

ALGORITMOS E PROGRAMAÇÃO DE COMPUTADORES I

Tipos de dados

TIPOS DE DADOS

Até o momento, vimos os seguintes tipos de dados suportados pela linguagem Python: int, float, bool, str, list e tuple.

Cada dado é armazenado na memória na forma de objetos.

Todo objeto possui um tipo e um valor.



Fonte: Perkovic, 2015

TIPOS DE DADOS

O tipo de um objeto indica quais valores ele pode armazenar e quais operações podem ser executadas.

A função `type()` pode ser usada para verificar o tipo de um objeto:

```
>>> type(3)
```

```
<class 'int'>
```

```
>>> type('Olá')
```

```
<class 'str'>
```

```
>>> type([1, 2, 3])
```

```
<class 'list'>
```

Importante lembrar que variáveis em Python não possuem um tipo! Elas simplesmente apontam temporariamente para um objeto na memória.

VALORES PARA TIPOS

Dado o tipo do objeto, há um conjunto de valores que esse objeto pode armazenar.

Inteiros podem conter valores arbitrariamente altos (depende da disponibilidade de memória do computador):

```
>>> 2**1024
```

```
11781361728633673532 ... 53056587776
```

Já o tipo float tem um limite atual de 64 bits:

```
>>> 2.0**1024
```

❑ ERRO

VALORES PARA TIPOS

No caso de float's, os valores também são aproximados:

```
>>> 2**100
```

```
1267650600228229401496703205376
```

```
>>> 2.0**100
```

```
1.2676506002282294e+30
```

```
>>> 2.0**-1075
```

```
0.0
```

CRIANDO OBJETOS

Para criar um objeto int com valor 3, fazemos:

```
>>> x = 3
```

Ou então:

```
>>> x = int(3)
```

A função `int()` é chamada de construtor, e é usada para instanciar um objeto `int` explicitamente.

O número entre parênteses é o valor que será associado ao objeto.

CRIANDO OBJETOS

Outros construtores: `float()`, `str()` e `list()`.

Valores padrão para `int`, `float`, `str` e `list`: `0`, `0.0`, `"` e `[]`.

Exemplos:

```
>>> x = int()
```

```
>>> x
```

```
0
```

```
>>> y = str()
```

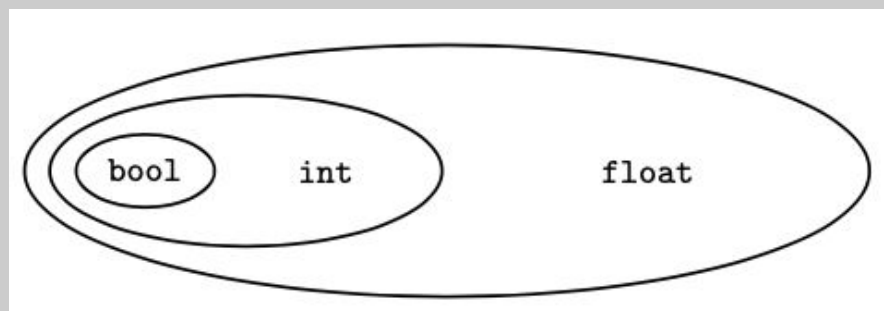
```
>>> y
```

```
"
```

CONVERSÃO IMPLÍCITA

Se em uma expressão há operandos de diferentes tipos, Python irá converter cada operando para o tipo que contém os outros

```
>>> True + 5      >>> 3 + 0.35  
6                 3.35
```



Fonte: (Perkovic, 2015)

CONVERSÃO EXPLÍCITA

Podemos usar os construtores para fazer a conversão explícita entre tipos:

```
>>> int(3.4)      >>> str(2.74)
```

```
3
```

```
'2.74'
```

```
>>> float(4)      >>> int('1.5')
```

```
4.0
```

```
→ ERRO
```

```
>>> float('5.6')  >>> bool(0)
```

```
5.6
```

```
False
```

ALGORITMOS E PROGRAMAÇÃO DE COMPUTADORES I

Tipos de dados