CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA DE MINAS GERAIS

Linguagens Formais e Autômatos

Formas Normais sobre GLCs

Andrei Rimsa Álvares
andrei@cefetmg.br

Sumário

- Formas normais
  - Chomsky
  - Greibach
- GLCs e autômatos de pilha

CEFET-MG CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA DE MINAS GERAIS

FORMAS NORMAIS

Linguagens Formais e Autômatos

Formas Normais

- Existem duas formas normais especialmente importantes
  - Forma normal de Chomsky
  - Forma normal de Greibach

Noam Chomsky ♥

Sheila Greibach ♦

♥: http://en.wikipedia.org/wiki/Noam_Chomsky

♦: http://en.wikipedia.org/wiki/Sheila_Greibach

Forma Normal de Chomsky

- Uma GLC $G = (V, \Sigma, R, P)$ está na forma normal de Chomsky (FNC) se cada uma de suas regras tem uma das formas

$P \rightarrow \lambda$ se $\lambda \in L(G)$

$X \rightarrow YZ$ para $X, Y, Z \in V$

$X \rightarrow a$ para $a \in \Sigma$

Transformação em FNC

- Seja uma GLC G, obter uma GLC equivalente à G tal que

$P \rightarrow \lambda$ se $\lambda \in L(G)$
$X \rightarrow a$ para $a \in \Sigma$
$X \rightarrow w$ para $|w| \geq 2$

Basta executar a sequência de eliminações na ordem que garanta a consistência

- Adicionar mais dois passos

1) Modificar cada regra $X \rightarrow w$, $|w| \geq 2$ de forma que ela contenha apenas variáveis
2) Substituir cada regra $X \rightarrow Y_1Y_2 \ldots Y_n$, $n \geq 3$, em que cada Y_i é uma variável, pelo conjunto de regras: $X \rightarrow Y_1Z_1$, $Z_1 \rightarrow Y_2Z_2$, ..., $Z_{n-2} \rightarrow Y_{n-1}Y_n$, em que Z_1, Z_2, ..., Z_{n-2} são novas variáveis

Exemplo

- Seja a GLC G = ({L, S, E}, {a, (,)}, R, L), em que R consta das regras

$$L \rightarrow (S)$$
$$S \rightarrow SE \mid \lambda$$
$$E \rightarrow a \mid L$$

1) Após eliminação de regras λ

Variáveis anuláveis:
$\mathcal{A}$ = { S }

$$L \rightarrow (S) \mid ()$$
$$S \rightarrow SE \mid E$$
$$E \rightarrow a \mid L$$

2) Após eliminação de regras unitárias

$$\text{enc}(L) = \{ L \}$$
$$\text{enc}(S) = \{ S, E, L \}$$
$$\text{enc}(E) = \{ E, L \}$$

$$L \rightarrow (S) \mid ()$$
$$S \rightarrow SE \mid a \mid (S) \mid ( )$$
$$E \rightarrow a \mid (S) \mid ( )$$

3) Não há eliminação de variáveis inúteis

Exemplo

- Seja a GLC G = ({L, S, E}, {a, (,)}, R, L), em que R consta das regras

$$L \rightarrow (\ S\ )$$
$$S \rightarrow SE \mid \lambda$$
$$E \rightarrow a \mid L$$

4) Regras da forma $X \rightarrow w$, $|w| \geq 2$ somente com variáveis

$$L \rightarrow (\ S\ ) \mid ()$$
$$S \rightarrow SE \mid a \mid (\ S\ ) \mid (\ )$$
$$E \rightarrow a \mid (\ S\ ) \mid (\ )$$

$\Rightarrow$

$$L \rightarrow ASB \mid AB$$
$$S \rightarrow SE \mid a \mid ASB \mid AB$$
$$E \rightarrow a \mid ASB \mid AB$$
$$A \rightarrow ($$
$$B \rightarrow)$$

Exemplo

- Seja a GLC G = ({L, S, E}, {a, (,)}, R, L), em que R consta das regras:

$$L \rightarrow (\ S\ )$$
$$S \rightarrow SE \mid \lambda$$
$$E \rightarrow a \mid L$$

5) Substituir regras da forma $X \rightarrow Y_1Y_2...Y_n$ $(n \geq 3)$

$$L \rightarrow ASB \mid AB$$
$$S \rightarrow SE \mid a \mid ASB \mid AB$$
$$E \rightarrow a \mid ASB \mid AB$$
$$A \rightarrow ($$
$$B \rightarrow)$$

$$\Rightarrow$$

$$L \rightarrow AX \mid AB$$
$$S \rightarrow SE \mid a \mid AX \mid AB$$
$$E \rightarrow a \mid AX \mid AB$$
$$A \rightarrow ($$
$$B \rightarrow)$$
$$X \rightarrow SB$$

Mais Manipulação de GLCs

- Antes de apresentar a forma normal de Greibach, serão vistos dois métodos de manipulação que são úteis não apenas como passos intermediários para obtenção de GLC's nesta forma normal, mas também em outros contextos

1) Eliminação de regras recursivas à esquerda
2) Eliminação de variável em regra

Eliminação de Regras Recursivas à Esquerda

- **Teorema:** para qualquer GLC, existe uma GLC equivalente sem regras recursivas à esquerda

  – Sejam, a seguir, todas as regras X de uma GLC G, onde nenhum w_i começa com X

  $$X \rightarrow Xy_1 \mid Xy_2 \mid \ldots \mid Xy_n \mid w_1 \mid w_2 \mid \ldots \mid w_k$$

  – Substituir recursão à esquerda por recursão à direita, em que Z é uma nova variável

  $$X \rightarrow w_1Z \mid w_2Z \mid \ldots \mid w_kZ$$
  $$Z \rightarrow y_1Z \mid y_2Z \mid \ldots \mid y_nZ \mid \lambda$$

  – Eliminando regras λ

  $$X \rightarrow w_1Z \mid w_2Z \mid \ldots \mid w_kZ \mid w_1 \mid w_2 \mid \ldots \mid w_k$$
  $$Z \rightarrow y_1Z \mid y_2Z \mid \ldots \mid y_nZ \mid y_1 \mid y_2 \mid \ldots \mid y_n$$

Exemplo

- Seja a GLC G = ({E}, {t, +, *, (,)}, R, E), com regras recursivas à esquerda, onde R é dado por

$$E \rightarrow E + E \mid E * E \mid (E) \mid t$$

- Remove-se a recursividade à esquerda

$$\begin{aligned} E &\rightarrow (E) \mid t \mid (E)Z \mid tZ \\ Z &\rightarrow +E \mid *E \mid +EZ \mid *EZ \end{aligned}$$

Eliminação de Variável em Regra

- **Teorema:** Seja uma GLC G = (V, Σ, R, P) tal que $X \longrightarrow uYv \in R$, onde $Y \in V$ e $Y \neq X$. Sejam $Y \longrightarrow w_1 \mid w_2 \mid \ldots \mid w_n$ **todas** as regras Y em R. Seja G' = (V, Σ, R', P) onde

$$R' = (R - \{X \rightarrow uYv\}) \cup \{X \rightarrow uw_1v \mid uw_2v \mid \ldots \mid uw_nv\}$$

L(G') = L(G)

Forma Normal de Greibach

- Uma GLC $G = (V, \Sigma, R, P)$ está na forma normal de Greibach (FNG) se cada uma de suas regras tem uma das formas

$$P \rightarrow \lambda \qquad \text{se } \lambda \in L(G)$$
$$X \rightarrow ay \qquad \text{para } a \in \Sigma \text{ e } y \in V^*$$

  – Uma forma sentencial de uma gramática na FNG, com exceção de λ, é da forma xy, em que $x \in \Sigma^+$ e $y \in V^*$

Qual o tamanho de uma derivação de uma palavra?

Transformação em FNG

- Seja uma GLC G, obter uma GLC equivalente à G tal que

$$P \rightarrow \lambda \text{ se } \lambda \in L(G)$$
$$X \rightarrow a \text{ para } a \in \Sigma$$
$$X \rightarrow w \text{ para } |w| \geq 2$$

Basta executar a sequência de eliminações na ordem que garanta a consistência

  – Para cada $X \rightarrow w$ para $|w| \geq 2$, substituir por variáveis todos os terminais de w a partir do segundo símbolo. Com isso, obtém-se regras da forma $X \rightarrow Yy$, sendo que $Y \in (V \cup \Sigma)$ e $y \in V^+$

Transformação em FNG

- Algoritmo
  1) Numerar sequencialmente as variáveis ($\#P = 1$)
  2) Para cada $A \in V$, começando com P, na ordem dada pela numeração, fazer enquanto possível

     2.1) Se existir $A \rightarrow By$, para $|y| \geq 1$, tal que $\#A > \#B$, substituir B

     2.2) Se existir $A \rightarrow Ay$, para $|y| \geq 1$, eliminar recursão à esquerda
  3) Seja B a variável de maior número em que há regra da forma $B \rightarrow Cy$, com $C \in V$, substituir C. Repetir até atingir a variável P
  4) Para regras da forma $Z \rightarrow Ay$, em que Z é variável nova (adicionada na eliminação de recursão à esquerda), substituir a variável A

Exemplo

- Seja a GLC G = ({A, B, C, D}, {b, c, d}, R, A), onde R é dado por

$$
\begin{aligned}
A &\rightarrow CB \\
B &\rightarrow BBD \mid b \\
C &\rightarrow BBC \mid Dc \\
D &\rightarrow AD \mid d
\end{aligned}
$$

Observação: essa gramática não possui regras λ e unitárias

- Elimina-se a regra $C \rightarrow Dc$, introduzindo as regras $C \rightarrow DE$ e $E \rightarrow c$

$$
\begin{aligned}
A &\rightarrow CB \\
B &\rightarrow BBD \mid b \\
C &\rightarrow BBC \mid DE \\
D &\rightarrow AD \mid d \\
E &\rightarrow c
\end{aligned}
$$

- Numeração: #A = 1, #B = 2, #C = 3, #D = 4, #E = 5 (passo 1)

Exemplo

- Removendo recursão à esquerda para as regras B (passo 2.2)

#1 $A \rightarrow CB$
#2 $B \rightarrow BBD \mid b$
#3 $C \rightarrow BBC \mid DE$
#4 $D \rightarrow AD \mid d$
#5 $E \rightarrow c$

$\Rightarrow$

$A \rightarrow CB$
* $B \rightarrow b \mid bZ_1$
$C \rightarrow BBC \mid DE$
$D \rightarrow AD \mid d$
$E \rightarrow c$
* $Z_1 \rightarrow BD \mid BDZ_1$

O asterisco (*) marca as novas regras

Exemplo

- Substitui-se a regra $C \rightarrow BBC$ (passo 2.1: $\#C > \#B$)

$$
\begin{aligned}
{}^{\#1}A &\rightarrow CB \\
{}^{\#2}B &\rightarrow b \mid bZ_1 \\
{}^{\#3}C &\rightarrow BBC \mid DE \\
{}^{\#4}D &\rightarrow AD \mid d \\
{}^{\#5}E &\rightarrow c \\
Z_1 &\rightarrow BD \mid BDZ_1
\end{aligned}
\quad \Rightarrow \quad
\begin{aligned}
A &\rightarrow CB \\
B &\rightarrow b \mid bZ_1 \\
* \quad C &\rightarrow bBC \mid bZ_1BC \mid DE \\
D &\rightarrow AD \mid d \\
E &\rightarrow c \\
Z_1 &\rightarrow BD \mid BDZ_1
\end{aligned}
$$

Exemplo

- Substitui-se a regra $D \rightarrow AD$ (passo 2.1: $\#D > \#A$)

$^{\#1}A \rightarrow CB$	$A \rightarrow CB$
$^{\#2}B \rightarrow b \mid bZ_1$	$B \rightarrow b \mid bZ_1$
$^{\#3}C \rightarrow bBC \mid bZ_1BC \mid DE$	$C \rightarrow bBC \mid bZ_1BC \mid DE$
$^{\#4}D \rightarrow AD \mid d$	* $D \rightarrow CBD \mid d$
$^{\#5}E \rightarrow c$	$E \rightarrow c$
$Z_1 \rightarrow BD \mid BDZ_1$	$Z_1 \rightarrow BD \mid BDZ_1$

Exemplo

- Substitui-se a regra $D \rightarrow CBD$ (passo 2.1: $\#D > \#C$)

#1 $A \rightarrow CB$
#2 $B \rightarrow b \mid bZ_1$
#3 $C \rightarrow bBC \mid bZ_1BC \mid DE$
#4 $D \rightarrow CBD \mid d$
#5 $E \rightarrow c$
$Z_1 \rightarrow BD \mid BDZ_1$

$A \rightarrow CB$
$B \rightarrow b \mid bZ_1$
$C \rightarrow bBC \mid bZ_1BC \mid DE$
* $D \rightarrow bBCBD \mid bZ_1BCBD \mid DEBD \mid d$
$E \rightarrow c$
$Z_1 \rightarrow BD \mid BDZ_1$

Exemplo

- Removendo recursão à esquerda para as regras D (passo 2.2)

#1 $A \rightarrow CB$
#2 $B \rightarrow b \mid bZ_1$
#3 $C \rightarrow bBC \mid bZ_1BC \mid DE$
#4 $D \rightarrow bBCBD \mid bZ_1BCBD \mid DEBD \mid d$
#5 $E \rightarrow c$
$Z_1 \rightarrow BD \mid BDZ_1$

$\Rightarrow$

$A \rightarrow CB$
$B \rightarrow b \mid bZ_1$
$C \rightarrow bBC \mid bZ_1BC \mid DE$
* $D \rightarrow bBCBD \mid bZ_1BCBD \mid d \mid bBCBDZ_2 \mid bZ_1BCBDZ_2 \mid dZ_2$
$E \rightarrow c$
$Z_1 \rightarrow BD \mid BDZ_1$
* $Z_2 \rightarrow EBD \mid EBDZ_2$

Exemplo

- Substitui-se a regra $C \rightarrow DE$ (passo 3)

#1 $A \rightarrow CB$

#2 $B \rightarrow b \mid bZ_1$

#3 $C \rightarrow bBC \mid bZ_1BC \mid DE$

#4 $D \rightarrow bBCBD \mid bZ_1BCBD \mid d \mid bBCBDZ_2 \mid$
$bZ_1BCBDZ_2 \mid dZ_2$

#5 $E \rightarrow c$

$Z_1 \rightarrow BD \mid BDZ_1$

$Z_2 \rightarrow EBD \mid EBDZ_2$

$\Rightarrow$

$A \rightarrow CB$

$B \rightarrow b \mid bZ_1$

$C \rightarrow bBC \mid bZ_1BC$

* $C \rightarrow bBCBDE \mid bZ_1BCBDE \mid dE \mid$
$bBCBDZ_2E \mid bZ_1BCBDZ_2E \mid dZ_2E$

$D \rightarrow bBCBD \mid bZ_1BCBD \mid d \mid bBCBDZ_2 \mid$
$bZ_1BCBDZ_2 \mid dZ_2$

$E \rightarrow c$

$Z_1 \rightarrow BD \mid BDZ_1$

$Z_2 \rightarrow EBD \mid EBDZ_2$

Exemplo

- Substitui-se a regra $A \rightarrow CB$ (passo 3)

#1 $A \rightarrow CB$
#2 $B \rightarrow b \mid bZ_1$
#3 $C \rightarrow bBC \mid bZ_1BC \mid bBCBDE \mid bZ_1BCBDE \mid$
$dE \mid bBCBDZ_2E \mid bZ_1BCBDZ_2E \mid dZ_2E$
#4 $D \rightarrow bBCBD \mid bZ_1BCBD \mid d \mid bBCBDZ_2 \mid$
$bZ_1BCBDZ_2 \mid dZ_2$
#5 $E \rightarrow c$
$Z_1 \rightarrow BD \mid BDZ_1$
$Z_2 \rightarrow EBD \mid EBDZ_2$

⇒

* $A \rightarrow bBCB \mid bZ_1BCB \mid bBCBDEB \mid$
$bZ_1BCBDEB \mid dEB \mid bBCBDZ_2 EB \mid$
$bZ_1BCBDZ_2EB \mid dZ_2EB$
$B \rightarrow b \mid bZ_1$
$C \rightarrow bBC \mid bZ_1BC \mid bBCBDE \mid bZ_1BCBDE \mid$
$dE \mid bBCBDZ_2E \mid bZ_1BCBDZ_2E \mid dZ_2E$
$D \rightarrow bBCBD \mid bZ_1BCBD \mid d \mid bBCBDZ_2 \mid$
$bZ_1BCBDZ_2 \mid dZ_2$
$E \rightarrow c$
$Z_1 \rightarrow BD \mid BDZ_1$
$Z_2 \rightarrow EBD \mid EBDZ_2$

Exemplo

- Finalmente, substitui-se as regras introduzidas pela eliminação de recursão à esquerda (passo 4)

#1 $A \rightarrow bBCB \mid bZ_1BCB \mid bBCBDEB \mid bZ_1BCBDEB \mid dEB \mid bBCBDZ_2EB \mid bZ_1BCBDZ_2EB \mid dZ_2EB$

#2 $B \rightarrow b \mid bZ_1$

#3 $C \rightarrow bBC \mid bZ_1BC \mid bBCBDE \mid bZ_1BCBDE \mid dE \mid bBCBDZ_2E \mid bZ_1BCBDZ_2E \mid dZ_2E$

#4 $D \rightarrow bBCBD \mid bZ_1BCBD \mid d \mid bBCBDZ_2 \mid bZ_1BCBDZ_2 \mid dZ_2$

#5 $E \rightarrow c$

$Z_1 \rightarrow BD \mid BDZ_1$

$Z_2 \rightarrow EBD \mid EBDZ_2$

$\Rightarrow$

$A \rightarrow bBCB \mid bZ_1BCB \mid bBCBDEB \mid bZ_1BCBDEB \mid dEB \mid bBCBDZ_2EB \mid bZ_1BCBDZ_2EB \mid dZ_2EB$

$B \rightarrow b \mid bZ_1$

$C \rightarrow bBC \mid bZ_1BC \mid bBCBDE \mid bZ_1BCBDE \mid dE \mid bBCBDZ_2E \mid bZ_1BCBDZ_2E \mid dZ_2E$

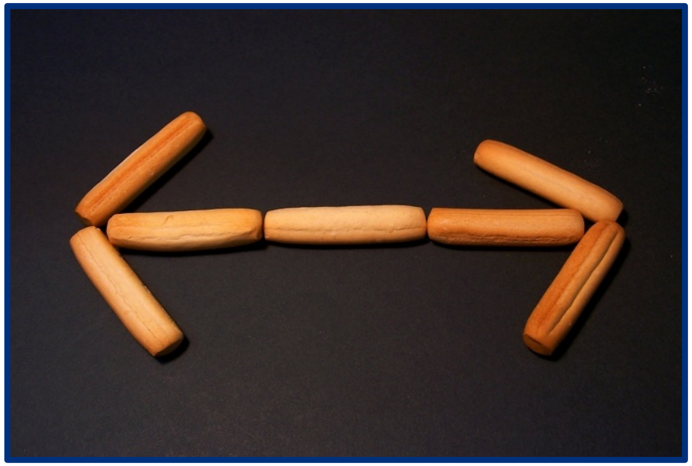
$D \rightarrow bBCBD \mid bZ_1BCBD \mid d \mid bBCBDZ_2 \mid bZ_1BCBDZ_2 \mid dZ_2$

$E \rightarrow c$

* $Z_1 \rightarrow bD \mid bZ_1D \mid bDZ_1 \mid bZ_1DZ_1$

* $Z_2 \rightarrow cBD \mid cBDZ_2$

GLCS E AUTÔMATOS DE PILHA



Linguagens Formais e Autômatos

Equivalência entre GLCs e APs

- Autômatos de pilha (APs) e gramáticas livres de contexto (GLCs) reconhecem a mesma classe de linguagem: as **linguagens livres de contexto**

- Será mostrado que

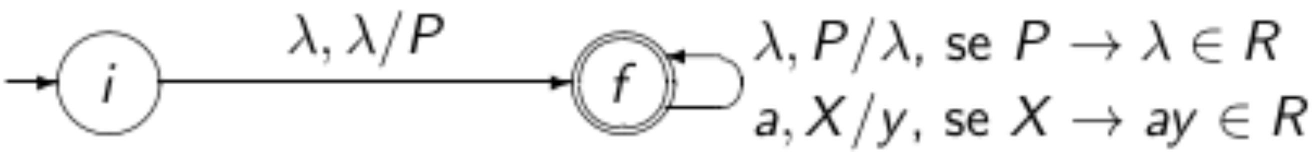
  1) Para qualquer GLC G existe um AP que reconhece $L(G)$
  2) Para qualquer APN M existe uma GLC que gera $L(M)$

GLC → AP

- **Teorema:** Para qualquer GLC G existe um AP que reconhece L(G)

Seja $G' = (V, \Sigma, R, P)$ uma GLC na FNG equivalente à G. Um APN que aceita L(G') é $M = (\{i, f\}, \Sigma, V, \delta, \{i\}, \{f\})$, onde δ consta das transições

- $\delta(i, \lambda, \lambda) = \{[f, P]\}$
- $\delta(f, \lambda, P) = \{[f, \lambda]\}$, se $P \rightarrow \lambda \in R$
- $\delta(f, a, X) = \{[f, y] \mid y \in V^* \text{ e } X \rightarrow ay \in R\}$,
  $\forall a \in \Sigma$ e $\forall X \in V$



Exemplo

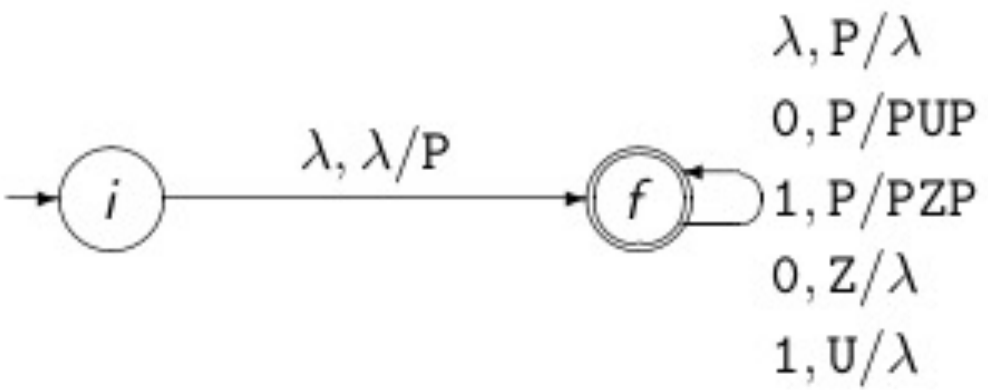
- Seja a GLC $G = (\{P\}, \{0, 1\}, R, P)$, onde R consta das regras

$$P \rightarrow 0P1P \mid 1P0P \mid \lambda$$

- A gramática G' equivalente à G na FNG consta das regras

$$P \rightarrow 0PUP \mid 1PZP \mid \lambda$$
$$Z \rightarrow 0$$
$$U \rightarrow 1$$

- Pode-se construir um autômato de pilha que reconhece L(G)



AP → GLC

- Antes de mostrar que é sempre possível obter uma GLC que gera a linguagem reconhecida por um AP, será mostrada a **ideia central** por trás do processo

Seja um, APN $M = (E, \Sigma, \Gamma, \delta, I, F)$. Dados $e, e' \in E$ e $A \in \Gamma \cup \{\lambda\}$, considere o conjunto $C(e, A, e')$ de todas as palavras $w \in \Sigma^*$ tais que o APN M, começando em e, com a pilha contendo A, termina no estado e' com a pilha vazia, após consumir w

$$C(e, A, e') = \{w \in \Sigma^* \mid [e, w, A] \overset{*}{\vdash} [e', \lambda, \lambda]\}$$

A linguagem reconhecida é dada por

$$L(M) = \bigcup_{(i,f) \in I \times F} C(i, \lambda, f)$$

AP → GLC

- Suponha que seja possível gerar o conjunto $C(e, A, e')$ por meio de uma GLC cuja variável de partida seja $[e, A, e']$

- Então L(M) pode ser gerada por uma GLC constituída de
  - Todas as regras $[e, A, e']$
  - $P \rightarrow [i, \lambda, f]$, para $\forall i \in I$ e $\forall f \in F$

Como produzir as regras?

AP → GLC

- **Teorema:** Para qualquer APN M existe uma GLC que gera L(M)

Seja o APN $M = (E, \Sigma, \Gamma, \delta, I, F)$. Será construída uma GLC com variáveis da forma $[e, A, e']$, para $e, e' \in E$ e $A \in \Gamma \cup \{\lambda\}$, de modo que para todo $w \in \Sigma^*$

$$[e, A, e'] \overset{*}{\Rightarrow} w \text{ se e somente se } [e, A, e'] \vdash^{*} [e', \lambda, \lambda]$$

AP → GLC

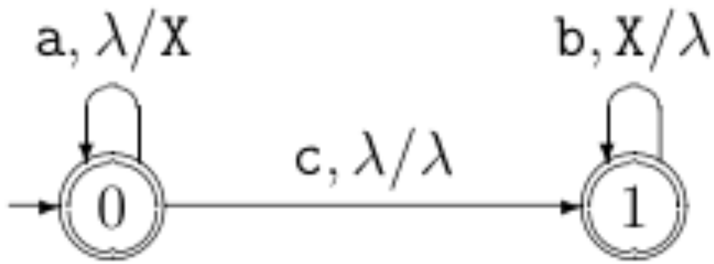
- A gramática G tal que $L(G) = L(M)$ é (V, Σ, R, P), onde $V = (\{P\} \cup E) \times (\Gamma \cup \{\lambda\}) \times E$ e R contém as regras
  - $\mathbf{P} \rightarrow [\boldsymbol{i}, \boldsymbol{\lambda}, \boldsymbol{f}]$, para cada $i \in I$ e $f \in F$
  - $[\boldsymbol{e}, \boldsymbol{\lambda}, \boldsymbol{e}] \rightarrow \boldsymbol{\lambda}$, para cada $e \in E$

  Para cada transição $[e', z] \in \delta(e, a, A)$, onde $a \in \Sigma \cup \{\lambda\}$, $z = B_1B_2 \ldots B_n$ ($B_i \in \Gamma$) e $A \in \Gamma \cup \{\lambda\}$, as seguintes regras (**tipo 1**)
  - Se $z = \lambda$: $[\boldsymbol{e}, \mathbf{A}, \boldsymbol{d}] \rightarrow \mathbf{a}[\boldsymbol{e'}, \boldsymbol{\lambda}, \boldsymbol{d}]$, para cada $d \in E$
  - Se $z \neq \lambda$: $[\boldsymbol{e}, \mathbf{A}, \boldsymbol{d_n}] \rightarrow \mathbf{a}[\boldsymbol{e'}, \mathbf{B_1}, \boldsymbol{d_1}] \ldots [\boldsymbol{d_{n-1}}, \mathbf{B_n}, \boldsymbol{d_n}]$, para cada $d_1, d_2, \ldots, d_n \in E$

  Ainda, se $A = \lambda$, tem-se adicionalmente as regras (**tipo 2**)
  - Se $z = \lambda$: $[\mathbf{e}, \mathbf{C}, \mathbf{d}] \rightarrow \mathbf{a}[\mathbf{e'}, \mathbf{C}, \mathbf{d}]$, para cada $C \in \Gamma$ e cada $d \in E$
  - Se $z \neq \lambda$: $[\boldsymbol{e}, \mathbf{C}, \boldsymbol{d_{n+1}}] \rightarrow \mathbf{a}[\boldsymbol{e'}, \mathbf{B_1}, \boldsymbol{d_1}] \ldots [\boldsymbol{d_{n-1}}, \mathbf{B_n}, \boldsymbol{d_n}][\boldsymbol{d_n}, \mathbf{C}, \boldsymbol{d_{n+1}}]$, para $C \in \Gamma$ e cada $d_1, d_2, \ldots, d_n, d_{n+1} \in E$

Exemplo

- Seja o APD que reconhece L = $\{a^n c b^n \mid n \geq 0\} \cup \{\lambda\}$



- Para a gramática, obtém-se as seguintes regras inicialmente
  - $P \rightarrow [0, \lambda, 0] \mid [0, \lambda, 1]$
  - $[0, \lambda, 0] \rightarrow \lambda$
  - $[1, \lambda, 1] \rightarrow \lambda$

Como são as outras regras?

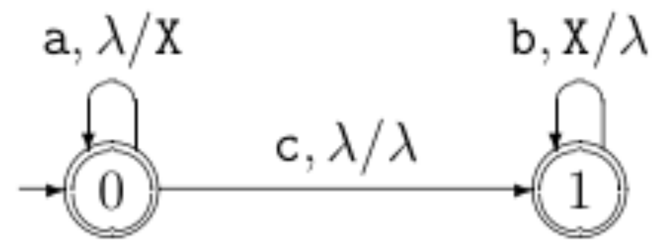
Exemplo

Como variáveis são inúteis?

- Para as transições
  - $[0, X] \in \delta(0, a, \lambda)$
    - $[0, \lambda, 0] \rightarrow a[0,X,0]$
    - $[0, \lambda, 1] \rightarrow a[0,X,1]$
    - $[0, X, 0] \rightarrow a[0, X, 0][0, X, 0] \mid a[0, X, 1][1, X, 0]$
    - $[0, X, 1] \rightarrow a[0, X, 0][0, X, 1] \mid a[0, X, 1][1, X, 1]$
  - $[0, X] \in \delta(0, a, \lambda)$
    - $[0, \lambda, 0] \rightarrow c[1, \lambda, 0]$
    - $[0, \lambda, 1] \rightarrow c[1, \lambda, 1]$
    - $[0, X, 0] \rightarrow c[1, X, 0]$
    - $[0, X, 1] \rightarrow c[1, X, 1]$
  - $[1, \lambda] \in \delta(1, b, X)$
    - $[1, X, 0] \rightarrow b[1, \lambda, 0]$
    - $[1, X, 1] \rightarrow b[1, \lambda, 1]$

Tipo 1

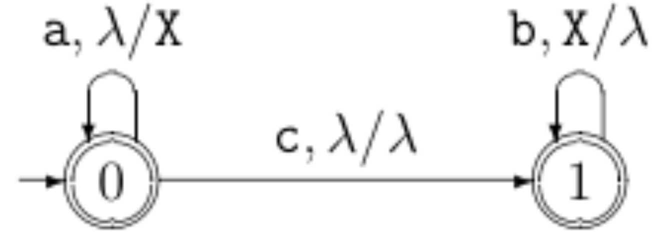
Tipo 2



Exemplo

- Eliminando as variáveis inúteis obtém-se as seguintes regras

$$
\begin{aligned}
&P \to [0, \lambda, 0] \mid [0, \lambda, 1] \\
&[0, \lambda, 1] \to a[0, X, 1] \mid c[1, \lambda, 1] \\
&[0, X, 1] \to a[0, X, 1][1, X, 1] \mid c[1, X, 1] \\
&[1, X, 1] \to b[1, \lambda, 1] \\
&[0, \lambda, 0] \to \lambda \\
&[1, \lambda, 1] \to \lambda
\end{aligned}
$$



ISSO É TUDO, PESSOAL!