

Grundlagen der Theoretischen Informatik

Sommersemester 2015

16.04.2015

Viorica Sofronie-Stokkermans

e-mail: sofronie@uni-koblenz.de

Bis jetzt

- Organizationalisches
- Literatur
- Motivation und Inhalt
- Kurzer Überblick: Logik
- Kurzer Überblick: Beweismethoden und mathematische Konzepte

Inhalt der Vorlesung

1. Terminologie
2. Endliche Automaten und reguläre Sprachen
3. Kellerautomaten und kontextfreie Sprachen
4. Turingmaschinen und rekursiv aufzählbare Sprachen
5. Berechenbarkeit, (Un-)Entscheidbarkeit
6. Komplexitätsklassen P und NP

Inhalt der Vorlesung

1. Terminologie
2. Endliche Automaten und reguläre Sprachen
3. Kellerautomaten und kontextfreie Sprachen
4. Turingmaschinen und rekursiv aufzählbare Sprachen
5. Berechenbarkeit, (Un-)Entscheidbarkeit
6. Komplexitätsklassen P und NP

Terminologie

In den folgenden Abschnitten führen wir die Begriffe

- **Sprache**
- **Grammatik**

ein.

Wir untersuchen insbesondere

1. wie man Probleme aus der Mathematik, Graphentheorie, Logik als **Probleme über Sprachen** formulieren kann.
2. wie man Klassen von Grammatiken von steigendem Schwierigkeitsgrad definiert: **Chomsky-Hierarchie**.
3. **wie viele** Grammatiken und Sprachen **es überhaupt gibt**
(so viele wie natürliche Zahlen, reelle Zahlen oder komplexe Zahlen?)

Sprache, Grammatik

Alphabete, Wörter

Definition (Alphabet)

Ein **Alphabet** ist eine Menge von Zeichen/Buchstaben

- Grundlage einer Sprache (die zur Verfügung stehenden Zeichen)
- Meist endlich

Alphabete, Wörter

Definition (Alphabet)

Ein **Alphabet** ist eine Menge von Zeichen/Buchstaben

- Grundlage einer Sprache (die zur Verfügung stehenden Zeichen)
- Meist endlich

Definition (Wort)

Ein **Wort** (über einem Alphabet Σ)

ist eine endliche Folge von Zeichen aus Σ

- $|w|$ bezeichnet Länge eines Wortes w
- ε bezeichnet das **leere Wort**
- $\#_a(w)$ ist die Anzahl der Vorkommen des Buchstabens a im Wort w .

Alphabete, Wörter

Beispiele

- $\Sigma_1 = \{0, 1\}$.

Wörter über Σ_1 : ε , 0, 1, 01, 1001, 100101

- $\Sigma_2 = \{a, b, c, d, e, \dots, x, y, z\}$.

Wörter über Σ_2 : ε , a, b, aba, baab, *informatik*

- $\Sigma_3 = \{(\,), +, -, *, a\}$.

Wörter über Σ_3 : ε , a, (a + a), (a + a) * (a - a), ++a)

Alphabete, Wörter

Operationen auf Wörtern

Verknüpfung (Konkatenation):

$$w \circ w'$$

assoziativ, oft geschrieben als ww'

i -te Potenz:

$$w^0 = \varepsilon, \quad w^{i+1} = ww^i$$

Reverse:

w^R = das Wort w rückwärts

Alphabete, Wörter

Beispiele

- $\Sigma_1 = \{0, 1\}$.

Wörter über Σ_1 : ε , 0, 1, 01, 1001, 100101

$$\varepsilon \circ 01001 = 01001, \quad 01 \circ 1001 = 011001$$

$$1001^3 = 100110011001$$

$$01101^R = 10110$$

- $\Sigma_2 = \{a, b, c, d, e, \dots, x, y, z\}$.

Wörter über Σ_2 : ε , *a*, *b*, *aba*, *baab*, *informatik*

$$ab \circ \varepsilon = ab, \quad aba \circ baab = ababaab$$

$$aba^2 = abaaba$$

$$informatik^R = kitamrofni.$$

- $\Sigma_3 = \{(,), +, -, *, a\}$.

Wörter über Σ_2 : ε , *a*, $(a + a)$, $(a + a) * (a - a)$, $++a$

Sprache

Definition (Sprache)

Eine Sprache L (über einem Alphabet Σ)
ist eine Menge von Wörtern über Σ .

Alphabete, Wörter

Beispiele

- $\Sigma_1 = \{0, 1\}$.
Gerade = Menge aller Wörtern über Σ_1 , die die binäre Representation einer geraden Zahl sind.
- $\Sigma_2 = \{a, b, c, d, e, \dots, x, y, z\}$.
ENGLISH = $\{w \mid w \text{ wort über } \Sigma_2, \text{ der auf English eine Bedeutung hat}\}$
- $\Sigma_3 = \{(,), +, -, *, a\}$.
EXPR = Menge der Wörter über Σ_2 , die korrekt geklammerten Ausdrücke sind.

Operationen auf Sprachen

Konkatenation:

$$L \circ M = \{w \circ w' \mid w \in L, w' \in M\}$$

oft geschrieben als LM

i -te Potenz:

$$L^0 = \{\varepsilon\}, \quad L^{i+1} := LL^i$$

Reverse:

$$L^R = \{w^R \mid w \in L\}$$

Operationen auf Sprachen

Kleene-Hülle

$$L^* = L^0 \cup L^1 \cup L^2 \cup \dots$$

Variante:

$$L^+ = LL^* = L^1 \cup L^2 \cup \dots$$

Operationen auf Sprachen

Kleene-Hülle

$$L^* = L^0 \cup L^1 \cup L^2 \cup \dots$$

Variante:

$$L^+ = LL^* = L^1 \cup L^2 \cup \dots$$

Σ^* bezeichnet die Menge aller Wörter über Σ

Operationen auf Sprachen

Kleene-Hülle

$$L^* = L^0 \cup L^1 \cup L^2 \cup \dots$$

Variante:

$$L^+ = LL^* = L^1 \cup L^2 \cup \dots$$

Σ^* bezeichnet die Menge aller Wörter über Σ

Genau genommen besteht ein Unterschied:

ein Buchstabe \neq Wort, das nur aus dem einen Buchstaben besteht

Darum ist Σ selbst keine Sprache über Σ

(Oft wird über diesen Unterschied hinweggesehen)

Reguläre Ausdrücke

Definition (Reguläre Ausdrücke)

Menge \mathfrak{Reg}_Σ der **regulären Ausdrücke** (über Σ) ist definiert durch:

1. 0 ist ein regulärer Ausdruck
2. Für jedes $a \in \Sigma$ ist a ein regulärer Ausdruck
3. Sind r und s reguläre Ausdrücke, so auch
 - $(r + s)$ (Vereinigung),
 - (rs) (Konkatenation),
 - (r^*) (Kleene Stern)

Klammern können weggelassen werden, dann

- * hat Vorrang vor Konkatenation
- Konkatenation hat Vorrang vor +

Reguläre Ausdrücke

Definition (Semantik regulärer Ausdrücke]

Ein regulärer Ausdruck r stellt eine Sprache $\mathcal{J}(r)$ über Σ wie folgt dar:

$$\begin{aligned}\mathcal{J}(0) &:= \emptyset \\ \mathcal{J}(a) &:= \{a\} \quad \text{für } a \in \Sigma \\ \mathcal{J}(r + s) &:= \mathcal{J}(r) \cup \mathcal{J}(s) \\ \mathcal{J}(rs) &:= \mathcal{J}(r)\mathcal{J}(s) \\ \mathcal{J}(r^*) &:= \mathcal{J}(r)^*\end{aligned}$$

Wir benutzen auch das Makro ...

$$1 := 0^*$$

Es gilt: $\mathcal{J}(1) = \{\varepsilon\}$

Reguläre Ausdruck

Übung

Welche Sprachen werden durch die folgenden regulären Ausdrücke dargestellt?

- aa
- $(a + b)^*$
- $aa^* + bb^*$

Reguläre Ausdruck

Übung

Welche Sprachen werden durch die folgenden regulären Ausdrücke dargestellt?

- aa
- $(a + b)^*$
- $aa^* + bb^*$

$$\mathcal{J}(aa) = \mathcal{J}(a)\mathcal{J}(a) = \{a\} \circ \{a\} = \{aa\}$$

$$\mathcal{J}((a + b)^*) = \mathcal{J}(a + b)^* = (\mathcal{J}(a) \cup \mathcal{J}(b))^* = (\{a\} \cup \{b\})^* = \{a, b\}^*$$

$$\mathcal{J}(aa^* + bb^*) = \mathcal{J}(aa^*) \cup \mathcal{J}(bb^*) = \{a\}\{a\}^* \cup \{b\}\{b\}^* = \{a\}^+ \cup \{b\}^+$$

Reguläre Ausdruck

Übung

Geben Sie einen regulären Ausdruck über $\Sigma = \{a, b, c\}$ an, der die Sprache darstellt, die genau die Wörter enthält, die mit b beginnen.

Antwort: $b(a + b + c)^*$