

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

Lista de Exercícios nº 1 (12/1)

- 1 – a) Diferencie e exemplifique problemas decidíveis e problemas indecidíveis, correlacionando-os com a teoria das linguagens formais.
b) De que forma problemas computacionais em geral, podem ser vistos como problemas de Linguagem? Explique e exemplifique.
c) Diferencie Compiladores e Interpretadores, citando vantagens e desvantagens de cada um e identifique situações (contextos) em que cada um é mais vantajoso.
d) Justifique os critérios mandatórios para escolha da forma de implementação de um compilador e identifique situações onde cada uma é mais vantajosa.
e) Diferencie e exemplifique aspectos léxicos, sintáticos e semânticos de linguagens de programação, correlacionando-os com os tipos de gramáticas estudados.
f) O que é Código Intermediário? Qual sua importância na implementação de processadores de Linguagens de programação? Em que consiste otimizá-lo?
- 2 – a) A União e a Concatenação de duas Linguagens (L_1 e L_2) de um mesmo tipo, resultam sempre em Linguagens desse mesmo tipo? Se sim, exemplifique; senão, justifique.
b) Mostre, genericamente e através de exemplos, que toda linguagem gerada por uma GLC com produções da forma $A \rightarrow x B \mid x$ (onde $A, B \in V_N \wedge x \in V_T^*$), é também uma LR.
c) Toda Linguagem finita pode ser gerada por uma GLC? E por uma GR? Justifique exemplificando ou contra-exemplificando.
d) Justifique, além da definição, o fato de que toda Linguagem tipo 1 é Recursiva.
- 3 – Defina, se possível, gramáticas G_1 e G_2 |
a) $L(G_1)$ seja Regular, $L(G_2)$ seja LC $\wedge L(G_2) \subseteq L(G_1)$
b) Construa, se possível, Gramáticas G_1 , G_2 e G_3 | G_3 seja GSC, G_2 seja GLC, G_1 seja regular $\wedge L(G_3) \subseteq L(G_2) \subseteq L(G_1) \wedge L(G_3)$ não seja LC $\wedge L(G_2)$ não seja Regular.
c) $L(G_1) \neq L(G_2) \wedge L(G_1) \cup L(G_2) = V_T^* \wedge L(G_1) \cap L(G_2) = \emptyset$
d) $L(G_1) \cap L(G_2) = L(G_3) \wedge G_1, G_2$ sejam GLC e G_3 seja GSC
- 4 - Construa uma gramática G , do maior tipo possível |
a) $L(G) = \{ a^i b^j c^k \mid i, j, k \geq 0 \wedge j \text{ seja par} \wedge i + k \text{ seja ímpar} \}$
b) $L(G) = \{ a^i b^j c^k \mid i, j, k \geq 0 \wedge k = i + j \text{ ou } i \neq k \}$
c) $L(G) = \{ a^n (b, c)^* d^m \mid n, m \geq 0 \wedge n > \#b's + \#c's > m \}$
d) $L(G) = \{ x \mid x \in (a, b)^* \wedge \# \text{ de ocorrências de "ab"} = \# \text{ de ocorrências de "ba"} \}$
e) $L(G) = \{ a^n b^m c^p d^q \mid n, m, p, q \geq 0 \wedge n + p = m + q \}$

5 - Construa uma Gramática Regular G |

- $L(G) = \{ x \mid x \in (0,1)^* \wedge x \text{ não possui os substrings "000" e "111"} \}$
- $L(G) = \{ x \mid x \in (a,b)^* c^n \wedge n + \#a's \text{ é ímpar} \wedge x \text{ não possui } \underline{b}'s \text{ consecutivos} \}$
- $L(G) = \{ x \mid x \in (a,b)^* \wedge |x| \text{ seja par} \wedge x \text{ não possui } \underline{b}'s \text{ consecutivos} \}$
- $L(G) = \{ 0^n x \mid x \in (0,1)^*, n \geq 1 \wedge x \text{ possui pelo menos } n \text{ 1's} \}$
- $L(G) = \{ x \mid x \in (1,2,3)^* \wedge \Sigma \text{ dos elementos de } x \text{ seja divisível por } 3 \}$

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

- $L(G) = \{ a^n b^m (c,d)^* \mid 0 \leq n \leq m \wedge \#c's = 2 * \#d's \}$
- $L(G) = \{ x \mid x \in (a,b)^* \wedge x \text{ não seja um palíndromo} \}$
- $L(G) = \{ x \mid x \in (a,b)^* \wedge |x| \text{ seja ímpar} \wedge \text{o primeiro símbolo de } x \text{ seja igual ao símbolo central de } x \}$
- $L(G) = \{ w c x \mid w, x \in (a,b)^* \wedge wR \text{ é uma sub-cadeia de } x \}$

7 - Construa uma GLC que especifique:

- a sintaxe da **declaração** de procedimentos/funções (métodos com/sem tipo), com ou sem parâmetros, de uma linguagem de programação qualquer que permita passagem de parâmetro por referência e por valor. OBS. Para simplificar, considere apenas parâmetros de tipos simples.
- a sintaxe da **chamada** de procedimentos/funções (com e sem parâmetros, os quais, se existirem, podem ser qualquer expressão aritmética válida).
- a sintaxe de **expressões aritméticas** envolvendo variáveis ("id"), constantes ("c"), parênteses ("(" e ")") e os operadores "+", "-", "/", "*", "**" e "***" (onde "***" é o operador de exponenciação).
- a sintaxe de uma seqüência/aninhamento dos comandos: if-then-else, while-do e atribuição.

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

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

9 - Seja G a seguinte gramática:

$$\begin{array}{ll} S \rightarrow a S \mid S C \mid c S A \mid b \\ A C \rightarrow C A & C A \rightarrow A C \\ b C \rightarrow b c & b A \rightarrow b a \\ a A \rightarrow a a & a C \rightarrow a c \\ c A \rightarrow c a & c C \rightarrow c c \end{array}$$

Pede-se:

- Verifique se $x = acba$ e $y = abca$ pertencem a $L(G)$, usando o algoritmo apropriado;
- Determine $L(G)$;
- Construa, se possível, uma GLC $G_1 \mid L(G_1) = L(G)$
- Construa $G_2 \mid G_2$ seja do mesmo tipo de $G \wedge L(G_2) = L(G) \cup \{ \varepsilon \}$

10 - Dada a gramática G :

$$S \rightarrow a S A \mid a S B \mid a S \mid S B \mid c$$
$$AB \rightarrow BA$$
$$BA \rightarrow AB$$
$$A \rightarrow a$$
$$B \rightarrow b$$

Pede-se:

- Verifique se $x = acba$ e $y = acaa \in L(G)$ - 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 $G_3 \mid L(G_3) = L(G) \cup \{ \varepsilon \}$

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

a) $S \rightarrow S a S \mid S b S \mid c$

(* $L(G)$ é também uma LR? *)

b) $S \rightarrow 0 S 1 \mid 1 S 0 \mid 0 1 \mid 1 0 \mid 0 1$

(* $L(G)$ é também uma LR? *)

c) $S \rightarrow a S B \mid S B C \mid B C$

(* $L(G)$ é também uma LLC? *)

$$C B \rightarrow B C$$
$$B C \rightarrow C B$$
$$B \rightarrow B B \mid b$$
$$C \rightarrow c$$

d) $S \rightarrow AD$

(* $L(G)$ é também uma LLC? *)

$$A \rightarrow a A C \mid a A \mid a$$
$$D \rightarrow B D d \mid D d \mid d$$
$$C B \rightarrow B C$$
$$B C \rightarrow C B$$
$$B \rightarrow b$$
$$C \rightarrow c$$

e) $S \rightarrow a A \mid b A \mid a \mid b$

$$A \rightarrow a B \mid b B$$
$$B \rightarrow a C \mid b C$$
$$C \rightarrow a D \mid b D$$
$$D \rightarrow a E \mid b E \mid a \mid b$$
$$E \rightarrow a S \mid b S$$

f) $S \rightarrow C D$

(* $L(G)$ é também uma LSC? *)

$$C \rightarrow a C A \mid b C B$$
$$A D \rightarrow a D$$
$$B D \rightarrow b D$$
$$A a \rightarrow a A$$
$$B a \rightarrow a B$$
$$A b \rightarrow b A$$
$$B b \rightarrow b B$$
$$C \rightarrow \varepsilon$$
$$D \rightarrow \varepsilon$$