Theoretische Grundlagen der Informatik
Tutorium XII & XIII (AVG, SR -109)

Tut Nr. 1 – Automaten, reg. Ausdrücke, Gram., Chomsky

David Münch

Karlsruher Institut für Technologie
Institut für Informatik
IKS Müller-Quade

29. Oktober 2009

Inhaltsverzeichnis

Inhaltsverzeichnis

Inhaltsverzeichnis

Inhaltsverzeichnis

Organisatorisches

Email: muenchdavid@gmail.com

`https://www.stud.uni-karlsruhe.de/~uhbro/`

Tutorium 12: Donnerstags 8:00 Uhr - AVG, Raum -109
Tutorium 13: Donnerstags 9:45 Uhr - AVG, Raum -109

Übungsblattabgabe Mittwochs 12:00 Uhr.

Organisatorisches

Deckblatt benutzen: `http://www.stud.uni-karlsruhe.de/~unbdh/deckblatt/index.php?course=6`

Gruppenarbeit erwünscht, aber jeder muss handschriftliche Lösung mit Namen aller Gruppenteilnehmer abgeben.

50% der Punkte sind notwendig für den Schein.

Was wollen wir heute erreichen?

Zu DEAs, regulären Ausdrücken, NEAs wollen wir:

Desweiteren:

Was wollen wir heute erreichen?

Zu DEAs, regulären Ausdrücken, NEAs wollen wir:

- Definitionen selbst formulieren.

Desweiteren:

Was wollen wir heute erreichen?

Zu DEAs, regulären Ausdrücken, NEAs wollen wir:

- Definitionen selbst formulieren.
- Aufgaben lösen.

Desweiteren:

Was wollen wir heute erreichen?

Zu DEAs, regulären Ausdrücken, NEAs wollen wir:

- Definitionen selbst formulieren.
- Aufgaben lösen.
- Zusammenhänge verstehen.

Desweiteren:

Was wollen wir heute erreichen?

Zu DEAs, regulären Ausdrücken, NEAs wollen wir:

- Definitionen selbst formulieren.
- Aufgaben lösen.
- Zusammenhänge verstehen.

Desweiteren:

- Überblick über die Chomksy-Hierarchie erhalten

Was wollen wir heute erreichen?

Zu DEAs, regulären Ausdrücken, NEAs wollen wir:

- Definitionen selbst formulieren.
- Aufgaben lösen.
- Zusammenhänge verstehen.

Desweiteren:

- Überblick über die Chomksy-Hierarchie erhalten
- Umformen von Typ-3-Grammatiken $\Leftrightarrow$ endlicher Automat

Was wollen wir heute erreichen?

Zu DEAs, regulären Ausdrücken, NEAs wollen wir:

- Definitionen selbst formulieren.
- Aufgaben lösen.
- Zusammenhänge verstehen.

Desweiteren:

- Überblick über die Chomksy-Hierarchie erhalten
- Umformen von Typ-3-Grammatiken $\Leftrightarrow$ endlicher Automat
- Beweisen, dass eine Grammatik eine konkrete Sprache erzeugt.

Definition: DEA

Definition: DEA

Ein **deterministischer endlicher Automat M** wird spezifiert durch ein 5-Tupel

$$M = (Q, \Sigma, \delta, q_0, F).$$

Hierbei bezeichnet Q die Menge der **Zustände**
und Σ ist das **Eingabealphabet**, $Q \cap \Sigma = \emptyset$.
Q und Σ müssen - wie der Name sagt - endliche Mengen sein.
$q_0 \in Q$ ist der **Startzustand**,
$F \subseteq Q$ ist die Menge der **Endzustände**
und $\delta : Q \times \Sigma \to Q$ heisst die **Überführungsfunktion**.

Aufgabe

Entwerfe einen DEA, der alle durch 4 teilbaren Binärzahlen akzeptiert.

induktive Definition: Reguläre Ausdrücke

induktive Definition: Reguläre Ausdrücke

- $\emptyset$ ist ein regulärer Ausdruck

induktive Definition: Reguläre Ausdrücke

- $\emptyset$ ist ein regulärer Ausdruck
- ε ist ein regulärer Ausdruck

induktive Definition: Reguläre Ausdrücke

- $\emptyset$ ist ein regulärer Ausdruck
- ε ist ein regulärer Ausdruck
- Für jedes $a \in \Sigma$ ist a ist ein regulärer Ausdruck

induktive Definition: Reguläre Ausdrücke

- $\emptyset$ ist ein regulärer Ausdruck
- ε ist ein regulärer Ausdruck
- Für jedes $a \in \Sigma$ ist a ist ein regulärer Ausdruck
- Wenn α und β reguläre Ausdrücke, dann auch $\alpha\beta$, $(\alpha \cup \beta)$, sowie $(\alpha)^*$

Statt $\cup$ auch $+$ möglich.

induktive Definition: Reguläre Ausdrücke

- $\emptyset$ ist ein regulärer Ausdruck
- ε ist ein regulärer Ausdruck
- Für jedes $a \in \Sigma$ ist a ist ein regulärer Ausdruck
- Wenn α und β reguläre Ausdrücke, dann auch $\alpha\beta$, $(\alpha \cup \beta)$, sowie $(\alpha)^*$

Statt $\cup$ auch $+$ möglich.
Beispiel:

$$(0 \cup (0 \cup 1)^* 00)$$

induktive Definition: Reguläre Sprache $L(\gamma)$

induktive Definition: Reguläre Sprache $L(\gamma)$

- Falls $\gamma = \emptyset$, so ist $L(\gamma) = \emptyset$

induktive Definition: Reguläre Sprache $L(\gamma)$

- Falls $\gamma = \emptyset$, so ist $L(\gamma) = \emptyset$
- Falls $\gamma = \varepsilon$, so ist $L(\gamma) = \{\varepsilon\}$

induktive Definition: Reguläre Sprache $L(\gamma)$

- Falls $\gamma = \emptyset$, so ist $L(\gamma) = \emptyset$
- Falls $\gamma = \varepsilon$, so ist $L(\gamma) = \{\varepsilon\}$
- Falls $\gamma = a$, so ist $L(\gamma) = \{a\}$

induktive Definition: Reguläre Sprache $L(\gamma)$

- Falls $\gamma = \emptyset$, so ist $L(\gamma) = \emptyset$
- Falls $\gamma = \varepsilon$, so ist $L(\gamma) = \{\varepsilon\}$
- Falls $\gamma = a$, so ist $L(\gamma) = \{a\}$
- Falls $\gamma = \alpha\beta$, so ist $L(\gamma) = L(\alpha)L(\beta)$

induktive Definition: Reguläre Sprache $L(\gamma)$

- Falls $\gamma = \emptyset$, so ist $L(\gamma) = \emptyset$
- Falls $\gamma = \varepsilon$, so ist $L(\gamma) = \{\varepsilon\}$
- Falls $\gamma = a$, so ist $L(\gamma) = \{a\}$
- Falls $\gamma = \alpha\beta$, so ist $L(\gamma) = L(\alpha)L(\beta)$

Beispiel:
Dem regulären Ausdruck $\gamma = (0 \cup (0 \cup 1)^*00)$ ist die reg. Sprache $L(\gamma) = \{0$ oder das Wort endet mit $00\}$ zugeordnet.

Aufgabe

Formuliere einen regulären Ausdruck für folgende Sprache:
Die Menge der Zeichenketten über dem Alphabet $\{a, b, c\}$, die mindestens ein a und ein b enthalten.

Aufgabe

Formuliere einen regulären Ausdruck für folgende Sprache:
Die Menge der Zeichenreihen aus Nullen und Einsen, deren Anzahl von Nullen durch 5 teilbar ist.

Satz: reg. Ausdrücke

Die Menge, der durch reguläre Ausdrücke beschreibbaren Sprachen, ist genau die Menge der regulären Sprachen.

Satz: reg. Ausdrücke

Die Menge, der durch reguläre Ausdrücke beschreibbaren Sprachen, ist genau die Menge der regulären Sprachen.

Satz: DEA $\Leftrightarrow$ reg. Sprache

Jede reguläre Sprache wird von einem DEA akzeptiert.
Jede durch DEAs erkennbare Sprache ist regulär.

Definition: NEA

Definition: NEA

Ein **nichtdeterministischer endlicher Automat M** wird spezifiert durch ein 5-Tupel

$$M = (Q, \Sigma, \delta, q_0, F).$$

Hierbei bezeichnet Q die Menge der **Zustände**
und Σ ist das **Eingabealphabet**, $Q \cap \Sigma = \emptyset$.
Q und Σ müssen - wie der Name sagt - endliche Mengen sein.
$q_0 \in Q$ ist der **Startzustand**,
$F \subseteq Q$ ist die Menge der **Endzustände**
und $\delta : Q \times (\Sigma \cup \{\varepsilon\}) \to \mathcal{P}(Q)$ heisst die **Überführungsfunktion**.

Definition: ε-Übergänge beim NEA

Definition: ε-Übergänge beim NEA

Für einen Zustand $q \in Q$ ist der ε-Abschluss $E(q)$ wie folgt definiert:
$E(q) := \{p \in Q | p$ ist von q durch eine Folge von ε-Übergängen erreichbar.$\}$

Aufgabe

Es ist folgender NEA M gegeben:

$M = (\{S, Z1, Z2, Z3\}, \{0, 1\}, \delta, S, \{Z2\})$ mit

$\delta : Q \times (\Sigma \cup \{\varepsilon\}) \rightarrow \mathcal{P}(Q)$

$S \times \varepsilon \rightarrow \{Z1, Z2\}$

$Z1 \times 1 \rightarrow Z2$

$Z2 \times 0 \rightarrow Z3$

$Z2 \times 1 \rightarrow Z3$

$Z3 \times \varepsilon \rightarrow Z1$

Aufgabe

Es ist folgender NEA M gegeben:

$M = (\{S, Z1, Z2, Z3\}, \{0, 1\}, \delta, S, \{Z2\})$ mit

$\delta : Q \times (\Sigma \cup \{\varepsilon\}) \rightarrow \mathcal{P}(Q)$

$S \times \varepsilon \rightarrow \{Z1, Z2\}$

$Z1 \times 1 \rightarrow Z2$

$Z2 \times 0 \rightarrow Z3$

$Z2 \times 1 \rightarrow Z3$

$Z3 \times \varepsilon \rightarrow Z1$

Sei L die Sprache, die von dem Automaten akzeptiert wird.

$01010 \in L$?

$1^n \in L \forall n \in \mathbb{N}_0$?

Es gibt ein $m \in \mathbb{N}_0$ mit $(10)^m 01^m \in L$?

Satz: DEA $\Leftrightarrow$ NEA

Jede von einem NFA akzeptierte Sprache ist auch durch einem DEA akzeptierbar. Und umgekehrt.

Satz: DEA $\Leftrightarrow$ NEA

Jede von einem NFA akzeptierte Sprache ist auch durch einem DEA akzeptierbar. Und umgekehrt.

Satz: DEA $\Leftrightarrow$ NEA $\Leftrightarrow$ reg. Sprache

DEA, NEA und reg. Sprache sind äquivalent.

Grammatiken und Chomsky-Hierarchie

Definition: Grammatik

Eine **Grammatik** G besteht aus vier Komponenten:

1. einem endlichen Alphabet T (**Terminalalphabet**)
2. einer endlichen Menge V mit $V \cap T = \emptyset$ von Variablen (**Nichtterminale**)
3. einem **Startsymbol** $S \in V$
4. einer endlichen Menge von **Ableitungsregeln** P (Produktionen).
   Dabei ist eine Ableitungsregel ein Paar (l, r) mit $l \in (V \cup T)^+$ und $r \in (V \cup T)^*$. Statt (l, r) schreiben wir auch $l \to r$.

λ ist das neutrale Element bezüglich der Konkatenation.

λ ist das neutrale Element bezüglich der Konkatenation.
Wir schreiben $w \rightarrow z$, wenn w durch eine Anwendung einer Ableitungsregel in z verwandelt wird und $w \xrightarrow{*} z$, wenn w durch eine Anwendung von mehreren Ableitungsregeln in z verwandelt wird.

λ ist das neutrale Element bezüglich der Konkatenation.
Wir schreiben $w \to z$, wenn w durch eine Anwendung einer Ableitungsregel in z verwandelt wird und $w \xrightarrow{*} z$, wenn w durch eine Anwendung von mehreren Ableitungsregeln in z verwandelt wird.

Definition: erzeugte Sprache

Die von einer Grammatik G **erzeugte Sprache** $L(G)$ ist die Menge aller Wöerter $z \in T^*$, für die $S \xrightarrow{*} z$ gilt.

Aufgabe 1

Gib eine Grammatik G an, mit der sich die korrekt geklammerten aussagenlogischen Formeln darstellen lassen.
Überprüfe ob $(\neg((v \vee v) \wedge (\neg v))) \in L(G)$.

Aufgabe 1

Gib eine Grammatik G an, mit der sich die korrekt geklammerten aussagenlogischen Formeln darstellen lassen.
Überprüfe ob $(\neg((v \vee v) \wedge (\neg v))) \in L(G)$.

Lösung:
$G = (\{A, B\}, \{(,), \vee, \wedge, \neg, 0, 1, v\}, B, P)$ mit $P = \{$
$B \rightarrow (B \vee B)$
$B \rightarrow (B \wedge B)$
$B \rightarrow (\neg B)$
$B \rightarrow A$
$A \rightarrow v$
$A \rightarrow 0$
$A \rightarrow 1\}$
Nicht die kürzeste Grammatik, aber die intuitiv einfachste!

Definition: Chomsky-Hierarchie

- Grammatiken ohne weitere Einschränkungen heissen Grammatiken vom **Typ 0**.

Definition: Chomsky-Hierarchie

- Grammatiken ohne weitere Einschränkungen heissen Grammatiken vom **Typ 0**.
- Grammatiken, bei denen alle Ableitungsregeln die Form

  $u \rightarrow v$ mit $u \in V^+, v \in ((V \cup T)\backslash\{S\})^+$ und $|u| \leq |v|$, oder $S \rightarrow \varepsilon$

  haben, heissen **kontextsensitiv** od. Grammatiken vom **Typ 1**.

Definition: Chomsky-Hierarchie

- Grammatiken ohne weitere Einschränkungen heissen Grammatiken vom **Typ 0**.
- Grammatiken, bei denen alle Ableitungsregeln die Form

  $u \to v$ mit $u \in V^+, v \in ((V \cup T) \backslash \{S\})^+$ und $|u| \leq |v|$, oder $S \to \varepsilon$

  haben, heissen **kontextsensitiv** od. Grammatiken vom **Typ 1**.
- Grammatiken, bei denen alle Ableitungsregeln die Form $A \to v$ mit $A \in V$ und $v \in (V \cup T)^*$ haben, heissen **kontextfrei** oder Grammatiken vom **Typ 2**.

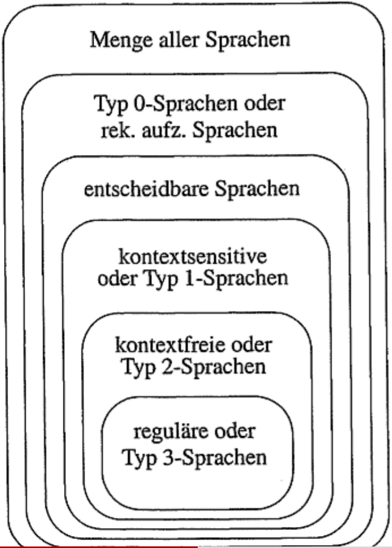
Definition: Chomsky-Hierarchie

- Grammatiken ohne weitere Einschränkungen heissen Grammatiken vom **Typ 0**.
- Grammatiken, bei denen alle Ableitungsregeln die Form

  $u \to v$ mit $u \in V^+, v \in ((V \cup T)\backslash\{S\})^+$ und $|u| \leq |v|$, oder $S \to \varepsilon$

  haben, heissen **kontextsensitiv** od. Grammatiken vom **Typ 1**.
- Grammatiken, bei denen alle Ableitungsregeln die Form $A \to v$ mit $A \in V$ und $v \in (V \cup T)^*$ haben, heissen **kontextfrei** oder Grammatiken vom **Typ 2**.
- Grammatiken, bei denen alle Ableitungsregeln die Form $A \to v$ mit $A \in V$ und $v = \varepsilon$ oder $v = aB$ mit $a \in T, B \in V$ haben, heissen **rechtslinear** oder Grammatiken vom **Typ 3**.

Grammatiken und Chomsky-Hierarchie



Satz

Die Klasse der von endichen Automaten akzeptierten Sprachen ist genau die Klasse der von Chomsky-3-Grammatiken erzeugten Sprachen.

Aufgabe

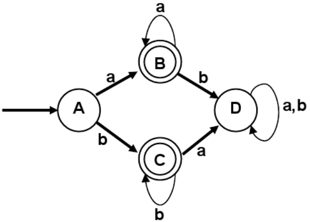
Gegeben ist die Grammatik $G = (T, V, S, P)$ mit $T = \{a, b, c\}$, $V = \{S, A\}$ und $P = \{S \rightarrow AA, A \rightarrow AAA|abA|aAb|bAa|baA|c\}$

1. Gebe den maximalen Typ von G in der Chomsky-Hierachie an und begründe deine Antwort.
2. Bestimme alle Wörter aus $L(G)$, die sich durch Anwendung von maximal vier Ableitungsregeln ergeben.
3. Ist das Wort $w = acbbca$ in $L(G)$ enthalten?
4. Zeige, dass die Sprache $L(\alpha)$, die durch den regulären Ausdruck

$$\alpha := ((ab \cup ba)^* c(ab \cup ba)^* c)^+$$

gegeben ist, in $L(G)$ enthalten ist.

Gegeben sei der folgende endliche Automat: $\mathcal{M} = (\mathcal{Q}, \Sigma, \delta, q_0, \mathcal{F})$
$\Sigma = \{a, b\}$, $\mathcal{Q} = \{A, B, C, D\}$, $q_0 = A$, $\mathcal{F} = \{B, C\}$



1. Geben Sie die von diesem Automaten akzeptierte Sprache als regulären Ausdruck an.
2. Konstruieren Sie einen äquivalenten endlichen Automaten, der nur einen einzigen Endzustand besitzt.
3. Geben Sie eine linkslineare Grammatik für die Sprache dieses Automaten an, die keine überflüssigen Nichtterminale und Regeln enthält.

1. Formulieren Sie einen regulären Ausdruck über dem Alphabet $\Sigma = \{0, 1\}$, der jedes beliebige Wort erfasst, wobei die vorletzte Ziffer 0 sein soll.
2. Geben Sie den dafür größtmöglichen Chomsky Typ und eine Grammatik an.
3. Geben Sie den dazugehörigen Automaten an, der diese Sprache akzeptiert.

Reflexion

Was haben wir heute gelernt?

Reflexion

Was haben wir heute gelernt?

- DEAs

Reflexion

Was haben wir heute gelernt?

- DEAs
- NEAs

Reflexion

Was haben wir heute gelernt?

- DEAs
- NEAs
- reguläre Ausdrücke, reguläre Sprache

Reflexion

Was haben wir heute gelernt?

- DEAs
- NEAs
- reguläre Ausdrücke, reguläre Sprache
- Übeblick über die Chomsky-Hierarchie

Reflexion

Was haben wir heute gelernt?

- DEAs
- NEAs
- reguläre Ausdrücke, reguläre Sprache
- Übeblick über die Chomsky-Hierarchie
- Chomsky-3-Sprachen

Reflexion

Was haben wir heute gelernt?

- DEAs
- NEAs
- reguläre Ausdrücke, reguläre Sprache
- Übeblick über die Chomsky-Hierarchie
- Chomsky-3-Sprachen
- Äquivalenz von Chomsky-3, DEA, NEA und reg. Sprachen

Noch Fragen?

Vorschau

Vorschau

- Pumping-Lemma

Vorschau

- Pumping-Lemma
- Minimierung von Automaten

Bis zum nächsten Mal