

Automatentheorie und Formale Sprachen

für die Studiengänge

- Angewandte Informatik
- Informatik - Technische Systeme

04 Reguläre Sprachen: Reguläre Ausdrücke

Prof. Dr. David Sabel
Sommersemester 2025

Stand der Folien: 20. Mai 2025

Reguläre Ausdrücke

- Reguläre Ausdrücke sind (wie Automaten und Grammatiken) ein Formalismus zur Repräsentation von Sprachen.
- Praktische Verwendung: Regex-Bibliotheken in Programmiersprachen oder bei der Shell-Programmierung zum Suchen und Ersetzen von Zeichenketten (verwenden meist **erweiterte** reguläre Ausdrücke)
- Aufbau regulärer Ausdrücke:
Basisausdrücke und Operatoren zum Zusammensetzen

Definition (Regulärer Ausdruck)

Sei Σ ein Alphabet. Ein **regulärer Ausdruck** über Σ ist induktiv definiert:

- $\emptyset$ ist ein regulärer Ausdruck
- ε ist ein regulärer Ausdruck
- a mit $a \in \Sigma$ ist ein regulärer Ausdruck
- Wenn α_1 und α_2 reguläre Ausdrücke sind, dann auch $\alpha_1\alpha_2$
- Wenn α_1 und α_2 reguläre Ausdrücke sind, dann auch $(\alpha_1|\alpha_2)$
- Wenn α regulärer Ausdruck ist, dann auch $(\alpha)^*$

Reguläre Ausdrücke (3)

Erzeugte Sprache

Die von einem regulären Ausdruck α **erzeugte Sprache** $L(\alpha)$ ist induktiv über dessen Struktur definiert:

$$\begin{aligned} L(\emptyset) &:= \emptyset \\ L(\varepsilon) &:= \{\varepsilon\} \\ L(a) &:= \{a\} \quad \text{für } a \in \Sigma \\ L(\alpha_1\alpha_2) &:= L(\alpha_1)L(\alpha_2) = \{uv \mid u \in L(\alpha_1), v \in L(\alpha_2)\} \\ L(\alpha_1|\alpha_2) &:= L(\alpha_1) \cup L(\alpha_2) \\ L((\alpha)^*) &:= L(\alpha)^* \end{aligned}$$

Für alle regulären Ausdrücke $\alpha_1, \alpha_2, \alpha_3$ gilt:

$$L((\alpha_1|\alpha_2)|\alpha_3) = L(\alpha_1|(\alpha_2|\alpha_3))$$

Daher lassen wir Klammern weg und schreiben $(\alpha_1|\alpha_2|\dots|\alpha_n)$.

Quiz 1

Welches Konstrukt ist eigentlich überflüssig, da es durch andere dargestellt werden kann?

- 1 $(\alpha)^*$
- 2 $(\alpha_1 \mid \alpha_2)$
- 3 ε
- 4 $\alpha_1 \alpha_2$
- 5 $\emptyset$

arsnova.hs-rm.de
6750 1376



Beachte...

$\emptyset^* = ?$

Beispiele

- $(a|b)^*aa(a|b)^*$
erzeugt alle Wörter über $\{a, b\}$, die **?**zwei aufeinanderfolgende a 's enthalten
- $(\varepsilon | ((a|b|c)^*a(a|b|c)(a|b|c)(a|b|c)))$
erzeugt alle Wörter über $\{a, b, c\}$, die **?**an viertletzter Stelle ein a haben und das leere Wort
- $((0|1|2|3|4|5|6|7|8|9)|1(0|1|2|3|4|5|6|7|8|9)|(2(0|1|2|3))) :$
 $((0|1|2|3|4|5)(0|1|2|3|4|5|6|7|8|9))$
erzeugt **?**alle Uhrzeiten im 24-Stunden-Format
- Eine endliche Sprache $S = \{w_1, \dots, w_n\}$ wird durch **?** $(w_1 | \dots | w_n)$ erzeugt.

Quiz 2

Welcher reguläre Ausdruck erzeugt die Sprache

$$\{u \in \{a, b\}^* \mid |u| = 4\}?$$

- 1 $(ab)^*(ab)^*(ab)^*(ab)^*$
- 2 $(a|b)(a|b)(a|b)(a|b)$
- 3 $(aaaa|bbbb)$
- 4 $(ab|ab|ab|ab)$
- 5 $(aa|ab|ba|bb)(aa|ab|ba|bb)$

arsnova.hs-rm.de
6750 1376



Quiz 3



Geben Sie einen regulären Ausdruck an, der die Sprache

$$\{u \in \{a, b\}^* \mid |u| \geq 4\}$$

erzeugt.

arsnova.hs-rm.de
6750 1376



Quiz 4



Geben Sie einen regulären Ausdruck an, der die Sprache

$$\{u \in \{a, b\}^* \mid |u| \leq 4\}$$

erzeugt.

arsnova.hs-rm.de
6750 1376



Komplement von regulären Ausdrücken



- Allgemein: Es gibt keinen „Komplementoperator“ für reguläre Ausdrücke.
- Harte Methode: Regulärer Ausdruck $\rightarrow$ NFA $\rightarrow$ DFA $\rightarrow$ NFA für das Komplement $\rightarrow$ DFA $\rightarrow$ regulärer Ausdruck
- Einfachere Methode:
 - 1.Schritt:** Regulärer Ausdruck $\rightarrow$ einfache Beschreibung der erzeugten Sprache
 - 2.Schritt:** $\rightarrow$ einfache Beschreibung des Komplements
 - 3.Schritt:** $\rightarrow$ regulärer Ausdruck

Aufgabe (Schritt für Schritt)



Ziel: *Finde regulären Ausdruck für das Komplement, der von 0^*10^* erzeugten Sprache*

Quiz 5



Welche Sprache wird von 0^*10^* erzeugt?

arsnova.hs-rm.de
6750 1376



Quiz 6



Was ist das Komplement der Sprache aller Wörter über $\{0, 1\}$, die genau eine 1 enthalten.

arsnova.hs-rm.de
6750 1376



Quiz 7



Geben Sie einen regulären Ausdruck an, der alle Wörter über $\{0, 1\}$ erzeugt, die keine Einsen enthalten.

arsnova.hs-rm.de
6750 1376



Quiz 8



Geben Sie einen regulären Ausdruck an, der alle Wörter über $\{0, 1\}$ erzeugt, die mindestens 2 Einsen enthalten.

arsnova.hs-rm.de
6750 1376



Ziel: *Finde regulären Ausdruck für das Komplement, der von 0^*10^* erzeugten Sprache*

Regulärer Ausdruck für das Komplement von 0^*10^* : $0^*|(0|1)^*1(0|1)^*$

Beispiel: grep

```
$ grep -E " d(er|ie|as) neue" faust.txt
Nein, er gefällt mir nicht, der neue Bürgermeister!
Allein der neue Trieb erwacht,
Da seh' ich auch die neue Wohnung,
Noch blendet ihn der neue Tag.
```

```
$ grep -E "(der|die|das) Q[a-z]*" faust.txt
Von dem der Quell sich ewig sprudelnd stürzt,
Vom ganzen Praß die Quintessenz.
```

```
$ grep -E "(( )*Gretchen[[:punct:]]*){2,}" faust.txt
Gretchen! Gretchen!
```

Satz von Kleene

Theorem 3.6.4 (Satz von Kleene)

Reguläre Ausdrücke erzeugen genau die regulären Sprachen.

Beweis in zwei Teilen:

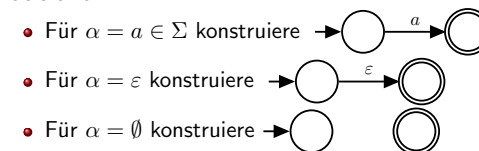
- ➊ Jede von einem regulären Ausdruck erzeugte Sprache ist regulär.
- ➋ Für jede reguläre Sprache gibt es einen regulären Ausdruck, der sie erzeugt.

Beweis: Satz von Kleene (1)

1. Jede von einem regulären Ausdruck erzeugte Sprache ist regulär.

Beweis:

- Wir konstruieren für regulären Ausdruck α einen NFA M_α mit ε -Übergängen und eindeutigen Start- und Endzuständen, sodass $L(M_\alpha) = L(\alpha)$.
- Induktion über die Struktur von α
- Basisfälle:



In allen Fällen ist $L(\alpha) = L(M_\alpha)$ offensichtlich

Beweis: Satz von Kleene (2)

- Induktionsschritt: Betrachte den Aufbau von α (3 Fälle)

- Für $\alpha = \alpha_1\alpha_2$, liefert die I.H. $M_{\alpha_1}, M_{\alpha_2}$.

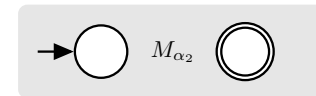


Konstruiere daraus M_α :

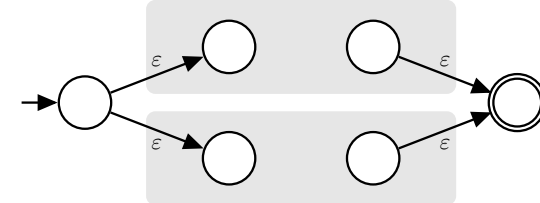


Beweis: Satz von Kleene (3)

- Für $\alpha = (\alpha_1|\alpha_2)$ liefert die I.H. $M_{\alpha_1}, M_{\alpha_2}$:



Konstruiere daraus M_α :

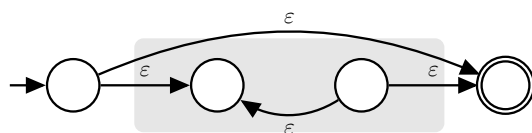


Beweis: Satz von Kleene (4)

- Für $\alpha = (\alpha_1)^*$ liefert die I.H. M_{α_1}



Konstruiere daraus M_α :

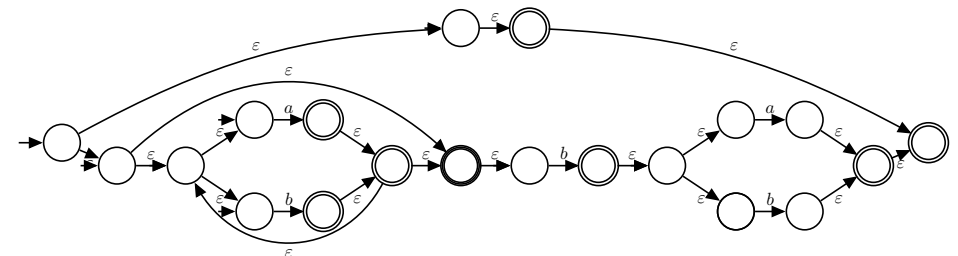


Beispiel: Regulärer Ausdruck $\rightarrow$ NFA mit ε -Übergängen

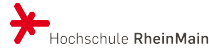
NFA zum regulären Ausdruck

$(\varepsilon|(a|b)^*b(a|b))$

konstruieren:



Beweis: Satz von Kleene (5)



2. Für jede reg. Sprache L gibt es einen regulären Ausdruck α mit $L(\alpha) = L$

Beweis:

- Sei DFA $M = (\{z_1, \dots, z_n\}, \Sigma, \delta, z_1, E)$ mit $L(M) = L$ gegeben.
- Für $w \in \Sigma^*$ und z_i, z_j mit $\hat{\delta}(z_i, w) = z_j$ sei $visit_i(w) = q_1, \dots, q_m$ die Folge der besuchten Zustände (wobei $q_1 = z_i$ und $q_m = z_j$).
- Wir definieren:

$$L_{i,j}^k = \left\{ w \in \Sigma^* \mid \begin{array}{l} \hat{\delta}(z_i, w) = z_j \text{ und } visit_i(w) = q_1, \dots, q_m, \\ \text{sodass für } 1 < l < m: \text{ wenn } q_l = z_p \text{ dann } p \leq k \end{array} \right\}$$

$L_{i,j}^k$ enthält die Wörter, die M von Zustand z_i zu Zustand z_j führen ohne dabei Zwischenzustände mit Index größer als k zu benutzen.

- Mit Induktion über k zeigen wir:

Es gibt reguläre Ausdrücke $\alpha_{i,j}^k$ mit $L(\alpha_{i,j}^k) = L_{i,j}^k$.

Beweis: Satz von Kleene (6)



Zur Erinnerung: $L_{i,j}^k$ enthält die Wörter, die M von Zustand z_i zu Zustand z_j führen ohne dabei Zwischenzustände mit Index größer als k zu benutzen.

Basis: $k = 0$

- Wenn $i \neq j$, dann ist $L_{i,j}^0 = \{a \in \Sigma \mid \delta(z_i, a) = z_j\}$.

Falls $L_{i,j}^0 = \{a_1, \dots, a_q\}$, dann gilt $L(\alpha_{i,j}^0) = L_{i,j}^0$ für $\alpha_{i,j}^0 = (a_1 | \dots | a_q)$.

Falls $L_{i,j}^0 = \emptyset$, dann gilt $L(\alpha_{i,j}^0) = L_{i,j}^0$ mit $\alpha_{i,j}^0 = \emptyset$.

- Wenn $i = j$, dann ist $L_{i,i}^0 = \{\varepsilon\} \cup \{a \in \Sigma \mid \delta(z_i, a) = z_i\}$.

Sei $L_{i,i}^0 = \{\varepsilon, a_1, \dots, a_q\}$.

Dann gilt $L(\alpha_{i,i}^0) = L_{i,i}^0$ für $\alpha_{i,i}^0 = (\varepsilon | a_1 | \dots | a_q)$.

Beweis: Satz von Kleene (7)



Zur Erinnerung: $L_{i,j}^k$ enthält die Wörter, die M von Zustand z_i zu Zustand z_j führen ohne dabei Zwischenzustände mit Index größer als k zu benutzen.

Induktionsschritt: $k \rightarrow k + 1$

$$L_{i,j}^{k+1} = L_{i,j}^k \cup L_{i,k+1}^k (L_{k+1,k+1}^k)^* L_{k+1,j}^k,$$

denn entweder läuft M ohne Zustand z_{k+1} zu besuchen, oder der Lauf kann in 3 Teile gespalten werden:

- 1 Lauf von z_i bis zum ersten Besuch des Zustands z_{k+1} (abgedeckt durch $L_{i,k+1}^k$)
- 2 Mehrmaliges, zyklisches Besuchen von $k + 1$ (beliebig oft) (abgedeckt durch $L_{k+1,k+1}^k$)
- 3 Letztmaliges Verlassen von z_{k+1} und Lauf bis zu z_j (abgedeckt durch $L_{k+1,j}^k$)

Daher gilt $\alpha_{i,j}^{k+1} = (\alpha_{i,k+1}^k | \alpha_{k+1,k+1}^k (\alpha_{k+1,k+1}^k)^* \alpha_{k+1,j}^k)$ und $L(\alpha_{i,j}^{k+1}) = L_{i,j}^{k+1}$.

Beweis: Satz von Kleene (8)



Zur Erinnerung: $L_{i,j}^k$ enthält die Wörter, die M von Zustand z_i zu Zustand z_j führen ohne dabei Zwischenzustände mit Index größer als k zu benutzen.

Damit haben wir bewiesen:

Es gibt reguläre Ausdrücke $\alpha_{i,j}^k$ mit $L(\alpha_{i,j}^k) = L_{i,j}^k$

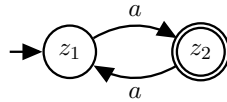
Jetzt müssen wir noch zeigen, dass es einen regulären Ausdruck gibt, der $L(M)$ erzeugt.

Sei die Menge der Endzustände $E = \{z_{i_1}, \dots, z_{i_r}\}$.

Dann gilt $L(\alpha_{1,i_1}^n | \alpha_{1,i_2}^n | \dots | \alpha_{1,i_r}^n) = \bigcup_{z_i \in E} L_{1,i}^n = L(M)$

□

Beispiel: DFA $\rightarrow$ regulärer Ausdruck



Regulärer Ausdruck dazu:

$$\begin{aligned}
 \alpha_{1,2}^2 &= (\alpha_{1,2}^1 | \alpha_{1,2}^1 (\alpha_{2,2}^1)^* \alpha_{2,2}^1) \\
 &= ((a|\varepsilon(\varepsilon)^*a) | (a|\varepsilon(\varepsilon)^*a)(\varepsilon|a(\varepsilon)^*a)^*(\varepsilon|a(\varepsilon)^*a)) \\
 &= (a|a(aa)^*) \text{ (durch Vereinfachung)}
 \end{aligned}$$

denn

$$\begin{aligned}
 \alpha_{1,1}^0 &= \varepsilon & \alpha_{2,2}^0 &= \varepsilon & \alpha_{1,2}^0 &= a & \alpha_{2,1}^0 &= a \\
 \alpha_{1,2}^1 &= (\alpha_{1,2}^0 | \alpha_{1,1}^0 (\alpha_{1,1}^0)^* \alpha_{1,2}^0) = (a|\varepsilon(\varepsilon)^*a) \\
 \alpha_{2,2}^1 &= (\alpha_{2,2}^0 | \alpha_{2,1}^0 (\alpha_{1,1}^0)^* \alpha_{1,2}^0) = (\varepsilon|a(\varepsilon)^*a)
 \end{aligned}$$

Zusammenfassung: Formalismen für reguläre Sprachen

