

Universidade Federal de Santa Catarina / Centro Tecnológico
Departamento de Informática e de Estatística
INE5421 - Linguagens Formais e Compiladores (13/2)

Lista de Exercícios nº 1

0 – Você deve saber ... (não precisa entregar a resposta desta questão)

- a) A diferença entre problemas decidíveis e problemas indecidíveis.
- b) De que forma problemas computacionais em geral, podem ser vistos como problemas de Linguagem.
- c) A relação entre problemas decidíveis e problemas indecidíveis com a teoria das linguagens formais.
- d) Diferenças entre aspectos léxicos, sintáticos e semânticos de linguagens de programação.
- e) A estrutura geral de um compilador e as funções básicas de cada fase.
- f) As formas de implementação de um compilador, e os critérios usados na escolha de uma delas para uma determinada implementação.
- g) A importância do uso de Código Intermediário na implementação de um compilador.
- g) As técnicas de otimização de código mais comuns.
- h) A definição formal de sentença, forma sentencial, gramática, linguagem, derivação e redução.
- i) Mostrar que GSC são recursivas.

1 – Diferencie Linguagem vazia, finita e infinita, exemplificando-as através de GR.

2 - Construa uma gramática G |

- a) G não seja recursiva mas possua uma GR equivalente $G1 \mid L(G1)$ seja Regular, finita e não contenha a sentença vazia.
- b) G seja recursiva, $L(G)$ não seja Livre de Contexto e $L(G)$ seja finita e contenha a sentença vazia..

3 – a) Mostre através de exemplos que a União e a Concatenação de duas Linguagens ($L1$ e $L2$) de um mesmo tipo, resultam sempre em Linguagens desse mesmo tipo.

b) Toda linguagem finita é uma L.L.C. ? é uma L. Regular ? Justifique sua resposta.

c) Proponha um algoritmo para verificar se a Linguagem gerada por uma dada GLC é vazia, finita ou infinita.

4 – a) Construa, se possível, Gramáticas $G1$ e $G2 \mid L(G1) \cup L(G2) = VT^* \wedge L(G1) \cap L(G2) = \varnothing$

b) Construa, se possível, Gramáticas $G1$, $G2$ e $G3 \mid G1$ seja GR, $G2$ seja GLC, $G3$ seja GSC $\wedge L(G3) \subseteq L(G2) \subseteq L(G1) \wedge L(G3)$ não seja LC $\wedge L(G2)$ não seja Regular.

c) Mostre, genericamente e através de exemplos, que toda linguagem gerada por uma GLC com produções da forma $A \rightarrow x B \mid x$ (onde $x \in VT^+$), também pode ser gerada por uma G.R..

5 – Construa uma gramática G, do maior tipo possível |

a) $L(G) = \{ a^n b^m c^k \mid n, m, k \geq 0 \wedge m \neq n = k \}$

b) $L(G) = \{ x \mid x \in (a, b)^* \wedge x$ não seja um palíndromo}

c) $L(G) = \{ x \mid x \in (a, b)^* \wedge \#$ de ocorrências de “ab” = # de ocorrências de “ba” }

d) $L(G) = \{ a^n b^m c^p \mid n, m, p \geq 0 \wedge n \leq (m + p) \leq 2 * n \}$

e) $L(G) = \{ a^n b^m c^p d^q \mid n, m, p, q \geq 0 \wedge n + q = m + p \}$

f) $L(G) = \{ 0^n x \mid x \in (0, 1)^*, n \geq 1 \wedge x$ possua pelo menos n 1's}

6 - Construa uma Gramática Regular G |

- a) $L(G) = \{ a^n (b,c)^* \mid n + \#c's \text{ é par} \wedge \text{ não existem } b's \text{ consecutivos} \}$
- b) $L(G) = \{ x \mid x \in (0,1)^* \wedge x \text{ seja um binário divisível por } 3 \}$
- c) $L(G) = \{ x \mid x \in (a,b)^* \wedge |x| \text{ seja par} \wedge \#b's \text{ seja ímpar} \}$
- d) $L(G) = \{ x \mid x \in (1,2,3)^* \wedge \Sigma \text{ dos elementos de } x \text{ seja divisível por } 3 \}$

7 - Construa uma Gramática Livre de Contexto (GLC) G |

- a) $L(G) = \{ a^n (b,c)^* d^m \mid 0 \leq n \leq m \wedge \#c's = 2 * \#b's \}$
- b) $L(G) = \{ a^n b^m c^k d^l \mid n,m,k,l \geq 0, n+k = m+l \}$
- c) $L(G) = \{ w c x \mid w, x \in (a,b)^* \wedge w^R \text{ é uma sub-cadeia de } x \}$
- d) $L(G) = \{ a^n b^m c^j \mid n,m,j \geq 0 \wedge n > m \vee m < n+j \}$

8 – Construa uma GLC que especifique:

- a) a sintaxe de uma Linguagem de programação contendo métodos (com e sem parâmetros) e outras declarações (generalizadas por “decl”) e chamadas (ativações) de métodos e outros comandos (generalizados por “com”). OBS.: diferencie sintaticamente parâmetros por “valor” e por “referência” e considere apenas parâmetros de tipos simples.
- b) a sintaxe de uma lista de comandos contendo qualquer combinação dos seguintes comandos: **atribuição, if-then, if-then-else e while-do**.

9 - Construa uma GSC (Gramática Sensível ao Contexto) G |

- a) $L(G) = \{ a^n b^m c^k d^l \mid n, m, k, l \geq 0 \wedge n = k \neq m+l \}$
- b) $L(G) = \{ a^n (b,c)^* d^m \mid n,m \geq 0 \wedge n > m \wedge \#c's + \#b's > m \}$
- c) $L(G) = \{ x \mid x \in (a,b,c)^* \wedge \#a's = \#b's \neq \#c's \}$
- d) $L(G) = \{ a^n (b,c,d)^* \mid n \geq 0 \wedge n < (\#b's + \#d's) > \#c's \}$

10 - Seja G a gramática definida por:

- $S \rightarrow C D$
- $C \rightarrow a C A \mid b C B$
- $A D \rightarrow a D$
- $B D \rightarrow b D$
- $A a \rightarrow a A$
- $A b \rightarrow b A$
- $B a \rightarrow a B$
- $B b \rightarrow b B$
- $C \rightarrow \epsilon$
- $D \rightarrow \epsilon$

Pede-se :

- a) Verifique se $x = aaaa$, $y = baab$ e $w = abab \in L(G)$:
- b) Determine $L(G)$;
- c) Existe algoritmo para verificar se uma sequência qualquer é gerada por G? Justifique.
- d) Construa, se possível, uma GSC $G_1 \mid L(G_1) = L(G)$

11 - Dada a gramática G :

$$\begin{aligned} S &\rightarrow a S A \mid a S B \mid a S \mid S B \mid c \\ AB &\rightarrow BA \\ BA &\rightarrow AB \\ A &\rightarrow a \\ B &\rightarrow b \end{aligned}$$

Pede-se:

- Verifique se $x = aacba$ e $y = acaba \in L(G)$ - usando o algoritmo dado em aula (seção II.7);
- Determine $L(G)$;
- Construa, se possível, uma GLC $G_1 \mid L(G_1) = L(G)$
- Construa, se possível, uma GR $G_2 \mid L(G_2) = L(G)$
- Construa $G_1 \mid L(G_1) = L(G) \cup \{ \epsilon \}$

12 - Determine $L(G)$ onde G é dada por:

a)
$$\begin{aligned} S &\rightarrow 0 A \mid 1 B \mid 0 \mid 1 \\ A &\rightarrow 0 C \mid 1 B \mid 0 \mid 1 \\ B &\rightarrow 0 A \mid 1 D \mid 0 \mid 1 \\ C &\rightarrow 1 B \mid 1 \\ D &\rightarrow 0 A \mid 0 \end{aligned}$$

b)
$$\begin{aligned} S &\rightarrow a A \mid b S \mid a \\ A &\rightarrow a B \mid b A \mid b \\ B &\rightarrow a C \mid b B \\ C &\rightarrow a D \mid b C \\ D &\rightarrow a E \mid b D \mid a \\ E &\rightarrow a S \mid b E \mid b \end{aligned}$$

c)
$$\begin{aligned} S &\rightarrow a A C \mid b B C \\ A &\rightarrow C A C \mid a \\ B &\rightarrow B C B \mid b \\ C &\rightarrow a \mid b \end{aligned}$$

$L(G)$ é também uma LR? Se sim, construa uma GR $G_1 \mid L(G_1) = L(G)$.

d)
$$\begin{aligned} S &\rightarrow a S \mid S C \mid c S A \mid b \\ AC &\rightarrow CA & CA &\rightarrow AC \\ bC &\rightarrow bc & bA &\rightarrow ba \\ aA &\rightarrow aa & aC &\rightarrow ac \\ cA &\rightarrow ca & cC &\rightarrow cc \end{aligned}$$

$L(G)$ é também uma LLC? Se sim, construa uma GLC $G_1 \mid L(G_1) = L(G)$.

e)
$$\begin{aligned} S &\rightarrow AD \\ A &\rightarrow a A C \mid a A \mid a \\ D &\rightarrow B D d \mid D d \mid d \\ CB &\rightarrow BC \\ BC &\rightarrow CB \\ B &\rightarrow b \\ C &\rightarrow c \end{aligned}$$

$L(G)$ é também uma LLC? Se sim, construa uma GLC $G_1 \mid L(G_1) = L(G)$.