

python

控制结构

贵州大学

hnzhang1@gzu.edu.cn

March 19, 2019

Overview

条件

控制结构包括：判断和循环。为了作出判断或控制循环，必须指定条件来做依据。

条件（也叫布尔表达式）是一种包含了关系运算符（>、>=、==、in、not in 等）和（或）逻辑运算符（*and*、*or*、*not*）的表达式。

条件的结果为：True 或 False。

ASCII

在应用关系运算符进行比较字符时的依据是这些字符的 ASCII¹值。

```
1 ord(c, /)
2     Return the ASCII value for a one-character string.
3 chr(i, /)
4     Return a string of the ASCII, ...
5 >>> print("ord('A')",ord('A'),"; ord('Z')",ord('Z'),\
6 ... "| ord('a')",ord('a'),"; ord('z')",ord('z'))
7 ord('A') 65 ; ord('Z') 90 | ord('a') 97 ; ord('z') 122
8 >>> print("chr(48)",chr(48),"; chr(57)",chr(57),\
9 ... "| chr(169)",chr(169),"; chr(248)",chr(248))
10 chr(48) 0 ; chr(57) 9 | chr(169) © ; chr(248) ø
```

¹ASCII (American Standard Code for Information Interchange, 美国信息交换标准代码) 是基于拉丁字母的一套电脑编码系统。

关系运算符小练习

判断下列条件的结果。

```
1 1<=1
2 'car'<'cat'
3 'Dog'<'dog'
4 'fun' in 'refunded'
5 'B' not in ('a','b','c')
6 let a=4,b=3,c='hello':
7     (len(c)-a)==(b/2)
```

逻辑运算符

当需要更复杂的条件时，就需要使用逻辑运算符，使用了这些运算符的条件称为复合条件。

判断方法

$a \text{ and } b$ a, b 均为真时，结果为真

$a \text{ or } b$ a, b 至少一个为真时，结果为真

$\text{not } a$ a 为假时，结果为真

短路求值

$a \text{ and } b$ 当 a 为假时，还有必要判断 b 吗？

$a \text{ or } b$ 当 a 为真时，还有必要判断 b 吗？

逻辑运算符小练习

```
1 let n = 4:  
2     (2 < n) or (n < 0)  
3     (n == 4) and (n > 5)  
4     (2 < n) and (n < 0)  
5     not (n > 2)
```

判断结构

判断结构（也称为分支结构）允许程序根据特定条件的真假来决定执行哪些语句。

if 的形式²

if

```
if condition:
    code...
```

if-else

```
if condition:
    code...
else:
    code...
```

if-elif

```
if condition:
    code...
elif condition:
    code...
```

if-elif-else

```
if condition:
    code...
elif condition:
    code...
else:
    code...
```

²if 可以嵌套，此处没有展示。

if 示例代码

```
a,b = 2,3
```

if

```
if a<b:  
    print("a<b")
```

if-else

```
if a<b:  
    print("a<b")  
else:  
    print("a>=b")
```

if-elif-else

```
if a<b:  
    print("a<b")  
elif a>b:  
    print("a>b")  
else:  
    print("a=b")
```

if 练习题

从终端输入两个字符，判断其是否为数值。

Tips

```
1 >>> help(str.isdigit)
2 isdigit(self, /)
3     Return True if the string is a digit string,
4     False otherwise.
```

if 练习题答案³

Q: 从终端输入两个字符，判断其是否为数值。

```
1 a=input("input 1st:")
2 b=input("input 2nd:")
3 if a.isdigit() and b.isdigit():
4     print("Both of them is digit.")
5 elif a.isdigit():
6     print("b is not a digit.")
7 elif b.isdigit():
8     print("a is not a digit.")
9 else:
10    print("Both of them is not digit.")
```

³这里用到了 `str` 模块中的方法，为何没有在程序起始位置 `import str`?

循环结构

循环也是程序设计中最重要结构之一，是程序中可以重复执行的一段代码。有两种循环形式：

- while
- for

while

语法

```
1 while condition:  
2     code...
```

第 1 行代码称为循环头，*code...*部分称为循环体。
当碰到 *while* 代码时，首先判断 *condition* 是否成立，若成立，则执行循环体，每执行完一次循环体后都会再次判断 *condition* 是否成立。

while 小练习

使用 while 循环，求出式子

$$2 + 4 + 6 + \dots + 100$$

的值。

for

特点

python 中的 *for* 循环用来迭代 (遍历) 一系列的值, 比如等差数列、字符串、列表、元组等序列中的元素。

语法

```
1 for var in sequence:  
2     code...
```

使用 *for* 遍历—字符串、列表、元组

字符串

```
s = 'gzu'
for e in s:
    print(e)
```

g
z
u

列表

```
mList = ['g','u',5]
for e in mList:
    print(e)
```

g
u
5

元组

```
mTuple = ('g','z',5)
for e in mTuple:
    print(e)
```

g
z
5

使用 *for* 遍历一等差数列

与 *c*、*java* 等语言不同，*python* 里的 *for* 循环，在处理类似下列代码所示情形时，需要通过遍历一个等差数列来实现。

```
1  int i, sum=0;
2  for(i=0; i<100; i++){
3      sum += i;
4  }
```

如何产生等差数列

在 *python* 中, 可以使用 *range()* 函数⁴来产生等差数列, 从而供 *for* 循环使用, 控制循环体。

```
1 >>> help(range)
2 Help on class range in module builtins:
3 class range(object)
4     range(stop) -> range object
5     range(start, stop[, step]) -> range object
6
7 Return an object that produces a sequence of integers
8 from start (inclusive) to stop (exclusive) by step.
9 Start defaults to 0, and stop is omitted!
10 range(4) produces 0, 1, 2, 3.
11 When step is given,
12 it specifies the increment (or decrement).
```

⁴*range* 类的构造方法

使用 *for* 遍历一等差数列 *range()*

range(stop)

```
l = range(3)
for e in l:
    print(e)
0
1
2
```

range(start, stop)

```
ll = range(2,5)
for e in ll:
    print(e)
2
3
4
```

range(start, stop, step)

```
lll = range(2,10,5)
for e in lll:
    print(e)
2
7
```

for 小练习

使用 for 循环，求出式子

$$1 + 3 + 5 + \dots + 99$$

的值。

break、*continue*⁵

break

在循环体中使用 *break* 关键字，会使程序**停止执行循环体**，并跳转到循环体外，执行后续代码。

continue

在循环体中使用 *continue* 关键字，**会终止本轮循环**，程序跳转到循环的头部。

⁵通常出现在 if 语句里面。

无限循环

某些特殊的情况下，会使用到无限循环。在使用无限循环的时候，要注意在适当的情况下，结束循环。

```
1 i = 0
2 while True:
3     i += 1
4     if i > 100:
5         print("termination.")
6         break
```

Homework

1. 求出前 10 个素数⁶。
2. 使用辗转相除法⁷求任意两个正整数的最大公约数。

⁶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。

⁷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