

# Introdução

## Compiladores



**INSTITUTO  
FEDERAL**  
Brasília

Prof. Daniel Saad Nogueira  
Nunes

IFB – Instituto Federal de Brasília,  
Campus Taguatinga



# Sumário

---

## 1 Introdução



# Introdução

---

- Expressões regulares são mecanismos que permitem descrever linguagens, isto é, conjuntos de palavras, de maneira compacta.
- O uso de expressões regulares pode ser utilizada para efetuar casamento de padrões, isto é, detectar padrões em texto.
- Uma aplicação imediata de expressões regulares é na construção de analisadores léxicos, visto que esse formalismo pode reconhecer e classificar os tokens.



# Introdução

---

- Nesta aula estudaremos este formalismo e criaremos expressões regulares para reconhecer diversos padrões.



# Sumário

---

## 2 Definições



# Definições: alfabeto

---

## Alfabeto

O alfabeto, denotado por  $\Sigma$ , corresponde à um conjunto **finito** de símbolos.



## Definições: alfabeto

---

- Vários programas em diversas linguagens são formadas inteiramente por símbolos ASCII: e.g. linguagem C.
- Outras linguagens, como Java e Python, consideram um alfabeto maior, como o Unicode.



# Definições: palavras

---

## Palavras

Palavras, ou strings, sobre um alfabeto  $\Sigma$ , são sequências **finitas** compostas unicamente de símbolos de  $\Sigma$ .





## Definições: palavras

---

### Exemplo

- As strings *cachorro*, *gato* e *papagaio* são palavras sobre o alfabeto  $\Sigma = \{a, \dots, z\}$ , isto é, o alfabeto das letras minúsculas.
- As strings *00100101*, *0110101* e *10* são palavras sobre o alfabeto  $\Sigma = \{0, 1\}$ .



## Definição: palavra vazia

---

### Palavra vazia

A palavra vazia, denotada por  $\varepsilon$  corresponde à uma sequência de símbolos de tamanho 0.



# Definições: linguagens

---

## Linguagens

Linguagens são conjuntos de palavras. Linguagens podem ter tamanho finito ou infinito.



## Definições: linguagens

---

### Exemplo

- $A = \{\text{cachorro}, \text{gato}, \text{papagaio}\}$  é uma linguagem formada por três palavras.  $\text{cachorro} \in A$ .
- $B = \{\varepsilon, 0, 1, 00, 01, 10, 11, 000, \dots\}$  é uma linguagem infinita contendo todas as palavras sobre o alfabeto  $\Sigma = \{0, 1\}$ .
- $\emptyset$  é a linguagem vazia.



# Definições: concatenação

---

## Concatenação

Se  $x = x_1 \dots x_n$  e  $y = y_1 \dots y_m$  são palavras, a concatenação de  $x$  e  $y$ , denotada por  $xy$ , corresponde à palavra  $xy = x_1 \dots x_n y_1 \dots y_m$ .  
Em especial  $\underbrace{xx \dots x}_k$  é denotado por  $x^k$



## Definições: concatenação

---

- Qualquer palavra concatenada com  $\varepsilon$  é a própria palavra:  
$$x\varepsilon = \varepsilon x = x.$$
- $x^0 = \varepsilon.$



## Definições: concatenação

---

### Concatenação

A definição de concatenação pode ser estendida para linguagens. Se  $A$  e  $B$  são linguagens,  $AB = \{xy \mid x \in A \text{ e } y \in B\}$ . Em especial,  $\underbrace{AA \dots A}_k$  é denotado por  $A^k$



## Definições: concatenação

---

- Temos que  $A^0 = \emptyset$ .





## Definições: concatenação

---

### Exemplo

- Se  $x = abra$  e  $y = cadabra$ ,  $xy = abracadabra$  e  $x^3 = abraabraabra$ .
- Se  $A = \{arroz, feijao\}$  e  $B = \{pure, batata\}$  então  $AB = \{arrozpure, feijaopure, arrozbatata, feijaobatata\}$  e  $A^2 = \{arrozarroz, arrozfeijao, feijaoarroz, feijaofeijao\}$ .



# Definições: fecho Kleene

---

## Fecho Kleene

O fecho Kleene para uma linguagem  $A$ , denotado por  $A^*$ , corresponde ao conjunto de palavras que pode ser formada pela concatenação de zero ou mais palavras de  $A$ . Formalmente temos:

$$A^* = \bigcup_{i \geq 0} A^i$$



## Definições: concatenação

---

### Exemplo

- Se  $A = \{0, 1\}$ , então  $A^*$  é o conjunto de todas as palavras binárias, isto é:  $A^* = \{\varepsilon, 0, 1, 00, 01, 10, 11, \dots\}$ .
- Se  $A = \{a, b, \dots, z\}$ , então  $A^*$  corresponde ao conjunto de todas as palavras que podem ser formadas por letras minúsculas (inclui  $\varepsilon$ ),



## Definições: fecho Kleene positivo

---

### Fecho Kleene

O fecho Kleene para uma linguagem  $A$ , denotado por  $A^+$ , corresponde ao conjunto de palavras que pode ser formada pela concatenação de **uma** ou mais palavras de  $A$ . Formalmente temos:

$$A^+ = \bigcup_{i \geq 1} A^i$$



## Definições: fecho Kleene positivo

---

### Exemplo

- Pelas definições anteriores, temos  $A^+ = A^*A$ .



## Definições: alternância

---

### Alternância

Se  $x_1, \dots, x_n$  são palavras, então  $x_1|x_2|\dots|x_n$  é o conjunto  $X = \{x_1, \dots, x_n\}$ .



## Definições: alternância

---

### Alternância

Essa noção pode ser estendida para linguagens. Se  $A$  e  $B$  são linguagens, então  $x \in A|B$  se e somente se  $x \in A$  **ou**  $x \in B$ . Esta noção é equivalente à união de conjuntos.



## Definições: fecho Kleene positivo

---

### Exemplo

- Se  $A = \{a, b, \dots, z\}$  e  $B = \{A, B, \dots, Z\}$  temos que  $A|B$  é o conjunto de todas as letras, minúsculas e maiúsculas.





# Sumário

---

## 3 Expressões regulares



# Expressões regulares

---

- Agora que temos os conceitos básicos definidos, podemos formalizar o que é uma expressão regular.



# Expressões regulares

---

## Definição

Tome um alfabeto  $\Sigma$ .  $R$  é uma expressão regular se:

- 1  $R = \{a\}$ , com  $a \in \Sigma$ .
- 2  $\{\varepsilon\}$ .
- 3  $\emptyset$ .
- 4  $(R_1|R_2)$  em que  $R_1$  e  $R_2$  são expressões regulares.
- 5  $R_1R_2$ , em que  $R_1$  e  $R_2$  são expressões regulares.
- 6  $R_1^*$ , em que  $R_1$  é uma expressão regular.
- 7  $R_1^+$ , em que  $R_1$  é uma expressão regular.

Por simplicidade, denotaremos os casos 1 e 2 com  $a$  e  $\varepsilon$ , respectivamente.



## Exemplos:

---

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

- $0^*10^*$ : conjunto de todas as palavras que possuem um único 1.
- $\Sigma^*1\Sigma^*$ : conjunto de todas as palavras binárias que possuem pelo menos um 1.
- $\Sigma^*001\Sigma^*$ : conjunto das palavras que possuem 001 como substring.
- $(\Sigma\Sigma)^*$  conjunto das palavras que possuem comprimento par.
- $(\Sigma\Sigma\Sigma)^*$  conjunto das palavras que possuem comprimento múltiplo de 3.
- $01|10 = \{01, 10\}$ .



## Exemplos:

---

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

- $0\Sigma^*0|1\Sigma^*1|0|1$ : conjunto de todas as palavras que iniciam e terminam com o mesmo símbolo.
- $(0|\varepsilon)(1|\varepsilon) : \{\varepsilon, 0, 1, 01\}$



# Sumário

---

## 4 Aplicações



# Aplicações

---

- Podemos projetar expressões regulares para definir classes de tokens, também chamada de lexemas, de maneira muito compacta.
- Extremamente úteis para projetar analisadores léxicos.



# Aplicações

---

## Exemplos

- Número decimais:  $D^+.D^+$ , em que  $D = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$ .
- Números inteiros:  $(+|-|\epsilon)D^+$ .
- Tipos:  $int|float|double|char|$ .
- Modificadores:  $long|short|long\ long$ .





# Sumário

---

## 5 Exercícios



# Exercícios

---

Utilizando uma linguagem de programação, crie expressões regulares para:

- Identificar um e-mail terminado com `.com` ou `.com.br`
- Hora no formato `HH:MM:SS`.
- Identificadores da linguagem C: começam com uma letra ou subscrito, e podem ser seguidos por letras, dígitos e subscritos.
- Números inteiros na linguagem C.
- Números decimais na linguagem C.
- Operadores binários da linguagem C.



# Exercícios

---

Utilizando uma linguagem de programação, crie expressões regulares para:

- Comentários multilinha da linguagem C.
- Comentários de única linha da linguagem C.
- Senha forte: pelo menos 12 caracteres com presença de: letras maiúsculas, minúsculas, dígitos e símbolos especiais.
- Endereços IPV4.
- Endereços MAC.