

Funções aninhadas e vários argumentos - Correção

NUMWORKS

Esta ficha foi redigida por **Claire Savinas**, professora da Escola Jean Vilar em Villeneuve-Lès-Avignon, França. A Claire é formadora em Python.

1 Objetivos da ficha


- programar utilizando uma função com vários argumentos
- criar uma função que utilize outra função
- descobrir a utilidade de uma função

2 Ficha de atividades para os alunos

3 Descobrir uma função com dois argumentos

(a) Vamos começar por criar um novo script chamado `atividade2.py`. Aqui está uma função chamada `maximo` que toma dois números como entrada e devolve o máximo dos dois valores.

```
gra PYTHON
1 from math import *
2 def maximo(a,b):
3     if a>b:
4         return a
5     else:
6         return b|
7
8
9
10
11
12
```

Insira a função no guião `atividade2.py` e de seguida seleccione **Executar o script** no menu de três pontos à direita do script. A função criada está disponível no menu da tecla . Experimente usar diferentes valores nesta função, por exemplo `maximo(12,15)`, `maximo(6,-8)`, ...

(b) No mesmo script adicione uma função chamada `minimo` que toma dois números como entrada e devolve o mínimo desses dois valores.

4 Descobrir uma função com mais de dois argumentos

(a) No mesmo script adicione uma função chamada `maximo3` que toma três números como entrada e devolve o máximo desses três valores.

(b) No mesmo script adicione uma função chamada `maximo4` que toma quatro números como entrada e devolve o máximo desses quatro valores.

Dica: Pode usar a função `maximo` e utilizar um esquema

(c) No mesmo script adicione uma função chamada `maximo8` que toma oito números como entrada e devolve o máximo dos oito valores, utilizando a função `maximo4`.

Dica: faça um esquema com o resultado de cada uma das funções

5 Correção da ficha

5.1 Descobrir uma função com dois argumentos

```
gra PYTHON
1 from math import *
2
3 def maximo(a,b):
4     if a>b:
5         m=a
6     else:
7         m=b
8     return m
9
10
11
12
```

(a) Poderíamos ter proposto o seguinte script que utiliza uma variável adicional.

(b) Aqui estão duas versões de correção.

```
gra PYTHON
1 from math import *
2 def minimo(a,b):
3     if a<b:
4         return a
5     else:
6         return b
7
8
9
10
11
12
```

Uma primeira versão que não utiliza uma variável suplementar.

```

gra PYTHON
1 from math import *
2 def minimo(a,b):
3     if a<b:
4         m=a
5     else:
6         m=b
7     return m
8
9
10
11
12

```

E uma segunda versão que utiliza a variável suplementar m.

5.2 Descobrir uma função com vários argumentos

(a) Há inúmeras possibilidades sem utilizar a função `maximo`, mas são longas e complexas.

```

gra PYTHON
1 from math import *
2 def maximo(a,b):
3     if a>b:
4         return a
5     else:
6         return b
7
8 def maximo3(a,b,c):
9     m=maximo(a,b)
10    return maximo(m,c)
11
12

```

Utilizando a função `maximo`, a nova função é relativamente simples.

```

gra PYTHON
1 from math import *
2 def maximo(a,b):
3     if a>b:
4         return a
5     else:
6         return b
7
8 def maximo3(a,b,c):
9     return maximo(maximo(a,b),c)
10
11
12

```

Aqui está uma outra resolução sem utilizar uma variável suplementar.

(b) Da mesma forma, reutilizamos a função `maximo`.

```
gra PYTHON
1 from math import *
2 def maximo(a,b):
3     if a>b:
4         return a
5     else:
6         return b
7
8 def maximo4(a,b,c,d):
9     m1=maximo(a,b)
10    m2=maximo(c,d)
11    return maximo(m1,m2)
12
13
```

Esta primeira versão utiliza duas variáveis auxiliares: `m1` e `m2`.

```
gra PYTHON
1 from math import *
2 def maximo(a,b):
3     if a>b:
4         return a
5     else:
6         return b
7
8 def maximo4(a,b,c,d):
9     return maximo(maximo(a,b),
10    maximo(c,d))
11
12
13
```

Esta segunda versão, no entanto, não utiliza variáveis complementares.

(c) Aqui reutilizamos a função `maximo4` que por sua vez recorre à função `maximo`.

```
gra PYTHON
1 from math import *
2 def maximo(a,b):
3     if a>b:
4         return a
5     else:
6         return b
7
8 def maximo4(a,b,c,d):
9     return maximo(maximo(a,b),
10    maximo(c,d))
11
12 def maximo8(a,b,c,d,e,f,g,h):
13    return maximo(
14    maximo4(a,b,c,d),
15    maximo4(e,f,g,h))

```

Uma versão que não utiliza uma variável suplementar.

```
gra PYTHON
5
6     return b
7
8 def maximo4(a,b,c,d):
9     m1=maximo(a,b)
10    m2=maximo(c,d)
11    return maximo(m1,m2)
12
13 def maximo8(a,b,c,d,e,f,g,h):
14     m1=maximo4(a,b,c,d)
15     m2=maximo4(e,f,g,h)
16     return maximo(m1,m2)
17
```

Uma outra versão com duas variáveis suplementares.