Charakterizace regulárních jazyků, kanonické automaty

Levý kvocient jazyka L podle slova u: $\quad u\backslash L = \{v \mid uv \in L\}$.

Věta. Jazyk $L \subseteq \Sigma^*$ je regulární právě tehdy, když je množina kvocientů $\{w\backslash L \mid w \in \Sigma^*\}$ konečná.

Důkaz v jednom směru stojí na pozorování, že když pro konečný automat $A = (Q, \Sigma, \delta, q_0, F)$ máme $q_0 \xrightarrow{w} q$, tak $w\backslash L(A) = L_q$, kde definujeme $L_q = \{u \in \Sigma^* \mid q \xrightarrow{u} F\}$.

Pro druhou implikaci si všimneme, že pro $\{w\backslash L \mid w \in \Sigma^*\} = \{L_0, L_1, \ldots, L_k\}$ můžeme $L_0, L_1, \ldots, L_k$ chápat jako stavy konečného automatu, kde klademe $L_i \xrightarrow{a} L_j$ právě když $a\backslash L_i = L_j$. Dodáme-li jako počáteční stav $L = \varepsilon\backslash L$ (řekněme, že je to L_0) a označíme-li jako přijímající stavy ty L_i, které obsahují ε, dostáváme konečný automat A, pro nějž platí $L(A) = L$ (jak lze snadno ověřit).

Konečný automat je *kanonickým automatem pro jazyk* $L \subseteq \Sigma^*$, jestliže je izomorfní výše uvedenému automatu se stavovou množinou $\{w\backslash L \mid w \in \Sigma^*\}$.

Příklad izomorfních automatů

A	0	1
$\leftrightarrow(q_1, r_1)$	(q_1, r_1)	(q_2, r_2)
$\leftarrow(q_1, r_2)$	(q_1, r_3)	(q_2, r_2)
$\leftarrow(q_1, r_3)$	(q_1, r_1)	(q_2, r_4)
(q_1, r_4)	(q_1, r_4)	(q_2, r_4)
(q_2, r_1)	(q_3, r_1)	(q_1, r_2)
(q_2, r_2)	(q_3, r_3)	(q_1, r_2)
(q_2, r_3)	(q_3, r_1)	(q_1, r_4)
(q_2, r_4)	(q_3, r_4)	(q_1, r_4)
(q_3, r_1)	(q_2, r_1)	(q_3, r_2)
(q_3, r_2)	(q_2, r_3)	(q_3, r_2)
(q_3, r_3)	(q_2, r_1)	(q_3, r_4)
(q_3, r_4)	(q_2, r_4)	(q_3, r_4)

A'	0	1	
$\leftrightarrow s_1$	s_1	s_2	$s_1 = (q_1, r_1)$
s_2	s_3	s_4	$s_2 = (q_2, r_2)$
s_3	s_5	s_6	$s_3 = (q_3, r_3)$
$\leftarrow s_4$	s_7	s_2	$s_4 = (q_1, r_2)$
s_5	s_8	s_4	$s_5 = (q_2, r_1)$
s_6	s_9	s_6	$s_6 = (q_3, r_4)$
$\leftarrow s_7$	s_1	s_9	$s_7 = (q_1, r_3)$
s_8	s_5	s_{10}	$s_8 = (q_3, r_1)$
s_9	s_6	s_{11}	$s_9 = (q_2, r_4)$
s_{10}	s_{12}	s_{10}	$s_{10} = (q_3, r_2)$
s_{11}	s_{11}	s_9	$s_{11} = (q_1, r_4)$
s_{12}	s_8	s_{11}	$s_{12} = (q_2, r_3)$

Co je to ten izomorfismus?

Dva konečné automaty
$A_1 = (Q_1, \Sigma, \delta_1, q_{01}, F_1)$, $A_2 = (Q_2, \Sigma, \delta_2, q_{02}, F_2)$
jsou *izomorfní*, jestliže

existuje zobrazení (funkce) $f : Q_1 \to Q_2$ takové, že

- f je bijekce
  ("prosté": $q \neq q' \Rightarrow f(q) \neq f(q')$,
  "na": $\forall r \in Q_2 \, \exists q \in Q_1 : f(q) = r$)
- $f(q_{01}) = f(q_{02})$
- $q \in F_1 \iff f(q) \in F_2$
- $q \xrightarrow{a} q' \iff f(q) \xrightarrow{a} f(q')$
  neboli $f(\delta_1(q, a)) = \delta_2(f(q), a)$

Tedy: Izomorfní automaty A_1, A_2 jsou "stejné, až na pojmenování stavů".

Kanonické a minimální automaty

Konečný automat A je minimální, jestliže neexistuje automat A', který má méně stavů než A a přitom platí $L(A) = L(A')$.

Věta. Následující podmínky pro konečný automat A jsou ekvivalentní:

1. A je minimální.
2. A je kanonický (pro jazyk $L(A)$).

Odstranění nedosažitelných stavů

Tvrzení. Ke každému KA A lze zkonstruovat KA A', v němž každý stav je dosažitelný a $L(A') = L(A)$.

A	0	1
$\leftrightarrow s_1$	s_1	s_2
s_2	s_3	s_4
s_3	s_5	s_6
$\leftarrow s_4$	s_7	s_2
s_5	s_8	s_4
s_6	s_9	s_6
$\leftarrow s_7$	s_1	s_9
s_8	s_5	s_{10}
s_9	s_6	s_{11}
s_{10}	s_{12}	s_{10}
s_{11}	s_{11}	s_9
s_{12}	s_8	s_{11}

$1/\ q_0 \in \mathit{Reach}(A),\ \ 2/\ (q \in \mathit{Reach}(A) \wedge q \longrightarrow q'\ ) \Longrightarrow q' \in \mathit{Reach}(A).$

Připomenutí: relace ekvivalence

Relace ρ na množině M je
ekvivalence $\Leftrightarrow_{df}$ ρ je:

- reflexivní $(\forall x \in M : x \rho x)$,
- symetrická
  $(\forall x, y \in M : x \rho y \Rightarrow y \rho x)$,
- tranzitivní $(\forall x, y, z \in M :$
  $(x \rho y \ \wedge \ y \rho z) \Longrightarrow x \rho z)$.

Ekvivalence ρ definuje na M *rozklad*

$$\{ [x]_\rho \mid x \in M \}$$

tj. systém vzájemně disjunktních množin, zvaných třídy ekvivalence, jejichž sjednocením je M, kde

$$[x]_\rho = \{ y \mid x \rho y \}.$$

Podílový automat (faktor-automat)

K $A = (Q, \Sigma, \delta, q_0, F)$ definujme ekvivalenci $\sim$ na množině Q takto:

$$q \sim q' \iff_{df} L_q = L_{q'}$$

Podílový automat podle ekvivalence $\sim$, označený $A_\sim$, definujeme takto (píšeme stručněji $[q]$ místo $[q]_\sim$):
$A_\sim = (Q_\sim, \Sigma, \delta_\sim, [q_0], F_\sim)$, kde
$Q_\sim = \{\, [q] \mid q \in Q \,\}$,
$F_\sim = \{\, [q] \mid q \in F \,\}$
$\delta_\sim([q], a) = [\delta(q, a)]$
Korektnost definice (co to je?) plyne z toho, že
$p \sim q$ implikuje $\delta(p, a) \sim \delta(q, a)$.
Jak ukážeme, že $L(A) = L(A_\sim)$?
Stačí (induktivně dle délky slova) dokázat

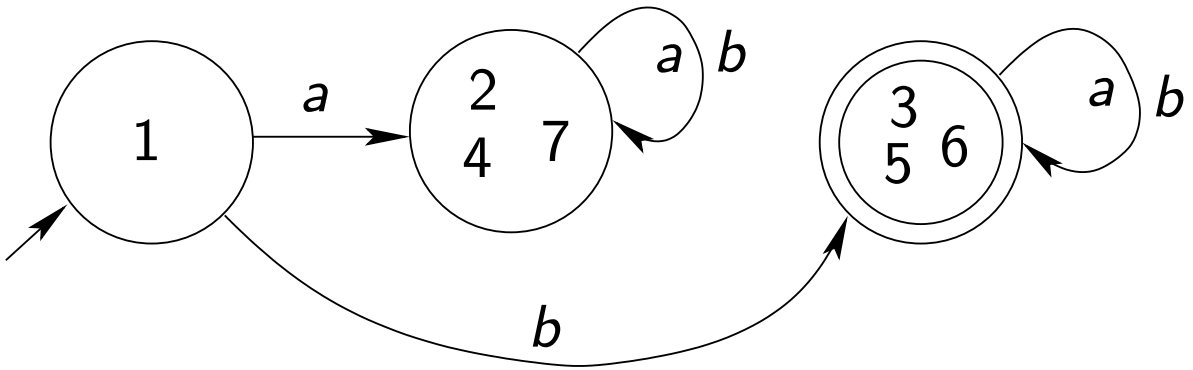
- $q \xrightarrow{w}_A q' \;\Rightarrow\; [q] \xrightarrow{w}_{A_\sim} [q']$
- $[q] \xrightarrow{w}_{A_\sim} [q'] \;\Rightarrow\; \exists r' : r' \sim q', q \xrightarrow{w}_A r'$

Příklad podílového automatu

	a	b
$\rightarrow$ 1	2	3
2	2	4
(3)	3	5
4	2	7
(5)	6	3
(6)	6	6
7	7	4

$$I = \{1\}$$
$$II = \{2, 4, 7\}$$
$$III = \{3, 5, 6\}$$

	a	b
$\rightarrow$ I	II	III
II	II	II
(III)	III	III



Jak ale zjistit, zda $L_q = L_{q'}$?

A	0	1
$\leftrightarrow s_1$	s_1	s_2
s_2	s_3	s_4
s_3	s_5	s_6
$\leftarrow s_4$	s_7	s_2
s_5	s_8	s_4
s_6	s_9	s_6
$\leftarrow s_7$	s_1	s_9
s_8	s_5	s_{10}
s_9	s_6	s_{11}
s_{10}	s_{12}	s_{10}
s_{11}	s_{11}	s_9
s_{12}	s_8	s_{11}

$\sim_0, \sim_1, \sim_2, \ldots$

Poznámka: koinduktivní přístup

Na naši konstrukci relace $\sim \subseteq Q \times Q$
definované vztahem

$$q \sim q' \Leftrightarrow L_q = L_{q'}$$

(pro automat $A = (Q, \Sigma, \delta, q_0, F)$), se lze dívat takto:

vezmeme (největší) relaci $R_0 = Q \times Q$,
postupně aplikujeme jistý monotónní funkcionál $\mathcal{F} : 2^{Q \times Q} \to 2^{Q \times Q}$
(pro $T_1 \subseteq T_2$ máme $\mathcal{F}(T_1) \subseteq \mathcal{F}(T_2)$)

dostáváme tak relace $R_0 \supseteq R_1 \supseteq R_2 \supseteq \cdots$
kde $R_1 = \mathcal{F}(R_0)$, $R_2 = \mathcal{F}(R_1)$, ...

až se dostaneme k (největšímu) pevnému bodu R (pro nějž $R = \mathcal{F}(R)$).

Rozhodování ekvivalence automatů

Věta. Existuje algoritmus, který pro zadané konečné automaty A_1, A_2 rozhodne, zda $L(A_1) = L(A_2)$.

Jedna idea algoritmu (umíte ji dotáhnout?):

$$\boxed{L(A_1) = L(A_2)} \iff \boxed{(L(A_1) - L(A_2)) \cup (L(A_2) - L(A_1)) = \emptyset}$$

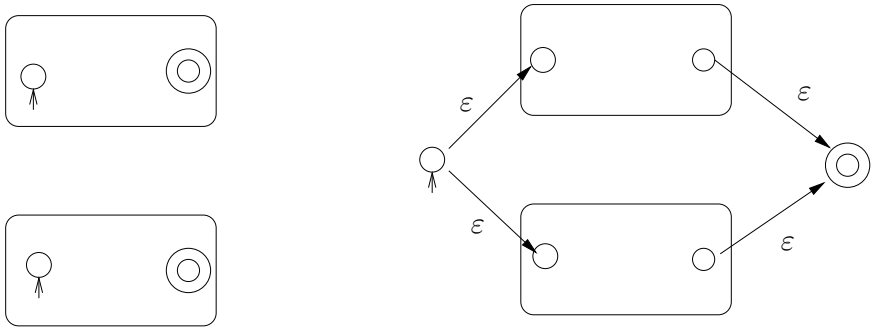
Máte jiný (lepší) nápad?
K A_1, A_2 zkonstruujeme ekvivalentní redukované automaty *v normovaném tvaru* a ty porovnáme.
(Když jsou stejné, tvrdíme $L(A_1) = L(A_2)$, jinak $L(A_1) \neq L(A_2)$.)
(Je to korektní??)

Věta. Následující podmínky pro konečný automat A jsou ekvivalentní:

1. A je minimální.
2. A je kanonický (pro jazyk $L(A)$).
3. A je redukovaný, tj. nemá nedosažitelné stavy a nemá různé stavy q, q' splňující $L_q = L_{q'}$.

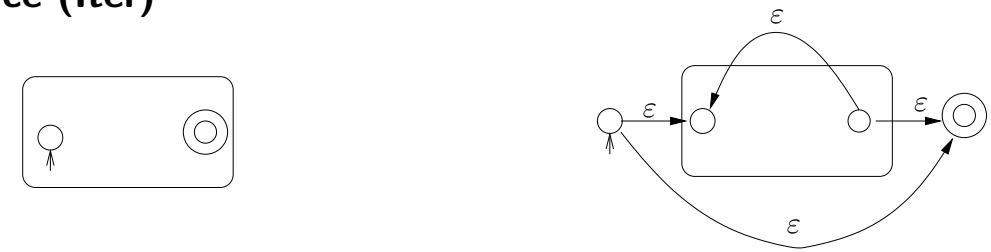
(Procedury) konstrukce ZNKA k regulárnímu výrazu

Sjednocení (Union)



Zřetězení (Conc)

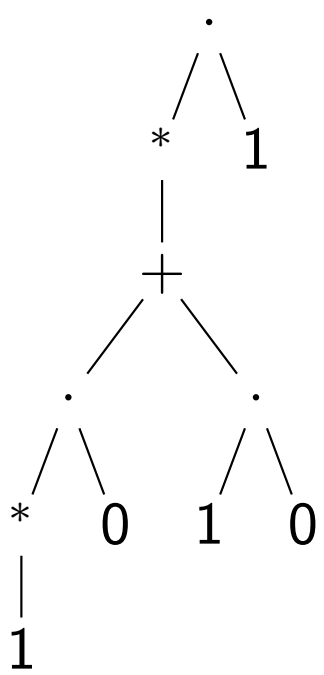
Iterace (Iter)

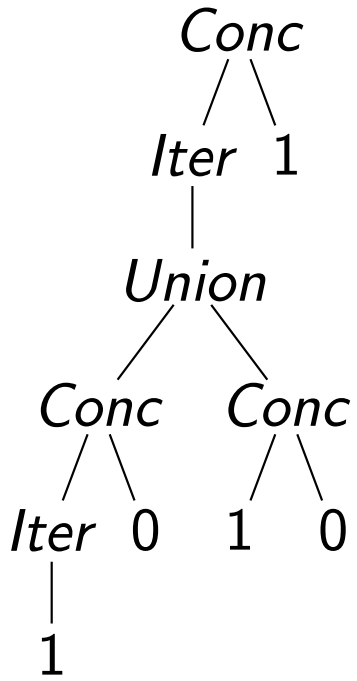


Syntaxí řízený překlad RV na ZNKA

regulární výraz $(1^*0 + 10)^*1$

syntaktický strom





lineární zápis

$$Conc(Iter(Union(Conc(Iter(1), 0), Conc(1, 0))), 1)$$

Bezkontextové gramatiky

Příklad: aritmetické výrazy v abecedě $\{ a, +, \times, (,) \}$
konkrétní řetězce: $a + a \times a$, $(a + a) \times a$
(a je zde atom, reprezentuje např. celé číslo)

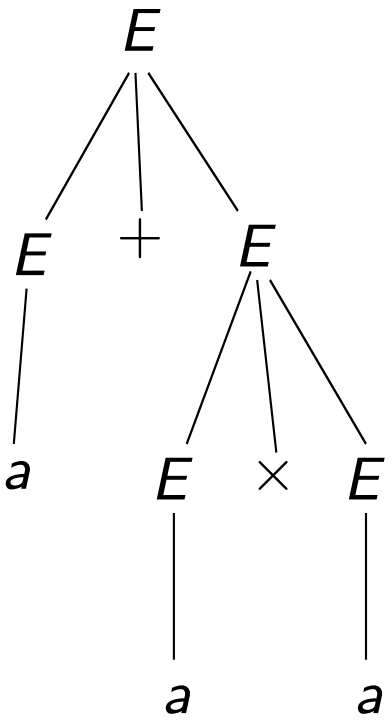
$$\langle EXPR \rangle \longrightarrow a$$
$$\langle EXPR \rangle \longrightarrow \langle EXPR \rangle + \langle EXPR \rangle$$
$$\langle EXPR \rangle \longrightarrow \langle EXPR \rangle \times \langle EXPR \rangle$$
$$\langle EXPR \rangle \longrightarrow (\langle EXPR \rangle)$$

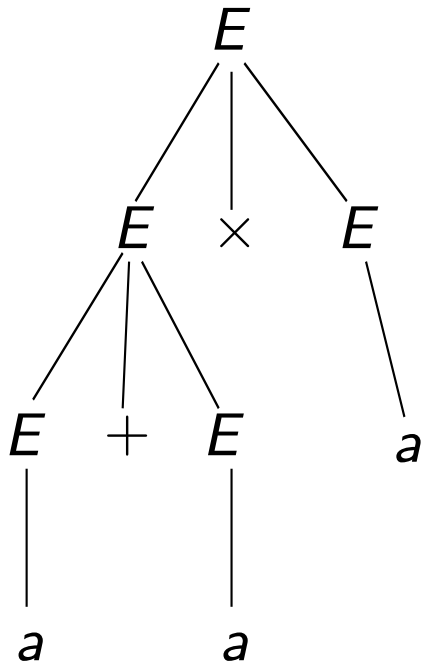
$$E \longrightarrow a \mid E + E \mid E \times E \mid (E)$$

Odvození (*derivace*), slova $a + a \times a$:
$E \Rightarrow E + E \Rightarrow a + E \Rightarrow a + E \times E \Rightarrow a + a \times E \Rightarrow a + a \times a$
levá derivace, *pravá* derivace ...
$E \Rightarrow E + E \Rightarrow E + E \times E \Rightarrow E + E \times a \Rightarrow E + a \times a \Rightarrow a + a \times a$
A ještě příklad derivace, která není ani levá ani pravá:
$E \Rightarrow E + E \Rightarrow E + E \times E \Rightarrow E + a \times E \Rightarrow a + a \times E \Rightarrow a + a \times a$

Derivační strom

strom odvození (*derivační strom*)





Těmto různým stromům odpovídají různé *levé* derivace:

$E \Rightarrow E + E \Rightarrow a + E \Rightarrow a + E \times E \Rightarrow a + a \times E \Rightarrow a + a \times a$

$E \Rightarrow E \times E \Rightarrow E + E \times E \Rightarrow a + E \times E \Rightarrow a + a \times E \Rightarrow a + a \times a$

Bezkontextová gramatika - formálně

Bezkontextová gramatika
$G = (\Pi, \Sigma, S, P)$:
Π ... konečná množina *neterminálů*,
Σ ... konečná množina *terminálů* $\quad (\Pi \cap \Sigma = \emptyset)$,
$S \in \Pi$ ... *počáteční neterminál*
P ... konečná množina pravidel typu

$$A \to \beta, \quad kde \quad A \in \Pi,\ \beta \in (\Pi \cup \Sigma)^*.$$

Relace $\Rightarrow$ (resp. $\Rightarrow_G$) na $(\Pi \cup \Sigma)^*$:

$$\gamma \Rightarrow \delta$$

jestliže $(\exists \mu_1, \mu_2, A, \beta :)$

$$\gamma = \mu_1 A \mu_2,\ \delta = \mu_1 \beta \mu_2,\ (A \to \beta) \in P$$

Bezkontextové jazyky

$\Rightarrow^*$ ... reflexivní a tranzitivní uzávěr relace $\Rightarrow$
(nejmenší relace, která obsahuje $\Rightarrow$ a je reflexivní a tranzitivní)
Tedy $\gamma \Rightarrow^* \delta \iff$ ex. posloupnost $\mu_0, \mu_1, \ldots, \mu_n$ tak, že
$\gamma = \mu_0 \Rightarrow \mu_1 \Rightarrow \cdots \Rightarrow \mu_n = \delta$.
(*Derivace* (délky n) slova δ ze slova γ.)

Jazyk generovaný gramatikou G: $L(G) = \{ w \in \Sigma^* \mid S \Rightarrow^* w \}$

Jazyk L je *bezkontextový* $\Leftrightarrow_{df}$ ex. BG G tak, že $L(G) = L$.

Značení:
$a, b, c, \ldots$ ... (proměnné, jejichž hodnoty jsou) terminály
$u, v, w, \ldots$ ... řetězce terminálů
$A, B, C, \ldots, X, Y, Z$ ... neterminály
$\alpha, \beta, \gamma, \ldots$ ... řetězce neterminálů a terminálů (prvky z $(\Pi \cup \Sigma)^*$)

Levá derivace

Pro $G = (\Pi, \Sigma, S, P)$

$\alpha \Rightarrow^L \beta$ (β vznikne z α *levým přepsáním*) $\Leftrightarrow_{df}$

$\exists u \in \Sigma^*, \delta \in (\Pi \cup \Sigma)^*, (X \to \gamma) \in P$: $\alpha = uX\delta$, $\beta = u\gamma\delta$

$\alpha_0 \Rightarrow \alpha_1 \Rightarrow \cdots \Rightarrow \alpha_n$ je *levou derivací* $\Leftrightarrow_{df}$ $\alpha_i \Rightarrow^L \alpha_{i+1}$ $(0 \leq i \leq n-1)$.

Derivační strom - formálně

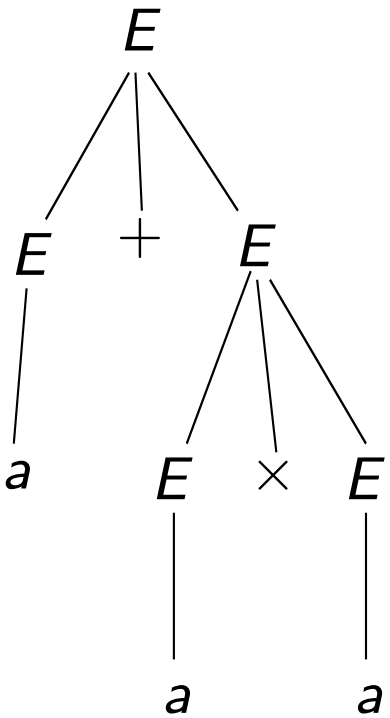
Derivační strom (vztahující se ke $G = (\Pi, \Sigma, S, P)$), je uspořádaný kořenový strom, v němž
- vrcholy ohodnoceny prvky $\Pi \cup \Sigma$,
- kořen ohodnocen S,
- vrchol ohodnocený $X\,(\in \Pi)$ má následníky ohodnocené $Y_1, Y_2, \ldots, Y_n$ $(Y_i \in \Pi \cup \Sigma)$, kde $(X \to Y_1 Y_2 \ldots Y_n) \in P$,
- vrchol ohodnocený $a\,(\in \Sigma)$ je listem.

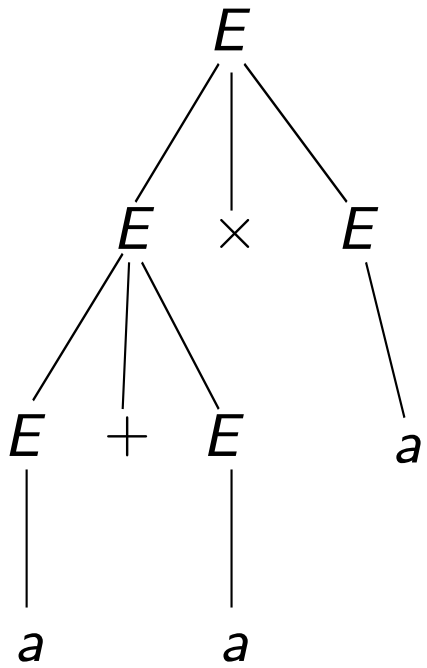
Derivační strom pro $w = a_1 a_2 \ldots a_n$ ... má listy zleva doprava ohodnoceny $a_1, a_2, \ldots, a_n$.

Věta. Každému derivačnímu stromu pro slovo w (v dané gramatice G) přirozeně odpovídá právě jedná levá derivace slova w (vyvozená z levého průchodu stromem); naopak každé levé derivaci slova w přirozeně odpovídá právě jeden derivační strom pro w.

Nejednoznačné gramatiky

Existence dvou různých derivačních stromů (levých derivací) pro jedno a totéž slovo ... pro překlad (vyhodnocení sémantiky) závada!
Připomeňme si





u gramatiky

$$E \longrightarrow a \mid E + E \mid E \times E \mid (E)$$

Jednoznačné gramatiky a jazyky

BG G je *jednoznačná* $\Leftrightarrow_{df}$ každé slovo z $L(G)$ má právě jeden derivační strom (tj. právě jednu levou derivaci).
V opačném případě je G *nejednoznačná* (či *víceznačná*).

Bezkontextový jazyk L je *jednoznačný* $\Leftrightarrow_{df}$ ex. jednoznačná G tž. $L(G) = L$; jinak se L nazývá (*vnitřně*) *nejednoznačný* (*víceznačný*).

Např.:
$L_1 = \{ a^n b^n \mid n \geq 0 \}$: $\quad S \longrightarrow aSb \mid \varepsilon$ (je jednoznačný)

$L_2 = \{ a^i b^j c^k \mid (i = j) \vee (j = k) \}$:

$$S \longrightarrow S_1 C \mid A S_2$$

$$S_1 \longrightarrow a S_1 b \mid \varepsilon \qquad S_2 \longrightarrow b S_2 c \mid \varepsilon$$
$$C \longrightarrow c\, C \mid \varepsilon \qquad A \longrightarrow aA \mid \varepsilon$$

Fakt: Neex. jednoznačná BG G tž. $L(G) = L_2$. (L_2 je víceznačný.)

Pozn.: problém jednoznačnosti bezkontextové gramatiky je algoritmicky nerozhodnutelný (ukážeme později ...).

Jednoznačné gramatiky a jazyky

K (nejednoznačné) gramatice

$$R \longrightarrow a \mid b \mid R + R \mid RR \mid R^* \mid (R)$$

lze sestrojit ekvivalentní gramatiku, která je jednoznačná:

$$\begin{aligned} R &\longrightarrow T + R \mid T \\ T &\longrightarrow FT \mid F \\ F &\longrightarrow F^* \mid (R) \mid C \\ C &\longrightarrow \ a \mid b \end{aligned}$$