AUTOMATY A GRAMATIKY

6

Pavel Surynek

Univerzita Karlova v Praze

Matematicko-fyzikální fakulta
Katedra teoretické informatiky a matematické logiky

Regulární výrazy

- Kleeneho věta poskytuje alternativní popis regulárních jazyků nad danou abecedou X
  - umožňuje zavedení **regulárních výrazů**, což jsou slova nad abecedou X∪{∅, λ ,+, ., *, (,)} vytvořené podle následujících pravidel:
    - předpokládáme, že {∅, λ ,+, ., *, (,)}∩X=∅
      - (i) ∅ a λ jsou regulární výrazy
      - (ii) x je regulární výraz pro každé x∈X
      - (iii) (α+β) je regulární výraz, když α, β jsou regulární výrazy
      - (iv) (α.β) je regulární výraz, když α, β jsou regulární výrazy
      - (v) α* je regulární výraz, když α je regulární výraz
      - (vi) každý regulární výraz vznikne konečným použitím pravidel (i)-(v)
  - regulární výraz α **reprezentuje** jazyk [α], kde
    - [∅]=∅, [λ]={λ}, [x]={x} pro x∈X
    - [(α+β)]=[α]∪[β]
    - [(α.β)]=[α].[β]
    - [α*]=[α]*
  - z Kleeneho věty vidíme, že reprezentovaný jazyk je regulární a naopak libovolný regulární jazyk lze reprezentovat nějakým regulárním výrazem

Př: X = {a,b,c,d}
a(bc)*a+cd
(ab)*(cd)*+ca

Př: [a(bc)*a+cd] =
[a(bc)*a]∪[cd] =
{a}.[bc]*.{a} ∪{cd} =
{aa, abca,abcbca, ...
abcbc...bca, cd}

Regulární **výraz** ⇒konečný **automat**

- regulárním jazykem **není** reprezentující regulární výraz určen **jednoznačně**
  - chceme rozpoznávat, zda dvojice regulárních výrazů reprezentuje stejný jazyk, tj. zda jsou *ekvivalentní*
    - převodem na konečný automat A=(QU{q_0}, X, δ, {q_0}, F)
      - regulární výrazy umožňují úspornou reprezentaci jazyka ⇒ rozhodnutí o ekvivalenci regulárních výrazů je PSPACE-úplný problém
    - 1. očíslujeme symboly v regulárním výrazu
      - množina stavů Q
    - 2. zjistíme, které očíslované symboly mohou stát **na začátku** reprezentovaného slova
      - počáteční stavy
    - 3. zjistíme, které dvojice očíslovaných symbolů mohou v reprezentovaném slově stát **vedle sebe**
      - přechodová funkce
    - 4. zjistíme, které očíslované symboly mohou stát **na konci** reprezentovaného slova
      - přijímající stavy
    - 5. speciálně ošetřit případ, kdy λ je reprezentováno regulárním výrazem

Př: X = {a,b,c,d}
a(bc)*a+cd
$a_1(b_2c_3)^*a_4+c_5d_6$
Q={a_1, b_2, c_3, a_4, c_5,d_6}

Př: začátek
{a_1, c_5}

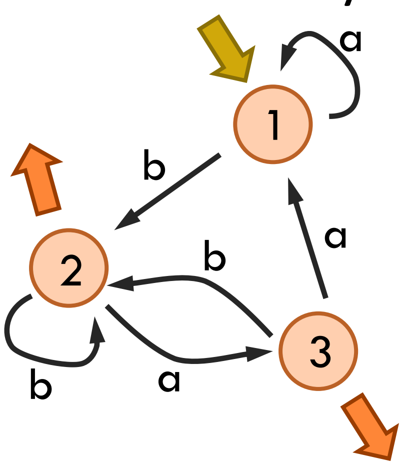
Př: sousedství
{[a_1,b_2], [b_2,c_3], [c_3,b_2], [c_3,a_4], [a_1,a_4], [c_5,d_6]}

Př: konec
{a_4, d_6}

Př: λ reprezentováno není
q_0 nebude v F

Konečný **automat** ⇒ regulární **výraz**

- mějme KA $A=(Q, X, \delta, q_1,F)$, kde $Q = \{q_1,q_2, \dots, q_n\}$
  - využijeme důkaz Kleeneho věty, tj. zkonstruujeme regulární výrazy pro $L_{i,j}$
    - induktivně podle: $L_{i,j}^{k+1} = L_{i,j}^{k} \cup L_{i,k+1}^{k}.(L_{k+1,k+1}^{k})^*. L_{k+1,j}^{k}$
    - výsledný regulární výraz sestavíme (aditivně) z regulárních výrazů pro $L_{1,i}$, kde $q_i \in F$



(a+b)*b + (a+b)*ba

$L_{i,j}^0$	1	2	3
1	a+λ	b	∅
2	∅	b+λ	a
3	a	b	λ

$L_{i,j}^1$	1	2	3
1	a*	a*b	∅
2	∅	b+λ	a
3	aa*	a*b	λ

$L_{i,j}^2$	1	2	3
1	a*	a*bb*	a*bb*a
2	∅	b*	b*a
3	aa*	a*bb*	a*bb*a+λ

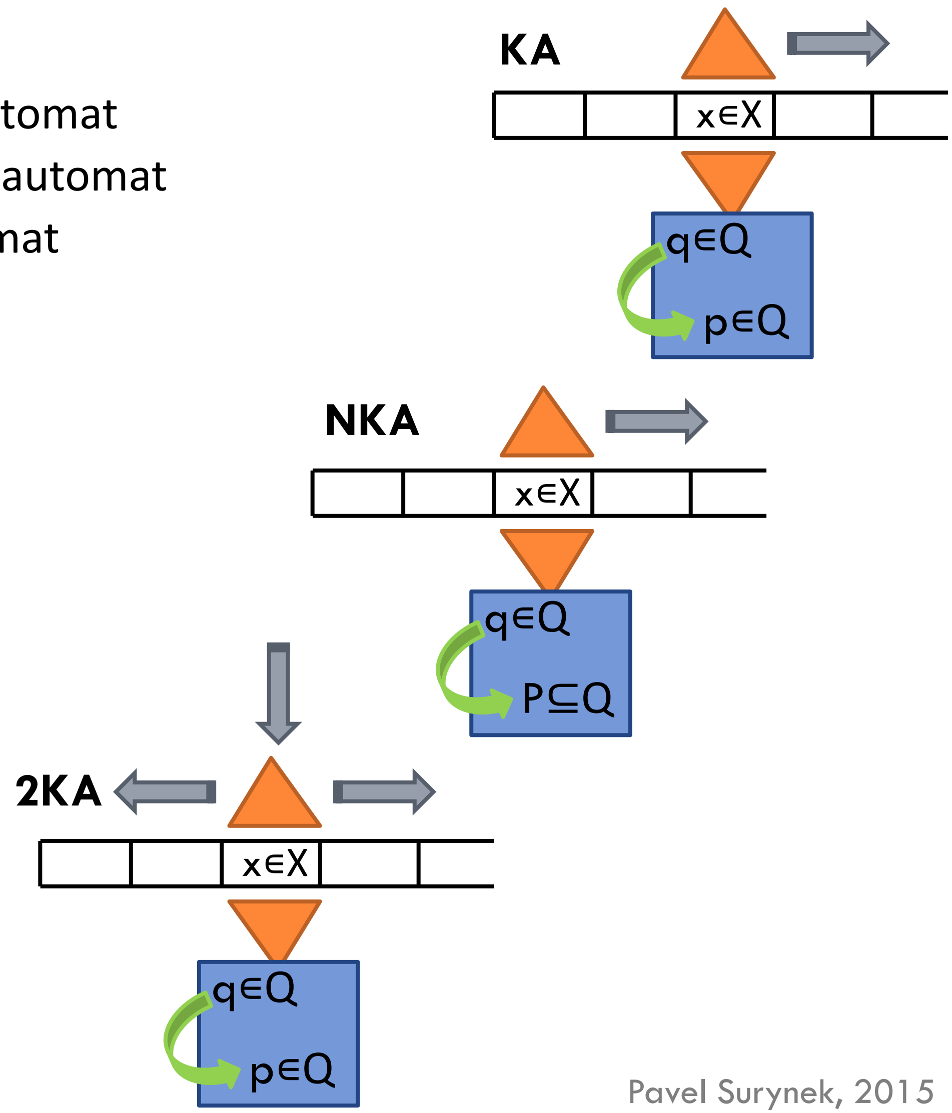
$L_{i,j}^3$	1	2	3
1	_	(a+b)*b	(a+b)*ba
2	_	_	_
3	_	_	_

Regularita regulární substituce

- snadno ukážeme, že $f(L)$, resp. $h(L)$ je regulární jazyk
  - pro f regulární substituci, resp. h homomorfismus a L regulární jazyk
    - připomenutí: $f: X \rightarrow 2^{Y^*}$, $h: X \rightarrow 2^{Y^*}$
      - $f(x)$ je regulární pro $x \in X$, $|h(x)|=1$ pro $x \in X$ (tedy $h(x)$ také regulární)
  - pro $x \in X$ existuje regulární výraz R_x, že $[R_x]=f(x)$
  - existuje regulární výraz R_L, že $[R_L]=L$
    - nechť $f(R_L)$ je regulární výraz vzniklý z R_L tak, že výskyt symbolu $x \in X$ v R_L je nahrazen výrazem R_x
    - $[f(R_L)]=f(L)$
      - indukcí podle složitosti R_L
        - $R_L=\emptyset$, pak $L=\emptyset$, $[f(\emptyset)]=f(\emptyset)$; $R_L=\lambda$, pak $L=\{\lambda\}$, $[f(\lambda)]=f(\{\lambda\})$
        - $R_L=x$ pro $x \in X$, pak $L=\{x\}$, $[f(x)]=[R_x]=f(\{x\})$
        - $R_L=(\alpha+\beta)$, přičemž víme, že $[f(\alpha)]=f([\alpha])$ a $[f(\beta)]=f([\beta])$
          - pak $[f(\alpha+\beta)]=[f(\alpha)+f(\beta)] =[f(\alpha)]\cup[f(\beta)]= f([\alpha])\cup f([\beta])=f([\alpha]\cup[\beta])=f([\alpha+\beta])$
        - $R_L=(\alpha.\beta)$, přičemž víme, že $[f(\alpha)]=f([\alpha])$ a $[f(\beta)]=f([\beta])$
          - pak $[f(\alpha.\beta)]=[f(\alpha).f(\beta)] =[f(\alpha)].[f(\beta)]= f([\alpha]).f([\beta])=f([\alpha].[\beta])=f([\alpha.\beta])$
        - $R_L=\alpha^*$, přičemž víme, že $[f(\alpha)]=f([\alpha])$
          - pak $[f(\alpha^*)]=[f(\alpha)^*] =[f(\alpha)]^*= f([\alpha])^*=f([\alpha]^*)=f([\alpha^*])$

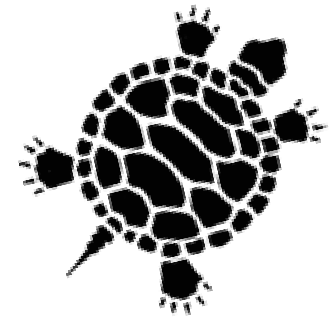
Konečné automaty shrnutí

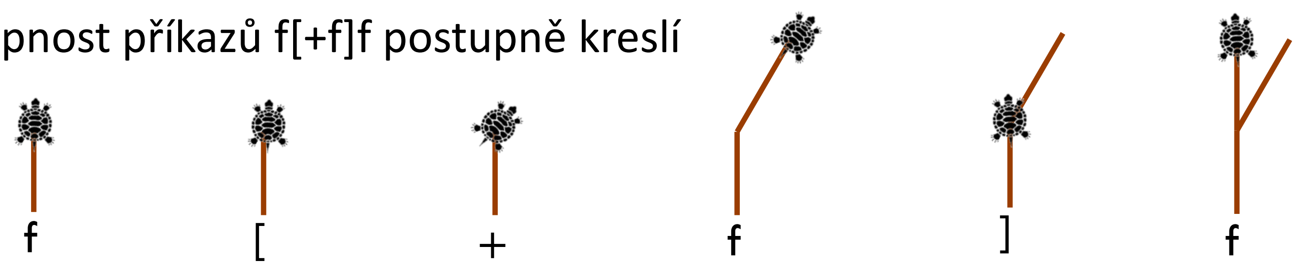
- **Varianty** konečných automatů
  - (D)KA deterministický konečný automat
  - NKA nedeterministický konečný automat
  - 2KA dvousměrný konečný automat
    - (D)2KA deterministický
    - N2KA nedeterministický
  - přijímají právě regulární jazyky
- **Regulární výrazy**
  - alternativní popis regulárních jazyků
- **Uzávěrové** vlastnosti
  - ∩,∪, . , *, kvocienty, substituce, ...
- **Charakterizace** regulárních jazyků
  - Myhill-Nerodova věta
  - pumping (iterační) lemma
  - Kleeneho věta



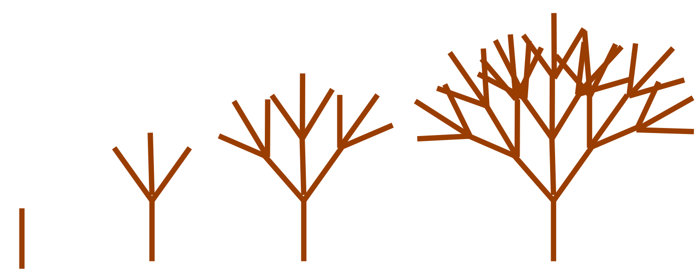
Grafická motivace ke **gramatikám**

- motivace **želví grafikou**
  - želva kreslí čáru
  - přijímá příkazy
    - f vpřed
    - \+ otočit po směru hodinových ručiček o 30°
    - \- otočit proti směru hodinových ručiček o 30°
    - [zapamatovat stav na zásobník (*push*)
    -] vyzvednout stav ze zásobníku a vrátit se do něj (*pop*)
  - posloupnost příkazů f[+f]f postupně kreslí





  - systém přepisovacích pravidel
    - začni s F
    - F přepiš na F[+F][-F]F, F přepiš na f
      - vytvoří posloupnosti příkazů pro kreslení stromů



Gramatiky formálně

- **gramatika** je čtveřice $G = (V_N, V_T, S, P)$
  - V_N - konečná neprázdná množina **neterminálních** symbolů
    - *neterminály*
  - V_T - konečná neprázdná množina **terminálních** symbolů
    - *terminály*
  - $S \in V_N$ - **počáteční** neterminál
  - P - konečná množina **přepisovacích** pravidel
    - pravidla jsou tvaru $u \rightarrow v$, kde $u,v \in (V_N \cup V_T)^*$ a u obsahuje <u>aspoň jeden neterminál</u>
- slovo $w \in (V_N \cup V_T)^*$ lze vzhledem ke gramatice G (přímo) ***přepsat*** na slovo $z \in (V_N \cup V_T)^*$
  - značíme $w \Rightarrow_G z$, jestliže existují $w_1, w_2 \in (V_N \cup V_T)^*$ a $u \rightarrow v \in P$, že $w = w_1 u w_2$ a $z = w_1 v w_2$
- slovo $z \in (V_N \cup V_T)^*$ lze vzhledem ke gramatice G ***odvodit*** ze slova $w \in (V_N \cup V_T)^*$
  - značíme $w \Rightarrow_G^* z$, jestliže existují $w_1, w_2, \ldots, w_n \in (V_N \cup V_T)^*$, kde $n \in \mathbb{N}$, taková, že
    - $w_1 = w$, $w_n = z$ a $w_i \Rightarrow_G w_{i+1}$ pro $i = 1, 2, \ldots, n-1$
  - posloupnost $w = w_1, w_2, \ldots, w_n = z$ se nazývá **odvozením** (*derivací*) slova z ze slova w vzhledem ke gramatice G

Př.: $G = (V_N, V_T, S, P)$, kde
$V_N = \{ F \}$
$V_T = \{ f, +, -, [,] \}$
$S = F$
$P = \{ F \rightarrow F[+F][-F]F,$
$F \rightarrow f \}$

Pozn.: *nedeterminismus* ve volbě pravidel

Jazyky a gramatiky

- $G = (V_N, V_T, S, P)$
  - jazyk L(G) generovaný gramatikou G
    - $L(G) = \{w \mid w \in V_T^* \text{ a } S \Rightarrow_G^* w\}$
  - $G_1 = (V_N^1, V_T, S_1, P_1)$ a $G_2 = (V_N^2, V_T, S_2, P_2)$ jsou **ekvivalentní**, jestliže $L(G_1) = L(G_2)$

Př.: $G = (V_N, V_T, S, P)$, kde
$V_N = \{ S \}$
$V_T = \{ 0, 1 \}$
$P = \{ S \to 0S1,$
$S \to \lambda \}$

$L(G) = \{ 0^i 1^i \mid i=0,1,2,\ldots \}$

Př.: $G = (V_N, V_T, V, P)$, kde
$V_N = \{ V, T, F \}$
$V_T = \{ 0, 1, (,), +, * \}$
$P = \{ V \to T + V \mid T,\ T \to F * V \mid F,\ F \to (V) \mid 1 \mid 0 \}$

L(G) = { jednoduché aritmetické výrazy nad konstantami 0 a 1 }

$0 + 1*0 + 1 \in L(G)$
$V \Rightarrow_G T + V \Rightarrow_G T + T + V \Rightarrow_G T + T + T \Rightarrow_G F + T + T \Rightarrow_G 0 + T + T \Rightarrow_G$
$0 + F * V + T \Rightarrow_G 0 + 1 * V + T \Rightarrow_G 0 + 1 * T + T \Rightarrow_G 0 + 1 * F + T$
$\Rightarrow_G 0 + 1 * 0 + T \Rightarrow_G 0 + 1 * 0 + F \Rightarrow_G \underline{0 + 1 * 0 + 1}$

Př.: $G = (V_N, V_T, S, P)$, kde
$V_N = \{ S, A, B, D \}$
$V_T = \{ a, b, c \}$
$P = \{ S \to \lambda \mid abc \mid ABSc$
$\mid ADc$
$BA \to AB,$
$BD \to Db,$
$AD \to ab,$
$Aa \to aa \}$

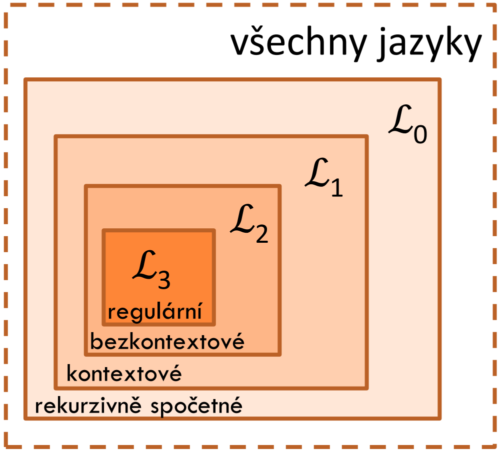
$L(G) = \{ a^i b^i c^i \mid i=0,1,2,\ldots \}$

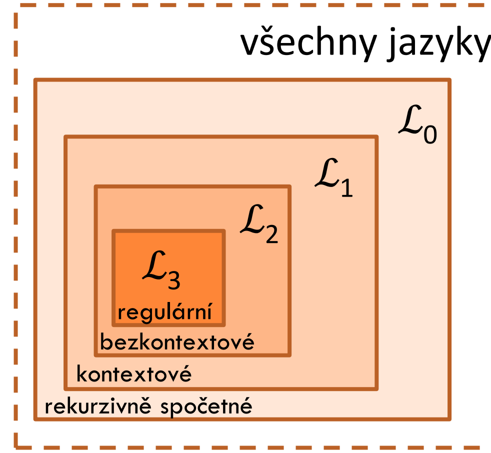
Chomského hierarchie

- $G = (V_N, V_T, S, P)$
  - různá omezení tvaru pravidel v gramatikách
    - klasifikace gramatik a jimi generovaných jazyků
  - gramatiky **typu 0**
    - rekurzivně spočetné jazyky, třída $\mathcal{L}_0$
      - žádné omezení gramatik
  - gramatiky **typu 1**
    - kontextové jazyky, třída $\mathcal{L}_1$
      - pravidla tvaru $\alpha X \beta \rightarrow \alpha w \beta$, kde $\alpha, \beta \in (V_N \cup V_T)^*$, $X \in V_N$ a $w \in (V_N \cup V_T)^*$ a $w \neq \lambda$
      - nebo $S \rightarrow \lambda$, pokud S není na pravé straně žádného jiného pravidla
    - alternativně jako **nezkracující** gramatika
      - pravidla tvaru $\alpha \rightarrow \beta$, kde $\alpha, \beta \in (V_N \cup V_T)^*$ a $|\alpha| \leq |\beta|$
      - nebo $S \rightarrow \lambda$, pokud S není na pravé straně žádného jiného pravidla
  - gramatiky **typu 2**
    - bezkontextové jazyky, třída $\mathcal{L}_2$
      - pravidla tvaru $X \rightarrow \alpha$, kde $X \in V_N$ a $\alpha \in (V_N \cup V_T)^*$
  - gramatiky **typu 3**
    - regulární jazyky (zprava lineární), třída $\mathcal{L}_3$
      - pravidla tvaru $X \rightarrow wY$ nebo $X \rightarrow w$, kde $X, Y \in V_N$ a $w \in V_T^*$

Pozn.: Lze každý jazyk generovat nějakou gramatikou?
Je $\mathcal{L}_0$ = všechny jazyky?

$\mathcal{L}_3 \subsetneq \mathcal{L}_2 \subsetneq \mathcal{L}_1 \subsetneq \mathcal{L}_0 \subsetneq$ vše





Jednoduché **vztahy**

- $\mathcal{L}_3 \subsetneq \mathcal{L}_2 \subsetneq \mathcal{L}_1 \subsetneq \mathcal{L}_0 \subsetneq$ vše
  - zatím snadno ověřitelné vztahy
  - $\mathcal{L}_1 \subseteq \mathcal{L}_0$
    - rekurzivně spočetné jazyky zahrnují kontextové
      - obecná pravidla zahrnují kontextová pravidla
  - $\mathcal{L}_3 \subseteq \mathcal{L}_2$
    - bezkontextové jazyky zahrnují regulární
      - pravidla tvaru $X \rightarrow wY$ nebo $X \rightarrow w$, kde $X, Y \in V_N$ a $w \in V_T^*$ jsou zároveň bezkontextová
  - skoro vidíme i $\mathcal{L}_2 \subseteq \mathcal{L}_1$
    - kontextové jazyky zahrnují bezkontextové
      - jsou-li pravidla tvaru $X \rightarrow \alpha$, kde $X \in V_N$ a $\alpha \in (V_N \cup V_T)^*$ s $|\alpha| \geq 1$, jsou zároveň kontextová
  - jsou-li $\mathcal{L}_3$ regulární jazyky, pak $\mathcal{L}_3 \neq \mathcal{L}_2$
    - $S \rightarrow 0S1$, $S \rightarrow \lambda$ je bezkontextová gramatika generující $\{ 0^i 1^i \mid i = 0,1,2,\dots \}$, který není regulární

Kontextové a bezkontextové (1) $\mathcal{L}_2 \subseteq \mathcal{L}_1$

- ošetření pravidel tvaru $X \rightarrow \lambda$ pro $X \in V_N$
  - bezkontextová gramatika je **nevypouštějící**, jestliže neobsahuje pravidla tvaru $X \rightarrow \lambda$ pro $X \in V_N$
  - ke každé bezkontextové gramatice $G=(V_N,V_T,S,P)$, existuje nevypouštějící gramatika $G'=(V_N,V_T,S,P')$, že $L(G')=L(G)-\{\lambda\}$
    - neterminál $X \in V_N$, pro který $X \Rightarrow_G^* \lambda$, místo přepisu na λ vůbec nevygenerujeme
      - $\Pi = \{X \mid X \in V_N \wedge X \Rightarrow_G^* \lambda\}$
        - pro pravidlo $Y \rightarrow w_1X_1w_2X_2...w_nX_nw_{n+1} \in P$, kde $X_i \in \Pi$ pro $i=1,2,...,n$ a $w_i \in ((V_N - \Pi) \cup V_T)^*$ dáme do P’ pravidla, která vzniknou z $Y \rightarrow w_1X_1w_2X_2...w_nX_nw_{n+1}$ vypuštěním libovolné podmnožiny neterminálů $\{X_1, X_2, ..., X_n\}$ na pravé straně
      - pravidla $X \rightarrow w \in P$, kde $w \neq \lambda$, přidáme do P’
  - ke každé bezkontextové gramatice G, kde $\lambda \in L(G)$, existuje bezkontextová gramatika $G''=(V_N \cup \{S''\},V_T,S'',P'')$, že $L(G'')=L(G)$ a jediné zkracující pravidlo v P’’ je $S'' \rightarrow \lambda$ s tím, že S’’ se nevyskytuje na pravé straně žádného pravidla
    - nejprve zkonstruujeme nevypouštějící bezkontextovou G’, že $L(G')=L(G)-\{\lambda\}$
      - přidáme nový neterminál S’’ (v pravidlech P’ se nevyskytuje)
      - změníme počáteční neterminál na S’’
      - $P''=P' \cup \{S'' \rightarrow \lambda, S'' \rightarrow S\}$

Kontextové a bezkontextové (2) $\mathcal{L}_2 \subseteq \mathcal{L}_1$

- **detaily** navržené konstrukce
  - $\Pi = \{X \mid X \in V_N \wedge X \Rightarrow_G^* \lambda\}$
    - $\Pi_1 = \{ X \mid X \in V_N \wedge X \Rightarrow_G \lambda\}$
    - $\Pi_{i+1} = \{ X \mid X \in V_N \wedge (\exists w \in \Pi_i^*) X \Rightarrow_G w\}$
    - $\Pi_1 \subseteq \Pi_2 \subseteq \ldots \subseteq V_N$
      - $(\exists k)\Pi_{k+1} = \Pi_k \quad \Pi = \Pi_k$
- $\lambda \neq w \in L(G) \Rightarrow w \in L(G')$
  - uvážíme derivaci ukazující $S \Rightarrow_G^* w$
    - použití pravidla $X \rightarrow \lambda$ pro $X \in V_N$, předcházelo použití pravidel, které vygenerovaly X, tj. $Y \rightarrow uZv$ pro $u,v \in (V_N \cup V_T)^+$, kde $Z \Rightarrow_G^* X$
    - použití pravidla $X \rightarrow \lambda$ vypustíme, použití pravidla $Y \rightarrow uZv$ nahradíme použitím pravidla $Y \rightarrow uv$, které bylo přidáno do P'
- $w \in L(G') \Rightarrow w \in L(G)$
  - $S \Rightarrow_{G'}^* w$
    - použití nových pravidel nahradíme původním pravidlem a vypouštěním

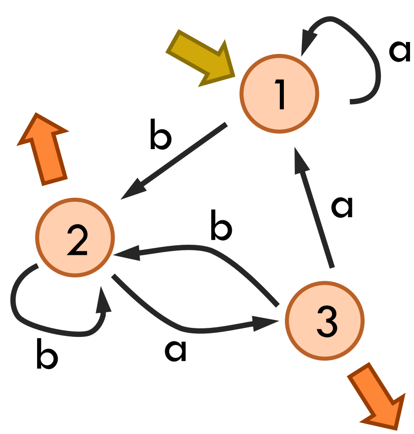
Př.: $G = (V_N, V_T, S, P)$, kde
$V_N = \{ S, A \}$
$V_T = \{ a, b, c \}$
$P = \{ S \rightarrow aSc \mid A$
$A \rightarrow bAc \mid \lambda \}$

$\Pi_1 = \{ A \}$
$\Pi_2 = \{ A, S \}$
$P' = \{ S \rightarrow aSc \mid ac \mid A$
$A \rightarrow bAc \mid bc \}$

$\mathcal{L}_3$

Konečný automat ⇒ gramatika typu 3

- KA $A=(Q, X, \delta, q_0, F)$
  - definujeme gramatiku $G = (Q, X, q_0, P)$, kde
    - $p \rightarrow xq \in P$, kdykoli $\delta(p, x) = q$
    - $p \rightarrow \lambda \in P$, kdykoli $p \in F$
- $w \in L(A) \Leftrightarrow w \in L(G)$
  - $x_1 x_2 \ldots x_n \in L(A) \Leftrightarrow (\exists q_0 q_1 q_2 \ldots q_n)\ \delta(q_{i-1}, x_i) = q_i$ a $q_n \in F$, právě když
  - $q_0 \Rightarrow_G x_1 q_1 \Rightarrow_G x_1 x_2 q_2 \Rightarrow_G \ldots \Rightarrow_G x_1 x_2 \ldots x_n q_n \Rightarrow_G x_1 x_2 \ldots x_n$



Př.: $G = (V_N, V_T, 1, P)$, kde
$V_N = \{ 1, 2, 3 \}$
$V_T = \{ a, b \}$
$P = \{ 1 \rightarrow a1 \mid b2$
$2 \rightarrow a3 \mid b2 \mid \lambda$
$3 \rightarrow a1 \mid b2 \mid \lambda \}$

Gramatika **typu 3** $\Rightarrow$ konečný automat (1)

- gramatika $G = (V_N,V_T,S,P)$ typu 3
  - zkonstruujeme gramatiku $G' = (V_N',V_T,S,P')$ typu 3, že $L(G')=L(G)$, kde
    - pravidla mají v P' mají tvar $X \rightarrow xY$ nebo $X \rightarrow \lambda$, kde $X, Y \in V_N$ a $x \in V_T$
      - pro pravidlo $\mathbf{X \rightarrow x_1 x_2 \ldots x_n Y} \in P$ s $x_i \in V_T$ pro $i=1,2,\ldots,n$ a $X,Y \in V_N$ dáme do P' pravidla
        - $X \rightarrow x_1Y_1$, $Y_1 \rightarrow x_2Y_2$, $Y_2 \rightarrow x_3Y_3$, …, $Y_{n-1} \rightarrow x_nY$, kde $Y_1,Y_2, \ldots,Y_{n-1}$ jsou nové neterminály do V_N'
      - podobně pro pravidlo $\mathbf{X \rightarrow x_1 x_2 \ldots x_n} \in P$ s $x_i \in V_T$ pro $i=1,2,\ldots,n$ a $X,Y \in V_N$ dáme do P' pravidla
        - $X \rightarrow x_1Z_1$, $Z_1 \rightarrow x_2Z_2$, $Z_2 \rightarrow x_3Z_3$, …, $Z_{n-1} \rightarrow x_nZ_n$, $Z_n \rightarrow \lambda$, kde $Z_1,Z_2, \ldots,Z_n$ jsou nové neterminály do V_N'
    - ošetření pravidel $\mathbf{X \rightarrow Y} \in P$ s $X,Y \in V_N$
      - zkonstruujeme $\Phi(X) = \{Y \mid Y \in V_N \wedge X \Rightarrow_G^* Y\}$
        - postupná konstrukce $\Phi_1(X) = \{Y \mid Y \in V_N \wedge X \Rightarrow_G Y\}$
        - $\Phi_{i+1}(X) = \Phi_i(X) \cup \{Y \mid Y \in V_N \wedge (\exists Z)[Z \in \Phi_i(X) \wedge Z \Rightarrow_G Y]\}$
      - do P' přidáme pravidla $X \rightarrow w$, kdykoli $Y \rightarrow w \in P'$ pro $Y \in \Phi(X)$

Gramatika **typu 3** ⇒ konečný automat (2)

- gramatika $G' = (V_N', V_T, S, P')$ typu 3, kde
  - pravidla mají v P' mají tvar $X \rightarrow xY$ nebo $X \rightarrow \lambda$, kde $X, Y \in V_N'$ a $x \in V_T$
    - definujeme NKA $A = (V_N', V_T, \delta, \{S\}, F)$, kde
      - $F = \{ X \mid X \in V_N' \wedge X \rightarrow \lambda \in P' \}$
      - $\delta(X, x) = \{ Y \mid Y \in V_N' \wedge X \rightarrow xY \in P' \}$ pro $X \in V_N'$ a $x \in V_T$
- $w \in L(G') \Leftrightarrow w \in L(A)$
  - $x_1x_2 \ldots x_n \in L(G') \Leftrightarrow$ existuje odvození $S \Rightarrow_{G'} x_1Y_1 \Rightarrow_{G'} x_1x_2Y_2 \Rightarrow_{G'} \ldots \Rightarrow_{G'} x_1x_2 \ldots x_nY_n \Rightarrow_{G'} x_1x_2 \ldots x_n$, kde $Y_1, Y_2, \ldots, Y_n \in V_N'$, právě když
  - $(\exists Y_1, Y_2, \ldots, Y_n \in V_N')$ $Y_{i+1} \in \delta(Y_i, x_{i+1})$ pro $i=1,2,\ldots,n-1$, $Y_1 \in \delta(S, x_1)$ a $Y_n \in F$