Pumping Lemma für reguläre Sprachen:

L DEA-Sprache $\Rightarrow$

$\exists\, N\in\mathbb{N} : \forall\, x\in L$ mit $|x|\geq N$: $\exists$ Unterteilung $x=uvw$ mit $|uv|\leq N$ und $|v|>0$: $\forall\, i\in\mathbb{N} : uv^iw\in L$

$\neg\big(\exists\, N\in\mathbb{N} : \forall\, x\in L$ mit $|x|\geq N$: $\exists$ Unterteilung $x=uvw$ mit $|uv|\leq N$ und $|v|>0$: $\forall\, i\in\mathbb{N} : uv^iw\in L\big)$

$\Rightarrow\quad \neg$ (L DEA-Sprache)

$\forall\, N\in\mathbb{N} : \exists\, x\in L$ mit $|x|\geq N$: $\forall$ Unterteilung $x=uvw$ mit $|uv|\leq N$ und $|v|>0$: $\exists\, i\in\mathbb{N} : uv^iw\notin L$

$\Rightarrow$ L ist **keine** DEA-Sprache

"Spiel" zum Zeigen, dass L keine DEA Sprache

1. Widersacher gibt eine Zahl $N\in\mathbb{N}$ vor.
2. Ich wähle ein $x\in L$ mit $|x|\geq N$.
3. Widersacher gibt eine Unterteilung $x = uvw$ vor mit $|uv|\leq N$, $|v|>0$.
4. Ich wähle ein $i\in\mathbb{N}$, sodass $uv^iw \notin L$.

$\forall\, N\in\mathbb{N} : \exists\, x\in L$ mit $|x|\geq N$: $\forall$ Unterteilung $x=uvw$ mit $|uv|\leq N$ und $|v|>0$: $\exists\, i\in\mathbb{N} : uv^iw \notin L$

$\Rightarrow$ L ist **keine** DEA-Sprache

"Spiel" zum Zeigen, dass L keine DEA Sprache

1. Widersacher gibt eine Zahl $N\in\mathbb{N}$ vor.
2. Ich wähle ein $x\in L$ mit $|x|\geq N$.
3. Widersacher gibt eine Unterteilung $x = uvw$ vor mit $|uv|\leq N$, $|v|>0$.
4. Ich wähle ein $i\in\mathbb{N}$, sodass $uv^iw \notin L$.

Bsp: $L = \{ a^p \mid p$ ist Primzahl $\}$ ist keine DEA Sprache

1. Widersacher gibt N
2. Ich wähle Primzahl $p>N$ und nehme $x=a^p \in L$
3. Widersacher unterteilt $x=uvw$ mit $|v|=k>0$, also $v=a^k$
4. $uv^iw = a^{(p-k)+ik}$, aber für $i=p+1$ ist das $a^{p(k+1)} \notin L$

Reguläre Ausdrücke

Mechanismus zum Spezifizieren von Sprachen über Alphabet Σ

	Ausdruck	Bedeutung
	$\emptyset$	$[\![\emptyset]\!] = \{\}$
	ε	$[\![\varepsilon]\!] = \{\varepsilon\}$
$a\in\Sigma$	a	$[\![a]\!] = \{a\}$
r_1, r_2 reg. Ausdrücke		
	(r_1+r_2)	$[\![(r_1+r_2)]\!] = [\![r_1]\!] \cup [\![r_2]\!]$
	$r_1\, r_2$	$[\![r_1\, r_2]\!] = [\![r_1]\!] \cdot [\![r_2]\!]$
	$(r_1)^*$	$[\![(r_1)^*]\!] = [\![r_1]\!]^*$

Reguläre Ausdrücke

Mechanismus zum Spezifizieren von Sprachen über Alphabet Σ

	Ausdruck	Bedeutung
	$\emptyset$	$[\![\emptyset]\!] = \{\}$
	ε	$[\![\varepsilon]\!] = \{\varepsilon\}$
$a \in \Sigma$	a	$[\![a]\!] = \{a\}$
r_1, r_2 reg. Ausdrücke		
	(r_1+r_2)	$[\![(r_1+r_2)]\!] = [\![r_1]\!] \cup [\![r_2]\!]$
	$r_1\, r_2$	$[\![r_1\, r_2]\!] = [\![r_1]\!] \cdot [\![r_2]\!]$
	$(r_1)^*$	$[\![(r_1)^*]\!] = [\![r_1]\!]^*$

Beispiel: alle Strings über $\{a,b\}$ ohne zwei direkt aufeinanderfolgende a's

$b^*(abb^*)^*(a+\varepsilon)$ oder auch $(ab+b+\varepsilon)^*(a+\varepsilon)$

Reguläre Ausdrücke definieren genau die DEA-Sprachen.

Satz: $L = [\![r]\!]$ für einen regulären Ausdruck r genau dann, wenn $L = L(M)$ für irgendeinen DEA M.

Deswegen heißen DEA-Sprachen üblicherweise ***reguläre Sprachen***.

Beweis: "$\Rightarrow$"

Idee: strukturelle Induktion; baue NEA mit ε-Übergängen

"$\Leftarrow$" Gegeben DEA $M=(\Sigma,Q,s,F,\Delta)$, Übergangsgraph G_M

$$Q = \{q_1,\cdots,q_n\},\ \ s = q_1,\ \ F = \{q_{j_1},\cdots,q_{j_f}\}$$

Für $0\le k\le n$, $1\le i,j\le n$ sei r^k_{ij} Ausdruck, der alle Beschriftungen von Pfaden in G_M beschreibt mit Anfangsknoten q_i, Endknoten q_j, und **internen** Knoten mit Index $\le k$.

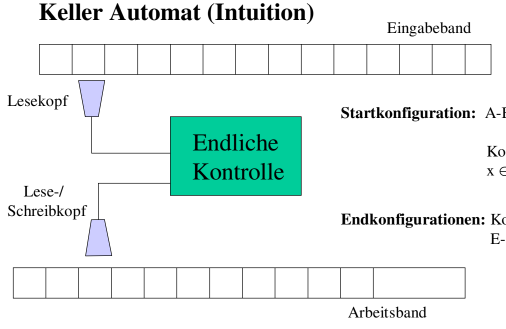
$$r^0_{ii} = \varepsilon + a_1 + \cdots a_t \quad \text{mit } (q_i,a_h,q_i)\in\Delta \text{ für } 1\le h\le t$$

$i\neq j$ $\qquad r^0_{ij} = a_1 + \cdots a_t \quad \text{mit } (q_i,a_h,q_j)\in\Delta \text{ für } 1\le h\le t$

$k>0$ $\qquad r^k_{ij} = r^{k-1}_{ij} + r^{k-1}_{ik}(r^{k-1}_{kk})^*\, r^{k-1}_{kj}$

$$L(M) = r^n_{1j_1} + r^n_{1j_2} + \cdots + r^n_{1j_f}$$

Keller Automat (Intuition)



Startkonfiguration: A-Band nur € auf linkester Zelle, dort auch A-Kopf
Kontrolle in Startzustand
$x \in \Sigma^*$ auf E-Band, E-Kopf ganz links,

Endkonfigurationen: Kontrolle in einem Endzustand
E-Kopf ganz rechts

Rechenschrittregeln:

derzeitiger Zustand, gelesenes A-Symbol, gelesenes E-Symbol $\longrightarrow$ schreibe auf A-Band, bewege E-Kopf nach rechts, neuer Zustand

Wenn A-Kopf nach links bewegt wird, hinterlässt er leere Zelle.

Formale Spezifikation von Kellerautomaten

Σ Eingabealphabet
Γ Kelleralphabet
$€\in\Gamma$ Kellerboden
Q Zustandsmenge (endlich)
$s\in Q$ Startzustand
$F\subset Q$ Endzustände

$\Delta \subset (Q\times\Gamma\times(\Sigma\cup\{\varepsilon\})) \times \Gamma^*\times Q$ Regeln (endlich)

derz. Zust. | derz. Keller-gipfel | konsumierte Eingabe | neuer Keller-gipfel | neuer Zustand

$(p , A , a , U , q) \in \Delta$ notiert durch

$p \xrightarrow[A/U]{a} q$

Formale Spezifikation von Kellerautomaten

Σ Eingabealphabet
Γ Kelleralphabet
$€\in\Gamma$ Kellerboden
Q Zustandsmenge (endlich)
$s\in Q$ Startzustand
$F\subset Q$ Endzustände

$\Delta \subset (Q\times\Gamma\times(\Sigma\cup\{\varepsilon\})) \times \Gamma^*\times Q$ Regeln (endlich)

derz. Zust. | derz. Keller-gipfel | konsumierte Eingabe | neuer Keller-gipfel | neuer Zustand

Konvention:

$a,b,c,.. \in \Sigma$ $A, B, C,... \in \Gamma$

$u,v,w,.. \in \Sigma^*$ $U, V, W,.. \in \Gamma^*$

$(p , A , a , U , q) \in \Delta$ notiert durch

$p \xrightarrow[A/U]{a} q$

Formale Spezifikation von Kellerautomaten

Σ Eingabealphabet
Γ Kelleralphabet
$€\in\Gamma$ Kellerboden
Q Zustandsmenge (endlich)
$s\in Q$ Startzustand
$F\subset Q$ Endzustände

Konvention:

$a,b,c,.. \in \Sigma$ $A, B, C,... \in \Gamma$

$u,v,w,.. \in \Sigma^*$ $U, V, W,.. \in \Gamma^*$

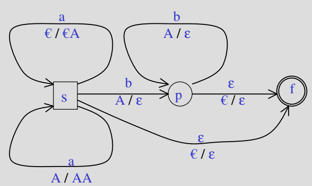
$\Delta \subset (Q\times\Gamma\times(\Sigma\cup\{\varepsilon\})) \times \Gamma^*\times Q$ Regeln (endlich)

derz. Zust. | derz. Keller-gipfel | konsumierte Eingabe | neuer Keller-gipfel | neuer Zustand

$(p , A , a , U , q) \in \Delta$ notiert durch

$p \xrightarrow[A/U]{a} q$

Bsp: NKA für $\{a^nb^n|n\in\mathbb{N}\}$



Konfiguration: $\Gamma^* \times Q \times \Sigma^*$

Kellerinhalt Zustand Eingaberest

Rechenschrittrelation:

$(WA , p , aw) \vdash_M (WU, q , w)$ g.d.w. $(p,A,a,U,q) \in \Delta$
$(WA , p , aw) \vdash_M (WU, q, aw)$ g.d.w. $(p,A,\varepsilon,U,q) \in \Delta$

$A\in\Gamma$, $W,U\in\Gamma^*$, $a\in\Sigma$, $w\in\Sigma^*$

Rechenrelation:

$\vdash_M^*$ reflexive, transitive Hülle von $\vdash_M$

Startkonfiguration für Eingabe x : $start_x = (\text{€} , s , x)$

Endkonfigurationen: $Fin = \{ (W , f , \varepsilon) \mid W \in \Gamma^* , f \in F \}$

(Akzeptanz durch Endzustand)

$Fin_\varepsilon = \{ (\varepsilon , q , \varepsilon) \mid q \in Q \}$

(Akzeptanz durch leeren Keller)

Von M akzeptierter Sprache:

$L(M) = \{ x \in \Sigma^* \mid start_x \vdash_M^* \varphi \text{ für irgendein } \varphi \in Fin \}$

$L_\varepsilon(M) = \{ x \in \Sigma^* \mid start_x \vdash_M^* \varphi \text{ für irgendein } \varphi \in Fin_\varepsilon \}$

Startkonfiguration für Eingabe x : $start_x = (\text{€} , s , x)$

Endkonfigurationen: $Fin = \{ (W , f , \varepsilon) \mid W \in \Gamma^* , f \in F \}$

(Akzeptanz durch Endzustand)

$Fin_\varepsilon = \{ (\varepsilon , q , \varepsilon) \mid q \in Q \}$

(Akzeptanz durch leeren Keller)

Von M akzeptierter Sprache:

$L(M) = \{ x \in \Sigma^* \mid start_x \vdash_M^* \varphi \text{ für irgendein } \varphi \in Fin \}$

$L_\varepsilon(M) = \{ x \in \Sigma^* \mid start_x \vdash_M^* \varphi \text{ für irgendein } \varphi \in Fin_\varepsilon \}$

Achtung: Um Wort x zu akzeptieren, muss das gesamte x konsumiert werden.

Konfiguration: $\underbrace{\Gamma^*}_{\text{Kellerinhalt}} \times \underbrace{Q}_{\text{Zustand}} \times \underbrace{\Sigma^*}_{\text{Eingaberest}}$

Rechenschrittrelation:

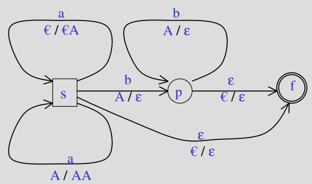
$(WA , p , aw) \vdash_M (WU, q , w)$ g.d.w. $(p,A,a,U,q) \in \Delta$
$(WA , p , aw) \vdash_M (WU, q, aw)$ g.d.w. $(p,A,\varepsilon,U,q) \in \Delta$

$A \in \Gamma$, $W,U \in \Gamma^*$, $a \in \Sigma$, $w \in \Sigma^*$

Rechenrelation:

$\vdash_M^*$ reflexive, transitive Hülle von $\vdash_M$

Bsp: NKA für $\{a^n b^n \mid n \in \mathbb{N}\}$



Akzeptierende Berechnung für aabb

$\underbrace{(\text{€} , s , aabb)}_{start_{aabb}} \vdash_M^* (\text{€}A , s , abb) \vdash_M^* (\text{€}AA , s , bb) \vdash_M^* (\text{€}A , p , b) \vdash_M^* (\text{€} , p , \varepsilon) \vdash_M^* (\varepsilon , f , \varepsilon)$

Satz: **$L=L(M)$ für irgendeinen NKA M**

⇔

$L=L_\varepsilon(M')$ für irgendeinen NKA M'

“Äquivalenz von Akzeptanz durch Endzustand und Akzeptanz durch leeren Keller.“

Def.: L heißt *NKA-Sprache* (oder *kontextfreie Sprache*), wenn $L=L(M)$ oder $L=L_\varepsilon(M)$ für irgendeinen NKA M.