456-330/1: Teoretická informatika (TI)
přednáška 4

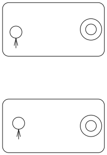
prof. RNDr Petr Jančar, CSc.

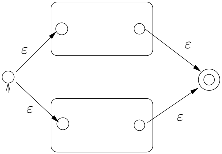
katedra informatiky FEI VŠB-TUO
www.cs.vsb.cz/jancar

LS 2009/2010

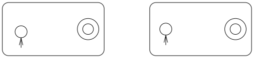
(Procedury) konstrukce ZNKA k regulárnímu výrazu

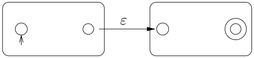
Sjednocení (Union)



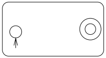


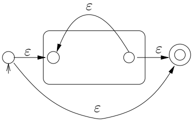
Zřetězení (Conc)





Iterace (Iter)

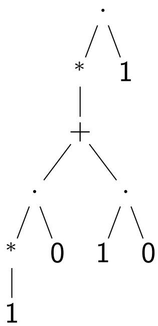


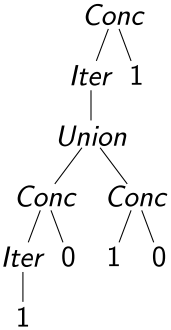


Syntaxí řízený překlad RV na ZNKA

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)

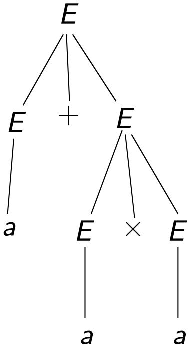
$$
\begin{aligned}
\langle EXPR \rangle &\longrightarrow a \\
\langle EXPR \rangle &\longrightarrow \langle EXPR \rangle + \langle EXPR \rangle \\
\langle EXPR \rangle &\longrightarrow \langle EXPR \rangle \times \langle EXPR \rangle \\
\langle EXPR \rangle &\longrightarrow (\langle EXPR \rangle)
\end{aligned}
$$

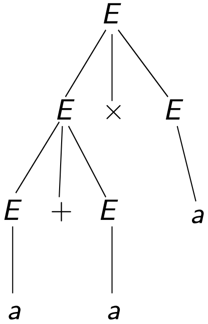
$$E \longrightarrow a \mid E + E \mid E \times E \mid (E)$$

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

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

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

$$\gamma \Rightarrow \delta$$

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

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

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*) $\Leftrightarrow_{df}$
$\exists u \in \Sigma^*$, $\delta \in (\Pi \cup \Sigma)^*$, $(X \rightarrow \gamma) \in P$: $\alpha = uX\delta$, $\beta = u\gamma\delta$

$\alpha_0 \Rightarrow \alpha_1 \Rightarrow \cdots \Rightarrow \alpha_n$ 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 $(X \rightarrow Y_1 Y_2 \ldots Y_n) \in P$,
- vrchol ohodnocený $a\,(\in \Sigma)$ je listem.

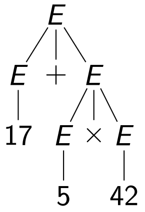
Derivační strom pro $w = a_1 a_2 \ldots a_n$ ... má listy zleva doprava ohodnoceny $a_1, a_2, \ldots, a_n$.

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

Syntaxe, sémantika

Gramatika slouží nejen pro popis syntaxe jazyka; syntaktická struktura (derivační strom) konkrétního slova je obvykle základem také pro vyhodnocení sémantiky (významu) slova.

Tím významem může být např. hodnota z nějaké množiny (domény). Např. u dříve uvedených aritmetických výrazů je to (celé) číslo či program (cílový kód) k jeho výpočtu … (interpret, kompilátor …)



Překlad RV na (ZN)KA na základě derivačního stromu BG

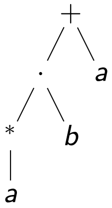
Omezme se na abecedu $\Sigma = \{a, b\}$. Množina regulárních výrazů nad Σ je dána např. následující gramatikou (symboly $\emptyset$, ϵ zde neuvažujeme)

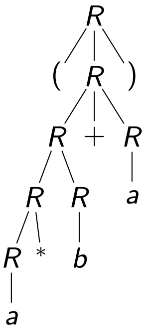
$$R \longrightarrow a \mid b \mid R + R \mid R \cdot R \mid RR \mid R^* \mid (R)$$

regulární výraz $\qquad (a^* b + a)$

derivační strom

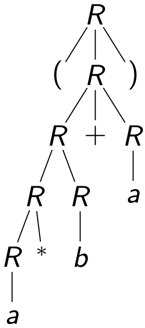
syntaktický strom





Konstrukce (ZN)KA na základě derivačního stromu

derivační strom

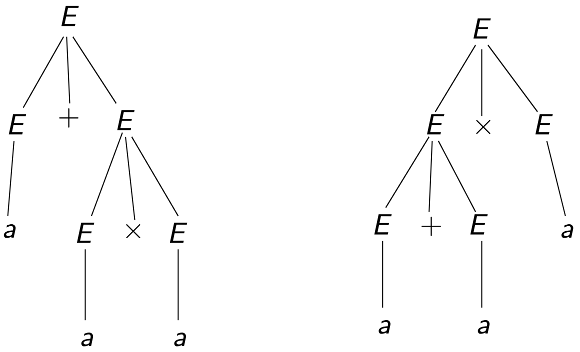


pro vrchol označený

- a, b: vydej příslušný (elementární) automat;
- R s následníky $R, +, R$: sestav automat A_1 pro 1. následníka, A_2 pro 3. následníka, a pak sestav a vydej automat A pro $L(A_1) \cup L(A_2)$;
- R s následníky $R, \cdot, R$ (podobně pro R, R): sestav automat A_1 pro 1. následníka, A_2 pro 3. následníka, a pak sestav a vydej automat A pro $L(A_1) \cdot L(A_2)$;
- R s následníky $R, ^*$: sestav automat A_1 pro 1. následníka, a pak sestav a vydej automat A pro $L(A_1)^*$;
- R s následníky $(, R,)$: vydej automat sestavený pro 2. následníka.

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)

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

$$S \longrightarrow S_1 C \mid AS_2$$

$$S_1 \longrightarrow aS_1 b \mid \varepsilon \qquad S_2 \longrightarrow bS_2 c \mid \varepsilon$$

$$C \longrightarrow c\, C \mid \varepsilon \qquad A \longrightarrow aA \mid \varepsilon$$

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:

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

Gramatika pro booleovské formule (cvičení)

Uvažujme jazyk sestávající ze všech booleovských formulí s proměnnými $x1, x2, \ldots$ a logickými spojkami $\neg, \wedge, \vee$; mohou se v nich používat závorky $(,)$, ale není nutné plně závorkovat. Každá taková formule je tedy řetězcem v abecedě

$$\Sigma = \{ x, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, \neg, \wedge, \vee, (,) \} ;$$

jako příklad může sloužit řetězec $(\neg x15 \vee x2 \wedge x5) \wedge \neg x21 \vee \neg(x2 \vee x5)$, který do jazyka patří. (Samozřejmě zde můžeme preferovat přehlednější zápis $(\neg x_{15} \vee x_2 \wedge x_5) \wedge \neg x_{21} \vee \neg(x_2 \vee x_5)$, ale to není podstatné.)
Navrhněte co nejjednodušší bezkontextovou gramatiku generující uvedený jazyk.
Takto navržená (jednoduchá) gramatika asi není jednoznačná; ověřte.
Zkonstruujte pak pro stejný jazyk jednoznačnou gramatiku, u níž derivační stromy přirozeně odpovídají obvyklé prioritě operátorů: negace váže silněji než konjunkce a konjunkce váže silněji než disjunkce.