INSTITUTO DE COMPUTAÇÃO

MATA50 – Linguagens Formais e Autômatos – 2023.1

Professor: Roberto Freitas Parente
Monitor: Fernando Franco de Lacerda Neto

Gabarito – Grupo 03

Exercício 1. *Prove que as seguintes linguagens a seguir não são regulares, construa uma gramática livre de contexto G e prove que $L(G)$ tem a mesma linguagem da esperada.*

1. $\{ww^R \colon w \in \{a, b, c\}^*\}$

Resposta. GLC $G = (\{S\}, \{a, b, c\}, P, S)$

- $S \to aSa \mid bSb \mid cSc \mid \varepsilon$

Prova por indução no tamanho da palavra de que $w = xx^R \implies w \in L(G)$:

Suponha que w tem essa forma, i.e., $w = xx^R$.
Base: Para $|w| = 0 \implies w = \varepsilon$ e como G tem a produção $P \to \varepsilon$ então $w \in L(G)$
Passo indutivo: Suponha que $|w| = n + 1 \geq 1$. Como $w = xx^R$ então ela começa e termina com o mesmo simbolo. Assim, $w = ayy^Ra$ ou $w = byy^Rb$ ou $w = cyy^Rc$. Por H.I. temos que $yy^R \in L(G)$, ou seja, $P \overset{*}{\Rightarrow} yy^R$ e como existem as produções $P \to aPa$, $P \to bPb$ e $P \to cPc$ podemos derivar w em cada um dos casos respectivamente, logo $w \in L(G)$.

Prova por indução na quantidade de derivações de que $w \in L(G) \implies w = xx^R$:

Suponha que $w \in L(G)$.
Base: Suponha que derivamos w com uma produção apenas,ou seja, utilizamos $P \to \varepsilon$ então $P \Rightarrow \varepsilon$ e ε é da forma que queremos.
Passo indutivo: Suponha que utilizamos $n+1$ derivações com $n > 0$. As únicas produções que permitem um derivação com pelo menos 2 passos são $P \to aPa$, $P \to bPb$ e $P \to cPc$ portanto sabemos que a primeira derivação foi uma delas.
Suponha que que utilizamos $P \to aPa$ assim

$$P \Rightarrow aPa \overset{*}{\Rightarrow} axa = w$$

Como $P \overset{*}{\Rightarrow} x$ em n passos temos, por H.I., que $x = yy^R \implies w = ayy^Ra = zz^R$.
Análogo para as outras produções. □

2. $\{w1^n \colon w \in \{0, 1\}^* \text{ e } n = |w|\}$

Resposta. GLC $G = (\{S\}, \{0, 1\}, P, S)$

- $S \to 0S1 \mid 1S1 \mid \varepsilon$

Prova por indução no tamanho da palavra de que $w = a1^{|a|} \implies w \in L(G)$:

Suponha que w tem essa forma.
Base: Para $|w| = 0 \implies w = \varepsilon$ e como G tem a produção $P \to \varepsilon$ então $w \in L(G)$.
Passo indutivo: Suponha que $|w| = n + 1 \geq 1$. Como $w = a1^{|a|}$ então $w = 0b1^{|b|+1}$ ou $w = 1b1^{|b|+1}$. Por H.I. temos que $b1^{|b|} \in L(G)$, ou seja, $P \overset{*}{\Rightarrow} b1^{|b|}$ e como existem as produções $P \to 0P1$ e $P \to 1P1$ podemos derivar w em cada um dos casos respectivamente, logo $w \in L(G)$.

Prova por indução na quantidade de derivações de que $w \in L(G) \implies w = a1^{|a|}$:

Suponha que $w \in L(G)$.

Base: Suponha que derivamos w com uma produção apenas,ou seja, utilizamos $P \to \varepsilon$ então $P \Rightarrow \varepsilon$ e ε é da forma que queremos.
Passo indutivo: Suponha que utilizamos $n+1$ derivações com $n > 0$. As únicas produções que permitem um derivação com pelo menos 2 passos são $P \to 0P1$ e $P \to 1P1$ portanto sabemos que a primeira derivação foi uma delas.
Suponha que que utilizamos $P \to 0P1$ assim

$$P \Rightarrow 0P1 \overset{*}{\Rightarrow} 0x1 = w$$

Como $P \overset{*}{\Rightarrow} x$ em n passos temos, por H.I., que $x = b1^{|b|} \implies w = 0b1^{|b|}1 = 0b1^{|b|+1} = c1^{|c|}$.
Análogo para o outro caso. □

3. $\{a^ib^jc^k \colon\ i \neq j \text{ ou } i \neq k\}$

Resposta. GLC $G = (\{S, A, B, C, X, Y\}, \{a, b, c\}, P, S)$

- $S \to XC \mid AY$
- $A \to aA \mid \varepsilon$
- $B \to bB \mid \varepsilon$
- $C \to cC \mid \varepsilon$
- $X \to aXb \mid aA \mid bB$
- $Y \to aYc \mid aAB \mid BcC$

Prova por indução no tamanho da palavra de que $w = a^ib^jc^k$ com $i \neq j$ ou $i \neq k \implies w \in L(G)$:

Suponha que w tem essa forma, i.e., $w = a1^{|a|}$.
Base: Para $|w| = 1 \implies w = a$ ou $w = b$ ou $w = c$ e como podemos ter as seguintes derivações

$$P \Rightarrow XC \Rightarrow aAC \Rightarrow aC \Rightarrow a$$

$$P \Rightarrow XC \Rightarrow bBC \Rightarrow bC \Rightarrow b$$

$$P \Rightarrow AY \Rightarrow Y \Rightarrow BcC \Rightarrow cC \Rightarrow c$$

então $w \in L(G)$.
Passo indutivo: Suponha que $|w| = n + 1 \geq 2$ e que é verdade para quando $|w| \leq n$. Temos alguns possiveis casos e # denota a quantidade de um simbolo:
(a) $\#a > \#b > \#c$
(b) $\#a > \#b = \#c$
(c) $\#a > \#b < \#c$
(d) $\#a = \#b > \#c$ - w = w'c não funciona pra esse
(e) $\#a = \#b < \#c$
(f) $\#a < \#b > \#c$
(g) $\#a < \#b = \#c$
(h) $\#a < \#b < \#c$

□

Exercício 2. *Mostre que toda linguagem regular é uma linguagem livre de contexto.* ***Dica:*** *Construa uma GLC por indução sobre o número de operadores na expressão regular.*

Resposta. Seja R a expressão regular para a linguagem regular.
Base: Para zero operadores em R temos que a linguagem reconhece apenas um simbolo a. Assim, podemos criar a seguinte GLC $G = (\{S\}, \Sigma, P, S)$

- $S \to a$

Hipótese indutiva: Agora suponha que vale para linguagens regulares com n ou menos operadores.
Passo indutivo: Agora para $n + 1$ operadores temos 3 casos

- União: $R = R_1 + R_2$. Sejam S_1 e S_2 as variaveis iniciais das GLC's de R_1 e R_2 respectivamente, que existem por H.I., criamos uma variavel inicial para R da seguinte maneira: $S \to S_1|S_2$. Se a primeira produção que usamos for $S \to S_1$ então derivamos todas as palavras derivadas por R_1, que, novamente, funciona por H.I.. Se a primeira produção que usamos for $S \to S_2$ então derivamos todas as palavras derivadas por R_2.

- Concatenação: $R = R_1R_2$. Sejam S_1 e S_2 as variaveis iniciais das GLC's de R_1 e R_2 respectivamente, que existem por H.I., criamos uma variavel inicial para R da seguinte maneira: $S \to S_1S_2$. Por H.I., S_1 deriva todas as palavras de R_1 e S_2 deriva todas as palavras de R_2. Assim, S irá produzir todas as palavras de R_1R_2.

- Estrela: $R = R_1^*$. Sejam S_1 a variavel inicial da GLC's de R_1, criamos uma variavel inicial para R da seguinte maneira: $S \to S_1S_1|\varepsilon$. Assim, podemos derivar a concatenação de $k, k \geq 0$ palavras derivadas a partir de S_1, ou seja, a *.

$\square$

Exercício 3. *Pegue as suas três gramáticas do "Exercício 1" acima e coloque-as na Forma Normal de Chomsky.*
Obs: *Se vc não fez uma delas, é um bom momento para tentar novamente, mas faça com as suas!*

Resposta.

1. GLC $G = (\{S\}, \{a, b, c\}, P, S)$
   - $S \to aSa \mid bSb \mid cSc \mid \varepsilon$

   Eliminando ε-produções
   - $S \to aSa \mid bSb \mid cSc \mid aa \mid bb \mid cc$

   Não temos produções únitarias e S é gerador e alcançável
   Colocando na Forma Normal de Chomsky
   - $S \to ASA \mid BSB \mid CSC \mid AA \mid BB \mid CC$
   - $A \to a$
   - $B \to b$
   - $C \to c$

   Passo 2:
   - $S \to AX_1 \mid BX_2 \mid CX_3 \mid AA \mid BB \mid CC$
   - $A \to a$
   - $B \to b$
   - $C \to c$
   - $X_1 \to SA$
   - $X_2 \to SB$
   - $X_3 \to SC$
2. GLC $G = (\{S\}, \{0, 1\}, P, S)$
   - $S \to 0S1 \mid 1S1 \mid \varepsilon$

   Eliminando ε-produções
   - $S \to 0S1 \mid 1S1 \mid 01 \mid 11$

   Não temos produções únitarias e S é gerador e alcançável
   Colocando na Forma Normal de Chomsky
   - $S \to ASB \mid BSB \mid AB \mid BB$
   - $A \to 0$
   - $B \to 1$

   Passo 2:

- $S \to AX_1 \mid BX_1 \mid AB \mid BB$
- $A \to 0$
- $B \to 1$
- $X_1 \to SB$

3. GLC $G = (\{S, A, B, C, X, Y\}, \{a, b, c\}, P, S)$
   - $S \to XC \mid Y$
   - $A \to aA \mid \varepsilon$
   - $B \to bB \mid \varepsilon$
   - $C \to cC \mid \varepsilon$
   - $X \to aXb \mid aA \mid bB$
   - $Y \to aYc \mid aAB \mid BcC$

   Eliminando ε-produções
   - $S \to XC \mid X \mid Y$
   - $A \to aA \mid a$
   - $B \to bB \mid b$
   - $C \to cC \mid c$
   - $X \to aXb \mid aA \mid a \mid bB \mid b$
   - $Y \to aYc \mid aAB \mid aB \mid aA \mid a \mid BcC \mid cC \mid Bc \mid c$

   Eliminando produções únitarias
   - $S \to XC \mid aXb \mid aA \mid a \mid bB \mid b \mid aYc \mid aAB \mid aB \mid aA \mid BcC \mid cC \mid Bc \mid c$
   - $A \to aA \mid a$
   - $B \to bB \mid b$
   - $C \to cC \mid c$
   - $X \to aXb \mid aA \mid a \mid bB \mid b$
   - $Y \to aYc \mid aAB \mid aB \mid aA \mid a \mid BcC \mid cC \mid Bc \mid c$

   Todas as variáveis são geradoras e alcançáveis
   Colocando na Forma Normal de Chomsky
   - $S \to XC \mid QXW \mid QA \mid Q \mid WB \mid W \mid QYE \mid QAB \mid QB \mid QA \mid BEC \mid EC \mid BE \mid E$
   - $A \to QA \mid Q$
   - $B \to WB \mid W$
   - $C \to EC \mid E$
   - $X \to QXW \mid QA \mid Q \mid WB \mid W$
   - $Y \to QYE \mid QAB \mid QB \mid QA \mid Q \mid BEC \mid EC \mid BE \mid E$
   - $Q \to a$
   - $W \to b$
   - $E \to c$

   Passo 2:
   - $S \to XC \mid QR_1 \mid QA \mid Q \mid WB \mid W \mid QR_2 \mid QR_3 \mid QB \mid QA \mid BR_4 \mid EC \mid BE \mid E$
   - $A \to QA \mid Q$
   - $B \to WB \mid W$
   - $C \to EC \mid E$
   - $X \to QR_1 \mid QA \mid Q \mid WB \mid W$
   - $Y \to QR_2 \mid QR_3 \mid QB \mid QA \mid Q \mid BR_4 \mid EC \mid BE \mid E$
   - $Q \to a$

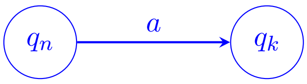
- $W \to b$
- $E \to c$
- $R_1 \to XW$
- $R_2 \to YE$
- $R_3 \to AB$
- $R_4 \to EC$

□

Exercício 4. *Uma outra forma de provar que toda linguagem regular é uma linguagem livre de contexto é propor uma estratégia que: Dado um autômato finito A apresentar uma G_A tal que a mesma simule o comportamente deste. Baseado nessa ideia faça o seguinte:*

1. *Dado um automato finito determinístico A crie uma gramática G_A baseada em seus estados e transições tal que $L(A) = L(G_A)$.*

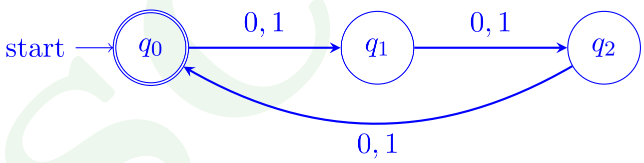
   Resposta. Construa uma GLC da seguinte maneira:
   Seja $A = (Q, \Sigma, q_0, F)$ o AFD de uma linguagem regular L, e G uma GLC , para cado estado de q_n de A temos uma variável correspondente Q_n em G , para cada transição de A da seguinte forma

   criamos uma regra, para a variável correspondente da seguinte maneira $Q_n \to aQ_k$, se q_n for um estado final a regra será $Q_n \to aQ_k \mid \epsilon$.
   E a variável inicial de G será Q_0.
   Assim, $G = (\{Q_n \mid q_n \in Q\}, \Sigma, P, Q_0)$. □

2. *Como exemplo do que propôs no item acima, dê um AFD para a linguagem $L = \{w \in \{0,1\}^* \colon |w| \cong 0 \mod 3\}$.*

   Resposta. AFD para L

   Pelo procedimento de .1 temos as produções

   - $Q_0 \to 0Q_1 \mid 1Q_1 \mid \varepsilon$
   - $Q_1 \to 0Q_2 \mid 1Q_2$
   - $Q_2 \to 0Q_0 \mid 1Q_0$

   E a GLC $G_A = (\{Q_0, Q_1, Q_2\}, \{0, 1\}, P, Q_0)$ □

3. *Prove que o seu procediemtno está corrento, ou seja, mostre que $L(A) = L(G_A)$.*

   Resposta. (1) $w \in L(A) \implies w \in L(G_A)$

   Indução no tamanho de w

   Base: $|w| = 0 \implies w = \varepsilon$ como $w \in L(A)$ e $\delta(q, \varepsilon) = q$ sabemos que o estado inicial q_0 também é final e por definição da gramática existe a produção $Q_0 \to \varepsilon$ e como Q_0 é a variavel inicial

   $$Q_0 \Rightarrow \varepsilon$$

   e $w \in L(G_A)$
   Passo indutivo: Suponha que $|w| = n + 1$ com $w = xa, x \in \Sigma^*$ e $a \in \Sigma$. Por H.I., $x \in L(A)$, ou seja, $\hat{\delta}(q_0, x) = q_{f_1}$ e q_{f_1} é um estado final, dessa maneira sabemos que a derivações na gramática chegam em Q_{f_1}.

Como $\delta(q_{f_1}, a) = q_{f_2}$ é um estado final, por definição da gramática, existem as produções $Q_{f_1} \to \varepsilon \mid aQ_{f_2}$, $Q_{f_2} \to \varepsilon$.
Usando elas temos

$$Q_0 \overset{*}{\Rightarrow} xQ_{f_1} \Rightarrow xaQ_{f_2} \Rightarrow xa = w$$

e $w \in L(G_A)$

(2) $w \in L(G_A) \implies w \in L(A)$

Indução no número de produções

Base: Pela maneira que a G_A é definida por utilizarmos apenas uma produção sabemos que a produção usada foi $Q_0 \to \varepsilon$

$$Q_0 \Rightarrow \varepsilon$$

Assim, sabemos que q_0, que é o estado inicial do AFD, é final e, como $\delta(q_0, \varepsilon) = q_0$, $\varepsilon \in L(A)$
Passo indutivo: Suponha que G_A deriva $w = xa, x \in \Sigma^*$ e $a \in \Sigma$ utilizando $n+1$ produções e que é valido para todas as derivações com menos passos.
Assim

$$Q_0 \overset{*}{\Rightarrow} xQ_{f_1} \Rightarrow xaQ_{f_2} \Rightarrow xa = w$$

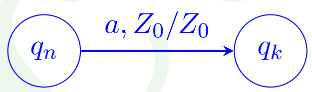
Por H.I., sabemos que $\hat{\delta}(q_0, x) = q_{f_1}$ e q_{f_1} é estado final. Como existem as produções $Q_{f_1} \to aQ_{f_2}$ e $Q_{f_2} \to \varepsilon$ temos a transição $\delta(q_{f_1}, a) = q_{f_2}$ e sabemos que q_{f_2} é estado final. Logo

$$\hat{\delta}(q_0, w) = \hat{\delta}(q_0, xa) = \delta(\hat{\delta}(q_0, x)a) = \delta(q_{f_1}, a) = q_{f_2}$$

e $w \in L(A)$ □

4. *Informalmente como poderíamos expandir a ideia do "item 1" para capturar autômatos de pilha?*

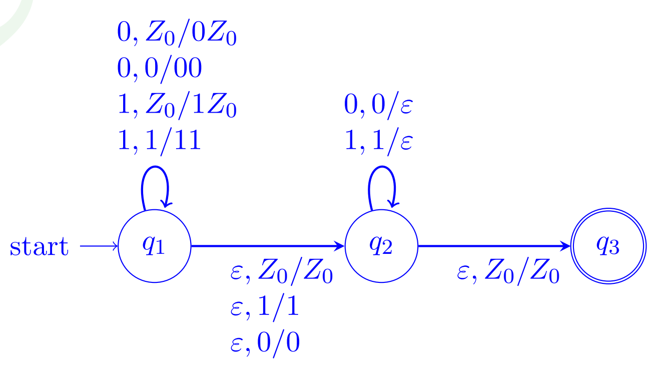
Resposta. Basta adicionar um ", Z_0/Z_0"para toda transição de A



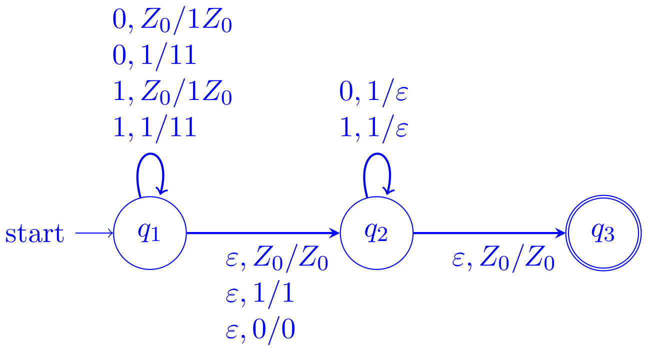
Assim, transformamos o AFD em PDA. □

Exercício 5. *Construa autômatos de pilha com "aceitação por estado final" para cada uma das linguagen do "Exercício 1".*

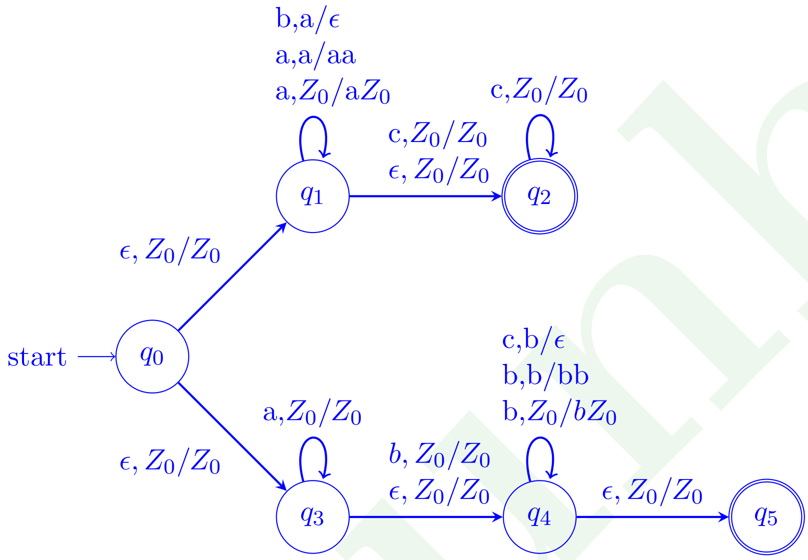
Resposta. 1. PDA por aceitação de estado final



2. PDA por aceitação de estado final



3. PDA por aceitação de estado final



□

Exercício 6. *Um PDA é chamado restrito se, em qualquer transição, ele pode aumentar a altura da pilha por no máximo um símbolo, ou seja, para $\delta(q, a, Z)$ conter (p, γ) deve ocorrer $|\gamma| \leq 2$. Mostre que se P é um PDA, então existe um PDA restrito P_3 tal que $L(P) = L(P_3)$.*

Resposta. □

Exercício 7. *Prove o seguinte teorema comentado em sala:*
Se P é um PDA, e se $(q, xw, \alpha) \overset{*}{\underset{P}{\vdash}} (p, yw, \beta)$, então também é verdade que $(q, x, \alpha) \overset{*}{\underset{P}{\vdash}} (p, y, \beta)$

Resposta. □

Exercício 8. *Suponha que você deseja fazer uma modificação no autômato de pilha para virar um autômato de fila. A fila é uma estrutura de dados em que os dados são organizados numa fila, i.e., o primeiro elemento que vc insere é o primeiro elemento a ser retirado, ou seja, diferentemente da pilha que retiramos sempre do "topo", na fila inserimos no "final" e retiramos na "frente". Resolva os seguintes exercícios:*

1. *Defina formalmente seu autômato de fila e explique como deve funcionar sua representação gráfica.*
   Obs: *Não esqueça de dizer como funciona o símbolo indicador de fila vazia.*
2. *Apresente como o autômato de fila deveria reconhecer palavras da linguagem $L = \{w1^{|w|} : w \in \{0,1\}^*\}$*
3. *Apresente duas linguagems N e M tal que existe um PDA para N, mas não existe um automato de fila para N e não existe um PDA para M e existe um automato de fila para M.*
   Obs: *Observe que a linguagem L não pode ser nem M nem N, pois L é reconhecida pelos dois modelos.*

Resposta. □