

## Lista de Exercícios #04

### Análise Descendente, Gramáticas LL(1) e Tabelas Preditivas

1. Como sabemos quando uma gramática não é LL(1)?
2. A gramática seguinte não pode ser utilizada em uma análise preditiva descendente. Identifique e corrija o problema, reescrevendo a gramática. Mostre que a nova gramática, corrigida, satisfaz a condição LL(1).
 
$$\begin{aligned} L &\rightarrow Ra \mid Qba \\ R &\rightarrow aba \mid caba \mid Rbc \\ Q &\rightarrow bbc \mid bc \end{aligned}$$
3. A gramática seguinte satisfaz a condição LL(1)? Justifique a sua resposta. Se ela não satisfizer, reescreva-a como uma gramática LL(1) equivalente, capaz de reconhecer a mesma linguagem.
 
$$\begin{aligned} A &\rightarrow Ba \\ B &\rightarrow dab \mid Cb \\ C &\rightarrow cB \mid Ac \end{aligned}$$
4. Para cada uma das gramáticas seguintes, defina analisadores preditivos descendentes através da construção de tabelas preditivas de análise. Algumas dessas gramáticas precisam ser, inicialmente, fatoradas à esquerda ou ter sua recursão à esquerda eliminada.
  - $S \rightarrow 0 S 1 \mid 0 1$
  - $S \rightarrow + S S \mid * S S \mid a$
  - $S \rightarrow S ( S ) S \mid \epsilon$
  - $S \rightarrow S + S \mid SS \mid ( S ) \mid S * \mid a$
  - $S \rightarrow ( L ) \mid a$  e  $L \rightarrow L, S \mid S$
  - $S \rightarrow aSbS \mid bSaS \mid \epsilon$
  - Gramática para expressões booleanas:
 
$$\begin{aligned} \text{bexpr} &\rightarrow \text{bexpr} \text{ or } \text{bterm} \mid \text{bterm} \\ \text{bterm} &\rightarrow \text{bfactor} \mid \text{bfactor} \\ \text{bfactor} &\rightarrow \text{not bfactor} \mid ( \text{bexpr} ) \mid \text{true} \mid \text{false} \end{aligned}$$
5. É possível realizar a construção de um analisador preditivo descendente para a gramática  $S \rightarrow SS + \mid SS * \mid a$  utilizada para descrever as expressões pós-fixadas com o operador  $a$ ? Quais são as modificações necessárias na gramática para efetuar tal tarefa?
6. A gramática seguinte satisfaz a condição LL(1)? Justifique a sua resposta. Qual tipo de linguagem ela descreve?
 
$$\begin{aligned} S &\rightarrow (L)p|q \\ L &\rightarrow LandS|LorS|S \end{aligned}$$
7. Implemente, em pseudo-código, um analisador recursivo descendente para a linguagem descrita no exercício 6., transformando a gramática em LL(1) se necessário.
8. Seja L uma linguagem cujas sentenças são formadas de qualquer texto. A seguinte sentença faz parte desta linguagem: “um (grande) animal chamado gato (podendo logicamente ser um leão ou um tigre {que são cada vez mais raros} ou ainda um tigre dente-de-sabre {que está extinto [ ler mais em Kurten ], o que é uma vergonha} ou leopardo) é um atraente (mas perigoso) amigo”. Você pode supor, por simplicidade, que o texto consiste apenas de letras e espaços. Defina uma gramática LL(1) que descreva esta linguagem e, por fim, construa uma tabela preditiva de análise descendente para a mesma. Mostre que a tabela funciona para uma entrada válida e uma entrada não válida.
9. Considere a seguinte gramática

expressão  $\rightarrow$  átomo | lista  
 átomo  $\rightarrow$  número | identificador  
 lista  $\rightarrow$  (sequência\_de\_expressões)  
 sequência\_de\_expressões  $\rightarrow$  expressão, sequência\_de\_expressão | expressão

- Fatore à esquerda esta gramática, se necessário, e elimine a recursão à esquerda, se existir.
- Construa uma tabela preditiva de análise descendente LL(1) para esta gramática.
- Mostre as ações do analisador LL(1) correspondente, considerando a entrada **(a,(b,(2)),(c))**

10. Considerando a gramática **g1**. O conjunto dos símbolos terminais é {a, b, c}, o conjunto dos símbolos não-terminais é {S, A, B, C}, o símbolo inicial é S, e as produções são as seguintes:

S  $\rightarrow$  cA | b  
 A  $\rightarrow$  cBC | bSA | a  
 B  $\rightarrow$  cc | Cb  
 C  $\rightarrow$  aS | ba

Considerando a gramática **g2**. O conjunto dos símbolos terminais é {a, b}, o conjunto dos símbolos não-terminais é {S, A}, o símbolo inicial é S, e as produções são as seguintes:

S  $\rightarrow$  abA | aa  
 A  $\rightarrow$  bb | bS

Considerando a gramática **g3**. O conjunto dos símbolos terminais é {a, b, c}, o conjunto dos símbolos não-terminais é {S, A, B}, o símbolo inicial é S, e as produções são as seguintes:

S  $\rightarrow$  AaS | B  
 A  $\rightarrow$  cS |  $\epsilon$   
 B  $\rightarrow$  b

- Para cada gramática **g1**, **g2** e **g3**, fatore à esquerda se necessário, e elimine a recursão à esquerda se houver.
- Construa tabelas preditivas LL(1) para a análise descendente para cada gramática **g1**, **g2** e **g3**.
- Para a gramática **g1**, mostre os passos da análise de **ccccba** utilizando a tabela construída.
- Para a gramática **g2**, mostre os passos da análise de **abbb** utilizando a tabela construída.
- Para a gramática **g3**, mostre os passos da análise de **acbab** utilizando a tabela construída.

11. Construa a tabela preditiva da seguinte gramática (já fatorada à esquerda e sem recursão à esquerda):

E  $\rightarrow$  TE'  
 E'  $\rightarrow$  ATE' |  $\epsilon$   
 T  $\rightarrow$  FT'  
 T'  $\rightarrow$  BFT' |  $\epsilon$   
 F  $\rightarrow$  (E) | id  
 A  $\rightarrow$  + | -  
 B  $\rightarrow$  \* | /

- Mostre os passos de análise para a entrada **id / id - id**
- Caso exista ambiguidade, resolva o problema modificando a tabela. Justifique cada escolha.
- Mostre os passos de análise para **id \* id / id** e **id + id - id**, mostrando o efeito das suas decisões.

12. Construa a tabela preditiva da seguinte gramática e mostre os passos de **ibtibtataeaea**

S  $\rightarrow$  iEtSS' | a  
 S'  $\rightarrow$  eS |  $\epsilon$   
 E  $\rightarrow$  b

13. Construa a tabela preditiva da seguinte gramática

S  $\rightarrow$  **if** ( E ) SS' | { S } | **a**  
 S'  $\rightarrow$  **else** S |  $\epsilon$   
 E  $\rightarrow$  b

- Mostre os passos de análise para **if(b){if(b)a}elsea**
- Mostre os passos de análise para **if(b){if(b){if(b)a}elsea}else{if(b)aelse{a}}**

14. Explique o porquê da gramática seguinte não ser ambígua, fazendo referência ao exercício 13.. Construa a tabela preditiva desta gramática e mostre os passos de análise para **{if{a}else{a}}**

S → { A }  
A → if SA' | a  
A' → else S