Linguagens Formais e Autômatos

Arthur do Prado Labaki – 11821BCC017

Segunda Lista de Exercícios

Parte 2

1) Lema do Bombeamento para linguagens regulares:

Exercício 4.1.1: Prove que as linguagens a seguir não são regulares.

d) $\{0^n1^m2^n \mid n \geq 1\} \mid n$ e m são inteiros arbitrários$\}$.

e) $\{0^n1^m \mid n \leq m\}$.

f) $\{0^n1^{2n} \mid n \geq 1\}$.

d) Assumindo que L é regular, o lema do bombeamento é valido.

Sendo p o valor dado pelo lema e w = 0^p2^p em que p >= 1.

Como |w| ≥ p, w pode ser escrito como uvz.

Qualquer que seja a divisão em que v ≥ 1 e |uv| ≤ p, temos v = 0^t, para algum t ≥ 1 e u = 0^r, para algum r ≥ 0.

Então w' = uvvz em que i = 2, temos que r+2t+p-r-t = p+t. Com isso temos que w' = $0^{p+t}2^p$ e como t tem seu tamanho mínimo de 1, nunca terá a mesma quantidade de 0's e 2's.

Logo uvvz não pertence a L, uma contradição do lema. Então L não é uma linguagem regular.

e) Assumindo que L2 é regular, o lema do bombeamento é valido.

Sendo p o valor dado pelo lema e $w = 0^p1^{p+1}$.

Como $|w| \geq p$, w pode ser escrito como uvz.

Qualquer que seja a divisão em que $v \geq 1$ e $|uv| \leq p$, temos $v = 0^t$, para algum $t \geq 1$ e $u = 0^r$, para algum $r \geq 0$.

Então w' = uvvz em que i = 2, temos que r+2t+p-r-t = p+t. Com isso temos que $w' = 0^{p+t}1^{p+1}$ e como t tem seu tamanho mínimo de 1 ainda pode satisfazer o lema, pois n <= m.

Então w'' = uvvvz em que i = 3, temos que r+3t+p-r-t = p+2t. Com isso temos que $w'' = 0^{p+2t}1^{p+1}$, agora sim, o tamanho mínimo do expoente do 0 será p+2, fazendo n > m.

Logo uvvvz não pertence a L2, uma contradição do lema. Então L2 não é uma linguagem regular.

f) Assumindo que L3 é regular, o lema do bombeamento é valido.

Sendo p o valor dado pelo lema e $w = 0^p1^{2p}$ em que p >= 1.

Como $|w| \geq p$, w pode ser escrito como uvz.

Qualquer que seja a divisão em que $v \geq 1$ e $|uv| \leq p$, temos $v = 0^t$, para algum $t \geq 1$ e $u = 0^r$, para algum $r \geq 0$.

Então w' = uvvz em que i = 2, temos que r+2t+p-r-t = p+t. Com isso temos que $w' = 0^{p+t}1^{2p}$ e como t tem seu tamanho mínimo de 1, a quantidade de 1's não será o dobro dos 0's.

Logo uvvz não pertence a L3, uma contradição do lema. Então L3 não é uma linguagem regular.

2) Simplificação e normalização de Gramáticas Livres de Contexto

* **Exercício 7.1.2:** Comece com a gramática:

$$\begin{aligned} S &\rightarrow ASB \mid \varepsilon \\ A &\rightarrow aAS \mid a \\ B &\rightarrow SbS \mid A \mid bb \end{aligned}$$

a) Elimine as ε-produções.

b) Elimine quaisquer produções unitárias na gramática resultante.

c) Elimine quaisquer símbolos inúteis na gramática resultante.

d) Coloque a gramática resultante na forma normal de Chomsky.

Exercício 7.1.3: Repita o Exercício 7.1.2 para a seguinte gramática:

$$\begin{aligned} S &\rightarrow 0A0 \mid 1B1 \mid BB \\ A &\rightarrow C \\ B &\rightarrow S \mid A \\ C &\rightarrow S \mid \varepsilon \end{aligned}$$

7.1.2)

a) S -> ASB | AB | ε, A -> aAS | aA | a, B -> SbS | b | A | bb

b) S -> ASB | AB | ε, A -> aAS | aA | a, B -> SbS | b | aAS | aA | a | bb

c) S -> ASB | AB | ε, A -> aAS | aA | a, B -> SbS | b | aAS | aA | a | bb

d) S -> X_{AS}B | AB | ε, A -> X_{aA}S | X_aA | a, B -> X_{Sb}S | b | X_{aA}S | X_aA | a | bb,

X_{AS} -> AS, X_{aA} -> X_aA, X_a -> a, X_{Sb} -> SX_b, X_b -> b

7.1.3)

a) S -> 0A0 | 00 | 1B1 | 11 | BB | ε, A -> C, B -> S | A, C -> S

b) S -> 0A0 | 00 | 1B1 | 11 | BB | ε, A -> 0A0 | 00 | 1B1 | 11 | BB | ε, B -> 0A0 | 00 | 1B1 | 11 | BB | ε, C -> 0A0 | 00 | 1B1 | 11 | BB | ε

c) S -> 0A0 | 00 | 1B1 | 11 | BB | ε, A -> 0A0 | 00 | 1B1 | 11 | BB | ε, B -> 0A0 | 00 | 1B1 | 11 | BB | ε

d) S -> $X_{0A}X_0$ | 00 | $X_{1B}X_1$ | 11 | BB | ε, A -> $X_{0A}X_0$ | 00 | $X_{1B}X_1$ | 11 | BB | ε, B -> $X_{0A}X_0$ | 00 | $X_{1B}X_1$ | 11 | BB | ε, X_{0A} -> X_0A, X_0 -> 0, X_{1B} -> X_1B, X_1 -> 1

3) **Ambiguidade**

Seja a gramática a seguir para expressões lógicas

G = ({S}, {∧ , ∨ , ¬ , →, a, b, c}, P, S) e

P = {S → S∧S | S∨S | S→S | ¬S | a | b | c}

a) Remova a ambiguidade considerando as seguintes precedências para os operadores lógicos: {¬ } > { → } > {∧ , ∨}.

Removendo a ambiguidade da gramatica utilizando as regras de precedências, temos:

G2 = ({S, T, F, I}, {∧ , ∨ , ¬, →, a, b, c}, P2, S) em que:

P2 = {S → S∧T | S∨T | T,

T → T → F |F,

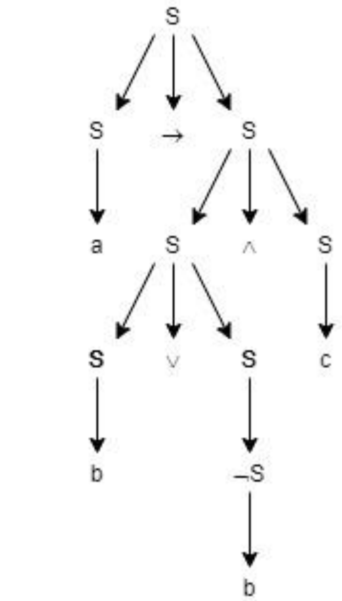
F → ¬ I | I,

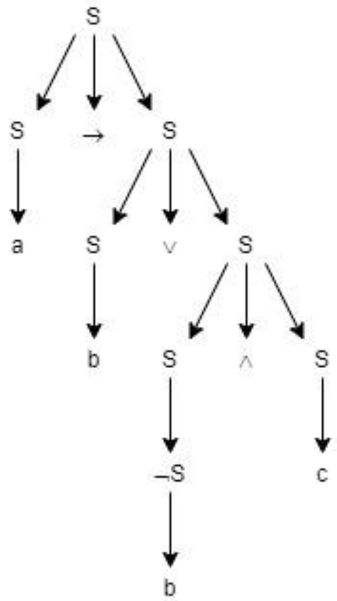
I → a | b | c}

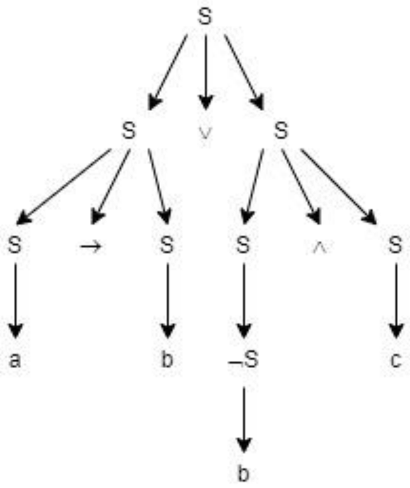
b) Mostre todas as árvores de derivação possíveis nas duas gramáticas (ambígua e não ambígua) para a expressão:

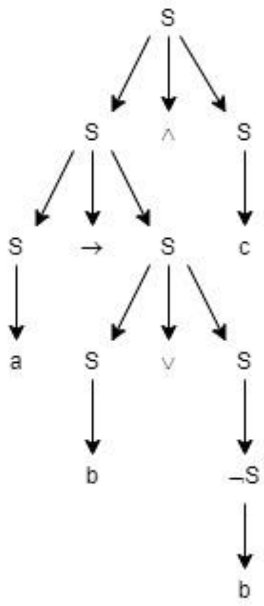
a → b ∨ ¬b ∧ c

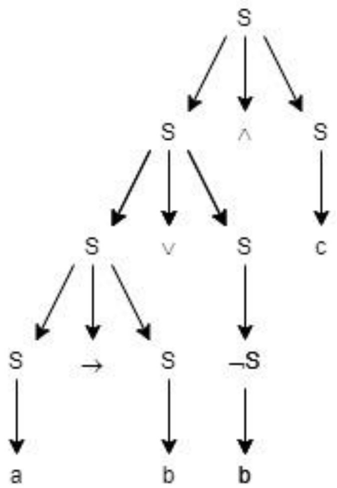
Ambíguas



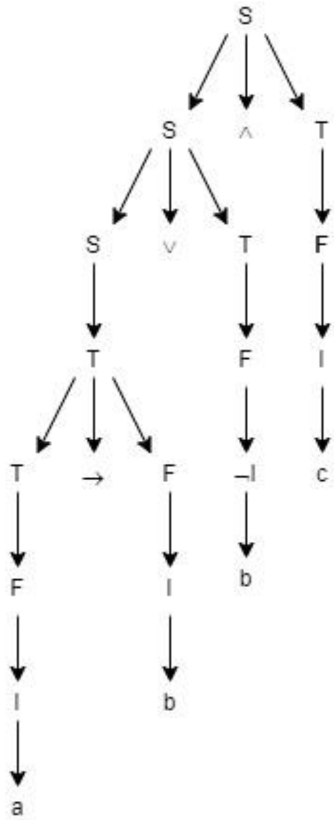








Não Ambígua

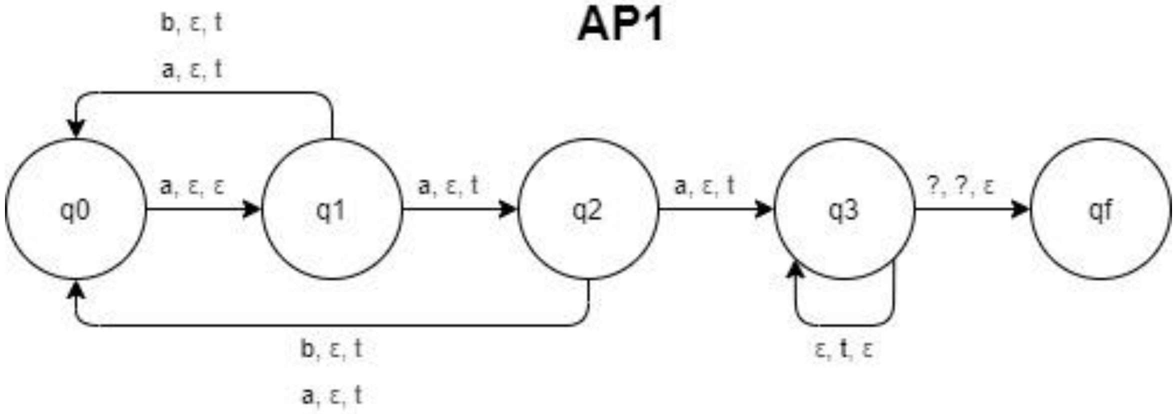


4) Autômato de Pilha

5.2 Construct a PDA equivalent to the following grammar.

$$S \to aAA, \qquad A \to aS \mid bS \mid a.$$

AP1



5.6 Give a grammar for the language $N(M)$ where

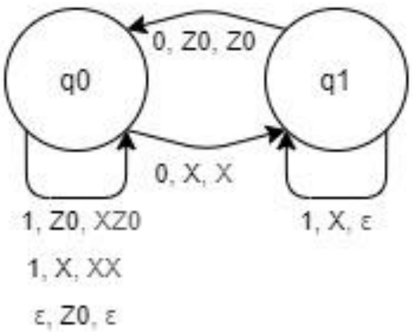
$$M = (\{q_0, q_1\}, \{0, 1\}, \{Z_0, X\}, \delta, q_0, Z_0, \varnothing)$$

and δ is given by

$$\delta(q_0, 1, Z_0) = \{(q_0, XZ_0)\}, \qquad \delta(q_0, \epsilon, Z_0) = \{(q_0, \epsilon)\},$$
$$\delta(q_0, 1, X) = \{(q_0, XX)\}, \qquad \delta(q_1, 1, X) = \{(q_1, \epsilon)\},$$
$$\delta(q_0, 0, X) = \{(q_1, X)\}, \qquad \delta(q_1, 0, Z_0) = \{(q_0, Z_0)\}.$$

AP2

L2 = {w | w é ε ou 1^n01^n0, em que n ≥ 1}



5) Lema do Bombeamento para Linguagens Livres de Contexto

6.1 Show that the following are not context-free languages.

a) $\{a^i b^j c^k \mid i < j < k\}$

d) the set of strings of a's, b's, and c's with an equal number of each

a) Suponha que é livre de contexto, então o lema do bombeamento vale.

Supondo que $w = a^p b^{p+1} c^{p+2}$, $|w| \geq p$, então $w = uxvyz$ em que:

$|xvy| \leq n$, $|xy| \geq 1$ e para todo $i \geq 0$, $ux^i vy^i z$.

Temos dois tipos de casos:

1- Se x e y tem apenas um símbolo cada:

$X = a^i$ e $y = a^j \mid b^j \mid \varepsilon$ $\qquad$ $X = b^i$ e $y = b^j \mid c^j \mid \varepsilon$

$X = c^i$ e $y = c^j \mid \varepsilon$ $\qquad$ $X = \varepsilon$ e $y = a^j \mid b^j \mid c^j$

Com $i, j \geq 1$. Em qualquer um desses casos, uvz terá quantidades não pertencente as regras, como ter mais a's do que b's.

2- Se x e y tem dois símbolos:

$X = a^i b^j$ e $y = b^k \mid \varepsilon$ $\qquad$ $X = b^i c^j$ e $y = c^k \mid \varepsilon$

$X = a^i \mid \varepsilon$ e $y = a^j b^k$ $\qquad$ $X = b^i \mid \varepsilon$ e $y = b^j c^k$

Com $i, j, k \geq 1$. Em qualquer um desses casos, uvz terá quantidade que também não pertence as regras.

Nos dois casos, não importa como w é dividida, temos que $ux^0 vy^0 z$ não pertence a linguagem. Logo a linguagem não é livre de contexto.

d) Suponha que é livre de contexto, então o lema do bombeamento vale.

Supondo que $w = a^p b^p c^p$, $|w| \geq p$, então $w = uxvyz$ em que:

$|xvy| \leq n$, $|xy| \geq 1$ e para todo $i \geq 0$, $ux^i vy^i z$.

Temos dois tipos de casos:

1- Se x e y tem apenas um símbolo cada:

X = a^i e y = a^j | b^j | ε

X = b^i e y = b^j | c^j | ε

X = c^i e y = c^j | ε

X = ε e y = a^j | b^j | c^j

Com i, j ≥ 1. Em qualquer um desses casos, ux^2vy^2z terá quantidades diferentes dos três símbolos.

2- Se x e y tem dois símbolos:

X = a^ib^j e y = b^k | ε

X = b^ic^j e y = c^k | ε

X = a^i | ε e y = a^jb^k

X = b^i | ε e y = b^jc^k

Com i, j, k ≥ 1. Em qualquer um desses casos, ux^2vy^2z terá os símbolos fora de ordem.

Nos dois casos, não importa como w é dividida, temos que ux^2vy^2z não pertence a linguagem. Logo a linguagem não é livre de contexto.

3.14 **Explique intuitivamente por que e prove que as seguintes linguagens não são Livres do Contexto:**

b) $L_{11} = \{a^nb^na^m \mid n \geq 0, m \geq 0 \text{ e } n \neq m\}$

Suponha que é livre de contexto, então o lema do bombeamento vale.

Supondo que w = a^pb^p, sendo m = 0, |w| ≥ p, então w = uxvyz em que:

|xvy| ≤ n, |xy| ≥ 1 e para todo i ≥ 0, ux^ivy^iz.

Temos dois tipos de casos:

1- Se x e y tem apenas um símbolo cada:

X = a^i e y = a^j | b^j | ε

X = b^i e y = b^j | ε

X = ε e y = a^j | b^j

Com i, j ≥ 1. Em qualquer um desses casos, ux^2vy^2z terá quantidades diferentes dos três símbolos.

2- Se x e y tem dois símbolos:

X = $a^i b^j$ e y = b^k | ε X = a^i | ε e y = $a^j b^k$

Com i, j, k ≥ 1. Em qualquer um desses casos, ux^2vy^2z terá os símbolos fora de ordem.

Bem semelhante ao exercício anterior, pois escolhi um w que facilita.

Nos dois casos, não importa como w é dividida, temos que ux^2vy^2z não pertence a linguagem. Logo a linguagem não é livre de contexto.

6) **Máquina de Turing**

4.4 Para cada uma das linguagens abaixo, desenvolva uma Máquina de Turing que a reconheça. Sugere-se que, pelo menos três sejam do tipo com Fita Limitada.

a) $L_1 = \{a^n b^n c^n \mid n \geq 0\}$

d) $L_4 = \{wcw \mid$ w é palavra de $\{a, b\}^*\}$

a)

M1 = ({q0, q1, q2, q3, q4, q5}, {a, b, c}, {a, b, c, x, y, z, B}, δ, q0, {q5}) onde:

δ(q0, a) = (q1, x, →), δ(q0, B) = (q5, B, →), δ(q1, a) = (q1, a, →),

δ(q1, b) = (q2, y, →), δ(q2, b) = (q2, b, →), δ(q2, c) = (q3, z, ←),

δ(q3, b) = (q3, b, ←), δ(q3, y) = (q3, y, ←), δ(q3, a) = (q3, a, ←),

δ(q3, x) = (q0, x, →), δ(q1, y) = (q1, y, →), δ(q2, z) = (q2, z, →),

δ(q3, z) = (q3, z, ←), δ(q0, y) = (q4, y, →), δ(q4, y) = (q4, y, →),

δ(q4, z) = (q4, z, →), δ(q4, B) = (q5, B, →).

O esquema dessa MT é mudar a para x, b para y e c para z, um caractere de cada por ida na fita, e voltar até o primeiro valor a, onde irá repetir. Quando não tiver mais a (também não deve ter b’s nem c’s) a fita vai para a direita e terminara em q5. Também aceita cadeia vazia.

b)

$M2 = (\{q0, q1, q2, q3, q4, q5, q6, q7\}, \{a, b, c\}, \{a, b, c, x, y, B\}, \delta, q0, \{q7\})$

$\delta(q0, a) = (q1, x, \rightarrow)$, $\delta(q0, b) = (q2, y, \rightarrow)$, $\delta(q0, x) = (q0, x, \rightarrow)$,

$\delta(q0, y) = (q0, y, \rightarrow)$, $\delta(q1, a) = (q1, a, \rightarrow)$, $\delta(q1, b) = (q1, b, \rightarrow)$,

$\delta(q1, c) = (q3, c, \rightarrow)$, $\delta(q2, a) = (q2, a, \rightarrow)$, $\delta(q2, b) = (q2, b, \rightarrow)$,

$\delta(q2, c) = (q4, c, \rightarrow)$, $\delta(q3, a) = (q5, x, \leftarrow)$, $\delta(q3, x) = (q3, x, \rightarrow)$,

$\delta(q3, y) = (q3, y, \rightarrow)$, $\delta(q4, b) = (q5, y, \leftarrow)$, $\delta(q4, x) = (q4, x, \rightarrow)$,

$\delta(q4, y) = (q4, y, \rightarrow)$, $\delta(q5, a) = (q5, a, \leftarrow)$, $\delta(q5, b) = (q5, b, \leftarrow)$,

$\delta(q5, x) = (q5, x, \leftarrow)$, $\delta(q5, y) = (q5, y, \leftarrow)$, $\delta(q5, c) = (q5, c, \leftarrow)$,

$\delta(q5, B) = (q0, B, \rightarrow)$, $\delta(q0, c) = (q6, c, \rightarrow)$, $\delta(q6, a) = (q6, a, \rightarrow)$,

$\delta(q6, b) = (q6, b, \rightarrow)$, $\delta(q6, x) = (q6, x, \rightarrow)$, $\delta(q6, y) = (q6, y, \rightarrow)$,

$\delta(q6, B) = (q7, B, \rightarrow)$.

O esquema dessa MT é ler a, ir até c, ler o próximo a, e voltar para o começo da fita. Os dois a’s lidos são mudados para x. O mesmo acontece caso ler b, mas eles são mudados para y. Quando terminar de ler w, a máquina irá buscar o final da fita, terminando em q7.