

[Aula 02] Linguagens e gramáticas

Prof. João F. Mari
joaof.mari@ufv.br

ROTEIRO

- Linguagem
- Alfabeto
- Palavra
- Linguagem formal
- Gramática
- Exercícios

LINGUAGEM

Linguagem

- Linguagem (Dicionário Aurélio)
 - *“O uso da palavra articulada ou escrita como meio de expressão e comunicação entre pessoas.”*
 - Insuficientemente precisa para o desenvolvimento matemático de uma teoria baseada em linguagens.
- **Linguagem:**
 - Conceito fundamental em Computação e Informática.
- Para definir linguagem:
 - **Alfabeto;**
 - **Palavra** ou cadeia de caracteres.

ALFABETO

Alfabeto

- Alfabeto:
 - Conjunto **finito** de símbolos ou caracteres.
- Símbolo ou Caractere:
 - Entidade **abstrata** básica, **não definida** formalmente. (*)
 - Base para definições:
 - **[EX]** letras e dígitos.
- Portanto:
 - Um conjunto infinito não é alfabeto.
 - \emptyset é um alfabeto.

[EX] Alfabeto

- São alfabetos
 - $\{ a, b, c \}$
 - \emptyset (conjunto vazio)
- Não são alfabetos (por quê?)
 - \mathbb{N} (conjunto dos números naturais)
 - $\{ a, b, aa, ab, ba, bb, aaa, \dots \}$

Símbolo, caractere ou letra

- Embora os **símbolos** também possam ser chamados de caracteres ou letras:
 - Eles **não precisam** ter necessariamente um único caractere.
 - Além disso, os símbolos de um alfabeto **não precisam** todos ter o mesmo número de caracteres.
 - A única **restrição** é que o tamanho do símbolo seja finito.

[EX] Símbolo, caractere ou letra

- Linguagem Natural:
 - Uma **palavra** em português equivale à um **símbolo**;
 - Uma **sentença** da língua portuguesa é uma cadeia composta por **vários símbolos**;
- **Alfabeto** de uma linguagem de programação como Pascal, C, ou Java.
 - O conjunto de todos os **símbolos válidos** no programa.
 - Cada **programa** escrito numa linguagem computacional corresponde a uma **cadeia de símbolos** que podem ser:
 - Identificadores: nomes de variáveis, funções, classes, ...;
 - Palavras reservadas;
 - Símbolos especiais e operadores;
 - Constantes numéricas.
- Alfabeto binário { a, b } ou { 0, 1 }
 - Domínio de valores de um **bit**;
 - Analogia com a representação interna dos computadores reais;
 - Poucos símbolos: simplifica as diversas abordagens desenvolvidas.

PALAVRA

Palavra

- **Palavra, Cadeia de Caracteres ou Sentença** sobre um alfabeto:
 - Sequência **finita** de símbolos **justapostos**.
- Cadeia sem símbolos:
 - ϵ - cadeia vazia ou palavra vazia
- **Prefixo, Sufixo e Subpalavra**
 - **Prefixo:**
 - qualquer sequência inicial de símbolos da palavra
 - **Sufixo:**
 - qualquer sequência final de símbolos da palavra
 - **Subpalavra:**
 - qualquer sequência de símbolos contíguos da palavra

[EX] Palavra, prefixo, sufixo e subpalavra

- **abcb** palavra sobre o alfabeto {a, b, c}
 - ϵ , **a**, **ab**, **abc**, **abcb** são todos os prefixos
 - ϵ , **b**, **cb**, **bcb**, **abcb** são todos os sufixos
 - Qualquer **prefixo** ou **sufixo** é uma **subpalavra**.

[EX] Palavra – linguagem de programação

- Em uma linguagem de programação como Pascal, C ou Java.
 - Uma **palavra** é um **programa**.

Alfabeto da linguagem Pascal

{program, var, integer, real, char, begin, end, if, then, else, for,..., ;, “”, :, :=, ., ...}



O código fonte de um programa corresponde à uma cadeia formada a partir de símbolos do alfabeto.

```
Program Teste;
Var
  i: integer;
Begin
  i:=0;
End.
```



LINGUAGEM
 Conjunto de todas as cadeias descritas a partir do alfabeto que respeitam um conjunto de regras sintáticas.

Concatenação de palavras

- Concatenação de Palavras (concatenação)
 - Operação binária sobre um conjunto de palavras;
 - Associa a cada par de palavras:
 - Palavra formada pela justaposição da primeira com a segunda.
- Notação:
 - justaposição dos símbolos que representam as palavras componentes.
- Propriedades:
 - Elemento Neutro: $\epsilon w = w = w \epsilon$
 - Associativa: $v(w t) = (v w)t$
 - Parênteses podem ser omitidos: $v w t$

[EX] Concatenação de palavras

- $\Sigma = \{ a, b \}$ um alfabeto.
- Para $v = baaaa$ e $w = bb$:
 - $v w = baaaabb$
 - $v \varepsilon = v = baaaa$

Concatenação sucessiva de palavras

- **Concatenação Sucessiva de uma Palavra:**
 - Ou simplesmente **Concatenação Sucessiva**;
 - Concatenação Sucessiva de uma Palavra com ela mesma;
 - w^n : n é o número de concatenações sucessivas.
- Indutivamente a partir da operação de concatenação:
 - $w^0 = \varepsilon$
 - $w^n = w w^{n-1}$, para $n > 0$

[EX] Concatenação sucessiva de palavras

- **w** palavra e **a** símbolo
 - $w^3 = www$
 - $w^1 = w$
 - $a^5 = aaaaa$
 - $a^n = aaa...a$
 - O símbolo **a** repetido n vezes.

Conjunto de todas as palavras

- Se Σ é um alfabeto
 - Σ^* conjunto de todas as palavras possíveis sobre Σ
 - Ou **Fechamento de um Alfabeto**;
 - Fecho transitivo e reflexivo
$$\Sigma^* = \Sigma^0 \cup \Sigma^1 \cup \Sigma^2 \cup \dots \cup \Sigma^n \cup \dots$$
 - $\Sigma^+ = \Sigma^* - \{ \epsilon \}$
 - Ou **Fechamento positivo de um Alfabeto**;
 - Fecho transitivo
- Σ^* é indutivamente definido
 - Base de Indução
 - $\epsilon \in \Sigma^*$;
 - Para qualquer $x \in \Sigma$, vale $x \in \Sigma^*$.
 - Passo de Indução
 - Se u e v são palavras de Σ^* ,
 - então a concatenação uv é uma palavra de Σ^* .

Palavra sobre um alfabeto

- Uma palavra sobre o alfabeto Σ é:
 - Qualquer elemento w de Σ^*
 - $w \in \Sigma^*$
- **[EX]** Conjunto de Todas as Palavras:
 - Se $\Sigma = \{ a, b \}$, então:
 - $\Sigma^+ = \{ a, b, aa, ab, ba, bb, aaa, \dots \}$
 - $\Sigma^* = \{ \epsilon, a, b, aa, ab, ba, bb, aaa, \dots \}$

Comprimento (tamanho) de uma palavra

- O comprimento de uma palavra w :
 - representado por $|w|$;
 - **número de símbolos** que compõem a palavra;
 - função com **domínio** em Σ^* e **codomínio** em \mathbb{N} .
- **[EX]** Comprimento de uma palavra
 - $|abcb| = 4$
 - $|\epsilon| = 0$

LINGUAGEM FORMAL

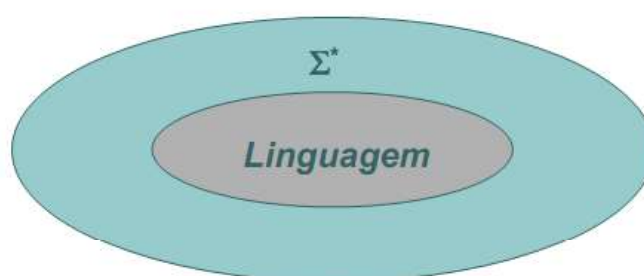
Linguagem formal

- **Linguagem Formal**

- dado o alfabeto Σ , o conjunto de palavras L é uma linguagem sobre Σ , se e somente se

$$L \subseteq \Sigma^*$$

- L é um **subconjunto** do conjunto de todas as palavras sobre Σ (Σ^*).



[EX] Linguagem formal

- \emptyset e $\{\varepsilon\}$ são linguagens sobre qualquer alfabeto:
 - $\emptyset \neq \{\varepsilon\}$
- Σ^* e Σ^+ são linguagens sobre um Σ qualquer:
 - $\Sigma^* \neq \Sigma^+$
- Conjunto de palíndromos sobre $\Sigma = \{a, b\}$:
 - $\varepsilon, a, b, aa, bb, aaa, aba, bab, bbb, aaaa, \dots$

Operações sobre linguagens

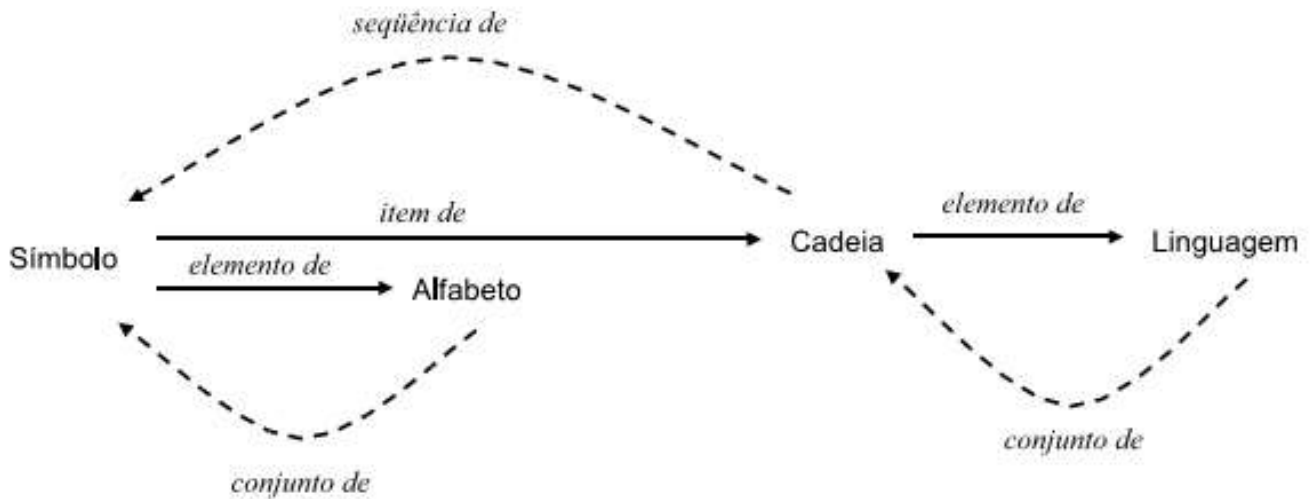
- Considere $L1$ e $L2$ linguagens definidas sobre Σ :
 - União:
 - $L1 \cup L2 = \{x \mid x \in L1 \text{ OU } x \in L2\}$
 - Intersecção:
 - $L1 \cap L2 = \{x \mid x \in L1 \text{ E } x \in L2\}$
 - Diferença:
 - $L1 - L2 = \{x \mid x \in L1 \text{ E } x \notin L2\}$
 - Concatenação:
 - $L1 L2 = \{x \mid x = yz, y \in L1 \text{ E } z \in L2\}$
 - Complemento:
 - $L1' = \{x \mid x \in \Sigma^* \text{ E } x \notin L1\}$

[EX] Operações sobre linguagens

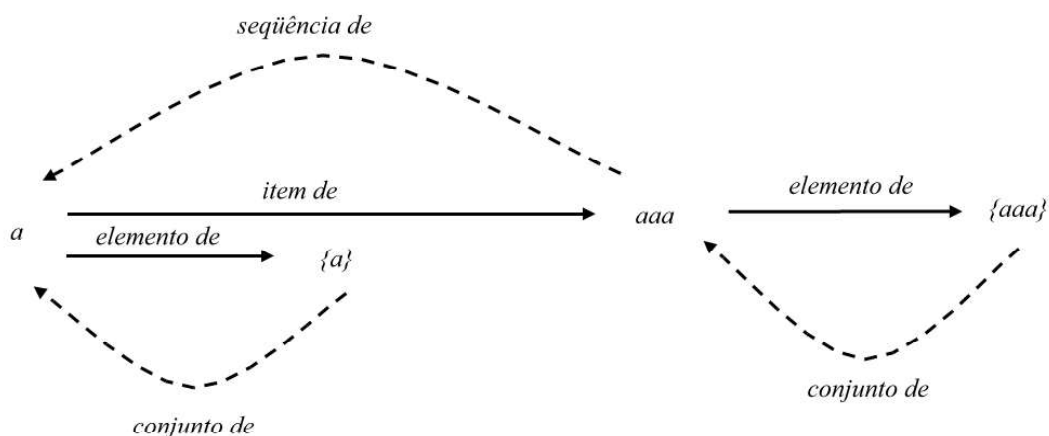
- Sejam $L1$ e $L2$ definidas sobre $\{0,1\}$:
 - $L1 = \{0, 11\}$
 - $L2 = \{0, 1, 00\}$
- $L1 \cup L2 = \{0, 1, 00, 11\}$
- $L1 \cap L2 = \{0\}$
- $L1 - L2 = \{11\}$
- $L1 L2 = \{00, 01, 000, 110, 111, 1100\}$

Conjunto de todas as linguagem sobre um alfabeto

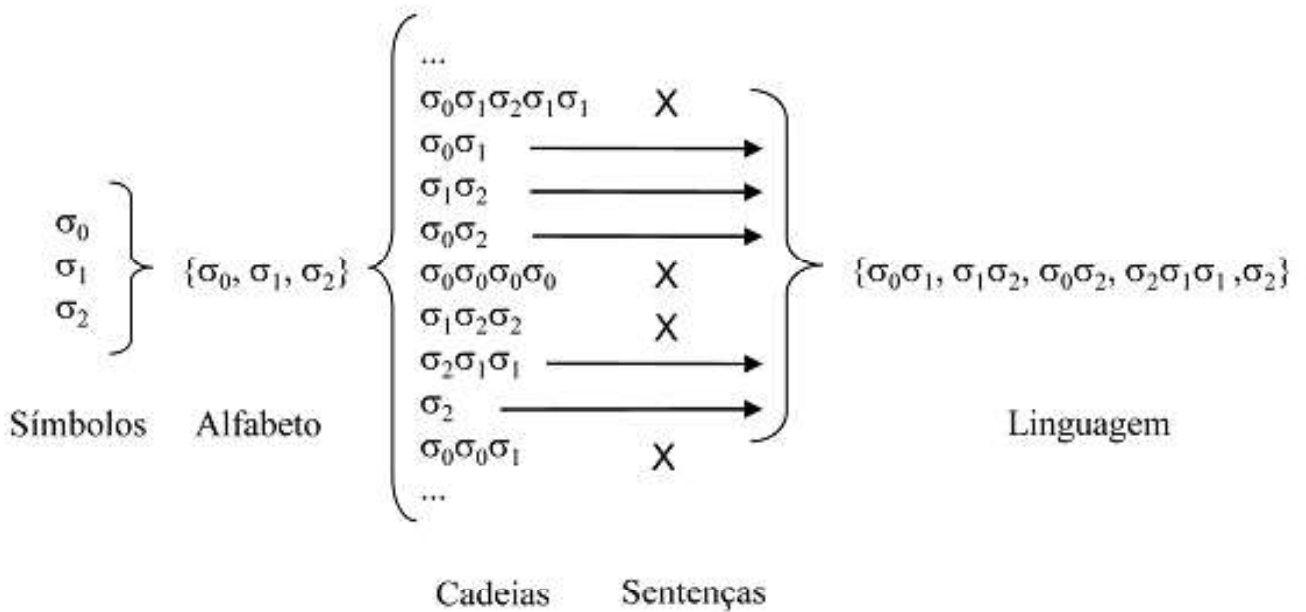
- É o **conjunto das partes** de Σ^*
 2^{Σ^*}
 - Ou seja, é o conjunto de todos os subconjuntos de Σ^*

[EX-1] Símbolos → Alfabeto → Palavras → Linguagem**[EX-1] Símbolos → Alfabeto → Palavras → Linguagem**

- O símbolo a é elemento do alfabeto $\{a\}$...
- E também um item da cadeia aaa ...
- Que por sua vez é elemento da linguagem $\{aaa\}$...
- Por outro lado, a linguagem $\{aaa\}$ é um conjunto que contém a cadeia aaa ...
- A cadeia aaa é uma sequência de símbolos a
- E o alfabeto $\{a\}$ contém o símbolo a ...



[EX-2] Símbolos → Alfabeto → Palavras → Linguagem



GRAMÁTICA

Gramática

- Linguagem de programação
 - Definida pelo conjunto de todos os programas (palavras).
- Linguagem de propósitos gerais (Pascal, C, Java, ...):
 - Conjunto de todos os programas é **infinito!**
 - Uma definição **inadequada** para implementação em computador.
- Formalismo **Gramática**
 - Uma maneira de **especificar**, de **forma finita**, linguagens (eventualmente) infinitas.

Gramática

- Gramática é, basicamente:
 - Conjunto finito de regras;
 - Quando aplicadas sucessivamente, geram palavras;
 - **Conjunto de todas as palavras geradas por uma gramática define a linguagem.**
- Gramáticas para linguagens naturais como Português:
 - As mesmas que as usadas para linguagens artificiais como Pascal, C ou Java.

Gramática

- Gramática, Gramática Irrestrita ou Gramática de Chomsky

$$G = (V, T, P, S)$$

- **V**: conjunto finito de **símbolos variáveis** ou não terminais;
- **T**: conjunto finito de **símbolos terminais** disjunto de V;
- **P**: Produções. Relação finita: $(V \cup T)^+ \rightarrow (V \cup T)^*$;
 - Par da relação: **regra de produção** ou **produção**;
- **S**: elemento distinguido de V: **símbolo inicial** ou **variável inicial**;
- Representação de uma **regra de produção** (α, β) :
 - $\alpha \rightarrow \beta$
- Representação abreviada para $\alpha \rightarrow \beta_1, \alpha \rightarrow \beta_2, \dots, \alpha \rightarrow \beta_n$:
 - $\alpha \rightarrow \beta_1 \mid \beta_2 \mid \dots \mid \beta_n$

Convenções

- **A, B, C, ..., S, T** para símbolos variáveis
- **a, b, c, ..., s, t** para símbolos terminais
- **u, v, w, x, y, z** para palavras de símbolos terminais
- **α, β, \dots** para palavras de símbolos variáveis ou terminais

Derivação

- Derivação:
 - É a **aplicação** de uma **regra de produção**;
 - **Aplicação sucessiva** de **regras de produção**:
 - Permite derivar palavras da linguagem;
 - Fecho transitivo da relação de derivação.

Derivação e relação de derivação

- Considere a gramática $\mathbf{G} = (\mathbf{V}, \mathbf{T}, \mathbf{P}, \mathbf{S})$
 - **Derivação** é um par da Relação de Derivação denotada por \Rightarrow
 - Domínio em $(V \cup T)^+$ e codomínio em $(V \cup T)^*$
 - $\langle \alpha, \beta \rangle$ é representado de forma infixada:
- $$\alpha \Rightarrow \beta$$
- \Rightarrow É indutivamente definida como segue:
 - Para toda produção da forma $S \rightarrow \beta$ (S é o símbolo inicial de G):

$$S \Rightarrow \beta$$
 - Para todo par $\eta \Rightarrow \rho \alpha \sigma$ da relação de derivação:
 - se $\alpha \rightarrow \beta$ é regra de P , então:

$$\eta \Rightarrow \rho \beta \sigma$$

Derivação e relação de derivação

- Derivação:
 - Substituição de uma **subpalavra** de acordo com uma regra de produção.
- Sucessivos passos de derivação
 - \Rightarrow^* Fecho transitivo e reflexivo da relação \Rightarrow
 - **Zero** ou mais passos de derivações sucessivos.
 - \Rightarrow^+ Fecho transitivo da relação \Rightarrow
 - **Um** ou mais passos de derivações sucessivos.
 - \Rightarrow^i
 - **Exatos i** passos de derivações sucessivos (i é natural).

Linguagem gerada

- Considere a gramática G :

$$G = (V, T, P, S)$$
- Linguagem Gerada por G :

$$L(G) \text{ ou } \text{GERA}(G)$$
- Palavras de símbolos terminais deriváveis a partir de S :

$$L(G) = \{ w \in T^* \mid S \Rightarrow^+ w \}$$
 - A linguagem gerada pela gramática G é formada por toda palavra (w) pertencente ao conjunto de todas as palavras possíveis sobre o alfabeto T (ou fechamento do alfabeto) tal que w tenha sido gerada por pelo menos 1 passo de sucessivas derivações da variável inicial S .

[EX] Números naturais

- Gramática $G = (V, T, P, N)$
 - $V = \{ N, D \}$
 - $T = \{ 0, 1, 2, \dots, 9 \}$
 - $P = \{ N \rightarrow D, N \rightarrow DN, D \rightarrow 0 \mid 1 \mid \dots \mid 9 \}$
- Gera, sintaticamente, o conjunto dos números naturais:
 - Se distinguem os zeros à esquerda.
 - **[EX]** 123 diferente de 0123.

[EX] Números naturais

- $G = (V, T, P, N)$
 - $V = \{ N, D \}$
 - $T = \{ 0, 1, 2, \dots, 9 \}$
 - $P = \{ N \rightarrow D, N \rightarrow DN, D \rightarrow 0 \mid 1 \mid \dots \mid 9 \}$
- Uma derivação do número 243
 - $N \Rightarrow N \rightarrow DN$
 - $DN \Rightarrow D \rightarrow 2$
 - $2N \Rightarrow N \rightarrow DN$
 - $2DN \Rightarrow D \rightarrow 4$
 - $24N \Rightarrow N \rightarrow D$
 - $24D \Rightarrow D \rightarrow 3$
 - 243

[EX] Números naturais

- Portanto:
 - $S \Rightarrow^* 243$
 - $S \Rightarrow^+ 243$
 - $S \Rightarrow^6 243$
- Interpretação indutiva da gramática
 - Base de Indução:
 - Todo dígito é natural
 - Passo de Indução:
 - Se n é natural, então a concatenação com qualquer dígito também é natural

[EX] Palavra duplicada

- $G = (\{ S, X, Y, A, B, F \}, \{ a, b \}, P, S)$
- Na qual:
 - $P = \{ S \rightarrow XY,$
 - $X \rightarrow XaA \mid XbB \mid F$
 - $Aa \rightarrow aA, Ab \rightarrow bA, AY \rightarrow Ya,$
 - $Ba \rightarrow aB, Bb \rightarrow bB, BY \rightarrow Yb,$
 - $Fa \rightarrow aF, Fb \rightarrow bF, FY \rightarrow \varepsilon \}$
- Gera a linguagem
 - $\{ ww \mid w \text{ é palavra de } \{ a, b \}^* \}$

[EX] Palavra duplicada

• Derivação de baba:

- S \Rightarrow S \rightarrow XY
- XY \Rightarrow X \rightarrow XaA
- XaAY \Rightarrow AY \rightarrow Ya
- XaYa \Rightarrow X \rightarrow XbB
- XbBaYa \Rightarrow Ba \rightarrow aB
- XbaBYa \Rightarrow BY \rightarrow Yb
- XbaYba \Rightarrow X \rightarrow F
- FbaYba \Rightarrow Fb \rightarrow bF
- bFaYba \Rightarrow Fa \rightarrow aF
- baFYba \Rightarrow FY \rightarrow ϵ
- baba

$$G = (\{S, X, Y, A, B, F\}, \{a, b\}, P, S)$$

$$P = \{ S \rightarrow XY, \\ X \rightarrow XaA \mid XbB \mid F, \\ Aa \rightarrow aA, \\ Ab \rightarrow bA, \\ AY \rightarrow Ya, \\ Ba \rightarrow aB, \\ Bb \rightarrow bB, \\ BY \rightarrow Yb, \\ Fa \rightarrow aF, \\ Fb \rightarrow bF, \\ FY \rightarrow \epsilon \}$$
[EX] Palavra duplicada

• Derivação de baba:

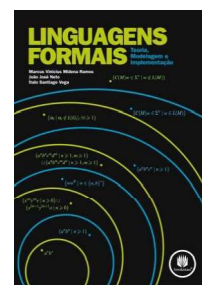
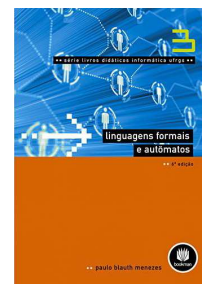
- S \Rightarrow S \rightarrow XY % Única transição possível.
- XY \Rightarrow X \rightarrow XaA % Gera um 'a' da primeira parte de b(a)ba.
- XaAY \Rightarrow AY \rightarrow Ya % Gera o outro 'a' bab(a).
- XaYa \Rightarrow X \rightarrow XbB % Gera um 'b' da primeira parte de (b)aba.
- XbBaYa \Rightarrow Ba \rightarrow aB % Caminha com 'B' para o início da outra metade.
- XbaBYa \Rightarrow BY \rightarrow Yb % Gera o outro 'b' de ba(b)a.
- XbaYba \Rightarrow X \rightarrow F % Transforma X em F para poder ser removido.
- FbaYba \Rightarrow Fb \rightarrow bF % Caminha com F até o início da segunda metade.
- bFaYba \Rightarrow Fa \rightarrow aF % Caminha com F até o início da segunda metade.
- baFYba \Rightarrow FY \rightarrow ϵ % Remove FY.
- baba

$$G = (\{S, X, Y, A, B, F\}, \{a, b\}, P, S)$$

$$P = \{ S \rightarrow XY, \\ X \rightarrow XaA \mid XbB \mid F, \\ Aa \rightarrow aA, \\ Ab \rightarrow bA, \\ AY \rightarrow Ya, \\ Ba \rightarrow aB, \\ Bb \rightarrow bB, \\ BY \rightarrow Yb, \\ Fa \rightarrow aF, \\ Fb \rightarrow bF, \\ FY \rightarrow \epsilon \}$$

BIBLIOGRAFIA

- MENEZES, P. B. Linguagens formais e autômatos, 6. ed., Bookman, 2011.
 - Capítulo 2.
 - + Slides disponibilizados pelo autor do livro.
- Bibliografia auxiliar
 - M.V.M. Ramos, J.J. Neto e I.S. Vega. Linguagens Formais: Teoria, Modelagem e Implementação. Bookman, 2009.
- Notas de aula do Prof. Yandre Costa
 - <https://sites.google.com/din.uem.br/prof-yandre-costa/lfa>



[FIM]

- FIM:
 - **[AULA 02]** Linguagens e Gramáticas
- Próxima aula:
 - **[AULA 03]** LINGUAGENS REGULARES – Autômato Finito Determinístico.