

1. Usando a gramática G_{inteiro} , desenvolva uma derivação mais à direita para o inteiro 4520.

2. Reescreva a BNF a seguir para dar ao operador + precedência sobre * e para forçar + a ser associativo à direita.

```
<assign> -> <id> = <expr>
<id> -> A | B | C
<expr> -> <expr> + <term>
          | <term>
<term> -> <term> * <factor>
          | <factor>
<factor> -> ( <expr> )
          | <id>
```

3. Reescreva a BNF a seguir para adicionar os operadores unários de Java ++e--.

```
<assign> -> <id> = <expr>
<id> -> A | B | C
<expr> -> <expr> + <term>
          | <term>
<term> -> <term> * <factor>
          | <factor>
<factor> -> ( <expr> )
          | <id>
```

4. Usando a gramática

```
<assign> -> <id> = <expr>
<id> -> A | B | C
<expr> -> <id> + <expr>
          | <id> * <expr>
          | ( <expr> )
          | <id>
```

, mostre uma árvore de análise sintática e uma derivação mais a esquerda para cada uma das seguinte sentenças:

- a) $A = A * (B + (C * A))$
- b) $B = C * (A * C + B)$
- c) $A = A * (B + (C))$

5. Usando a gramática

$\langle \text{assign} \rangle \rightarrow \langle \text{id} \rangle = \langle \text{expr} \rangle$
 $\langle \text{id} \rangle \rightarrow A \mid B \mid C$
 $\langle \text{expr} \rangle \rightarrow \langle \text{expr} \rangle + \langle \text{term} \rangle$
 $\quad \mid \langle \text{term} \rangle$
 $\langle \text{term} \rangle \rightarrow \langle \text{term} \rangle * \langle \text{factor} \rangle$
 $\quad \mid \langle \text{factor} \rangle$
 $\langle \text{factor} \rangle \rightarrow (\langle \text{expr} \rangle)$
 $\quad \mid \langle \text{id} \rangle$

, mostre uma árvore de análise sintática e uma derivação mais a esquerda para cada uma das seguinte sentenças: a. $A=(A+B)*C$

b. $A=B+C+A$

c. $A=A*(B+C)$

d. $A=B*(C*(A+B))$

6. Prove que a seguinte gramática é ambígua:

$\langle S \rangle \rightarrow \langle A \rangle$

$\langle A \rangle \rightarrow \langle A \rangle + \langle A \rangle \mid \langle \text{id} \rangle$

$\langle \text{id} \rangle \rightarrow a \mid b \mid c$

7. Modifique a gramática do Exemplo 3.4 para adicionar um operador unário de subtração que tenha precedência mais alta que + ou *.

8. Considere a seguinte gramática:

$\langle S \rangle \rightarrow \langle A \rangle a \langle B \rangle b \quad \langle A \rangle \rightarrow \langle A \rangle b \mid b \quad \langle B \rangle \rightarrow a \langle B \rangle \mid$

a Quais das sentenças abaixo estão na linguagem gerada por essa gramática?

a. baab

b. bbbab

c. bbaaaaa

d. bbaab

9. Converta de BNF para EBNF a seguinte gramática:

$\langle \text{program} \rangle \rightarrow \mathbf{begin} \langle \text{stmt_list} \rangle \mathbf{end}$

$\langle \text{stmt_list} \rangle \rightarrow \langle \text{stmt} \rangle$

$\quad \mid \langle \text{stmt} \rangle ; \langle \text{stmt_list} \rangle$

$\langle \text{stmt} \rangle \rightarrow \langle \text{var} \rangle = \langle \text{expression} \rangle$

$\langle \text{var} \rangle \rightarrow A \mid B \mid C$

$\langle \text{expression} \rangle \rightarrow \langle \text{var} \rangle + \langle \text{var} \rangle$

$\quad \mid \langle \text{var} \rangle - \langle \text{var} \rangle$

$\quad \mid \langle \text{var} \rangle$

10. Converta de BNF para EBNF a seguinte gramática:

$\langle \text{assign} \rangle \rightarrow \langle \text{id} \rangle = \langle \text{expr} \rangle$

$\langle \text{id} \rangle \rightarrow \text{A} \mid \text{B} \mid \text{C}$

$\langle \text{expr} \rangle \rightarrow \langle \text{expr} \rangle + \langle \text{expr} \rangle$
| $\langle \text{expr} \rangle * \langle \text{expr} \rangle$
| $(\langle \text{expr} \rangle)$
| $\langle \text{id} \rangle$