \{\underline{a}^{2n}\underline{b}^{3k-2}\underline{a}^{2n} \mid n, k \geq 1\}$

Erzeugt G_1 die obige Sprache L?

$$G_1 = (\{S, A\}, \{a, b\}, \{S \to \underline{aa}S\underline{aa} \mid A, A \to \underline{b} \mid \underline{bbb}A\}, S)$$

Grammatiken: Pitfalls

Gegeben ist $L = \{\underline{a}^{2n}\underline{b}^{3k-2}\underline{a}^{2n} \mid n, k \geq 1\}$

Erzeugt G_1 die obige Sprache L?

$$G_1 = (\{S, A\}, \{a, b\}, \{S \to \underline{aa}S\underline{aa} \mid A, A \to \underline{b} \mid \underline{bbb}A\}, S)$$

Nein, das Wort $\underline{b}$ wird von G erzeugt, ist aber nicht in L.

$S \implies A \implies \underline{b}$

Grammatiken: Pitfalls

Gegeben ist $L = \{\underline{a}^{2n}\underline{b}^{3k-2}\underline{a}^{2n} \mid n, k \geq 1\}$

Grammatiken: Pitfalls

Gegeben ist $L = \{\underline{a}^{2n}\underline{b}^{3k-2}\underline{a}^{2n} \mid n, k \geq 1\}$

Erzeugt G_2 die obige Sprache L?

$$G_2 = (\{S, A\}, \{a, b\}, \{S \to \underline{aa}S\underline{aa} \mid A, A \to \underline{bbb}A\}, S)$$

Grammatiken: Pitfalls

Gegeben ist $L = \{\underline{a}^{2n}\underline{b}^{3k-2}\underline{a}^{2n} \mid n, k \geq 1\}$

Erzeugt G_2 die obige Sprache L?

$$G_2 = (\{S, A\}, \{a, b\}, \{S \rightarrow \underline{aa}S\underline{aa} \mid A, A \rightarrow \underline{bbb}A\}, S)$$

Nein, da $\mathcal{L}(G) = \{\}$, da kein Wort erzeugt werden kann, das nur aus Terminalsymbolen besteht.

Grammatiken: Ableitungen

Gegeben ist
$G = (\{S, A\}, \{\underline{a}, \underline{b}\}, \{S \to \underline{aa}S\underline{aa} \mid \underline{aa}A\underline{aa}, A \to \underline{b} \mid \underline{bbb}A\}, S)$

Grammatiken: Ableitungen

Gegeben ist
$G = (\{S, A\}, \{\underline{a}, \underline{b}\}, \{S \to \underline{aa}S\underline{aa} \mid \underline{aa}A\underline{aa}, A \to \underline{b} \mid \underline{bbb}A\}, S)$

Gesucht: Linksableitung des Wortes $\underline{aabbbbaa} \in \mathcal{L}(G)$

Grammatiken: Ableitungen

Gegeben ist
$G = (\{S, A\}, \{\underline{a}, \underline{b}\}, \{S \to \underline{aa}S\underline{aa} \mid \underline{aa}A\underline{aa}, A \to \underline{b} \mid \underline{bbb}A\}, S)$

Gesucht: Linksableitung des Wortes $\underline{aabbbbaa} \in \mathcal{L}(G)$

$$S \implies \underline{aa}A\underline{aa} \implies \underline{aabbb}A\underline{aa} \implies \underline{aabbbbaa}$$

Grammatiken: Ableitungen

Gegeben ist
$G = (\{S, A\}, \{\underline{a}, \underline{b}\}, \{S \to \underline{aa}S\underline{aa} \mid \underline{aa}A\underline{aa}, A \to \underline{b} \mid \underline{bbb}A\}, S)$

Gesucht: Linksableitung des Wortes $\underline{aabbbbaa} \in \mathcal{L}(G)$

$$S \implies \underline{aa}A\underline{aa} \implies \underline{aabbb}A\underline{aa} \implies \underline{aabbbbaa}$$