

ACH2043  
INTRODUÇÃO À TEORIA DA COMPUTAÇÃO

**Aula 10b**

Cap. 2.1 – Gramáticas Livres de Contexto

Profa. Ariane Machado Lima  
ariane.machado@usp.br

# Introdução a gramáticas livres de contexto

# Aula 8

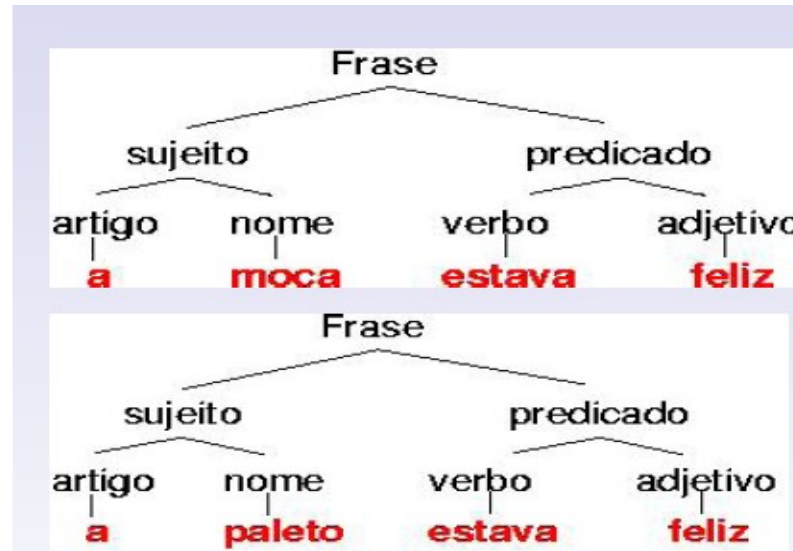
- Conceitos básicos de gramáticas e hierarquia de Chomsky

# Gramáticas

conjunto de produções

símbolo inicial

Frase	→	sujeito	predicado
sujeito	→	artigo	nome
artigo	→	a	
artigo	→	o	
nome	→	paletó	
nome	→	moça	
nome	→	dia	
predicado	→	verbo	adjetivo
verbo	→	é	
verbo	→	estava	
adjetivo	→	feliz	
adjetivo	→	azul	



símbolos não-terminais

símbolos terminais

# Gramáticas

- Definição: uma gramática  $G$  é uma quádrupla  $(V, \Sigma, S, P)$ , onde
  - $V$  é o conjunto de símbolos não-terminais (variáveis)
  - $\Sigma$  é o conjunto de símbolos terminais
  - $S$  é o símbolo inicial
  - $P$  é o conjunto de produções da forma
$$(\Sigma \cup V)^* V (\Sigma \cup V)^* \rightarrow (\Sigma \cup V)^*$$

# Hierarquia de Chomsky

$\alpha \rightarrow \beta$

Linguagens irrestritas  
(tipo 0)

$\alpha \in (V \cup \Sigma)^* V (V \cup \Sigma)^*$   
 $\beta \in (V \cup \Sigma)^*$

Linguagens sensíveis ao contexto  
(tipo 1)

$\alpha \in (V \cup \Sigma)^* V (V \cup \Sigma)^*$   
 $\beta \in (V \cup \Sigma)^*$   
 $|\alpha| \leq |\beta|$

Linguagens livres de contexto  
(tipo 2)

$\alpha \in V$

Linguagens regulares  
(tipo 3)

$\beta \in (V \cup \Sigma)^*$

$\alpha \in V$

$\beta \in \Sigma_\varepsilon, \beta \in V, \beta \in (V \cup \Sigma)^*$

# Hierarquia de Chomsky

$$\alpha \rightarrow \beta$$

Linguagens irrestritas  
(tipo 0)

$$\alpha \in (V \cup \Sigma)^* V (V \cup \Sigma)^* \\ \beta \in (V \cup \Sigma)^*$$

Linguagens sensíveis ao contexto  
(tipo 1)

$$\alpha \in (V \cup \Sigma)^* V (V \cup \Sigma)^* \\ \beta \in (V \cup \Sigma)^* \\ |\alpha| \leq |\beta|$$

Linguagens livres de contexto  
(tipo 2)

$$\alpha \in V \\ \beta \in (V \cup \Sigma)^*$$

Linguagens regulares  
(tipo 3)

$$\alpha \in V$$

$$\beta \in \Sigma_\varepsilon, \beta \in V, \beta \in (V \Sigma \cup \Sigma V)$$

# Aula de hoje

$$\alpha \rightarrow \beta$$

Linguagens irrestritas  
(tipo 0)

$$\alpha \in (V \cup \Sigma)^* V (V \cup \Sigma)^* \\ \beta \in (V \cup \Sigma)^*$$

Linguagens sensíveis ao contexto  
(tipo 1)

$$\alpha \in (V \cup \Sigma)^* V (V \cup \Sigma)^* \\ \beta \in (V \cup \Sigma)^* \\ |\alpha| \leq |\beta|$$

Linguagens livres de contexto  
(tipo 2)

$$\alpha \in V$$

$$\beta \in (V \cup \Sigma)^*$$

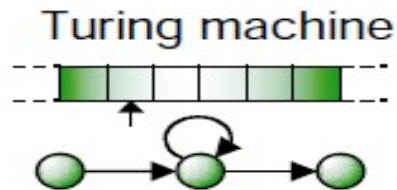
Linguagens regulares  
(tipo 3)

$$\alpha \in V$$

$$\beta \in \Sigma_\varepsilon, \beta \in V, \beta \in (V \Sigma \cup \Sigma V)$$



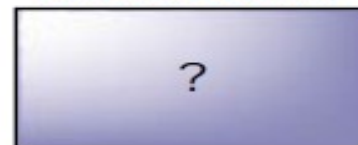
Recursively enumerable languages



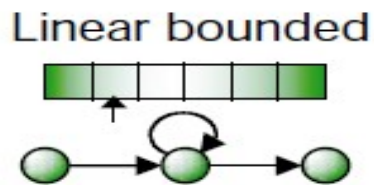
Unrestricted

$$Baa \rightarrow A$$

Undecidable



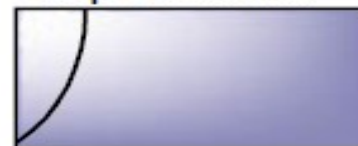
Context-sensitive languages



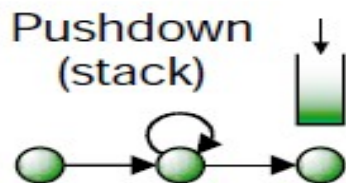
Context sensitive

$$At \rightarrow aA$$

Exponential?



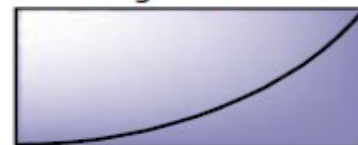
Context-free languages



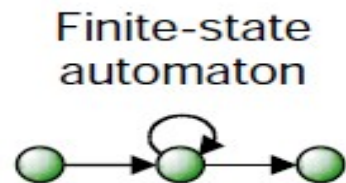
Context free

$$S \rightarrow gSc$$

Polynomial



Regular languages



Regular

$$A \rightarrow cA$$

Linear



# Gramáticas Livres de Contexto

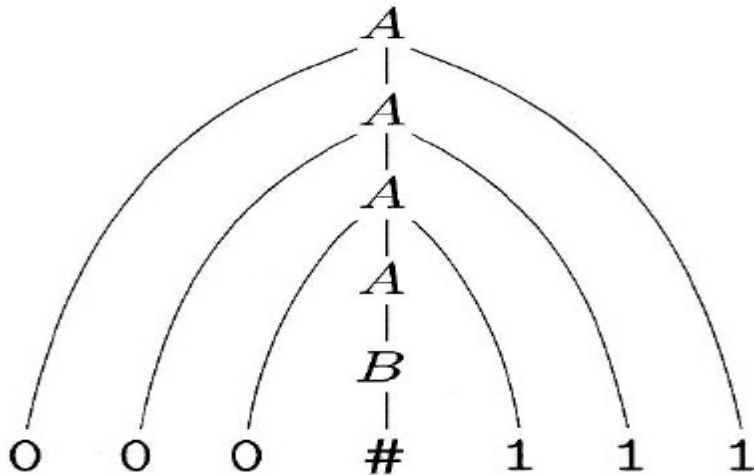
$$A \rightarrow 0A1$$

$$A \rightarrow B$$

$$B \rightarrow \#$$

Por exemplo, a gramática  $G_1$  gera a cadeia  $000\#111$ .

$$A \Rightarrow 0A1 \Rightarrow 00A11 \Rightarrow 000A111 \Rightarrow 000B111 \Rightarrow 000\#111$$



Árvore sintática  
ou  
Árvore de derivação

# Projetando gramáticas livres de contexto

Lembram-se da linguagem  $A = \{0^n 1^n \mid n \geq 0\}$ ?

Era regular?

# Projetando gramáticas livres de contexto

Lembram-se da linguagem  $A = \{0^n 1^n \mid n \geq 0\}$ ?

Era regular?

É livre de contexto! (**Estritamente** livre de contexto, pois não é regular)

Qual seria a gramática que a gera?

# Projetando gramáticas livres de contexto

Lembram-se da linguagem  $A = \{0^n 1^n \mid n \geq 0\}$ ?

Era regular?

É livre de contexto! (**Estritamente** livre de contexto, pois não é regular)

Qual seria a gramática que a gera?

$S \rightarrow 0 S 1$

$S \rightarrow \varepsilon$

# Projetando gramáticas livres de contexto

Lembram-se da linguagem  $A = \{0^n 1^n \mid n \geq 1\}$ ?



Era regular?

É livre de contexto!

Qual seria a gramática que a gera?

$S \rightarrow 0 S 1$

$S \rightarrow \varepsilon$

# Projetando gramáticas livres de contexto

Lembram-se da linguagem  $A = \{0^n 1^n \mid n \geq 1\}$ ?

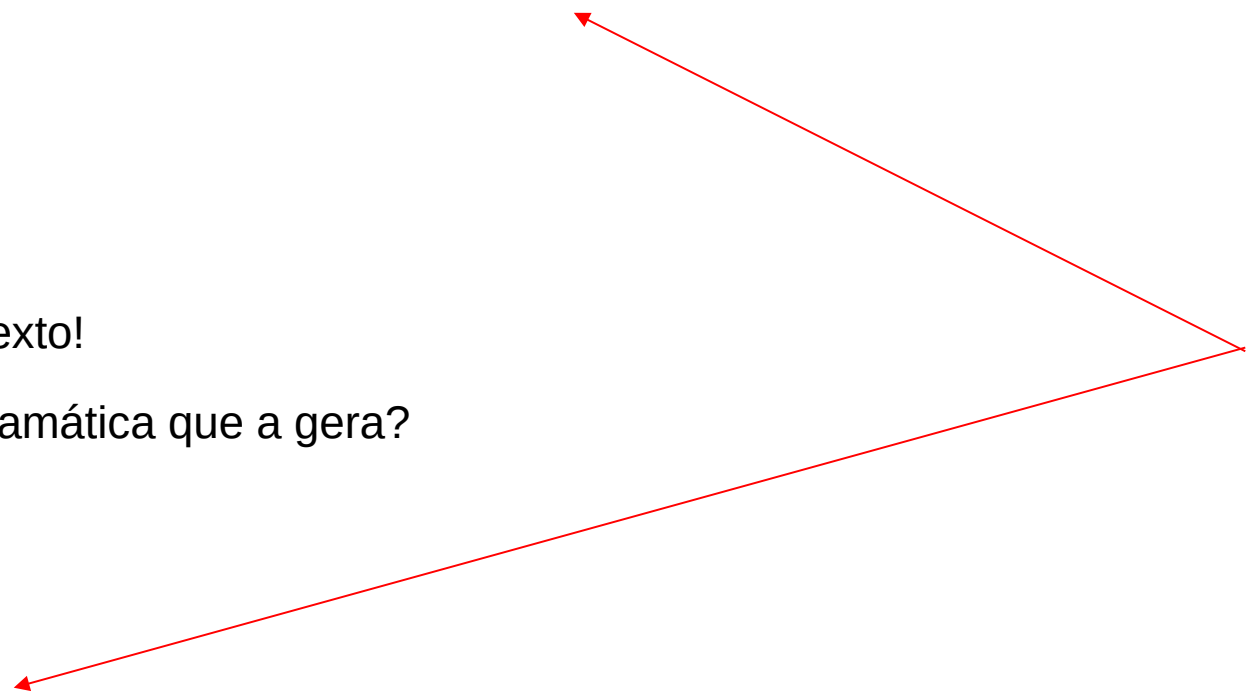
Era regular?

É livre de contexto!

Qual seria a gramática que a gera?

$S \rightarrow 0 S 1$

$S \rightarrow 01$



# Dicas para projetar gramáticas livres de contexto

- É a união de linguagens mais simples?

Por exemplo, para obter uma gramática para a linguagem  $\{0^n 1^n \mid n \geq 0\} \cup \{1^n 0^n \mid n \geq 0\}$ , primeiro construa a gramática

$$S_1 \rightarrow 0S_1 1 \mid \epsilon$$

para a linguagem  $\{0^n 1^n \mid n \geq 0\}$  e a gramática

?

para a linguagem  $\{1^n 0^n \mid n \geq 0\}$  e então adicione a regra

para dar a gramática



# Dicas para projetar gramáticas livres de contexto

- É a união de linguagens mais simples?

Por exemplo, para obter uma gramática para a linguagem  $\{0^n 1^n \mid n \geq 0\} \cup \{1^n 0^n \mid n \geq 0\}$ , primeiro construa a gramática

$$S_1 \rightarrow 0S_1 1 \mid \epsilon$$

para a linguagem  $\{0^n 1^n \mid n \geq 0\}$  e a gramática

$$S_2 \rightarrow 1S_2 0 \mid \epsilon$$

para a linguagem  $\{1^n 0^n \mid n \geq 0\}$  e então adicione a regra  $S \rightarrow S_1 \mid S_2$  para dar a gramática

# Dicas para projetar gramáticas livres de contexto

- É a união de linguagens mais simples?

Por exemplo, para obter uma gramática para a linguagem  $\{0^n 1^n \mid n \geq 0\} \cup \{1^n 0^n \mid n \geq 0\}$ , primeiro construa a gramática

$$S_1 \rightarrow 0S_1 1 \mid \epsilon$$

para a linguagem  $\{0^n 1^n \mid n \geq 0\}$  e a gramática

$$S_2 \rightarrow 1S_2 0 \mid \epsilon$$

para a linguagem  $\{1^n 0^n \mid n \geq 0\}$  e então adicione a regra  $S \rightarrow S_1 \mid S_2$  para dar a gramática

$$\begin{aligned} S &\rightarrow S_1 \mid S_2 \\ S_1 &\rightarrow 0S_1 1 \mid \epsilon \\ S_2 &\rightarrow 1S_2 0 \mid \epsilon . \end{aligned}$$

# Dicas para projetar gramáticas livres de contexto

- Definições recursivas

## EXEMPLO 2.3

---

Considere a gramática  $G_3 = (\{S\}, \{a, b\}, R, S)$ . O conjunto de regras,  $R$ , é

$$S \rightarrow aSb \mid SS \mid \epsilon.$$

# Dicas para projetar gramáticas livres de contexto

- Definições recursivas

## EXEMPLO 2.3

---

Considere a gramática  $G_3 = (\{S\}, \{a, b\}, R, S)$ . O conjunto de regras,  $R$ , é

$$S \rightarrow aSb \mid SS \mid \epsilon.$$

Dá para representar RNAs com isso!!!!

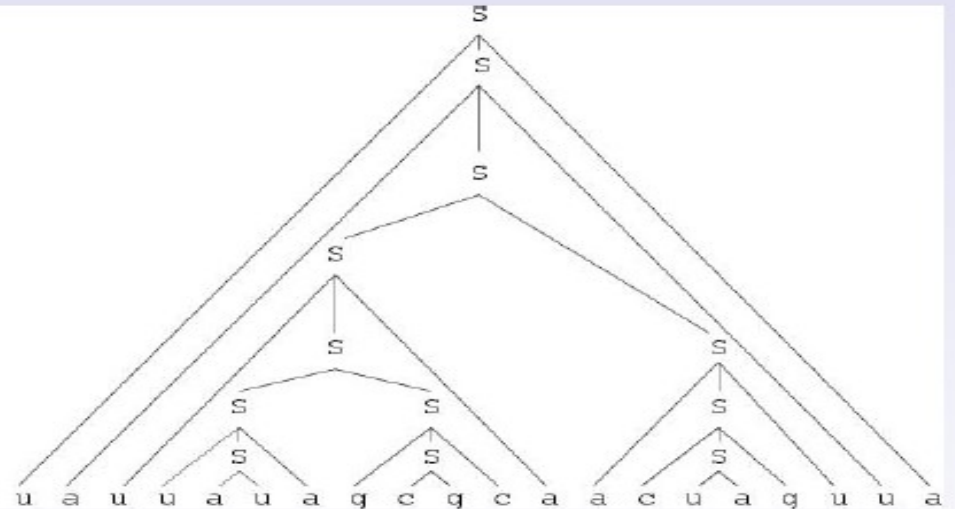
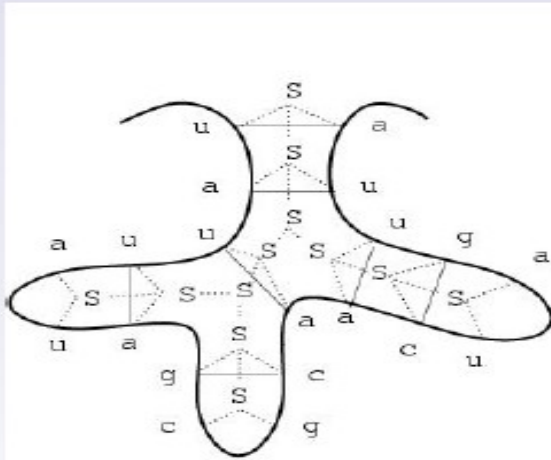
# Aplicações de gramáticas livres de contexto

Abrindo parêntesis ....

para exemplificar algumas aplicações importantes de gramáticas livres de contexto....

# Estrutura secundária de RNAs

S →	aSu [0.1]	uSa [0.1]	cSg [0.1]	gSc [0.1]	
S →	aS [0.1]	uS [0.1]	cS [0.1]	gS [0.1]	
S →	SS [0.04]	a [0.04]	u [0.04]	c [0.04]	g [0.04]





Search Rfam Search

### Rfam 14.3 (September 2020, 3446 families)

The Rfam database is a collection of RNA families, each represented by **multiple sequence alignments, consensus secondary structures** and **covariance models (CMs)**. [More...](#)



Search Rfam Search

Examples: [SAM](#), [Homo sapiens](#), [snoRNA](#), [author:"Weinberg"](#)

Browse [Families](#), [Clans](#), [Motifs](#), [Genomes](#), or [Families with 3D structures](#)

- QUICK LINKS**
- SEQUENCE SEARCH** YOU CAN FIND DATA IN RFAM IN VARIOUS WAYS...  
Analyze your RNA sequence for Rfam matches
- VIEW AN RFAM FAMILY** View Rfam family annotation and alignments
- VIEW AN RFAM CLAN** View Rfam clan details
- KEYWORD SEARCH** Query Rfam by keywords
- TAXONOMY SEARCH** Fetch families or sequences by NCBI taxonomy
- JUMP TO** enter any accession or ID

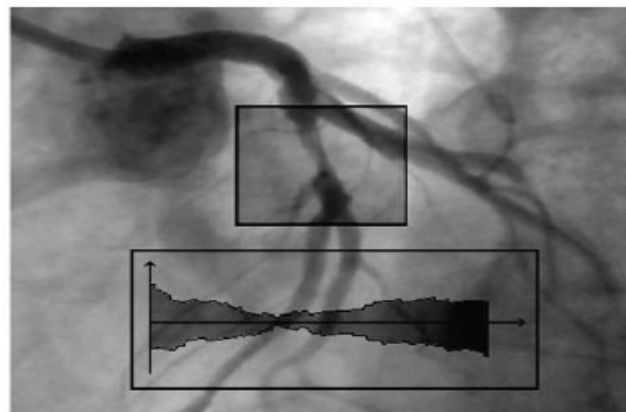


## Syntactic reasoning and pattern recognition for analysis of coronary artery images

Marek R. Ogiela\*, Ryszard Tadeusiewicz

*Institute of Automatics 30 Mickiewicza Avenue, University of Mining and Metallurgy,  
PL-30-059, Krakow, Poland*

Received 5 March 2002; received in revised form 23 April 2002; accepted 23 April 2002



$$V_N = \{\text{SYMPTOM}, \text{STENOSIS}, H, V, NV\}$$

$$V_T = \{h, v, nv\} \text{ for } h \in (-10^\circ, 10^\circ), v \in (11^\circ, 90^\circ), nv \in (-11^\circ, -90^\circ)$$

1. SYMPTOM  $\rightarrow$  STENOSIS
2. STENOSIS  $\rightarrow$  NV H V
3. STENOSIS  $\rightarrow$  NV VINV H
4. V  $\rightarrow$  v|V v
5. NV  $\rightarrow$  nv|NV nv
6. H  $\rightarrow$  h|H h

# Linguagens de programação

- Gramática que define uma linguagem de programação
- Compilador que checa se o programa está de acordo com a gramática (se o programa pertence à linguagem gerada pela gramática)

1  $S \rightarrow S ; S$   
2  $S \rightarrow \text{id} := E$   
3  $S \rightarrow \text{print} ( L )$

4  $E \rightarrow \text{id}$   
5  $E \rightarrow \text{num}$   
6  $E \rightarrow E + E$   
7  $E \rightarrow ( S , E )$

8  $L \rightarrow E$   
9  $L \rightarrow L , E$

Outro exemplo:

[http://pubs.opengroup.org/onlinepubs/7908799/xcu/awk.html#tag\\_000\\_000\\_108\\_016](http://pubs.opengroup.org/onlinepubs/7908799/xcu/awk.html#tag_000_000_108_016)

# Using Grammars for Pattern Recognition in Images: A Systematic Review

RICARDO WANDRÉ DIAS PEDRO, FÁTIMA L. S. NUNES,  
and ARIANE MACHADO-LIMA, School of Arts, Sciences and Humanities, University of Sao Paulo,  
Brazil

26

Grammars are widely used to describe string languages such as programming and natural languages and, more recently, biosequences. Moreover, since the 1980s grammars have been used in computer vision and related areas. Some factors accountable for this increasing use regard its relatively simple understanding and its ability to represent some semantic pattern models found in images, both spatially and temporally. The objective of this article is to present an overview regarding the use of syntactic pattern recognition methods in image representations in several applications. To achieve this purpose, we used a systematic review process to investigate the main digital libraries in the area and to document the phases of the study in order to allow the auditing and further investigation. The results indicated that in some of the studies retrieved, manually created grammars were used to comply with a particular purpose. Other studies performed a learning process of the grammatical rules. In addition, this article also points out still unexplored research opportunities in the literature.

Categories and Subject Descriptors: F.4.2 [Mathematical Logic and Formal Language]: Grammars and Other Rewriting Systems—*Decision problems*; I.4.8 [Image Processing and Computer Vision]: Scene Analysis; I.4.10 [Image Processing and Computer Vision]: Image Representation; I.5.1 [Pattern Recognition]: Models—*Structural*

General Terms: Algorithms, Theory

Additional Key Words and Phrases: Image grammars, computer vision, image representation, formal languages, syntactic methods, pattern recognition

## ACM Reference Format:

Pedro, R. W. D., Nunes, F. L. S., and Machado-Lima, A. 2013. Using grammars for pattern recognition in images: A systematic review. *ACM Comput. Surv.* 46, 2, Article 26 (November 2013), 34 pages.  
DOI: <http://dx.doi.org/10.1145/2543581.2543593>



# Towards an approach using grammars for automatic classification of masses in mammograms

Ricardo Wandré Dias Pedro<sup>1</sup>  | Ariane Machado-Lima<sup>2</sup> |  
Fátima L. S. Nunes<sup>1,2</sup>

... fechando parêntesis.

# Exercício – árvore sintática (baseado no ex 2.1 do Sipser 2<sup>a</sup>. ed)

Construa as árvores sintáticas das cadeias abaixo segundo a gramática:

$$E \rightarrow E + T \mid T$$

$$T \rightarrow T \times F \mid F$$

$$F \rightarrow (E) \mid a$$

- a) a
- b) a + a + a
- c) ((a))
- d) a + a x a

# Exercício – projeto de gramática (baseado nos ex 4 e 7 do cap 4 do livro de RAMOS)

Construa gramáticas livres de contexto para as seguintes linguagens:

1)  $\{a^i b^{2^i} \mid i \geq 1\}$

2)  $\{ww^R \mid w \in \{a, b, c\}^*\}$

3)  $\{w c w^R \mid w \in \{a, b\}^*\}$

4)  $\{a^m b^m c^n d^n \mid m \geq 0, n \geq 1\}$

5)  $\{a^m b^n c^n d^m \mid m \geq 1, n \geq 0\}$

6)  $\{a a^i b^i c^j \mid i \geq 1, j \geq i\}$

# Referências

SIPSER, M. Introdução à Teoria da Computação. Ed. Thomson.  
Cap 2.1

RAMOS, M. V. M.; NETO, J. J.; VEGA, I. S. Linguagens Formais.  
Ed. Bookman, Cap 4



ACH2043  
INTRODUÇÃO À TEORIA DA COMPUTAÇÃO

**Aula 10b**

Cap. 2.1 – Gramáticas Livres de Contexto

Profa. Ariane Machado Lima  
ariane.machado@usp.br