



---

---

---

---

---

---

---

### Plano de Ensino

- Apresentação. Revisão de Funções.
- **Expressões Regulares.**
- Gramática Regular.
- Autômatos Finitos Determinísticos.
- Conversão entre GR e AFD.
- Minimização de Autômatos.
- Autômatos Finitos Não-Determinísticos.
- Conversão de Autômatos AFD para AFND.
- Autômatos com Pilha.
- Máquinas de Turing.

---

---

---

---

---

---

---

### Livro-Texto

- Bibliografia Básica:
  - » MENEZES, Paulo Fernando Blauth. **Linguagens Formais e Autômatos**. 5ª ed. Porto Alegre: Bookman, 2008.
- Bibliografia Complementar:
  - » LEWIS, Ricki. **Elementos da Teoria da Computação**. 2ª ed. Porto Alegre: Bookman, 2004.
  - » HOPCROFT, John E; ULLMAN, Jeffrey D; MOTWANI, Rajeev, SOUZA. **Introdução a Teoria dos Autômatos, Linguagens e Computação**. 1ª ed. São Paulo: CAMPUS, 2003.

---

---

---

---

---

---

---

## 2. Introdução – Alfabeto



- Alfabeto ( $\Sigma$ )  $\rightarrow$  é um conjunto não vazio e finito de símbolos. Sendo assim, um conjunto também é considerado um alfabeto. Letras e dígitos são exemplos de símbolos usados frequentemente.

$\Sigma = \{a, e, i, o, u\}$

$\Sigma = \{a, b, c, d, e, \dots, z\}$

$\Sigma = \{0, 1\}$

---

---

---

---

---

---

---

## 2. Introdução – Palavra



- Palavra, cadeia de caracteres ou sentença  $\rightarrow$  é uma sequência finita de símbolos (do alfabeto) justapostos. Uma palavra sem símbolo ( $\epsilon \rightarrow$  vazia).

- Seja  $\Sigma = \{a, e, i, o, u\}$

» Palavra vazia ( $\epsilon$ )  $\rightarrow$  palavra sem símbolos  $\rightarrow$   
 $\Sigma = \{\epsilon\}$

» Conjunto de todas as palavras possíveis  $\rightarrow$   
 $\Sigma^* = \{\epsilon, a, ae, aei, aaeaa, aeioa, aeioou, \dots\}$

» Conjunto de todas as palavras possíveis excetuando-se a palavra vazia  $\rightarrow$   
 $\Sigma^+ = \{a, ae, aei, aaeaa, aeioa, aeioou, \dots\}$  ou  $\Sigma^+ = \Sigma^* - \{\epsilon\}$

---

---

---

---

---

---

---

## 2. Introdução – Palavra



- Tamanho de uma palavra  $\rightarrow$  o tamanho ou comprimento de uma palavra  $w$ , representado por  $|w|$  é o número de símbolos que compõem a palavra.

- Seja  $\Sigma = \{a, e, i, o, u\}$

» Se  $w = aei$  então  $|w| = 3$

» Se  $w = aeioouae$  então  $|w| = 8$

» Se  $w = \epsilon$  então  $|w| = 0$  (sentença vazia)

---

---

---

---

---

---

---

## 2. Introdução – Palavra



- Prefixo, Sufixo e Subpalavra  $\rightarrow$  é qualquer seqüência de símbolos inicial (prefixo) ou final (sufixo) da palavra. Qualquer prefixo ou sufixo de uma palavra é uma subpalavra.
- Seja uma palavra  $w = abcb$  em  $\Sigma = \{a, b, c\}$ 
  - » Prefixos:  $\epsilon, a, ab, abc, abcb$ .
  - » Sufixos:  $\epsilon, b, cb, bcb, abcb$ .

---

---

---

---

---

---

---

## 2. Introdução – Linguagem



- Uma linguagem formal é um conjunto de palavras sobre um alfabeto.
- Sendo  $\Sigma = \{a, b, c\}$ :
  - » O conjunto vazio e o conjunto formado pela palavra vazia são linguagens sobre  $\Sigma$  ( $\{\} \neq \{\epsilon\}$ )
  - » O conjunto de palíndromos (mesma leitura de ambos os lados) sobre  $\Sigma$  é um exemplo de linguagem infinita ( $\Sigma = \{\epsilon, a, b, aa, bb, aaa, bbb, aba, bab, aaaa, \dots\}$ ).

---

---

---

---

---

---

---

## 2. Introdução – Linguagem



- Concatenação  $\rightarrow$  é uma justaposição dos símbolos que representam as palavras componentes.
  - » Associatividade:  $v(wt) = (vw)t$
  - » Elemento neutro:  $\epsilon w = w = w\epsilon$
- Seja o alfabeto  $\Sigma = \{a, b, c\}$  e as palavras  $v = baaaa$  e  $w = bb$ .
  - »  $vw = baaaabb$
  - »  $v\epsilon = v = baaaa$

---

---

---

---

---

---

---

## 2. Introdução – Linguagem



- Concatenação sucessiva  $\rightarrow$  é uma justaposição com os símbolos da própria palavra de forma sucessiva; é representada na forma de expoente, ou seja,  $w^n$ , onde  $w$  é a palavra e  $n$  o número de concatenações consecutivas.
  - »  $w^0 = \varepsilon$
  - »  $w^n = ww^{n-1}$ , para  $n > 0$
- Seja  $w$  uma palavra. Então:
  - »  $w^1 = w$
  - »  $w^2 = ww$
  - »  $w^3 = www$
  - »  $w^4 = wwww$
  - »  $w^n = www...w$  ( $n$  vezes)

---

---

---

---


---

---

---

## 2. Introdução – Formalismo



- Frase em Português: “É **nóis**”.
- Frase em Inglês: “**They** **needs** to do this”.
- Placa: 
- Sintaxe em linguagem C: **valor** **:=** **valor** + 1;

---

---

---

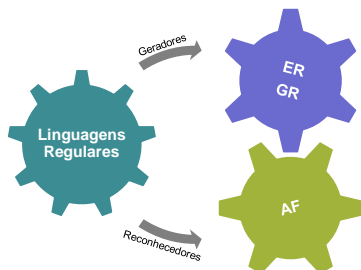
---

---

---

---

## 2. Introdução – Formalismo



---

---

---

---

---

---

---

## 2. Expressões Regulares



- Toda linguagem regular pode ser descrita por uma expressão simples, denominada Expressão Regular (ER).
- Trata-se de um formalismo gerador, pois expressa como construir (gerar) as palavras da linguagem.
- Uma ER é definida recursivamente a partir de conjuntos (linguagens) básicas e operação de concatenação e união.

---

---

---

---

---

---

---

---

## 2. Expressões Regulares



- Dado um alfabeto  $\Sigma$ :
  - » Os símbolos do alfabeto são expressões regulares, incluindo-se o vazio ou  $\epsilon$ .
  - » Se  $R_1$  e  $R_2$  são ER, então  $(R_1 \cup R_2)$  é uma ER.
    - »  $(R_1 | R_2)$  representa a união de linguagens.
  - » Se  $R_1$  e  $R_2$  são ER, então  $(R_1 R_2)$  é uma ER.
    - »  $R_1 R_2$  representa concatenação de linguagens
  - » Se  $R_1$  é uma ER, então  $(R_1)^*$  é uma ER;
    - »  $(R_1)^*$  representa a linguagem formada pela concatenação de zero ou mais palavras de  $R_1$
  - » Se  $R_1$  é uma ER, então  $(R_1)^+$  é uma ER;
    - »  $(R_1)^+$  representa a linguagem formada pela concatenação de um ou mais palavras de  $R_1$
    - » Obs:  $R_1^+ = R_1 R_1^*$

---

---

---

---

---

---

---

---

## 2. Expressões Regulares



- Dado um alfabeto  $\Sigma = \{a, b\}$ ; e as expressões regulares a seguir, teremos a linguagem gerada, conforme tabela:

ER	Linguagem Gerada
a	{a}
ab	{ab}
(a   b)	{a, b}
ba*	{b, ba, baa, baaa, ...}
(a)*	{ $\epsilon$ , a, aa, aaa, ...}
(a   b)*	{ $\epsilon$ , a, b, aa, ab, bb, abaa, ...}
(a (a   b))*	{ $\epsilon$ , aa, ab, aaaa, abaa, aaab, ...}
(a (a   b)*)	{aa, ab, aaa, aba, aab, ...}
((a   b)*   (a   b))*	{ $\epsilon$ , a, b, ab, aa, bb, aaa, aba, abb, ...}

---

---

---

---

---

---

---

---



**Linguagens Formais e  
Autômatos**

**Ciência da Computação**  
[clayton.valdo@anhanguera.com](mailto:clayton.valdo@anhanguera.com)



---

---

---

---

---

---

---