Gramática livre-do-contexto

↳ Um método poderoso de descrever linguagens

↳ Estudamos AF e REGEX, descrevem muitas linguagens, mas algumas linguagens simples não podem ser representadas:

ex: $L = \{0^n 1^n \mid n \geq 0\}$

↳ As gramáticas livres-do-contexto podem descrever certas características que têm uma estrutura recursiva! — VAV —

↳ foram primeiro utilizadas no estudo de linguagens humanas.

↳ Uma maneira de entender o relacionamento de termos tais como: nome, verbo e preposição e suas respectivas frases levam a uma recursão natural, porque frases nominais podem aparecer dentro de frases verbais e vice-versa.

↳ Aplicação: Especificação e compilação de linguagens de programação.

↳ Uma gramática de linguagem de programação frequentemente aparece como uma referência para pessoas tentando aprender a sintaxe da linguagem.

primeiramente, veremos Autômatos com <u>pilha</u>

↳ São como autômatos finitos não-determinísticos, mas tem um componente extra chamado PILHA

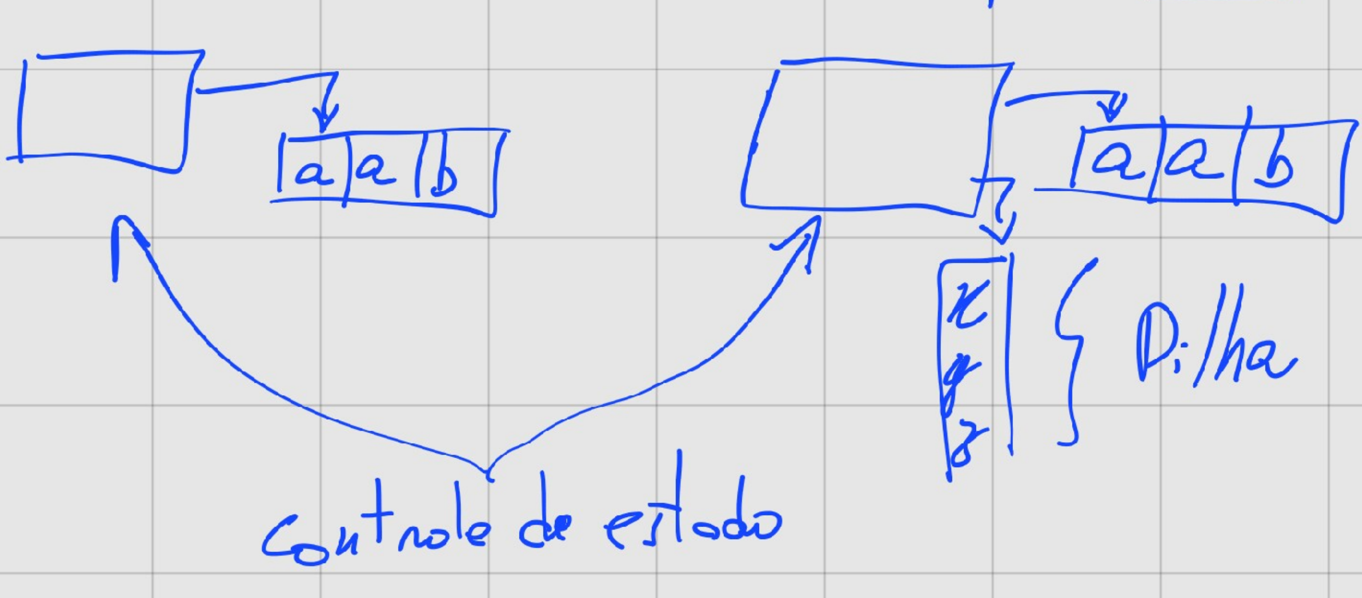
↳ A pilha provê memória adicional além da quantidade finita disponível no controle. A pilha permite que o autômato com pilha reconheça algumas linguagens não regulares.

↳ São equivalentes a gramáticas livre-do-contexto

```
for(int i=0; i<strlen(buf); i++)
{
}
```

-O2

AF AP



Um AP pode escrever símbolos na fita e lê-los de volta mais tarde.

↳ Em qualquer momento, o símbolo no topo da pilha pode ser lido e removido

↳ Escrever na pilha: empilhar (push)

↳ Remover símbolo na pilha: desempilhando (pop)

↳ Quais posições podem ser lidas e modificadas?

topo!

Definição formal

Um autômato com pilha é uma 6-upla $(Q, \Sigma, \Gamma, \delta, q_0, F)$, onde Q, Σ, Γ e F são todos conjuntos finitos, e

1. Q é conjunto de estados
2. Σ é alfabeto
3. Γ é alfabeto de pilha
4. $\delta : Q \times \Sigma_\varepsilon \times \Gamma_\varepsilon \rightarrow P(Q, \Gamma_\varepsilon)$ é função de transição
5. $q_0 \in Q$ é o estado inicial
6. $F \subseteq Q$ é conjunto de estados finais

Um AP $M = (Q, \Sigma, \Gamma, \delta, q_0, F)$ computa da seguinte maneira. Ele aceita a entrada w se w puder ser escrita como

$w = w_1 w_2 \ldots w_m$

onde cada $w_i \in \Sigma_\varepsilon$ e existem uma sequência de estados $r_0, r_1, \ldots, r_m \in Q$ e cadeias $s_0, s_1, \ldots, s_m \in \Gamma^*$ que satisfazem as três condições a seguir.

1. $r_0 = q_0$ e $s_0 = \varepsilon$

   ↳ M inicia no estado inicial e com a pilha vazia

2. Para $i = 0, \ldots m-1$, Temos $(r_{i+1}, b) \in \delta(r_i, w_{i+1}, a)$ onde $s_i = at$ e $s_{i+1} = bt$ para algum $a, b \in \Gamma_\varepsilon$ e $t \in \Gamma^*$

↳ Esta condição afirma que M se move conforme o estado, a pilha e o próximo símbolo de entrada.

3. $r_m \in F$

Exemplo

$L = \{0^n 1^n \mid n \geq 0\}$. $M_1 = (Q, \Sigma, \Gamma, \delta, q_1, F)$

$Q = \{q_1, q_2, q_3, q_4\}$

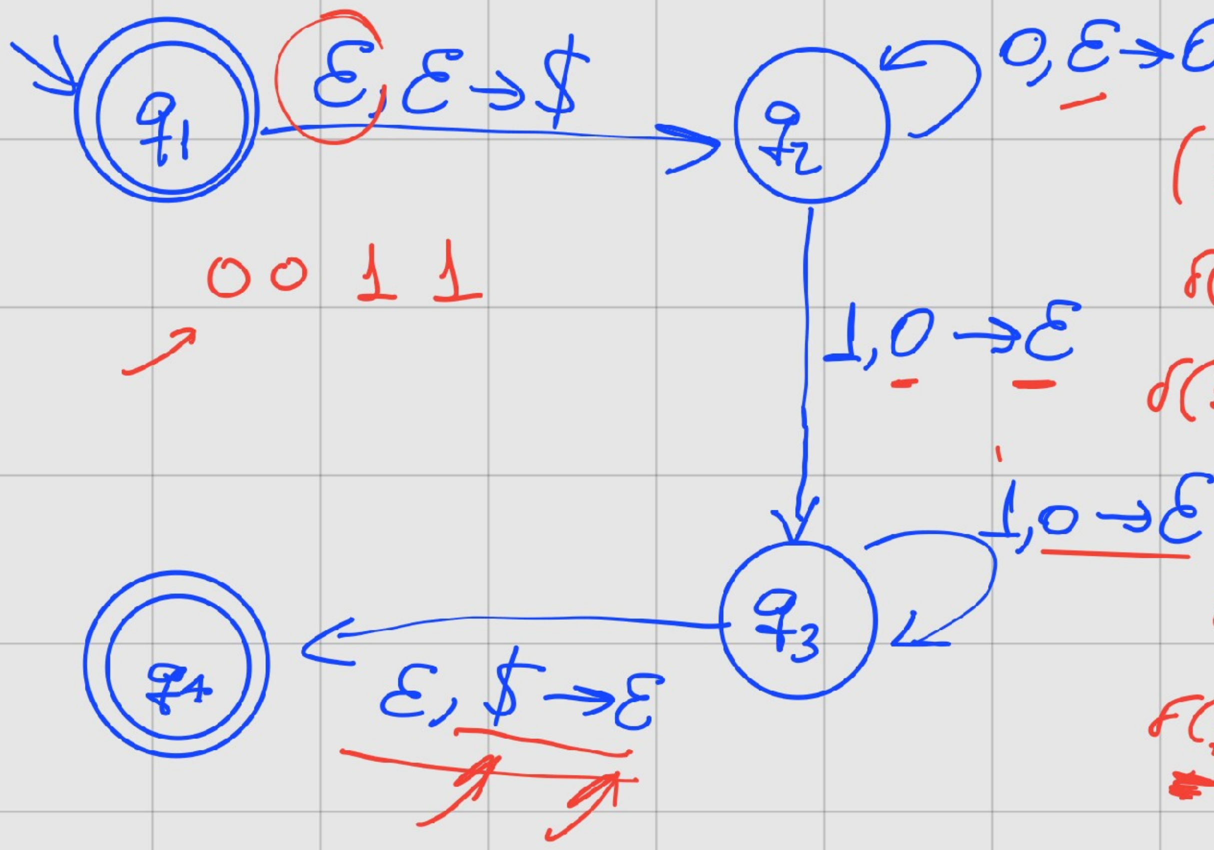
$\Sigma = \{0, 1\}$

$\Gamma = \{0, \$\}$

$F = \{q_1, q_4\}$

δ

	0			1			ε		
	0	\$	ε	0	\$	ε	0	\$	ε
q_1									$\{(q_2, \$)\}$
q_2			$\{(q_2, 0)\}$	$\{(q_3, \varepsilon)\}$					
q_3				$\{(q_3, \varepsilon)\}$			$\{(q_4, \varepsilon)\}$		
q_4									



S

$(q_2, 0011, \$)$ | $

$\delta(q_2, 011, 0\$)$

$\delta(q_2, 11, 00\$)$

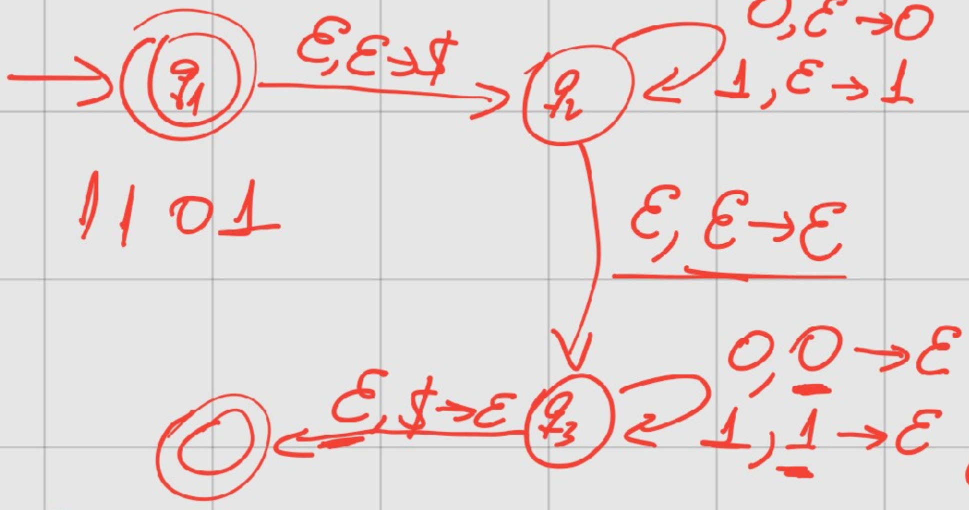
$\delta(q_3, 1, 0\$)$

$\delta(q_3, \varepsilon, \$)$

$\delta(q_4, \varepsilon, \varepsilon)$

exemplo

$L = \{ w w^R \mid w \in \{0,1\}^* \}$



1101

0110

δ(q2, 0110, $)

δ(q2, 110, 0$)

δ(q2, 10, 10$)

δ(q3, ε10, 10$)

δ(q3, 0, 0$)

δ(q3, ε, $)

δ(q2, 1101, $)

δ(q2, 101, 1$)

δ(q3, 101, 1$)

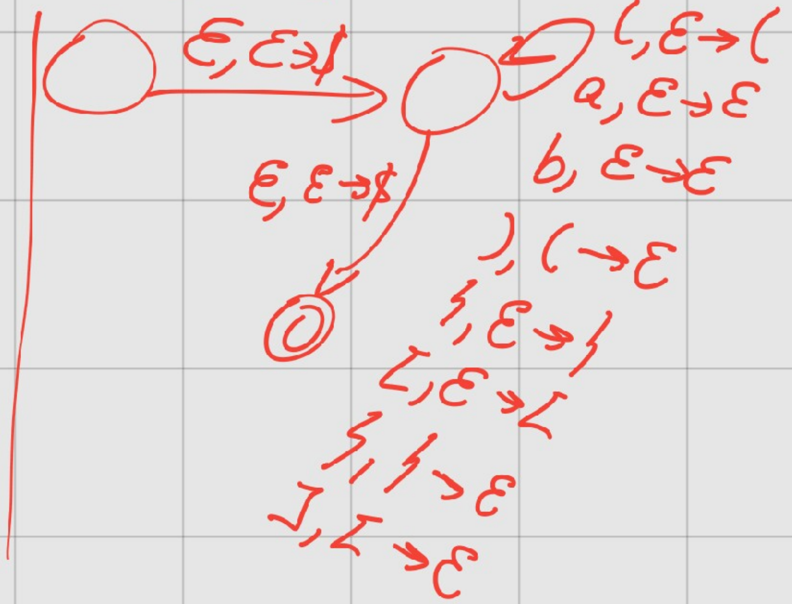
δ(q3, 01, $)

δ(q4, 01, ε)

$L = \{ (^n w)^n \mid w \in \{ (,), a, b \}$, os parentese estão balanceados $\}$

$\{ \quad \}$

$[\quad]$



(a(

(a[a{a}])

( ] (}

Gramática Livres-do-contexto

ex:

$$A \to 0A1$$
$$A \to B$$
$$B \to \#$$

→ Uma gramática consiste de uma coleção de REGRAS de Substituição, também denominadas PRODUÇÕES.

→ Cada regra aparece como uma linha na gramática, compreendendo um símbolo e uma cadeia separados por uma seta

→ O símbolo é chamado de VARIÁVEL

→ A cadeia é constituída de variáveis e outros símbolos chamados Terminais.

→ geralmente VARIÁVEL é representado por uma letra maiúscula

→ terminais são análogos ao alfabeto de entrada

→ Uma variável é designada como variável inicial.

→ Você usa a gramática para descrever uma linguagem gerando cada cadeia da seguinte maneira:

1. Escreva a variável inicial
2. Encontre uma variável que esteja escrita e uma regra que começa com essa variável. Substitua a variável escrita pelo lado direito dessa regra.
3. Repita o passo 2 até que não reste nenhuma variável.

→ A sequência de substituição é denominada

DERIVAÇÃO

exemplo: 000#111 é aceita por G1?

G1:

$A \rightarrow 0A1$

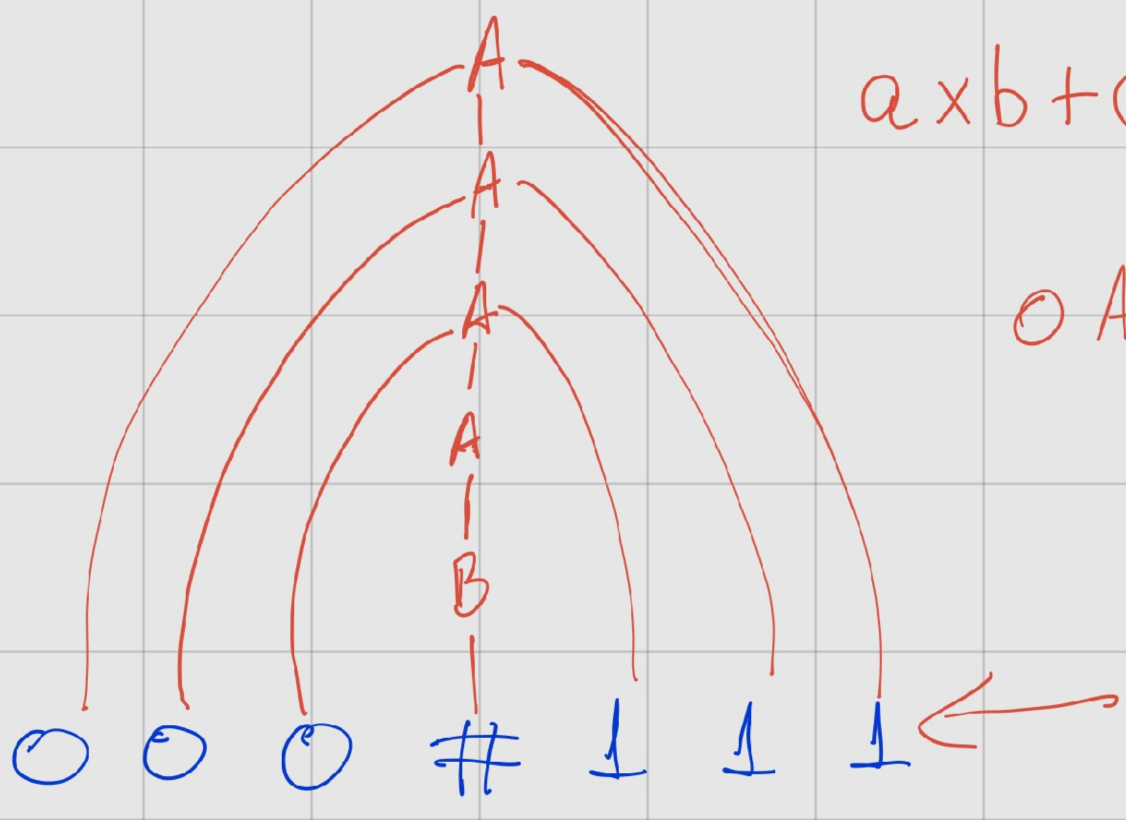
$A \rightarrow B$

$B \rightarrow \#$

Uma derivação

$A \Rightarrow 0A1 \Rightarrow 00A11 \Rightarrow 000A111 \Rightarrow 000B111 \Rightarrow 000\#111$

Também podemos representar com uma árvore sintática:



a x b + c

0A1

O conjunto de todos cadeias gerados dessa maneira constitui a Linguagem da gramática. Escrevemos $L(G1)$ para a linguagem da gramática G_1. $L(G_n)$

$L(G_1)$ é $\{0^n \# 1^n / n \geq 0\}$

Qualquer linguagem que pode ser gerada por alguma gramática livre-do-contexto é chamada de uma linguagem livre-do-contexto (LLC).

Podemos abreviar várias regras com a mesma variável no lado esquerdo em uma única linha, usando o símbolo "|"

G1:

$A \to 0A1$

$A \to B$

$B \to \#$

G1:

$A \to 0A1 \mid B$

$B \to \#$

Temos G2 como um exemplo de LLC, que descreve um fragmento da língua inglesa

⟨SENTENCE⟩ → ⟨NOUN-PHRASE⟩⟨VERB-PHRASE⟩ 1
⟨NOUN-PHRASE⟩ → ⟨CMPLX-NOUN⟩ | ⟨CMPLX-NOUN⟩⟨PREP-PHRASE⟩ 2 3
⟨VERB-PHRASE⟩ → ⟨CMPLX-VERB⟩ | ⟨CMPLX-VERB⟩⟨PREP-PHRASE⟩ 4 5
⟨PREP-PHRASE⟩ → ⟨PREP⟩⟨CMPLX-NOUN⟩ 6
⟨CMPLX-NOUN⟩ → ⟨ARTICLE⟩⟨NOUN⟩ 7
⟨CMPLX-VERB⟩ → ⟨VERB⟩ | ⟨VERB⟩⟨NOUN-PHRASE⟩ 8 9
⟨ARTICLE⟩ → a | the 10 11
⟨NOUN⟩ → boy | girl | flower 12 13 14
⟨VERB⟩ → touches | likes | sees 15 16 17
⟨PREP⟩ → with 18

→ G2 possui:

→ 10 variáveis

→ 27 Terminais (alfabeto inglês-padrão + caracter de espaço)

→ 18 regras

$\langle$SENTENCE$\rangle \rightarrow \langle$NOUN-PHRASE$\rangle\langle$VERB-PHRASE$\rangle$
$\langle$NOUN-PHRASE$\rangle \rightarrow \langle$CMPLX-NOUN$\rangle \mid \langle$CMPLX-NOUN$\rangle\langle$PREP-PHRASE$\rangle$
$\langle$VERB-PHRASE$\rangle \rightarrow \langle$CMPLX-VERB$\rangle \mid \langle$CMPLX-VERB$\rangle\langle$PREP-PHRASE$\rangle$
$\langle$PREP-PHRASE$\rangle \rightarrow \langle$PREP$\rangle\langle$CMPLX-NOUN$\rangle$
$\langle$CMPLX-NOUN$\rangle \rightarrow \langle$ARTICLE$\rangle\langle$NOUN$\rangle$
$\langle$CMPLX-VERB$\rangle \rightarrow \langle$VERB$\rangle \mid \langle$VERB$\rangle\langle$NOUN-PHRASE$\rangle$
$\langle$ARTICLE$\rangle \rightarrow$ a | the
$\langle$NOUN$\rangle \rightarrow$ boy | girl | flower
$\langle$VERB$\rangle \rightarrow$ touches | likes | sees
$\langle$PREP$\rangle \rightarrow$ with

exemplos:

a boy sees

the boy sees a flower

a girl with a flower likes the boy

$\langle$SENTENCE$\rangle \Rightarrow \langle$NOUN-PHRASE$\rangle\langle$VERB-PHRASE$\rangle$
$\Rightarrow \langle$CMPLX-NOUN$\rangle\langle$VERB-PHRASE$\rangle$
$\Rightarrow \langle$ARTICLE$\rangle\langle$NOUN$\rangle\langle$VERB-PHRASE$\rangle$
$\Rightarrow$ a $\langle$NOUN$\rangle\langle$VERB-PHRASE$\rangle$
$\Rightarrow$ a boy $\langle$VERB-PHRASE$\rangle$
$\Rightarrow$ a boy $\langle$CMPLX-VERB$\rangle$
$\Rightarrow$ a boy $\langle$VERB$\rangle$
$\Rightarrow$ a boy sees

Definição formal:

Uma gramática livre-do-contexto é uma 4-upla (V, Σ, R, S), onde

1. V é o conjunto finito de variáveis
2. Σ é um conjunto finito, disjunto de V, de terminais
3. R é o conjunto de regras
4. $S \in V$ é a variável inicial.

$u\,v\,w \qquad A \to w$

$\Rightarrow uAv \Rightarrow uwv$

Se u, v e w são cadeias de variáveis e terminais, e $A \to w$ é uma regra da gramática, dizemos que

uAv **ORIGINA** uwv, escrito

$uAv \Rightarrow uwv$.

Dizemos que u deriva v, escrito

$u \overset{*}{\Rightarrow} v$, se $u = v$ ou se existe

uma sequência $u_1, u_2, \ldots, u_k$ para $k \geq 0$

$$u \Rightarrow u_1 \Rightarrow u_2 \Rightarrow \ldots \Rightarrow u_k \Rightarrow v$$

$u \to v$

A linguagem da gramática é

$$\{ w \in \Sigma^* \mid S \overset{*}{\Rightarrow} w \}$$

$G_3 = (\{S\}, \{a,b\}, R, S)$.

Sendo R

$$S \rightarrow aSb \mid SS \mid \varepsilon$$

ab aabb a~~b~~a ___

Como montar uma gramática para a linguagem $\boxed{\{0^n 1^n \mid n \geq 0\}} \cup \{1^n 0^n \mid n \geq 0\}$?

$S_1 = 0 S_1 1 \mid \varepsilon$

$S_1 \Rightarrow 0 S_1 1 \Rightarrow 0 \varepsilon 1 \Rightarrow 01$

$S_1 \Rightarrow 0 S_1 1 \Rightarrow 0 0 S_1 1 1 \Rightarrow 0 0 \varepsilon 1 1 \Rightarrow 0011$

$S = S_1 \mid S_2$

$S_2 = 1 S_2 0 \mid \varepsilon$

Forma Normal de Chomsky

Uma GLC está na forma normal de Chomsky se toda regra é da forma

$$A \to BC$$
$$A \to a$$

Onde a é qualquer terminal e A, B e C são quaisquer variáveis — exceto que B e C não podem ser a variável inicial.

Adicionalmente permitimos a regra $S \to \varepsilon$, onde S é a variável inicial.

$$S \to ASA \mid \mathtt{a}B$$
$$A \to B \mid S$$
$$B \to \mathtt{b} \mid \varepsilon$$

$$S_0 \to S$$
$$S \to ASA \mid \mathtt{a}B$$
$$A \to B \mid S$$
$$B \to \mathtt{b} \mid \varepsilon$$

2. Remova as regras ε $B \to \varepsilon$, mostrado à esquerda, e $A \to \varepsilon$, mostrado à direita.

$$S_0 \to S$$
$$S \to ASA \mid \mathtt{a}B \mid \mathbf{a}$$
$$A \to B \mid S \mid \boldsymbol{\varepsilon}$$
$$B \to \mathtt{b} \mid \cancel{\varepsilon}$$

$$S_0 \to S$$
$$S \to ASA \mid \mathtt{a}B \mid \mathtt{a} \mid \boldsymbol{SA} \mid \boldsymbol{AS} \mid \boldsymbol{S}$$
$$A \to B \mid S \mid \cancel{\varepsilon}$$
$$B \to \mathtt{b}$$

3a. Remova regras unitárias $S \to S$, mostrado à esquerda, e $S_0 \to S$, mostrado à direita.

$$S_0 \to S$$
$$S \to ASA \mid \mathtt{a}B \mid \mathtt{a} \mid SA \mid AS \mid \cancel{S}$$
$$A \to B \mid S$$
$$B \to \mathtt{b}$$

$$S_0 \to \cancel{S} \mid \boldsymbol{ASA} \mid \mathbf{a}\boldsymbol{B} \mid \mathbf{a} \mid \boldsymbol{SA} \mid \boldsymbol{AS}$$
$$S \to ASA \mid \mathtt{a}B \mid \mathtt{a} \mid SA \mid AS$$
$$A \to B \mid S$$
$$B \to \mathtt{b}$$

3b. Remova as regras unitárias $A \to B$ e $A \to S$.

$$S_0 \to ASA \mid \mathtt{a}B \mid \mathtt{a} \mid SA \mid AS$$
$$S \to ASA \mid \mathtt{a}B \mid \mathtt{a} \mid SA \mid AS$$
$$A \to \cancel{B} \mid S \mid \mathbf{b}$$
$$B \to \mathtt{b}$$

$$S_0 \to ASA \mid \mathtt{a}B \mid \mathtt{a} \mid SA \mid AS$$
$$S \to ASA \mid \mathtt{a}B \mid \mathtt{a} \mid SA \mid AS$$
$$A \to \cancel{S} \mid \mathtt{b} \mid \boldsymbol{ASA} \mid \mathbf{a}\boldsymbol{B} \mid \mathbf{a} \mid \boldsymbol{SA} \mid \boldsymbol{AS}$$
$$B \to \mathtt{b}$$

4. Converta as regras remanescentes para a forma apropriada acrescentando variáveis e regras adicionais. A gramática final em forma normal de Chomsky, a seguir, é equivalente a G_6. (Na realidade, o procedimento dado no Teorema 2.9 produz diversas variáveis U_i juntamente com várias regras $U_i \to \mathtt{a}$. Simplificamos a gramática resultante usando uma única variável U e a regra $U \to \mathtt{a}$.)

$$S_0 \to AA_1 \mid UB \mid \mathtt{a} \mid SA \mid AS$$
$$S \to AA_1 \mid UB \mid \mathtt{a} \mid SA \mid AS$$
$$A \to \mathtt{b} \mid AA_1 \mid UB \mid \mathtt{a} \mid SA \mid AS$$
$$A_1 \to SA$$
$$U \to \mathtt{a}$$
$$B \to \mathtt{b}$$