

# python function

贵州大学

*hnzhang1@gzu.edu.cn*

March 14, 2019

# Overview

introduce of function

define a function

module

homework

Q&A

# introduce of function

函数（方法）具有以下特点：

- 是一段代码
- 能完成特定功能
- 可以在其他地方被调用
- 可以接收参数或不接收参数
- 可以有返回值或没有返回值

函数用于将一个复杂的问题分解为若干个简单的子问题。

# 函数的类别

内建函数 int()、chr()、ord()、round() 等

自定义函数

## 函数的定义

```
1 def functionName(parameter1, parameter2, ...):  
2     functionBlock  
3     return aValue
```

第 1 行称为函数签名，用于指定函数名称以及函数的每个形式参数变量名，注意要有结尾的英文冒号。

`def` 定义函数的关键字

`functionName` 函数名

`parameter1...` 参数<sup>1</sup>

`functionBlock` 函数体

`return` 表明此函数是有返回值的<sup>2</sup>

`aValue` 函数的返回值

---

<sup>1</sup>函数可以不需要参数

<sup>2</sup>函数可以不返回任何值

## 函数的调用

```
1 import math
2
3 def myGCD(a,b):
4     c, d = a, b
5     while a % b != 0:
6         a, b = b, a % b
7     print(c, ' 和 ', d, ' 的最大公约数是:', b)
8
9 a = int(input("a:"))
10 b = int(input("b:"))
11 myGCD(a,b)
12 print(math.gcd(a, b))
```

一系列的  
import

一系列的  
函数定义

任意数量的  
全局代码

程序的主体

## 默认 (可选) 参数

可以给一个  
参数指定默  
认值 (在函  
数签名中为  
参数赋值),  
这样的参数  
也叫可选参  
数。可选参  
数可以有多个, 但所有  
的可选参数  
都要定义在  
必选参数之  
后。

```
def is_prime(first_n=10):  
    prime_list = [2]  
    i = 3  
    while True:  
        for d in range(2, i):  
            if i % d == 0:  
                break  
            elif (i % d != 0) and (d == (i-1)):  
                prime_list.append(i)  
        if len(prime_list) == first_n:  
            break  
        i += 1  
    return prime_list  
  
print(is_prime())  
print(is_prime(5))
```

## 多态性

```
1 import math
2
3 def my_power(a, b):
4     return a**b
5
6 n1 = my_power(2,5)
7 n2 = my_power(math.pi, 1)
8 print(n1, n2)
9
10
11 32 3.141592653589793
```

从 `my_power()` 这个函数的返回值可以看出，传递 `int` 值的时候，返回的也是 `int` 值；如果传递的是浮点类型的值，返回值就是浮点类型的。

*python* 的这个特性体现了其灵活性，即多态性。多态性允许定义一个函数，供不同类型的对象使用。

但是要注意传递恰当类型的对象，如果向 `my_power()` 函数传递字符串类型的对象的话，程序会产生 `TypeError` 错误。



## 返回多个值

```
1 import math
2
3 def move(x, y, distance, angle = 0):
4     new_x = x + distance * math.cos(angle)
5     new_y = y + distance * math.sin(angle)
6     return new_x, new_y
7
8 x, y = move(0, 0, 1.414, math.pi/4)
9 print(x, y)
10 z = move(0, 0, 2, math.pi/6)
11 print(z)
12
13 0.9998489885977783 0.999848988597778
14 (1.7320508075688774, 0.9999999999999999)
```

第 8 行形式上返回了两个值，通过第 10 可以看出其实是一个元组。

## 列表解析

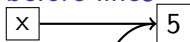
当想要对一个序列中的每个元素都执行某个特定的操作时，通常可以使用 *for* 循环来实现。在 *python* 中，可以使用列表解析。

```
1 import math
2
3 l = [2, 3, 4]
4 m = [pow(x, 2) for x in l]
5 n = [math.sqrt(y) for y in m]
6 print(m)
7 print(n)
8
9 [4, 9, 16]
10 [2.0, 3.0, 4.0]
```

## 向函数传递不可变对象<sup>3</sup>

```
1 def double(l):
2     print('before:: l=', l, 'id(l)=', id(l))
3     l = l * 2
4     print('after:: l=', l, 'id(l)=', id(l))
5
6 x = 5
7 print('x=', x, 'id(x)=:', id(x))
8 double(x)
9 print('x=', x)
10
11 x= 5 id(x)=: 4385389696
12 before:: l= 5 id(l)= 4385389696
13 after:: l= 10 id(l)= 4385389856
14 x= 5
```

before line3



after line3

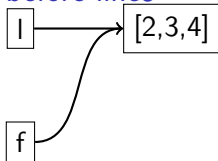


<sup>3</sup>[https://www.python-course.eu/passing\\_arguments.php](https://www.python-course.eu/passing_arguments.php)

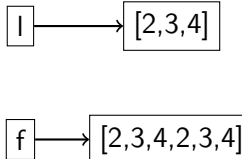
## 向函数传递可变对象 /

```
1 def double(f):  
2     print('before:: f=',f,'id(f)=',id(f))  
3     f = f * 2  
4     print('after:: f=',f,'id(f)=',id(f))  
5  
6 l = [2, 3, 4]  
7 print('l=',l,'id(l)=:', id(l))  
8 double(l)  
9 print('l:', l)  
10  
11 l= [2, 3, 4] id(l)=: 4386889608  
12 before:: f= [2, 3, 4] id(f)= 4386889608  
13 after:: f= [2, 3, 4, 2, 3, 4]  
14 id(f)= 4388025352  
15 l: [2, 3, 4]
```

before line3



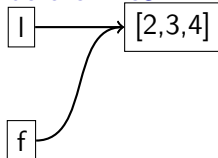
after line3



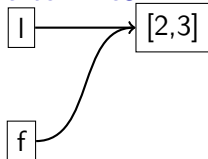
## 向函数传递可变对象 <sup>114</sup>

```
1 def double_2(f):
2     print('before:: f=',f,'id(f)=',id(f))
3     del f[-1]
4     print('after:: f=',f,'id(f)=',id(f))
5
6 l = [2, 3, 4]
7 print('l=',l,'id(l)=',id(l))
8 double_2(l)
9 print('l:', l)
10
11 l= [2, 3, 4] id(l)= 4377313160
12 before:: f= [2, 3, 4] id(f)= 4377313160
13 after:: f= [2, 3] id(f)= 4377313160
14 l: [2, 3]
```

before line3



after line3

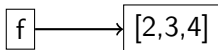
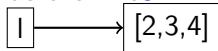


<sup>4</sup>Side effects: A function is said to have a side effect if, in addition to producing a value, it modifies the caller's environment in other ways.

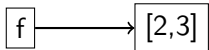
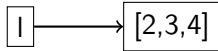
## 如何避免函数改变主程序中变量的值

```
1 def double_2(f):
2     print('before:: f=',f,'id(f)=',id(f))
3     del f[-1]
4     print('after:: f=',f,'id(f)=',id(f))
5
6 l = [2, 3, 4]
7 print('l=',l,'id(l)=',id(l))
8 double_2(l[:])
9 print('l:', l)
10
11 l= [2, 3, 4] id(l)= 4537892744
12 before:: f= [2, 3, 4] id(f)= 4539028552
13 after:: f= [2, 3] id(f)= 4539028552
14 l: [2, 3, 4]
```

before line3



after line3



introduce of function  
○○

define a function  
○○○○○○○○○○●

module  
○

homework  
○

Q&A  
○

# lamda

introduce of function  
○○

define a function  
○○○○○○○○○○○○

module  
●

homework  
○

Q&A  
○

# 模块



# Homework

1. 求出前 10 个素数<sup>5</sup>。
2. 使用辗转相除法<sup>6</sup>求任意两个数的最大公约数。

---

<sup>5</sup>质数 (prime number) 又称素数，有无限个。质数定义为在大于 1 的自然数中，除了 1 和它本身以外不再有其他因数。

<sup>6</sup>用较大数除以较小数，再用出现的余数（第一余数）去除除数，再用出现的余数（第二余数）去除第一余数，如此反复，直到最后余数是 0 为止。如果是求两个数的最大公约数，那么最后的除数就是这两个数的最大公约数。

# Q&A