

# **Textos**

## **(cadeia de caracteres)**

Fundamentos de Lógica e Algoritmos

# Sumário

- Definição
- Opções do comando
  - Condição no início
  - Condição no final
- Exemplo
- Desafios

# Definição

# Cadeia de caractere

- Wikipedia [https://pt.wikipedia.org/wiki/Cadeia de caracteres](https://pt.wikipedia.org/wiki/Cadeia_de_caracteres)
- Na programação de computadores, uma cadeia de caracteres é uma sequência de caracteres, geralmente utilizada para representar palavras, frases ou textos de um programa
- Sinônimos
  - Texto, String

**Operações**

# Operações com textos

- Uso geral
- Concatenação e subtração
- Subtextos
- Conversão

# Uso geral (1)

## Potigol

```
### atribuindo texto a variável  
var minha_var := "Meu Texto"
```

```
### tamanho do texto  
escreva minha_var.tamanho
```

```
### maiúsculas e minúsculas  
escreva minha_var.maiúsculo  
escreva minha_var.minúsculo
```

```
### verificando existência  
escreva minha_var.posição('e')  
escreva minha_var.contém('a')
```

## Python

```
### atribuindo texto a variável  
minha_var = "Meu Texto"
```

```
### tamanho do texto  
print( len(minha_var) )
```

```
### maiúsculas e minúsculas  
print( minha_var.upper() )  
print( minha_var.lower() )
```

```
### verificando existência  
print( minha_var.find('e') )  
print(minha_var.find('a') >= 0)
```

# Uso geral (2)

## Potigol

```
### "cbA"  
"Abc".inverta
```

```
### "abc"  
"cab".ordene
```

```
### "a-b-c"  
"abc".junte("-")
```

```
# "[a, b, c]"  
"abc".junte("[", ", ", "]")
```

## Python

```
### "cbA"  
"Abc"[::-1]
```

```
### "abc"  
sorted("cab")
```

```
### "a-b-c"
```

```
# "[a, b, c]"
```

# Concatenação e subtração

Potigol

Python

```
### "abcd" (concatenação)  
escreva "ab" + "cd"
```

```
### "acb" (subtração)  
escreva "abcb" - "bd"
```

```
### "abcd" (concatenação)  
print( "ab" + "cd" )
```

```
### "ac" (subtração)  
print( "abcb".replace("b", "") )
```

# Sub textos (1)

## Potigol

```
### 'b'  
escreva "abc"[2]  
### 'a'  
escreva "abc".cabeça  
### "bc"  
escreva "abc".cauda  
### 'c'  
escreva "abc".último  
### "abc"  
escreva "abcde".pegue(3)  
### "de"  
escreva "abcde".descarte(3)
```

## Python

```
### 'b'  
print( "abc"[1] )  
### 'a'  
print( "abc"[0] )  
### "bc"  
print( "abc"[1:] )  
### 'c'  
print( "abc"[-1] )  
### "abc"  
print( "abcde"[0:3] )  
### "de"  
print( "abcde"[3:] )
```

# Sub textos (2)

## Potigol

```
### x = "ac"  
x = "abc".remova(2)  
  
### y = "abdc"  
y = "abc".insira(3, 'd')  
  
### z = "abdefc"  
z = "abc".insira(3, "def")
```

## Python

```
t = "abc"  
  
### x = "ac"  
  
### y = "abdc"  
y = t[:2] + "d" + t[2:]  
  
### z = "abdefc"  
z = t[:2] + "def" + t[2:]
```

# Conversão

## Potigol

```
### 123
escreva "123".inteiro

### 123.4
escreva "123.4".real

### ["1", "2", "3.0"]
escreva "1 2 3.0".divida(" ")

### "123"
escreva 123.texto
escreva 85.caractere
```

## Python

```
### 123
int( "123" )

### 123.4
float( "123.4" )

### ["1", "2", "3.0"]
"1 2 3.0".split(" ")

### "123"
str( 123 )
```

**Exemplos**

# URI Online Judge

## 1238 Combinador

```
1  numero_de_textos = leia_inteiro
2
3  para i de 1 até numero_de_textos faça
4      palavras = leia_texto.divida(" ")
5      primeira_palavra = palavras[1]
6      segunda_palavra = palavras[2]
7      var combinador := ""
8      var tamanho := segunda_palavra.tamanho
9      se (primeira_palavra.tamanho < segunda_palavra.tamanho) então
10         tamanho := primeira_palavra.tamanho
11     fim
12     para x de 1 até tamanho faça
13         combinador := combinador + primeira_palavra[x] + segunda_palavra[x]
14     fim
15     se (primeira_palavra.tamanho < segunda_palavra.tamanho) então
16         combinador := combinador + segunda_palavra.descarte(primeira_palavra.tamanho)
17     senãose (primeira_palavra.tamanho > segunda_palavra.tamanho) então
18         combinador := combinador + primeira_palavra.descarte(segunda_palavra.tamanho)
19     fim
20     escreva combinador
21 fim
```

# 1º Exemplo

# Saída

**Desafios**

**Qual a saída?**

# Links

- <http://potigol.github.io>
- <https://wiki.python.org.br/>
- [https://pt.wikipedia.org/wiki/Cadeia de caracteres](https://pt.wikipedia.org/wiki/Cadeia_de_caracteres)