

Funções predefinidas e Bibliotecas

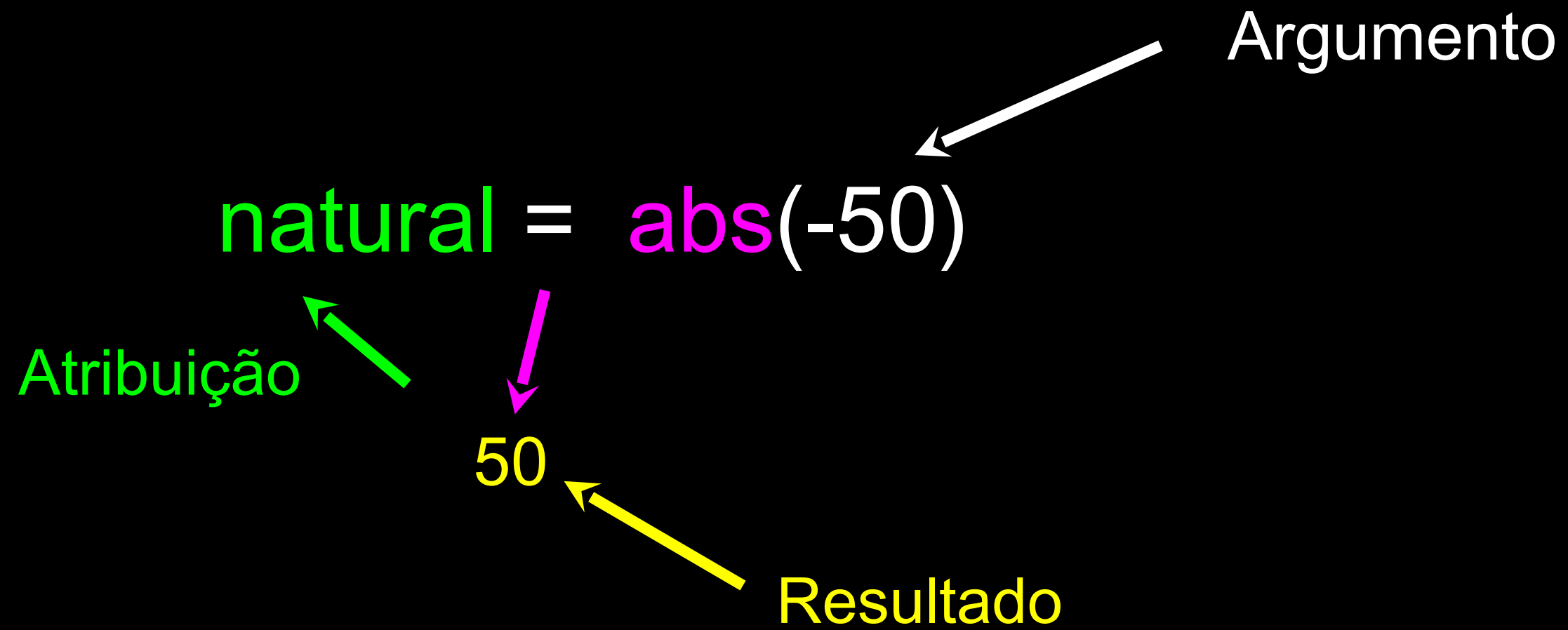
Prof. Alberto Costa Neto
Programação em Python

Definição de Funções

- Em Python, assim como em outras linguagens de programação, uma **função** é um código fonte reusável que recebe **argumento(s)** como entrada, computa algo, e então retorna um resultado ou resultados

Funções em Python

- Há 2 tipos de **funções** em Python.
 - **Funções Built-in** que são providas como parte da linguagem Python - **input()**, **type()**, **float()**, **int()** ...
 - > **Funções** que nós **definimos** e então as utilizamos
- Tratamos os nomes das **funções** built-in como “novas” **palavras reservadas** (ou seja, evitamos usá-los como nomes de variáveis)

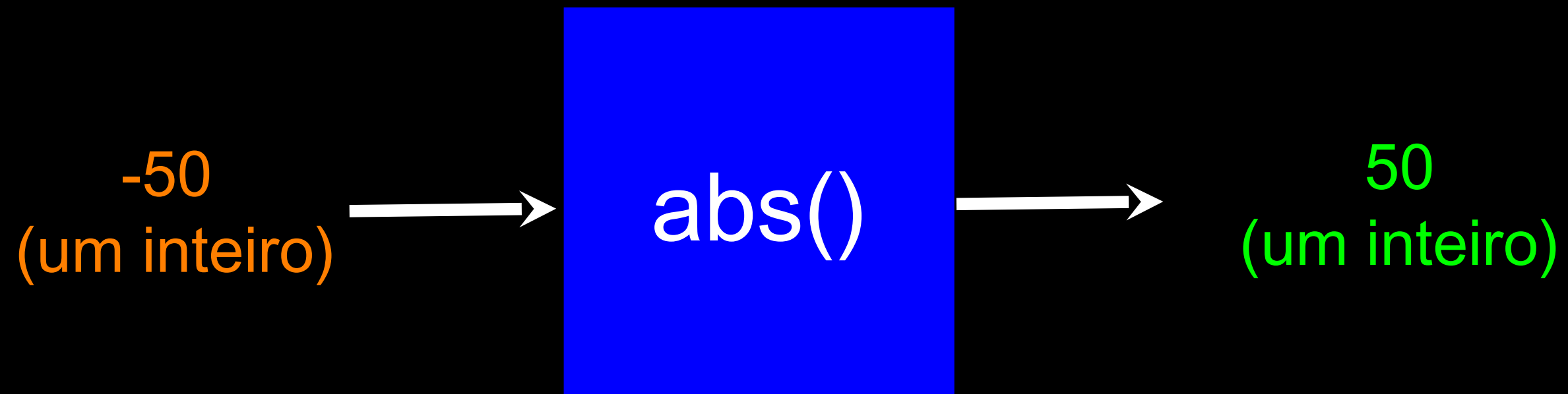


```
>>> natural = abs(-50)
>>> print(natural)
50
```

Função abs

```
>>> natural = abs(-50)
>>> print(natural)
50
```

Uma função é um código fonte armazenado que podemos usar. Uma função recebe uma entrada e produz uma saída.



Função abs

```
>>> natural = abs(-50)
>>> print(natural)
50
```

Uma função é um código fonte armazenado que podemos usar. Uma função recebe uma entrada e produz uma saída.

-50
(um inteiro)



```
def abs(v):
    if v < 0:
        return -v
    else:
        return v
```



50
(um inteiro)

Bibliotecas de funções em Python

- Há um número crescente de bibliotecas disponíveis para **várias finalidades**
- Para usar uma biblioteca, é preciso usar o comando **import** seguido do nome da biblioteca
- Por exemplo, para usar a constante **pi**, é preciso **importar a biblioteca math**
- Junto com a constante pi, há várias funções matemáticas

Biblioteca math

- sin, cos, tan, asin, acos, atan
- log, log2, log10
- pow, sqrt
- floor, ceil

```
>>> import math
>>> x = math.pow(3,4)
>>> print(x, math.sqrt(x))
81 9.0
>>> print(math.floor(2.8))
2
>>> print(math.ceil(2.8))
3
```

- Para saber mais sobre a biblioteca math, acesse:
docs.python.org/3.4/library/math.html

Outras bibliotecas

- Python fornece um conjunto de bibliotecas para vários domínios de aplicação
- Para ver uma lista mais completa, acesse:
 - <https://docs.python.org/3.4/library/>

Usar funções é muito bom

- Organiza o código fonte em “parágrafos” - capture um raciocínio completo e escolha um bom nome para a função
- DRY - *Don't repeat yourself* – Faça apenas uma vez e reuse
- Se algo ficou muito grande e complexo, quebre em pedaços lógicos e coloque estes pedaços em funções
- Crie uma biblioteca (*library*) de coisas comuns que você faz repetidamente – talvez compartilhar com seus amigos...