Teoria da Computação:
GLC, LLC e AP

Tópicos II

Thiago Silva Vilela

Gramáticas Livres do Contexto

- Uma gramática livre do contexto (GLC) é uma gramática (V, ∑, R, P):
  - V: variáveis (símbolos não terminais);
  - ∑: alfabeto (símbolos terminais);
  - R: conjunto de regras;
  - P: variável de partida

- Cada regra tem a forma $X \rightarrow w$, onde $X \in V$ e $w \in (V \cup \Sigma)^*$.
  - cada regra de uma GLC tem, do lado esquerdo, somente uma variável;
  - o lado direito de cada regra pode ter qualquer palavra constituída de variáveis e/ou terminais;

Gramáticas Livres do Contexto

- Uma gramática regular é uma gramática livre do contexto especial, onde somente uma única variável pode ser expandida em qualquer forma sentencial.

- Linguagens regulares são um subconjunto das GLCs.
  - Existem linguagens não regulares que são linguagens livres do contexto;
  - Não existem linguagens regulares que não são livres do contexto.

Gramáticas Livres do Contexto

Exemplo: a linguagem não regular $\{0^n1^n \mid n \in \mathbf{N}\}$ é gerada pela GLC G = ({P}, {0, 1}, R, P), onde R consta das duas regras:

$P \rightarrow 0P1 \mid \lambda$

- Essa gramática só gera palavras por *n* aplicações da regra $P \rightarrow 0P1$, seguidas de uma aplicação da regra $P \rightarrow \lambda$.

Gramáticas Livres do Contexto

- **Exemplo**: A gramática G a seguir gera os palíndromos sobre {0, 1}, ou seja, $L(G) = \{w \in \{0, 1\}^* \mid w = w^R\}$.

    - $G = (\{P\}, \{0, 1\}, R, P)$, onde R consta de:
    - $P \rightarrow 0P0 \mid 1P1 \mid 0 \mid 1 \mid \lambda$

- Aplicando-se as 2 primeiras regras gera-se qualquer forma sentencial do tipo wPw^R.
- Para finalizar a geração da palavra, aplica-se uma das 3 últimas regras.

Linguagens Livres do Contexto

- Uma linguagem é dita ser uma linguagem livre do contexto se existe uma gramática livre do contexto que a gera.

- São importantes, pois são linguagens para as quais há reconhecedores eficientes.

Derivações e Ambiguidade

- Em geral, existem várias derivações de uma mesma palavra da linguagem gerada por uma gramática.

- Uma árvore de derivação (AD) captura a essência de uma derivação:
  - a cada derivação irá corresponder uma única AD;
  - a uma AD podem corresponder uma grande quantidade de derivações.

- AD's particionam o conjunto de todas as derivações de uma GLC em derivações "equivalentes".
  - duas derivações são equivalentes se, e somente se, correspondem à mesma AD.

Derivações e Ambiguidade

Considere a gramática G = ({E, T, F}, {t, +, *}, R, E), onde R consta das seguintes regras:

$E \rightarrow E + T \mid T$

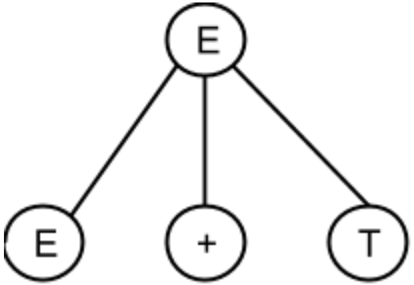
$T \rightarrow T * F \mid F$

$F \rightarrow (E) \mid t$

Qual a árvore de derivação da forma sentencial E + T?

Derivações e Ambiguidade

Qual a árvore de derivação da forma sentencial E + T?

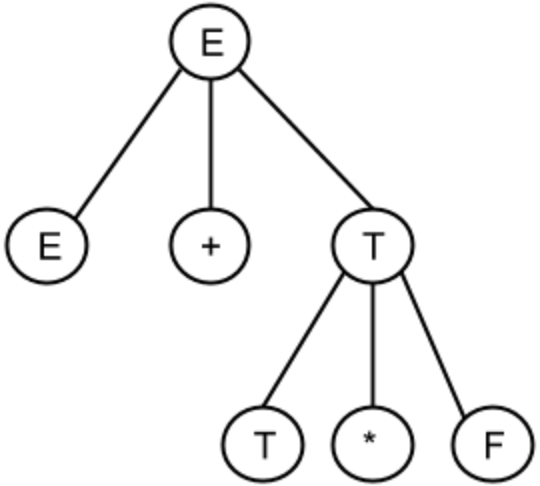


Derivações e Ambiguidade

Qual a árvore de derivação da forma sentencial E + T * F?

Derivações e Ambiguidade

Qual a árvore de derivação da forma sentencial E + T * F?

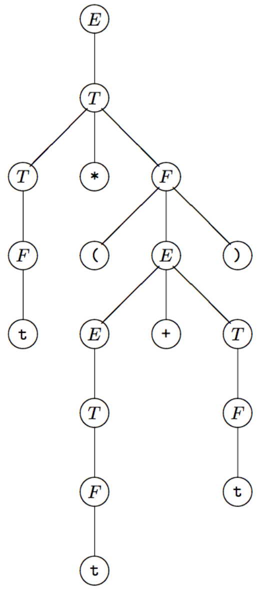


Derivações e Ambiguidade

Qual a árvore de derivação da forma sentencial t*(t+t)?

Derivações e Ambiguidade

Qual a árvore de derivação da forma sentencial t*(t+t)?



Derivações e Ambiguidade

- Para gerar uma árvore de derivação basta construir uma árvore através de aplicações das regras das gramáticas.

- O número de passos de qualquer derivação que leva a uma AD X é o número de vértices internos de X.
  - cada vértice interno corresponde à aplicação de uma regra.

Derivações e Ambiguidade

- **Ambiguidade**: Uma GLC é dita ambígua quando existe mais de uma árvore de derivação para alguma sentença que ela gera.

- Note que a gramática é dita ambígua e não a linguagem.
  - podem existir duas GLCs equivalentes (que geram a mesma linguagem) onde uma é ambígua e a outra não.

- A gramática para expressões aritméticas vista anteriormente, por exemplo, não é ambígua.

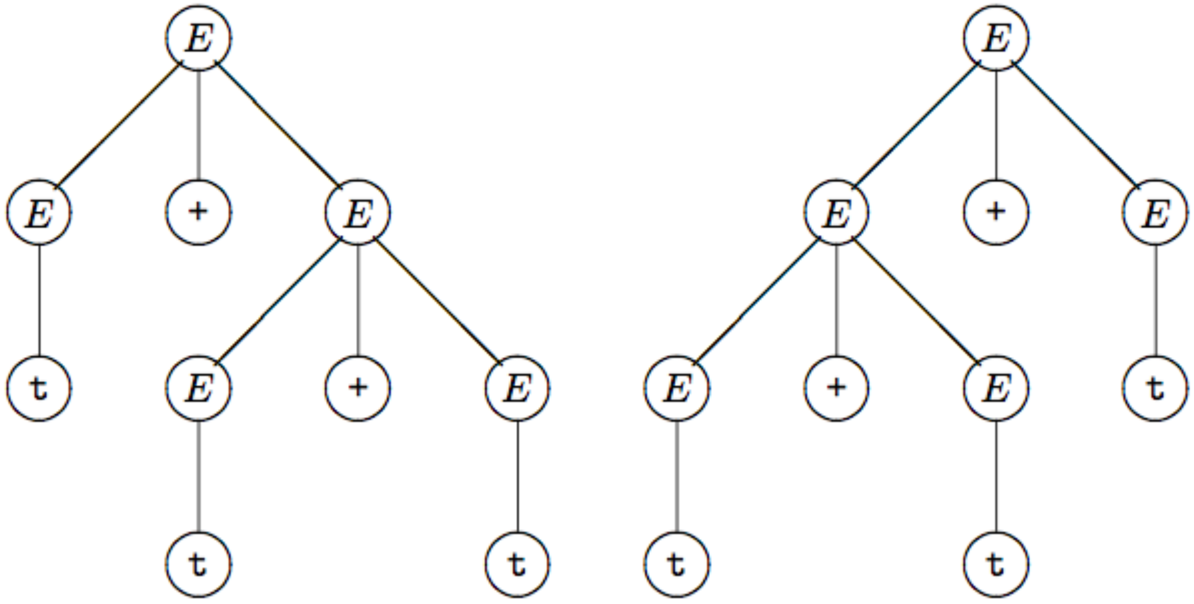
Derivações e Ambiguidade

Exemplo: Vamos mostrar que a gramática G = ({E}, {t, +, *}, R, E), com as seguintes regras R:

$E \rightarrow E + E \mid E * E \mid (E) \mid t$

é ambígua. Essa gramática gera a mesma linguagem de expressões aritméticas vista anteriormente. Vamos mostrar sua ambiguidade encontrando duas árvores de derivação para a forma sentencial t+t+t.

Derivações e Ambiguidade



Derivações e Ambiguidade

- Existem dois tipos particularmente importantes de derivações:
  - Derivação mais à esquerda (DME);
  - Derivação mais à direita (DMD);

- Elas se diferenciam na escolha da variável a ser expandida:
  - DME’s expandem sempre a variável mais à esquerda;
  - DMD’s expandem sempre a variável mais à direita;

- Uma outra definição de ambiguidade normalmente utilizada é:
  - uma GLC é ambígua se e somente se existe mais de uma DME ou mais de uma DMD para alguma sentença que ela gera.

Derivações e Ambiguidade

Exemplo: Duas DMD's para a sentença t+t+t, usando a gramática do exemplo anterior, são:

$E \Rightarrow_D E+E$	$E \Rightarrow_D E+E$
$\Rightarrow_D E+E+E$	$\Rightarrow_D E+t$
$\Rightarrow_D E+E+t$	$\Rightarrow_D E+E+t$
$\Rightarrow_D E+t+t$	$\Rightarrow_D E+t+t$
$\Rightarrow_D t+t+t$	$\Rightarrow_D t+t+t$

Derivações e Ambiguidade

- Existem linguagens livres do contexto para as quais existem somente gramáticas ambíguas. Essas linguagens são chamadas de **linguagens inerentemente ambíguas**.

- O problema de determinar se uma GLC arbitrária não é ambígua é indecidível.

- Gramáticas livres do contexto e autômatos com pilha (AP) reconhecem a mesma classe de linguagens.
  - para qualquer GLC G existem um AP que reconhece L(G).

Autômatos com pilha

- Acrescentam memória (na forma de pilha) ao autômato.
- Autômatos com pilha reconhecem exatamente as linguagens livres do contexto.
- Certa palavra é reconhecida quando, após sua leitura completa, um estado final é alcançado e a pilha se encontra vazia.
- Podem também ser divididos em autômatos com pilha determinísticos e autômatos com pilha não determinísticos.

Autômatos com pilha determinísticos

Exemplo: Vamos construir um autômato com pilha determinístico para a linguagem $\{a^n b^n \mid n \in \mathbf{N}\}$.

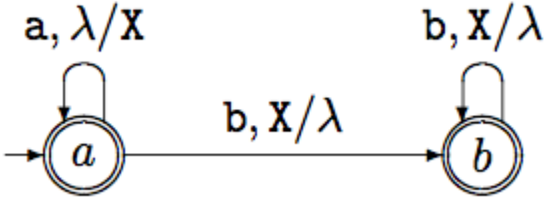
As transições de um autômato com pilha determinístico são da forma:

□(e, a, b) = [e’, z]

Isso significa que, estando no estado e, se o próximo símbolo de entrada for a e o símbolo no topo da pilha for b, há uma transição para o estado e’, b é desempilhado e z é empilhado.

Autômatos com pilha determinísticos

Exemplo: Vamos construir um autômato com pilha determinístico para a linguagem $\{a^n b^n \mid n \in \mathbf{N}\}$.

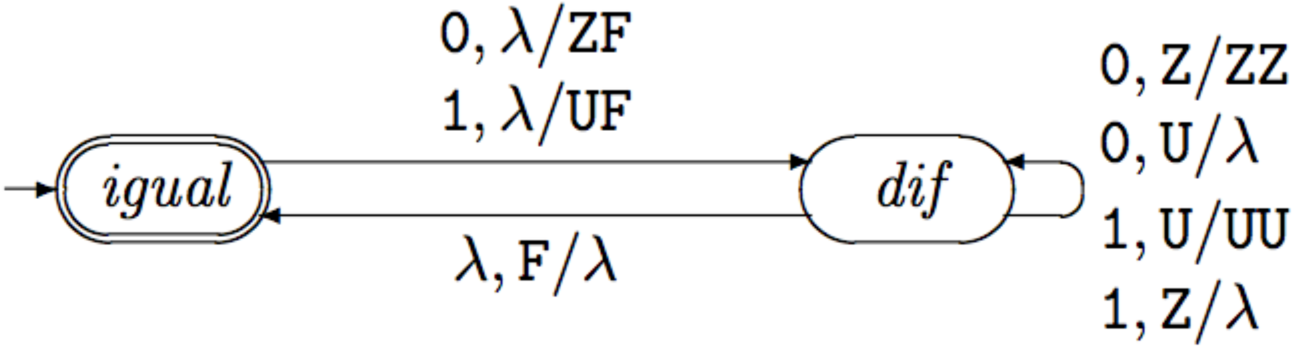


Autômatos com pilha determinísticos

Exemplo: Vamos construir um autômato com pilha determinístico para a linguagem $\{w \in \{0, 1\}^* \mid$ o número de 0's em w é igual ao número de 1's$\}$.

Autômatos com pilha determinísticos

Exemplo: Vamos construir um autômato com pilha determinístico para a linguagem {w ∈ {0, 1}* | o número de 0's em w é igual ao número de 1's}.



Autômatos com pilha não determinísticos

- Autômatos com pilha não determinísticos (APN’s) são mais poderosos que APD’s.
  - existem linguagens que não podem ser reconhecidas por APD’s e podem ser reconhecidas por APN’s.

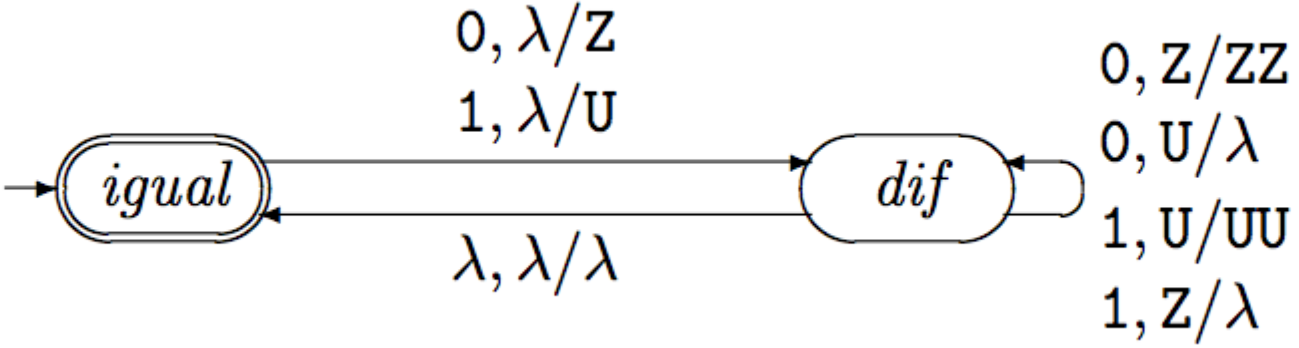
- APN’s, assim como AFN’s, podem ter duas transições possíveis a um mesmo momento, e são capazes de escolher aquela que levará a um reconhecimento de certa palavra.

Autômatos com pilha não determinísticos

Exemplo: vimos, anteriormente, um APD para a linguagem $\{w \in \{0, 1\}^* \mid$ o número de 0's em w é igual ao número de 1's}. Vamos fazer um APN para essa mesma linguagem.

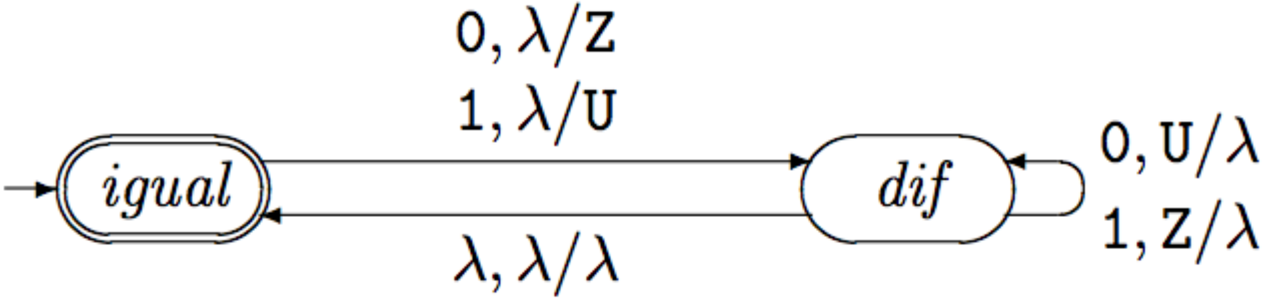
Autômatos com pilha não determinísticos

Exemplo: vimos, anteriormente, um APD para a linguagem {w ∈ {0, 1}* | o número de 0's em w é igual ao número de 1's}. Vamos fazer um APN para essa mesma linguagem.



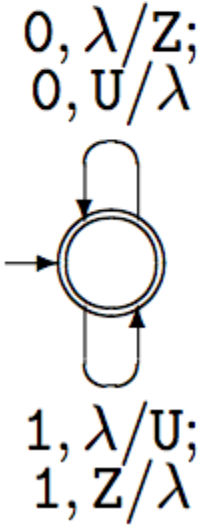
Autômatos com pilha não determinísticos

Como simplificar o autômato anterior?



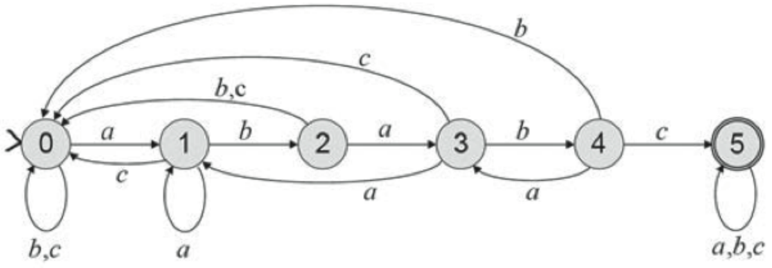
Autômatos com pilha não determinísticos

Podemos simplificar ainda mais?



Exemplos de questões

A figura abaixo apresenta um autômato que reconhece sequências sobre o alfabeto ∑={a,b,c} e uma gramática livre de contexto que gera um subconjunto de ∑*, em que λ representa o *string* vazio.



S → aS | bS | cS | abA
A → abA | abcB
B → aB | bB | cB | λ

Analisando a gramática e o autômato acima, conclui-se que:
a) a linguagem gerada pela gramática é inerentemente ambígua.
b) a gramática é regular e gera uma linguagem livre de contexto.
c) a linguagem reconhecida pelo autômato é a mesma gerada pela gramática.
d) o autômato reconhece a linguagem sobre ∑ em que os *strings* possuem o prefixo *ababc*.
e) a linguagem reconhecida pelo autômato é a mesma que a representada pela expressão regular (a + b + c)*(ab)*abc(a + b + c)*.

Exemplos de questões

(A) A linguagem gerada por essa gramática é regular e, portanto, não faz sentido definir ambiguidade para ela.

(B) Essa afirmativa é um pouco controversa. Alguns autores consideram a gramática mostrada como regular, e outros preferem considerar como gramáticas regulares somente aquelas com regras que tenham somente um terminal do lado direito. A linguagem gerada por essa gramática é regular. No entanto, gramáticas regulares podem ser vistas como um caso especial de gramáticas livres do contexto.

(C) Verdadeira. Tanto o autômato como a gramática reconhecem a linguagem $\{a, b, c\}^*\{ab\}^+abc\{a, b, c\}^*$.

(D) Falso.

(E) Falso. A expressão regular $(a + b + c)^*(ab)^*abc(a + b + c)^*$ está levemente incorreta. Essa expressão indica que, por exemplo, a palavra abc deveria ser reconhecida pelo autômato, uma vez que é gerada por essa expressão, o que é incorreto. A expressão regular correta seria $(a + b + c)^*(ab)^+abc(a + b + c)^*$.

Exemplos de questões

Considere a gramática G definida pelas regras de produção abaixo, em que os símbolos não-terminais são S, A e B, e os símbolos terminais são a e b.

S → AB
AB → AAB
A → a
B → b

Com relação a essa gramática, é correto afirmar que

(A) a gramática G é ambígua.
(B) a gramática G é uma gramática livre de contexto.
(C) a cadeia aabbb é gerada por essa gramática.
(D) é possível encontrar uma gramática regular equivalente a G.
(E) a gramática G gera a cadeia nula.

Exemplos de questões

(A) a gramática G sequer é livre do contexto. A literatura não define ambiguidade para gramáticas que não sejam livres do contexto e, portanto essa afirmativa é falsa.

(B) a gramática G **não** é uma gramática livre de contexto. É fácil perceber isso pela regra AB → AAB, que possui dois símbolos não terminais do lado esquerdo.

(C) a gramática gera senteças da forma $a^n b$, com n >= 1. Logo a cadeia aabbb não pode ser gerada pela gramática.

(D) a linguagem gerada por essa gramática é uma linguagem regular. É facil pensar em um AFD que reconhece tal linguagem. Logo, é possível encontrar uma gramática regular equivalente a G. Uma possível gramática seria: S → aS | ab.

(E) a gramática G não possui nenhuma regra que permite a geração da cadeia nula.

Exemplos de questões

Na gramática livre do contexto a seguir, que pode ser usada para gerar expressões aritméticas binárias parentizadas, os símbolos não terminais E, S, O e L representam expressões, subexpressões, operadores e literais, respectivamente, e os demais símbolos das regras são terminais.

E → (S O S)
S → L | E
O → + | - | * | /
L → a | b | c | d | e

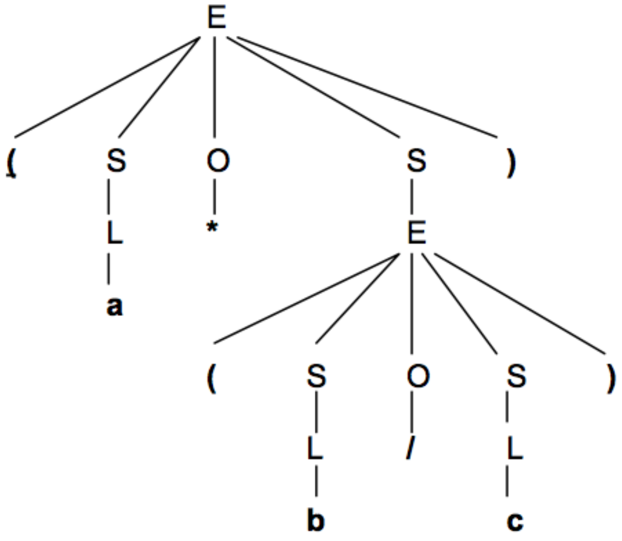
Tendo como referência as informações acima, faça o que se pede a seguir.

(A) Mostre que a expressão (a * (b / c)) pode ser obtida por derivações das regras acima. Para isso, desenhe a árvore de análise sintática correspondente.

(B) Existem diferentes derivações para a expressão (((a + b) * c) + (d * e)). É correto, então, afirmar que a gramática acima é ambígua? Justifique sua resposta.

Exemplos de questões

(A) A árvode de análise sintática é a árvore de derivação. Ela pode ser construída através de aplicações das regras fa gramática.



Exemplos de questões

(B) Note que o exercício disse que existem diferentes **derivações** para a expressão dada, e não diferentes **árvores de derivação**. Uma sequencia de derivação sempre define uma única árvore, mas uma árvore de derivação pode ter várias derivações associadas. A expressão mostrada realmente possui várias derivações possíveis. No entanto, **todas elas correspondem à mesma árvore de derivação**.

Uma gramática é ambígua quando possui diferentes árvores de derivação para uma mesma sentença. Portanto, não podemos dizer que a gramática mostrada é ambígua por possuir diferentes derivações para uma mesma sentença. De fato, a gramática mostrada não é ambígua, mas a prova disso não é necessária para responder a questão.