

## 4.7 Exercícios

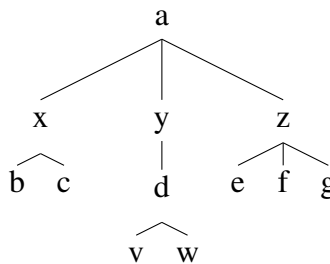
- 4.1 Mostre, para a sentença 00001111 na gramática  $G_1$  (Seção 2.3), a sua árvore sintática, as derivações canônicas mais à esquerda e mais à direita e as correspondentes seqüências de reconhecimento.
- 4.2 Mostre a árvore sintática, as derivações canônicas e as correspondentes seqüências de reconhecimento para a sentença  $aabb$  na gramática  $G_b = \{V_t, V_n, P, S\}$ , com  $V_t = \{a, b\}$ ,  $V_n = \{A, S\}$  e as produções  $P = \{S \rightarrow A, A \rightarrow aAb, A \rightarrow ab\}$ .
- 4.3 Mostre a árvore sintática, as derivações canônicas e as correspondentes seqüências de reconhecimento para a sentença  $xxxyyzzxxz$  na gramática  $G_c$ , com  $V_n = \{S, A, B, C\}$ ,  $V_t = \{x, y, z\}$ , símbolo sentencial  $S$  e produções  $S \rightarrow AxByC, A \rightarrow xAx, A \rightarrow \varepsilon, B \rightarrow By, B \rightarrow \varepsilon, C \rightarrow zAz$ .
- 4.4 Dada a gramática com símbolo não-terminal e sentencial  $S$ , símbolos terminais  $a, b$  e produções

$$S \rightarrow aSbS \mid bSaS \mid \epsilon$$

mostre, usando a sentença *abab*, que esta é ambígua. Para tanto, apresente para a sentença:

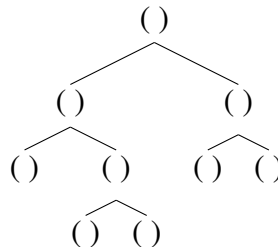
- (a) Duas derivações canônicas mais à direita;
- (b) Duas derivações canônicas mais à esquerda;
- (c) Duas árvores sintáticas.

#### 4.5 Para a árvore



- (a) Qual é a seqüência de apresentação dos seus elementos quando a estratégia de varredura pré-ordem é adotada?
- (b) Qual é a seqüência de apresentação dos seus elementos quando a estratégia de varredura pós-ordem é adotada?

#### 4.6 Uma árvore binária com nove nós apresenta a seguinte estrutura:



Atribua a cada nó, preservando essa estrutura, os dígitos de 1 a 9 de forma que a varredura da árvore apresente os valores em ordem ascendente quando a estratégia adotada é:

- (a) pré-ordem;
- (b) intra-ordem;
- (c) pós-ordem.

4.7 Apresente todas as árvores sintáticas para as seguintes sentenças em  $G_2$  (Seção 4.1):

- (a)  $v \times v$
- (b)  $v \times v + v$
- (c)  $v \times (v + v)$
- (d)  $v + v \times v + v$
- (e)  $(v + v) \times (v + v)$

4.8 Considere a gramática  $G_g$  com símbolos não-terminais  $\{S, L\}$ , símbolo sentencial  $S$ , símbolos terminais  $\{(\,, \,), \,a, \,\Delta\}$  e produções

$$\begin{aligned} S &\rightarrow (L) \\ S &\rightarrow a \\ L &\rightarrow L \Delta S \\ L &\rightarrow S \end{aligned}$$

(a) Apresente as árvores sintáticas para as sentenças:

- $(a \Delta a)$
- $(a \Delta (a \Delta a))$
- $(a \Delta ((a \Delta a) \Delta (a \Delta a)))$

(b) Monte a tabela sintática para um analisador sintático preditivo para reconhecer sentenças nessa linguagem e mostre a operação do analisador sintático preditivo para cada uma das sentenças do item  $a$ .

(c) Obtenha as relações de Wirth-Weber entre os símbolos da gramática  $G_g$ .

(d) Se possível, construa o analisador de precedência fraca para a gramática  $G_g$  e mostre sua operação para cada uma das sentenças do item  $a$ . Caso contrário, indique qual condição é violada.

4.9 Para a gramática  $G_h = \{V_n, V_t, P, S\}$ , com  $V_n = \{A, S\}$ ,  $V_t = \{a, b\}$  e produções  $P = \{S \rightarrow A, A \rightarrow aAb, A \rightarrow ab\}$ .

(a) Monte a tabela sintática para o analisador preditivo, se possível.

(b) Obtenha as relações de Wirth-Weber e mostre que a gramática é de precedência fraca.

- (c) Monte a tabela DR para o analisador de precedência fraca.
- (d) Mostre a operação do analisador no reconhecimento da sentença  $aabbb$ , com a correspondente construção da árvore sintática a cada passo.

4.10 Considere a gramática  $G_i$  cujas produções são apresentadas a seguir, com símbolo sentencial  $S$  e símbolos terminais  $\{a, e, o, l, x\}$ :

$$\begin{aligned} S &\rightarrow ABe \\ A &\rightarrow a \\ A &\rightarrow o \\ B &\rightarrow x \\ B &\rightarrow l \end{aligned}$$

- (a) Construa o analisador sintático preditivo.
- (b) Construa o analisador sintático de precedência fraca.
- (c) Mostre a operação dos dois analisadores no reconhecimento da sentença  $axe$ .

4.11 O seguinte fragmento de um arquivo yacc define os símbolos e produções para uma gramática  $G_j$ :

```
%token  T1, T2, T3, T4
%%
n1      :  n2 T1 n2 T2   |   T2 ;
n2      :  n3 |   n4 ;
n3      :  n4 T3 n4 |   T3 n4 ;
n4      :  n4 T4 |   T4 ;
```

- (a) Mostre a representação formal para  $G_j$ .
- (b) Mostre a representação de  $G_j$  em BNF.
- (c) Mostre a representação gráfica de  $G_j$  na notação de diagramas sintáticos.

4.12 O seguinte fragmento de uma especificação yacc tenta representar o comando condicional (IF) de alguma linguagem:

```
%token CMD COND ELSE IF
%%
    stmt : CMD | ifstmt ;
    ifstmt : IF '(' COND ')' stmt
           | IF '(' COND ')' stmt ELSE stmt ;
```

Ao compilar essa especificação, o gerador de analisador sintático produz uma mensagem de aviso para indicar que a gramática especificada tem uma ambigüidade que produz um conflito de deslocamento ou redução.

- (a) Qual é a causa desse conflito na especificação?
  - (b) Complemente a especificação de modo que seja possível executar um exemplo que permita descobrir qual é a ação padrão tomada pelo analisador gerado numa dessas situações de conflito.
- 4.13 Complemente o exemplo da Seção 4.6.3 de forma a contemplar as quatro operações básicas e a incluir valores negativos com o operador unário `-`.
- 4.14 Complemente o exemplo da Seção 4.6.3 de forma a aceitar valores reais além de inteiros.
- 4.15 Complemente o exemplo da Seção 4.6.3 de forma a definir uma calculadora científica que aceita valores inteiros e reais e que, além das quatro operações aritméticas e de valores negativos, reconheça as operações da biblioteca matemática de C, especificadas no arquivo `cmath`, como as funções `pow`, `log` e `sin`.