

Compiladores — Folha laboratorial 3

DCC/FCUP

Outubro 2024

Gramáticas independentes de contexto

Exercício 1

- (a) Considere a gramática apresentada na aula teórica com símbolos terminais $\{a, b\}$, não-terminais $\{S, B\}$, símbolo inicial S e as produções seguintes.

$$\begin{array}{ll} S \rightarrow aSB & B \rightarrow Bb \\ S \rightarrow \varepsilon & B \rightarrow b \\ S \rightarrow B & \end{array}$$

Descreva a linguagem reconhecida por esta gramática numa frase. (Sugestões: Onde podem ocorrer as e bs ? Qual relação entre as suas contagens?)

- (b) Escreva uma gramática para a linguagem de palavras com o mesmo número de as e bs (em qualquer ordem)
- (c) Escreva uma gramática para a linguagem de parêntesis casados. Alguns exemplos:

Palavras aceites	Palavras não aceites
ε (palavra vazia)	(
())
(()))(
()()	((
(())())	(())

- (d) Escreva uma gramática para a linguagem da expressão regular $((ab^*a) | (ba^*b))$
- (e) Escreva uma gramática para a linguagem da expressão regular $((0|1)^+ \cdot "(0|1)^* | ((0|1)^* \cdot "(0|1)^+)$

Exercício 2

Considere uma gramática de programas sequências apresentada na aula teórica:

$$\begin{array}{ll} S \rightarrow S ; S & E \rightarrow \text{ident} \\ S \rightarrow \text{ident} = E & E \rightarrow \text{num} \\ S \rightarrow \text{ident} ++ & E \rightarrow E + E \end{array}$$

- (a) Mostre que a gramática acima é ambígua, i.e., encontre uma palavra com duas derivações correspondentes a árvores distintas. Consegue encontrar mais do que um exemplo?
- (b) Re-escreva a gramática de forma a eliminar a ambiguidade.

Exercício 3

- (a) Mostre que a gramática de parêntesis simples

$$A \rightarrow (A) \quad | \quad a$$

é $LR(0)$ construindo o autômato e a tabela de *parsing* LR.

- (b) Mostre que a gramática

$$\begin{array}{ll} T \rightarrow R & R \rightarrow \varepsilon \\ T \rightarrow aTc & R \rightarrow bR \end{array}$$

é $SLR(1)$ construindo o autômato e a tabela de *parsing* LR. Justifique que esta não é gramática $LR(0)$ mostrando que existem conflitos na tabela de *parsing* $LR(0)$.

Exercício 4

Considere a seguinte extensão à gramática de parêntesis:

$$\begin{array}{ll} E \rightarrow (L) & | \quad a \\ L \rightarrow L, E & | \quad E \end{array}$$

- (a) Mostre uma derivação para a palavra $((a), a, (a, a))$
- (b) Construa o autômato de *parsing* $LR(0)$ da gramática
- (c) Será que a gramática é $LR(0)$? Se não, mostre qual o conflito; se sim, construa a tabela de *parsing*.

Exercício 5

Considere a seguinte gramática simplificada de declarações de variáveis na linguagem C; os terminais são `int`, `float`, vírgula (`,`), ponto-e-vírgula (`;`) e identificadores (*ident*):

$$\begin{aligned}Decl &\rightarrow Type\ Varlist\ ; \\Type &\rightarrow \texttt{int} \mid \texttt{float} \\Varlist &\rightarrow Varlist, ident \mid ident\end{aligned}$$

- (a) Construa o autômato LR e a tabela de parsing $LR(0)$
- (b) Simule os passos de execução do autômato para a entrada `int x,y,z;`