

ALGORITMOS E PROGRAMAÇÃO DE COMPUTADORES I

Strings

STRINGS

Além de variáveis do tipo numérico (int e float), e do tipo lógico (bool), Python permite definir variáveis de tipos mais complexos.

O tipo str é usado para representar e manipular texto ou uma sequência de caracteres, incluindo espaço em branco, pontuação e símbolos diferentes.

STRINGS

Uma string é criada como uma sequência de caracteres envolvida por aspas (simples ou duplas):

```
>>> a = 'Algoritmos e Programação'
```

```
>>> a
```

```
'Algoritmos e Programação'
```

OPERADORES

É possível usar alguns operadores já vistos com strings:

```
>>> s = 'abc'
```

```
>>> s == 'abc'
```

```
True
```

```
>>> t = 'def'
```

```
>>> s < t
```

```
True
```

```
>>> s + t
```

```
'abcdef'
```

```
>>> s * t
```

```
→ ERRO
```

```
>>> s * 2
```

```
'abcabc'
```

```
>>> ch = 'b'
```

```
>>> ch in s
```

```
True
```

```
>>> ch is str('b')
```

```
True
```

OPERADORES

Operadores comuns que não funcionam com strings: -, /, //, %, **

Outros operadores importantes:

len() → retorna o tamanho da string

[] → operador de indexação

EXEMPLOS

```
>>> s = 'abcd'
```

```
>>> len(s)
```

```
4
```

```
>>> s[0]
```

```
'a'
```

```
>>> s[3]
```

```
'd'
```

```
>>> s[len(s)]
```

```
ERRO!
```

INDEXAÇÃO

Apesar do índice variar de 0 até o tamanho da string-1, é possível usar índices negativos

Índices negativos acessam os caracteres do lado direito da string

Exemplos:

```
>>> s = 'abcd'
```

```
>>> s[-1]
```

```
'd'
```

```
>>> s[-2]
```

```
'c'
```

```
>>> s[-5]
```

```
ERRO!
```

INDEXAÇÃO

É possível também acessar substrings por meio da indexação:

```
>>> s = 'abcd'
```

```
>>> s[0:2]
```

```
'ab'
```

```
>>> s[-4:-2]
```

```
'ab'
```

```
>>> s[:3]
```

```
'abc'
```

```
>>> s[-1:]
```

```
'd'
```


MÉTODOS

Python oferece alguns métodos úteis para manipulação de strings:

s.find(p) → retorna o índice em que a substring p aparece em s

s.count(p) → retorna a frequência em que a substring p aparece em s

s.replace(p, q) → substitui a substring p pela substring q em s

s.capitalize() → substitui primeiro caractere de s em maiúscula

s.upper() → substitui todos os caracteres de s em maiúscula

s.lower() → substitui todos os caracteres de s em minúscula

s.strip() → remove espaços em branco em excesso

ALGORITMOS E PROGRAMAÇÃO DE COMPUTADORES I

Strings