

CS61A学习笔记

水一寿

2024 年 1 月 20 日

1 Function

1.1 ch.1 expressions

在powershell环境中给出了：

- pi 从 math中import: `from math import pi, sin`
- 第一个Python自带的函数: `max()`
- 自定义函数: `def f(x):`
eg. `return mul(x,x)`

实现参数与参数的相互链接：通过函数实现。

在第一节的最最后给出了一系列函数之间的赋值和计算的一个题目。

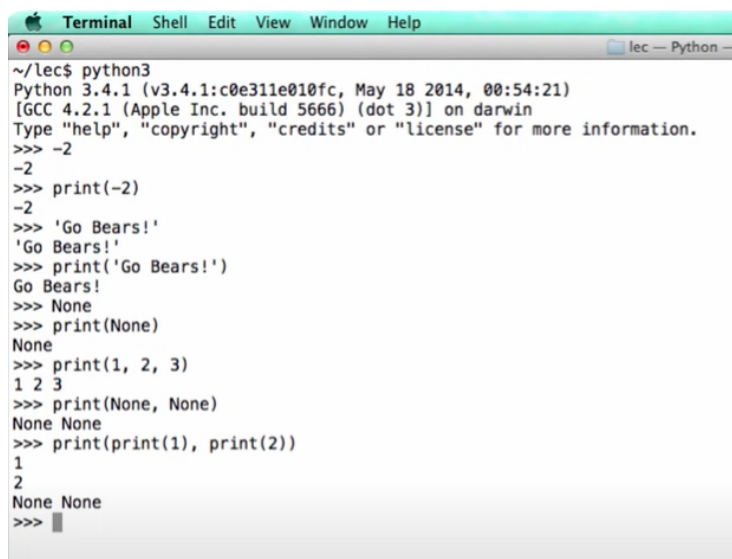
1.2 ch.2 environment diagrams

程序员常常在各种盒子之间画箭头：想想肖老师的灵魂画手。

code on the left and frames on the right

the course of execution.

在赋值语言中，python的赋值规则是：先计算右边的表达式，表达式从左算到右，再按照顺序赋值到左边。



```

~/lec$ python3
Python 3.4.1 (v3.4.1:c0e311e010fc, May 18 2014, 00:54:21)
[GCC 4.2.1 (Apple Inc. build 5666) (dot 3)] on darwin
Type "help", "copyright", "credits" or "license" for more information.
>>> -2
-2
>>> print(-2)
-2
>>> 'Go Bears!'
'Go Bears!'
>>> print('Go Bears!')
Go Bears!
>>> None
>>> print(None)
None
>>> print(1, 2, 3)
1 2 3
>>> print(None, None)
None None
>>> print(print(1), print(2))
1
2
None None
>>>

```

图 1: 教学示例

1.3 ch.3 defining function

前面提到的def

signature:

Listing 1: 函数定义的一般形式

```
def <name>(<formal parameters>)
    return <return expression>
```

函数内部的实际运算发生在函数被调用的时候。当你调用一个函数时，*Python*会执行函数体中的代码，进行相应的运算。在函数被调用之前，函数体内的代码不会执行。

调用的时候直接用:f() **built-in function VS User-defined function**

1.4 ch.4 print and none

解释：先进行括号里面的print(1)和print(2),得到1和2，然后最外面的大括号内进行运算。 print函数的返回值是none，所以返回两个None。

None代表了nothing 如果一个函数并不明显地返回什么值，那么它返回None。

函数的分类：

pure functions(有返回值的函数): eg:abs(-2) pow(2,100)

Non-pure functions(无返回值的函数): print()函数

疑难：

python中如何理解”environment is a sequence of frames”

”Environment is a sequence of frames” 这个说法是指在Python中，执行代码时，变量和函数等符号的查找是通过一个被称为环境（environment）的数据结构来实现的。这个环境可以被视为一系列帧（frames）的序列。

在这个上下文中，一个“frame”通常对应于一个作用域，比如函数、模块或者类。每个frame都包含了在该作用域内定义的变量、函数等信息。而整个环境则是这些帧的序列。

当你在代码中引用一个变量或者调用一个函数时，Python解释器会在当前环境的帧序列中查找相应的符号。它会从当前帧开始查找，如果找不到，就会顺着帧序列往外层查找，直到找到为止。

举个简单的例子，考虑下面的代码：

Listing 2: 示例代码

```
python
x = 10

def foo():
    y = 5
    print(x + y)

foo()
```

在这个例子中，有两个帧：全局作用域的帧包含变量‘x’，而‘foo’函数的帧包含变量‘y’。当‘foo’函数中执行‘print(x + y)’时，解释器首先在当前帧（‘foo’函数的帧）中查找变量‘x’和‘y’，如果找不到，它会继续在外层的帧（全局作用域的帧）中查找。在这种方式下，环境的帧序列形成了一个符号查找的链。

总的来说，”environment is a sequence of frames” 可以理解为Python中符号查找的机制，其中变量、函数等的作用域由帧构成，而环境则是这些帧的有序序列。