TEORIA DA COMPUTAÇÃO

UNIDADE 2: AUTÔMATOS E LINGUAGENS

AULA 3: AMBIGUIDADE E REGRA DE CHOMSKY

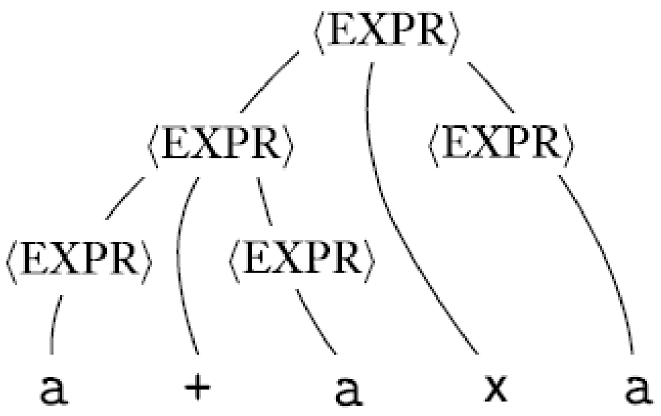
PROFESSOR: LUCAS CAMBUIM

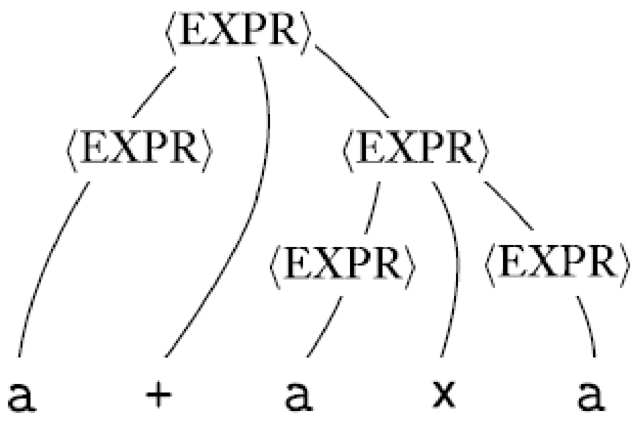
Ambiguidade

- Vejamos a gramática abaixo:
- <EXPR> → <EXPR> + <EXPR> | <EXPR> x <EXPR>
  | (<EXPR>) | a
- Para a cadeia **a + a x a** temos as seguintes árvores sintáticas:

Ambiguidade

- Vejamos a gramática abaixo:
- <EXPR> → <EXPR> + <EXPR> | <EXPR> x <EXPR>
  | (<EXPR>) | a
- Para a cadeia **a + a x a** temos as seguintes árvores sintáticas:





Ambiguidade

- Esta gramática gera a palavra **a + a x a** de forma ambígua;

Ambiguidade

- Esta gramática gera a palavra **a + a x a** de forma ambígua;
- Pois é possível obter duas árvores sintáticas diferentes;

Ambiguidade

- Esta gramática gera a palavra **a + a x a** de forma ambígua;
- Pois é possível obter duas árvores sintáticas diferentes;
- Se uma gramática gera alguma cadeia ambiguamente, dizemos que a gramática é ambígua.

Ambiguidade

- Esta gramática gera a palavra **a + a x a** de forma ambígua;
- Pois é possível obter duas árvores sintáticas diferentes;
- Se uma gramática gera alguma cadeia ambiguamente, dizemos que a gramática é ambígua.
- As derivações podem mudar alterando a ordem de substituição das variáveis;

Ambiguidade

- Esta gramática gera a palavra **a + a x a** de forma ambígua;
- Pois é possível obter duas árvores sintáticas diferentes;
- Se uma gramática gera alguma cadeia ambiguamente, dizemos que a gramática é ambígua.
- As derivações podem mudar alterando a ordem de substituição das variáveis;
- Perceba que estamos falando que a palavra tem duas árvores sintáticas diferentes, não duas derivações diferentes;

Ambiguidade

- Esta gramática gera a palavra **a + a x a** de forma ambígua;
- Pois é possível obter duas árvores sintáticas diferentes;
- Se uma gramática gera alguma cadeia ambiguamente, dizemos que a gramática é ambígua.
- As derivações podem mudar alterando a ordem de substituição das variáveis;
- Perceba que estamos falando que a palavra tem duas árvores sintáticas diferentes, não duas derivações diferentes;
- Ambiguidade pode ser indesejável para certas aplicações, como em linguagens de programação, onde um dado programa deve ter uma única interpretação.

Ambiguidade

- Para focarmos na estrutura, vamos definir um tipo de derivação que substitui as variáveis em uma ordem fixa;

Ambiguidade

- Para focarmos na estrutura, vamos definir um tipo de derivação que substitui as variáveis em uma ordem fixa;
- **Uma derivação de uma palavra w em uma gramática G é a derivação mais a esquerda se a cada etapa, a variável mais a esquerda é a substituída**;

Ambiguidade

- Para focarmos na estrutura, vamos definir um tipo de derivação que substitui as variáveis em uma ordem fixa;
- **Uma derivação de uma palavra w em uma gramática G é a derivação mais a esquerda se a cada etapa, a variável mais a esquerda é a substituída**;
- No caso da palavra **a + a x a**, ela possui duas árvores sintáticas, utilizando o mesmo tipo de derivação (a mais a esquerda);

Ambiguidade

- **Uma palavra w é derivada ambiguamente em uma gramática livre de contexto G se tem duas ou mais derivações a esquerda. A gramática G é ambígua se gera alguma palavra ambiguamente;**

Ambiguidade

- **Uma palavra w é derivada ambiguamente em uma gramática livre de contexto G se tem duas ou mais derivações a esquerda. A gramática G é ambígua se gera alguma palavra ambiguamente;**
- Em alguns casos, quando temos uma gramática ambígua, podemos encontrar uma gramática não ambígua que gera a mesma linguagem.

Ambiguidade

- **Uma palavra w é derivada ambiguamente em uma gramática livre de contexto G se tem duas ou mais derivações a esquerda. A gramática G é ambígua se gera alguma palavra ambiguamente;**
- Em alguns casos, quando temos uma gramática ambígua, podemos encontrar uma gramática não ambígua que gera a mesma linguagem.
- Porém, algumas linguagens livres de contexto só podem ser geradas através de gramáticas ambíguas;

Forma Normal de Chomsky

- Nós podemos simplificar gramáticas livres de contexto;

Forma Normal de Chomsky

- Nós podemos simplificar gramáticas livres de contexto;
- Uma das formas mais simples e úteis é chamada de **Forma Normal de Chomsky**;

Forma Normal de Chomsky

- Nós podemos simplificar gramáticas livres de contexto;
- Uma das formas mais simples e uteis é chamada de **Forma Normal de Chomsky**;
- Definição: Uma gramática está na Forma Normal de Chomsky se todas as regras estão na forma:

$$A \rightarrow BC$$

$$A \rightarrow a$$

onde a é um terminal e A, B e C são variáveis.

- B e C não podem ser a variável inicial;
- Em outras palavras a variável inicial S não pode aparecer no lado direito de nenhuma regra
- Somente a variável inicial pode ter a regra S → ε.

Forma Normal de Chomsky

- **Teorema:** Qualquer linguagem livre de contexto é gerada por uma gramática livre de contexto na Forma Normal de Chomsky;
- **Ideia de Prova:** Podemos converter uma gramática G na Forma Normal de Chomsky. A conversão é feita em etapas, onde as regras que violam as condições são substituídas por equivalentes e que são satisfatórias.
  - Elimine todas as regras da forma $A \rightarrow \varepsilon$;
  - Elimine todas as regras da forma $A \rightarrow B$ (unárias);
  - Converta o restante das regras na forma proposta;

Forma Normal de Chomsky

- **Prova:**
  - Primeiro criamos uma nova variável inicial S_0 e a regra $S_0 \rightarrow S$, onde S era a variável inicial;
  - Removemos as regras da forma $A \rightarrow \varepsilon$, onde A não é a variável inicial.
  - Então para cada ocorrência de A no lado direito de uma regra, adicionamos uma nova regra com a ocorrência removida.
  - Ex.: $B \rightarrow aA \mid A \mid AaA$ se torna $B \rightarrow a \mid \varepsilon \mid Aa \mid aA$;

Forma Normal de Chomsky

- **Prova:**
  - Removemos agora as regras do tipo $A \rightarrow B$;
    - Ex.: $A \rightarrow B$ e $B \rightarrow aA \mid \varepsilon \mid C$
    - $A \rightarrow aA \mid \varepsilon \mid C$;
  - Por fim, nós substituímos as regras da forma $A \rightarrow u_1...u_n$, onde $n > 2$ e cada u_i sendo um terminal ou variável por regras $A \rightarrow u_1A_1$, $A_1 \rightarrow u_2A_2$,…., $A_{k-2} \rightarrow u_{k-1}u_k$;

Exemplo

$S \rightarrow ASA \mid aB$

$A \rightarrow B \mid S$

$B \rightarrow b \mid \varepsilon$

Exemplo

Adicionamos a variável inicial S_0 e a regra S0 → **S**

S_0 → S

S → ASA | aB

A → B | S

B → b | ε

Exemplo

Remova as regras ϵ

Antes	Depois
$S_0 \rightarrow S$	$S_0 \rightarrow S$
$S \rightarrow ASA \mid aB$	$S \rightarrow ASA \mid aB \mid \mathbf{a}$
$A \rightarrow B \mid S$	$A \rightarrow B \mid S \mathbf{\mid \varepsilon}$
$B \rightarrow b \mid \varepsilon$	$B \rightarrow b \cancel{\mid \varepsilon}$

Exemplo

Remova as regras ϵ

Antes	Depois
$S_0 \to S$	$S_0 \to S$
$S \to ASA \mid aB \mid \mathbf{a}$	$S \to ASA \mid aB \mid a \mid \mathbf{SA \mid AS \mid S}$
$A \to B \mid S \mid \boldsymbol{\varepsilon}$	$A \to B \mid S \cancel{\mid \varepsilon}$
$B \to b \cancel{\mid \varepsilon}$	$B \to b$

Exemplo

Remova as regras unárias

$S_0 \rightarrow \cancel{S \mid}$ **ASA | aB | a | SA | AS**

$S \rightarrow ASA \mid aB \mid a \mid SA \mid AS \cancel{\mid S}$

$A \rightarrow B \mid S$

$B \rightarrow b$

Exemplo

Remova as regras unárias

$S_0 \rightarrow ASA \mid aB \mid a \mid SA \mid AS$

$S \rightarrow ASA \mid aB \mid a \mid SA \mid AS$

$A \rightarrow \cancel{B} \mid S \mid \mathbf{b}$

$B \rightarrow b$

Exemplo

Remova as regras unárias

$S_0 \rightarrow ASA \mid aB \mid a \mid SA \mid AS$

$S \rightarrow ASA \mid aB \mid a \mid SA \mid AS$

$A \rightarrow B \mid \cancel{S} \mid b \mid \mathbf{ASA \mid aB \mid a \mid SA \mid AS \mid}$

$B \rightarrow b$

Exemplo

Converta as regras remanescentes para a forma apropriada acrescentando variáveis e regras adicionais.

$S_0 \rightarrow A\mathbf{A_1} \mid \mathbf{A_2}B \mid a \mid SA \mid AS$

$S \rightarrow A\mathbf{A_1} \mid \mathbf{A_2}B \mid a \mid SA \mid AS$

$A \rightarrow b \mid A\mathbf{A_1} \mid \mathbf{A_2}B \mid a \mid SA \mid AS$

$B \rightarrow b$

$\mathbf{A_1} \rightarrow \mathbf{SA}$

$\mathbf{A_2} \rightarrow \mathbf{a}$

Exemplo 2

$S \longrightarrow aSb \mid bSa \mid SS \mid \epsilon$

1) Criação de nova variável inicial:

$S0 \longrightarrow S$

$S \longrightarrow aSb \mid bSa \mid SS \mid \epsilon$

2) Eliminação de substituições ϵ :

$S0 \longrightarrow S \mid \epsilon$

$S \longrightarrow aSb \mid bSa \mid SS \mid ab \mid ba \mid S$

Exemplo 2

3) Eliminação de regras unitárias:

$S0 \longrightarrow \epsilon \mid aSb \mid bSa \mid SS \mid ab \mid ba$

$S \longrightarrow aSb \mid bSa \mid SS \mid ab \mid ba$

4) Padronização:

$S0 \longrightarrow \epsilon \mid AT \mid BU \mid SS \mid AB \mid BA$

$S \longrightarrow AT \mid BU \mid SS \mid AB \mid BA$

$T \longrightarrow SB$

$U \longrightarrow SA$

$A \longrightarrow a$

$B \longrightarrow b$