

# PYTHON程序设计

张树兵

20230307改

# 和董传留别

## 苏轼

粗缯大布裹生涯，  
腹有诗书气自华。  
厌伴老儒烹瓠叶，  
强随举子踏槐花。  
囊空不办寻春马，  
眼乱行看择婿车。  
得意犹堪夸世俗，  
诏黄新湿字如鸦。

## 注释：

- **粗缯**，拼音是cū zēng，意思为粗制的丝织品。
- **烹瓠叶**pēng hù yè，瓠子。一年生攀缘草本植物。葫芦的变种

While循环+穷举法+一题多解

# While循环 概念 与for循环比较

- 适用：不知道循环次数的情况

- 例如：

- 如果我国现有人口13亿，按人口增长率0.8%计算，问多少年后，我国人口超过26亿？

- 多少年呢？ 87年

# While循环 概念

- Python中的while循环是一种重复执行某个操作，直到条件不再满足的结构。例如，当我们想要打印从1到10的数字时，我们可以使用while循环来实现。代码如下：

**i = 1**

**while i <= 10:**

**print("i=",i)**

**i+=1#必须有， why?**

```
>>> %Run 7_1while.py
```

```
i= 1  
i= 2  
i= 3  
i= 4  
i= 5  
i= 6  
i= 7  
i= 8  
i= 9  
i= 10
```

```
>>>
```

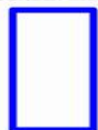
- 以上代码中，我们定义了一个变量i，并将其初始化为1。在while循环中，我们判断i是否小于等于10，如果是，则打印i的值，并将i加1。当i的值变为11时，条件不再满足，循环结束。
- 简单来说，Python中的while循环就是一个重复执行某个操作的工具，直到条件不再满足。
- **条件总是成立，就构成一个死循环**

# While循环流程图1

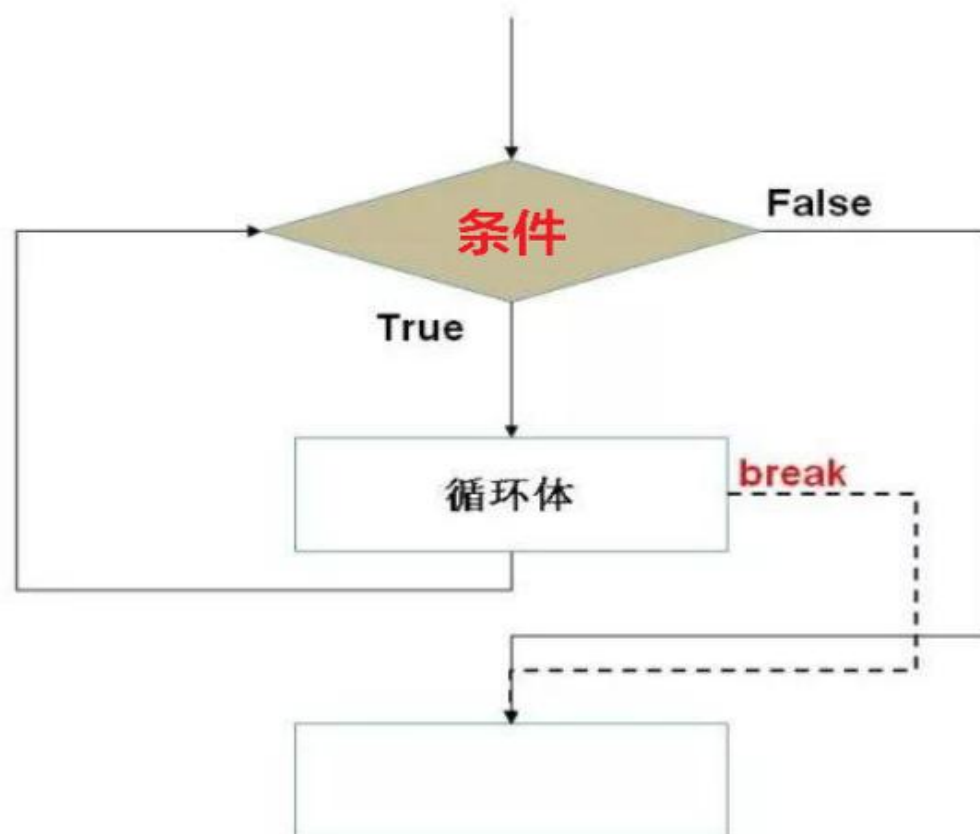
While循环

while循环流程一

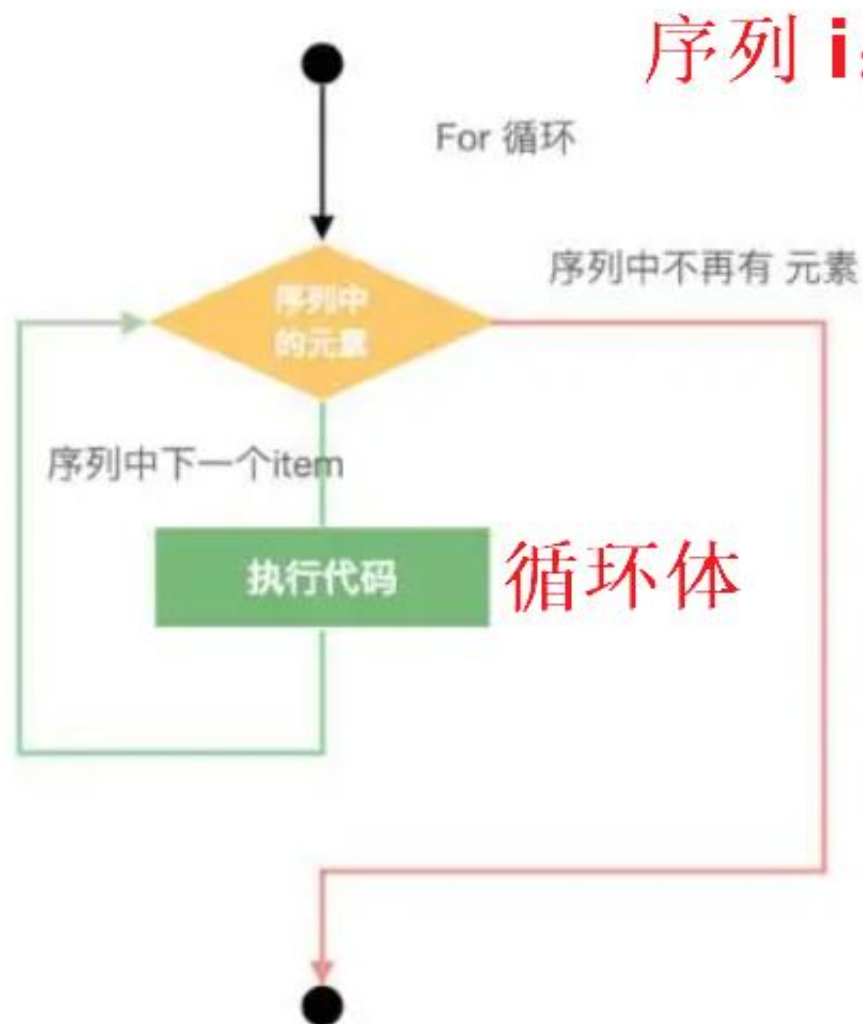
*while* 表达式:



循环体



# 对比: for\_1:流程图



## For循环 语法总结

### 遍历循环

*for* <循环变量> *in* <遍历结构> :  
    <语句块>

- 由保留字for和in组成，完整遍历所有元素后结束
- 每次循环，所获得元素放入循环变量，并执行一次语句块



# While循环

- Python中的while是一个循环语句，它可以让计算机重复执行一段代码，直到某个条件不再满足为止。
- 比如，我们可以编写一个程序来计算从1到10的累加和：

```
sum=0
```

```
i = 1
```

```
while i <= 10:
```

```
    print("i=",i)
```

```
    sum+=i
```

```
    i+=1#必须有，why?
```

```
print("sum=",sum)
```

```
>>> %Run '7_2while sum.py'
```

```
i= 1
```

```
i= 2
```

```
i= 3
```

```
i= 4
```

```
i= 5
```

```
i= 6
```

```
i= 7
```

```
i= 8
```

```
i= 9
```

```
i= 10
```

```
sum= 55
```

- 这个程序中，sum表示累加和，i表示当前的数字，while i <= 10表示只要i小于等于10，就一直执行下面的代码，每次执行都会将i加1，并将i加到sum中。
- 这样，当i变成11时，while条件不再满足，循环就会结束，程序会输出sum的值，也就是1到10的累加和。

# While循环：字符串遍历

- Python 如何访问字符串中的每个字符?假设我们有一个字符串 `s`，我们可以使用一个 `while` 循环来访问每个字符：

```
s = "民航大学路线： 666 678 511 M4"
```

```
index = 0
```

```
while index < len(s):
```

```
    print(s[index])
```

```
    index += 1
```

```
>>> %Run -c
```

```
民航大学路线：
6
6
6
6
7
```

- 这是因为 `while` 循环每次都会访问字符串中的下一个字符。我们使用一个变量 `index` 来跟踪我们现在正在访问哪个字符。我们使用 `len(string)` 函数来获取字符串中字符的数量，以便我们知道何时停止循环。在循环的每个迭代中，我们打印出当前字符并将 `index` 增加 1，以便下次循环中访问下一个字符。

## 循环 split() join()

**s="中国民航大学 CAUC: 空管学院 2022级 牛牛牛"**

**#空格为标志切分**

**s2=s.split(" ")**

**for k in s2:**

**print(k)**

**#,作为分隔符, 连接成一个新的字符串**

**b=",".join(s2)**

**print(b)**

```
>>> %Run 7_6.py
```

```
中国民航大学
```

```
CAUC:
```

```
空管学院
```

```
2022级
```

```
牛牛牛
```

```
中国民航大学,CAUC:,空管学院,2022级,牛牛牛
```

# While循环流程图2

While循环

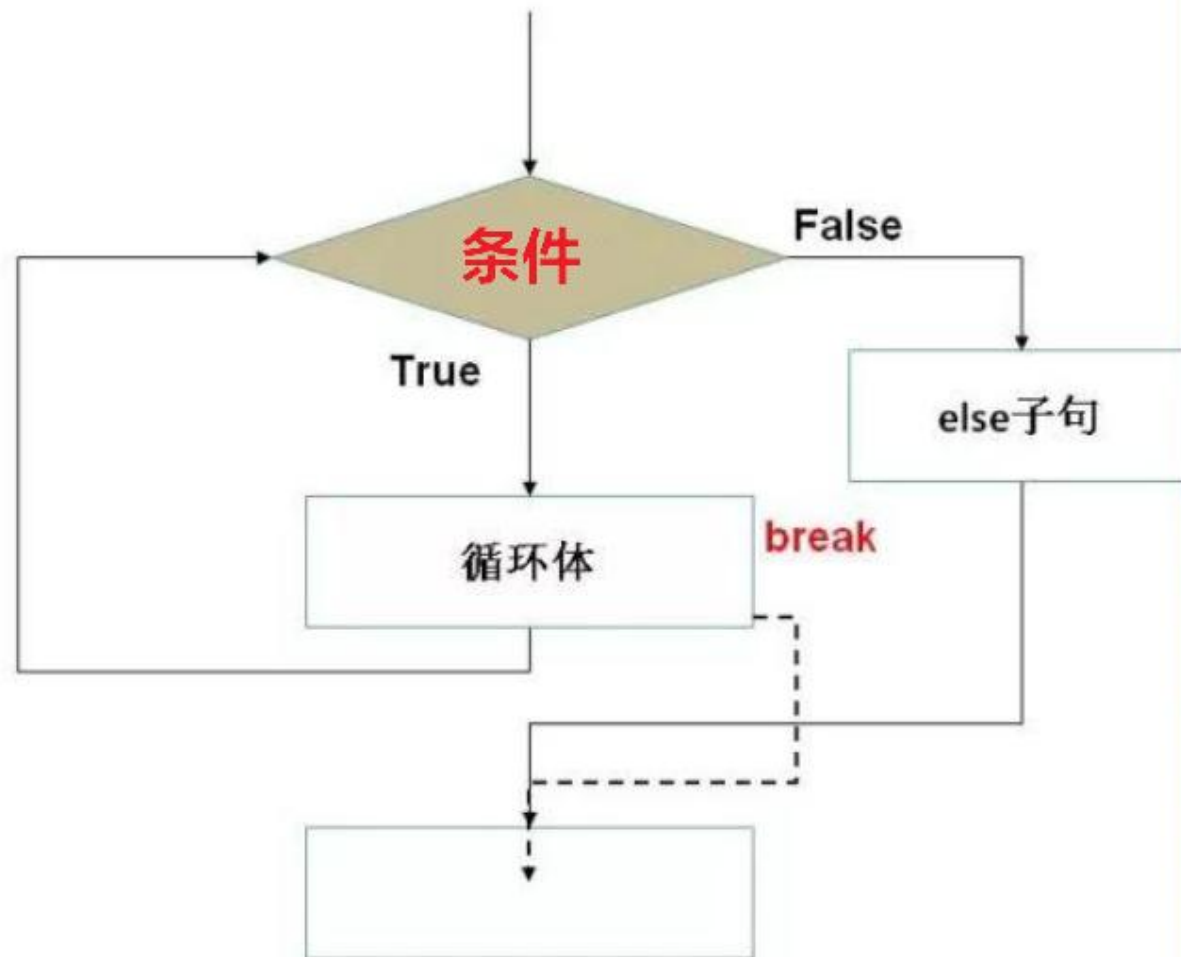
while循环流程二

*while* 表达式:

□ 循环体

*else*:

□ *else*子句



# While else循环:

- Python中的while else是一个循环结构。它的意思是当while语句中的条件不再成立时，会执行else语句块中的代码。
- 例如，我们可以写一个程序来计算1到10的和。代码如下：

```
sum=0
```

```
i = 1
```

```
while i <= 10:
```

```
    print("i=",i)
```

```
    sum+=i
```

```
    i+=1  #必须有， why?
```

```
else:
```

```
    print("sum=",sum)
```

- 在这个例子中，while语句会一直执行，直到i的值大于10。当i等于11时，条件不再成立，程序会跳出while循环，执行else语句块中的代码，输出1到10的和。

# While else循环:

- Python的while else语句可以用来重复执行一段代码，直到满足某个条件后跳出循环，然后执行else语句中的代码。举个例子，假设我们要输出1到5之间的所有偶数，可以使用以下代码：

```
num = 1
```

```
while num <= 10:
```

```
    if num % 2 == 0:
```

```
        print(num)
```

```
    num += 1
```

```
else:
```

```
    print("循环结束")
```

```
>>> %Run '7_3while sum2.py'
2
4
6
8
10
循环结束
```

- 在这个例子中，我们使用了while循环来重复执行代码块，如果num能够被2整除，就输出它，然后num加1，直到num大于5时跳出循环，执行else语句中的代码，即输出"循环结束"。

# 人口增长问题代码如何写？

- 课堂完成

# 辨析：缩进的作用

```
1
2 num = 1
3 while num <= 10:
4     if num % 2 == 0:
5         print(num)
6     num += 1
7 else:
8     print("循环结束")
9
```

Shell ×

```
>>> %Run '7_3while sum2.py'
```

```
2
4
6
8
10
循环结束
```

```
>>>
```

```
1 num = 1
2 while num <= 10:
3     if num % 2 == 0:
4         print(num)
5         num += 1    #死循环
6 else:
7     print("循环结束")
```

Shell ×

```
>>> %Run '7_3while死循环.py'
```



# While else循环：猜数

- Python中的while else是一个循环语句，它用于重复执行特定的代码，直到满足特定的条件为止。如果条件不满足，则执行else语句块中的代码。
- 例如，我们要写一个程序来猜数字游戏。程序会随机生成一个1到10之间的数字，玩家需要猜出这个数字。如果猜错了，程序会提示玩家猜的数字是太大还是太小。如果玩家在5次机会内没有猜出数字，则游戏结束，程序会显示答案。
- 在这个例子中，while循环用于重复执行玩家的猜测，直到猜对了数字或者达到了5次机会。如果玩家在5次机会内没有猜出数字，则else语句块会执行，显示正确的答案。
- 总之，Python中的while else语句是一个循环结构，它允许我们在循环结束时执行一些特定的代码。

# While else循环：猜数

```
1 import random
2 number = random.randint(1, 10)
3 count = 0
4 while count < 5:
5     guess = int(input("猜一下数字是多少？ (1-10) : "))
6     count += 1
7     if guess == number:
8         print("你猜对了！")
9         break
10    elif guess < number:
11        print("你猜的数字太小了！")
12    else:
13        print("你猜的数字太大了！")
14 else:
15    print("很遗憾，你没有猜对，数字是：" + str(number))
```

名称	值
count	5
guess	2
number	2
random	<modu

助手 ×

英 ' 表情 键盘 语音 帮助 打印 分享

If it is not working as it should, please check the general debugging technique.

Shell ×

```
>>> %Run '7_3while caishu.py'
```

```
猜一下数字是多少？ (1-10) : 6
你猜的数字太大了！
猜一下数字是多少？ (1-10) : 5
你猜的数字太大了！
猜一下数字是多少？ (1-10) : 4
你猜的数字太大了！
猜一下数字是多少？ (1-10) : 3
```

绘图器可以把输出到  
Shell中的数字可视化

详细请查看帮助。

激活 Windows  
转到“设置”以激活 Windows。

# 穷举法：水仙花数1

- Python中的while语句可以帮助我们找到水仙花数。水仙花数是指一个三位数，它的每个位上的数字的立方和等于它本身。比如153就是一个水仙花数，因为 $1^3+5^3+3^3=153$ 。现在我们来写一个程序，找出所有的水仙花数：

```
num = 100
```

```
while num < 1000:
```

```
    b = num // 100
```

```
    s = (num // 10) % 10
```

```
    g = num % 10
```

```
    if b ** 3 + s ** 3 + g ** 3 == num:
```

```
        print(num)
```

```
    num+=1
```

```
>>> %Run '7_4水仙花1.py'
```

```
153
```

```
370
```

```
371
```

```
407
```

```
>>>
```

- 这个程序的意思是，从100到999的所有数字都要尝试一遍。对于每个数字，我们把它三个位数分别取出来，然后计算它们的立方和。如果立方和等于这个数字本身，那么就把它打印出来。最后，num要加1，以便进行下一轮循环。

# 穷举法：水仙花数2

- Python是一种编程语言，可以用来编写程序来解决各种问题。其中一个有趣的
- 趣的问题是找出所有的“水仙花数”。
- 水仙花数是指一个三位数，它的每个数字的立方和等于它本身。例如，
- 153就是一个水仙花数，因为1的立方加上5的立方加上3的立方等于153。
- 下面是一个简单的Python程序，可以找出所有的水仙花数：

```
for i in range(100, 1000):
```

```
    if i == (i // 100)**3 + ((i // 10) % 10)**3 + (i % 10)**3:  
        print(i)
```

```
>>> %Run '7_4水仙花3.py'  
153  
370  
371  
407  
>>>
```

- 这个程序使用了一个for循环来遍历所有的三位数。对于每个三位数，它
- 检查它是否等于它的每个数字的立方和。如果是，则打印该数字。

# 穷举法：水仙花数3

- 水仙花数是指一个三位数，它的各位数字立方和等于它本身。比如153就是一个水仙花数，因为1的立方加上5的立方加上3的立方等于153。在Python中，我们可以用一个循环来找出所有的水仙花数，然后把它们放在一个列表里面。代码如下：

```
shuihua=[]
```

```
for num in range(100,1000):
```

```
    b = num // 100
```

```
    s= (num // 10) % 10
```

```
    g = num % 10
```

```
    if b ** 3 + s ** 3 + g ** 3 == num:
```

```
        shuihua.append(num)
```

```
print(shuihua)
```

```
>>> %Run '7_4水仙花2.py'
[153, 370, 371, 407]
```

- 这就是一个简单的Python程序，它可以找出所有的水仙花数，并将它们放在一个列表里面。

# 穷举法：水仙花数4

- 水仙花数是一种数学概念，指的是一个三位数，它的各位数字的立方和等于这个数本身。比如说，153 就是一个水仙花数，因为  $1^3 + 5^3 + 3^3 = 153$ 。在 Python 中，我们可以用字符串来表示这个数，然后用循环和判断语句来找出所有的水仙花数。比如下面这段代码：

```
for num in range(100, 1000):
```

```
    s = str(num)
```

```
    if int(s[0])**3 + int(s[1])**3 + int(s[2])**3 == num:
```

```
        print(num)
```

```
>>> %Run '7_4水仙花4.py'
```

```
153
370
371
407
```

- 这段代码会从 100 到 999 的所有数中找出水仙花数，并把它们打印出来。



# While循环嵌套

**Python while 循环嵌套语法:**

```
while expression:  
    while expression:  
        statement(s)  
    statement(s)
```

# 循环概念

需要程序多次做同一件事情的时候，可以使用迭代语句。一般的迭代（也称循环）机制如图2-4矩形框中所示。与条件语句类似，它从一个测试条件开始。如果测试条件取值为True，程序就执行一次循环体，然后重新检查测试条件。一直重复这个过程，直到测试条件为False，此后程序控制权就传递给迭代语句后面的代码。

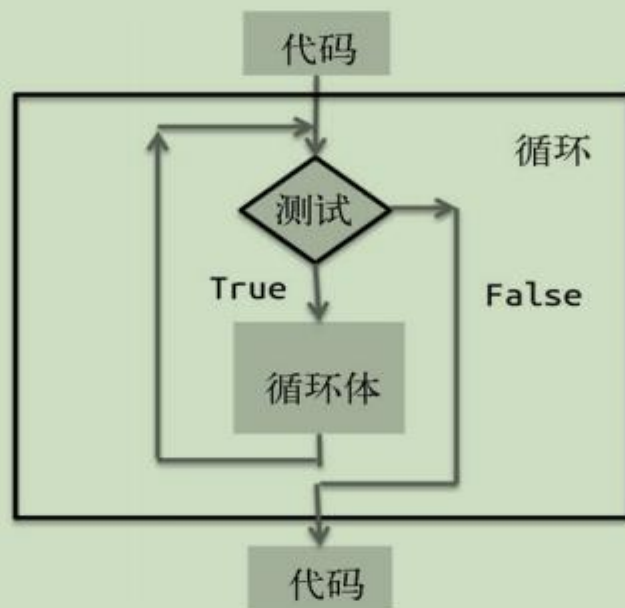


图2-4 迭代流程图

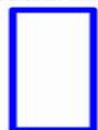


# While循环流程图1

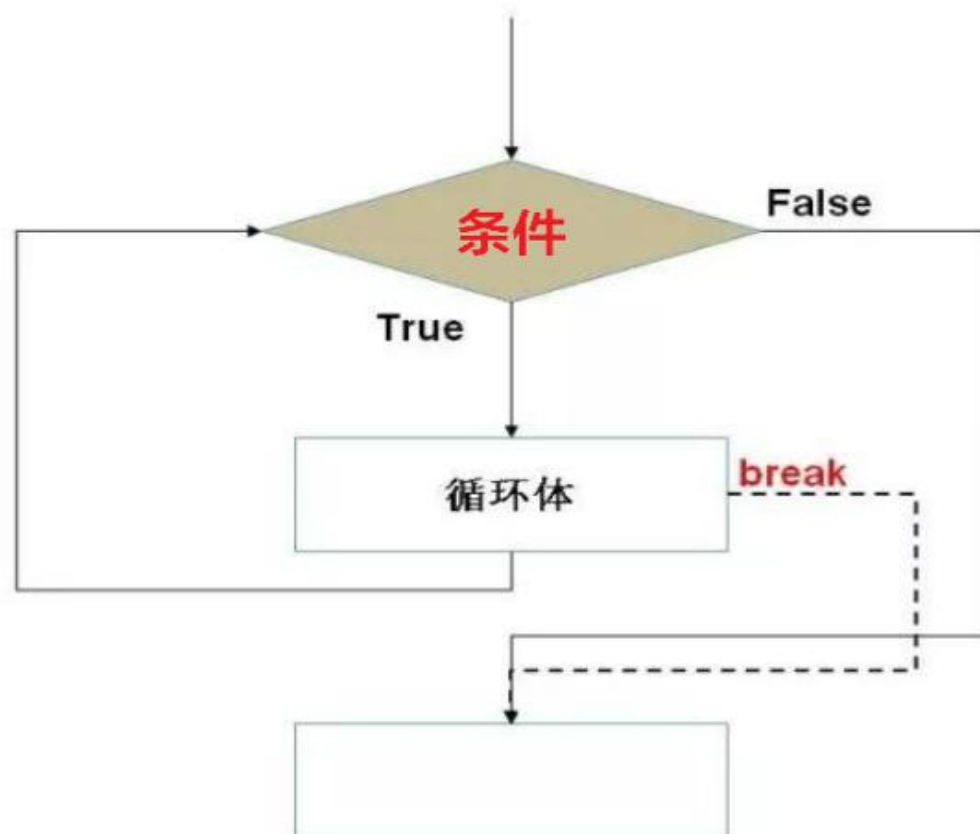
While循环

while循环流程一

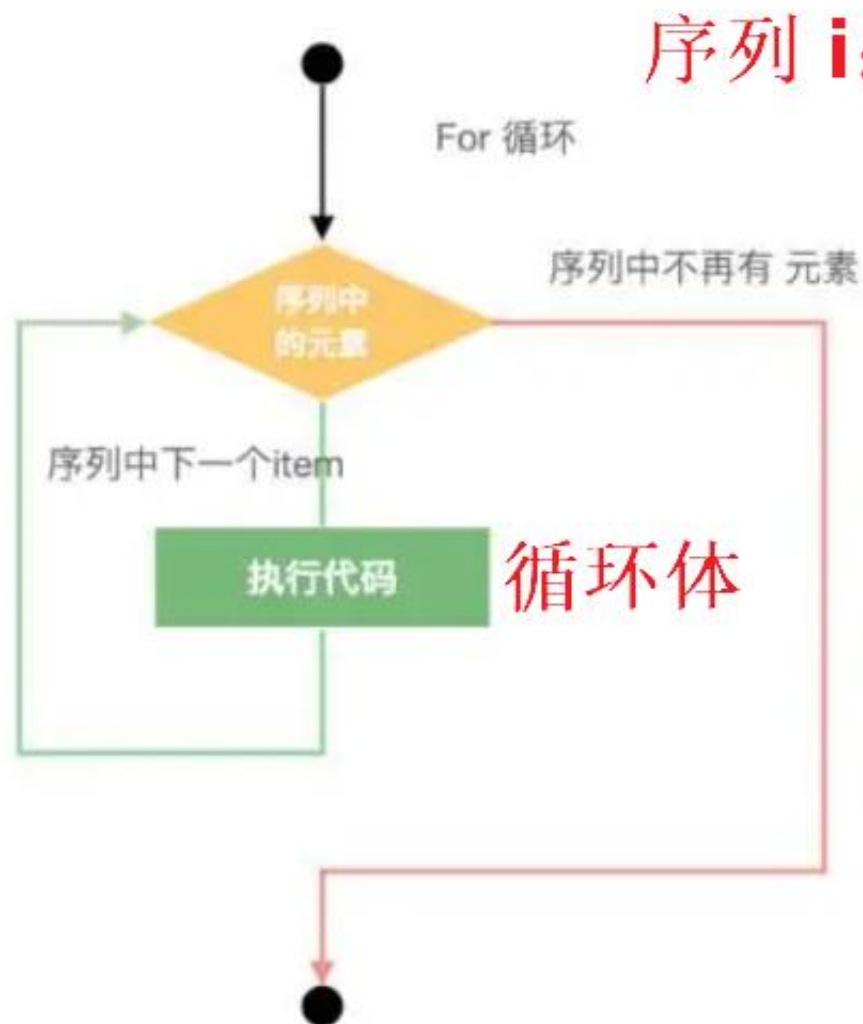
*while* 表达式:



循环体



# 对比: for\_1:流程图



## For循环 语法总结

### 遍历循环

*for* <循环变量> *in* <遍历结构> :  
    <语句块>

- 由保留字for和in组成，完整遍历所有元素后结束
- 每次循环，所获得元素放入循环变量，并执行一次语句块

# While循环流程图2

While循环

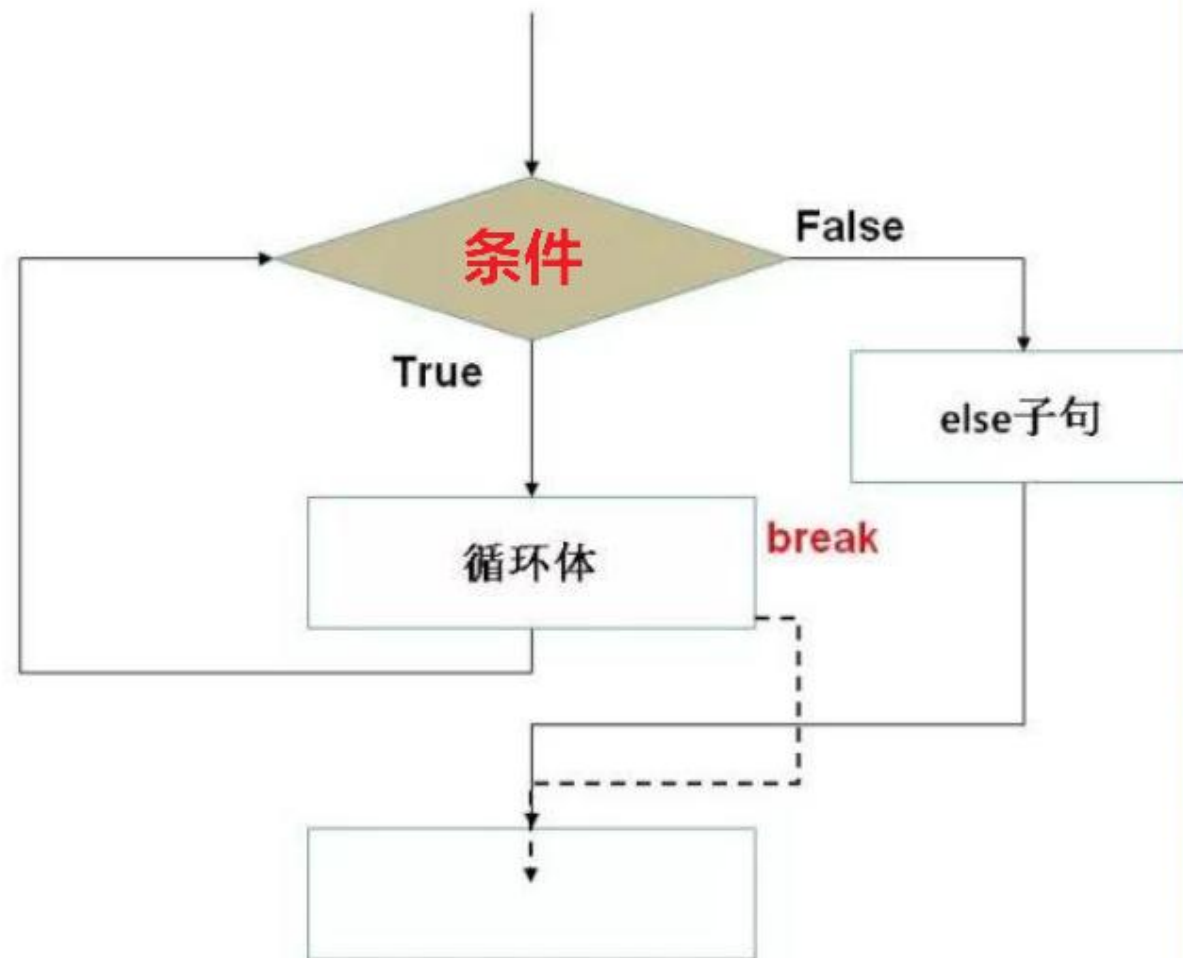
while循环流程二

*while* 表达式:

□ 循环体

*else*:

□ *else*子句



# 趣味编程1：爱心显示

```
for i in range(6):  
    for j in range(7):  
        if (i==0 and j%3!=0) or (i==1 and j%3==0) or (i-j==2) or (i+j==8):  
            print("*", end="")  
        else:  
            print(" ", end="")  
    print()
```

>>> %Run 7\_7.py

```
  **  **  
*   *  *  
*       *  
  *     *  
    *  *  
      *
```

# 趣味编程2

200440024王玮楠.py

进入循环的条件判断:



```
1 while (1):
2     province=input("您来自哪个省/区/市? ")
3     if province=="山东":
4         city=input("您来自哪个城市? ")
5         if city=="菏泽":
6             county=input("您来自哪个区/县? ")
7             if county=="曹县":
8                 print("山东菏泽曹县, 牛批, 666, 我滴宝贝!!! ")
9             else:
10                print("欢迎您来到中国民航大学! ")
11        else:
12            print("欢迎您来到中国民航大学! ")
13    else:
14        print("欢迎您来到中国民航大学! ")
15
16 | 条件不成立, 退出循环
```



问题？