# 循环结构

王肖燕

# 内容回忆

① 单分支、双分支和多分支结构的语法。

② 多分支的else分支一定要有吗?

③ 关系运算和逻辑运算出现在分支结构的判断表达式中，其结果为布尔值：True或False。

④ 有从属关系的结构怎么书写?

- ✓ 缩进，纯Tab
- ✓ 包含if, else, elif关键字的语句要以冒号作为结束

# 内容介绍

## 一．引例

## 二．循环结构

- ✓ while语句，else扩展模式

## 三．改变循环状态的语句

- ✓ break语句、continue语句

## 四．循环嵌套

## 五．基本应用

# 引例1

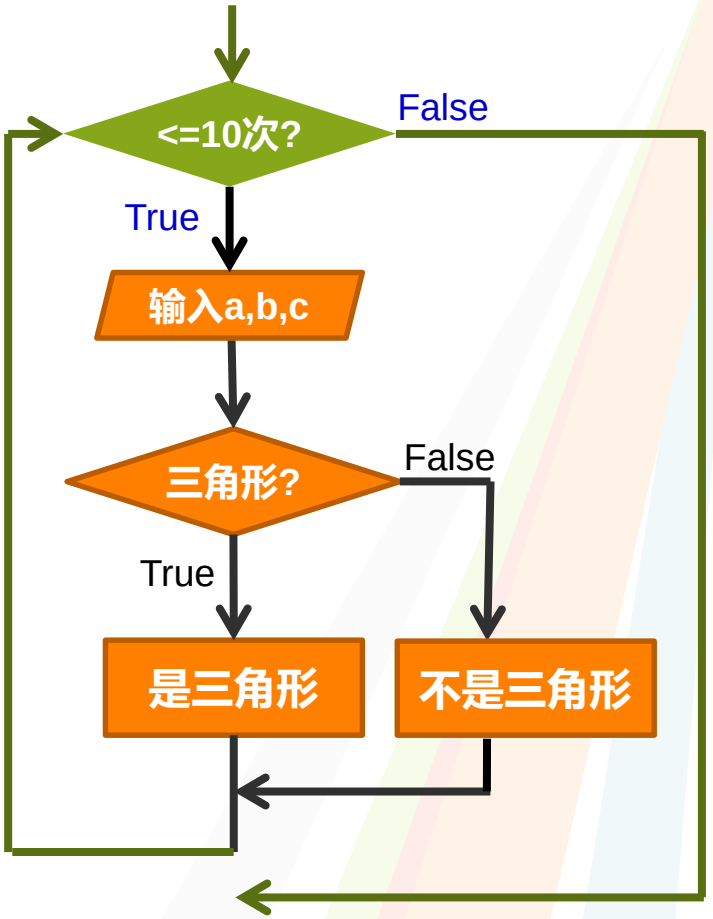
- **有10次输入三个数的机会，分别判断每次输入的三个数作为边能否组成三角形？能组成，打印“可以组成三角形”，不能组成，打印“不能组成三角形”。**

  ① 如何让输入和判断做10次?

  ② 再上节课编写代码的基础上如何完成上述功能?

# 引例1——思路

- **同一件事情，有条件地反复做**
  - ① 确定判断条件“<=10次”
  - ② 操作范围，什么功能要反复做?



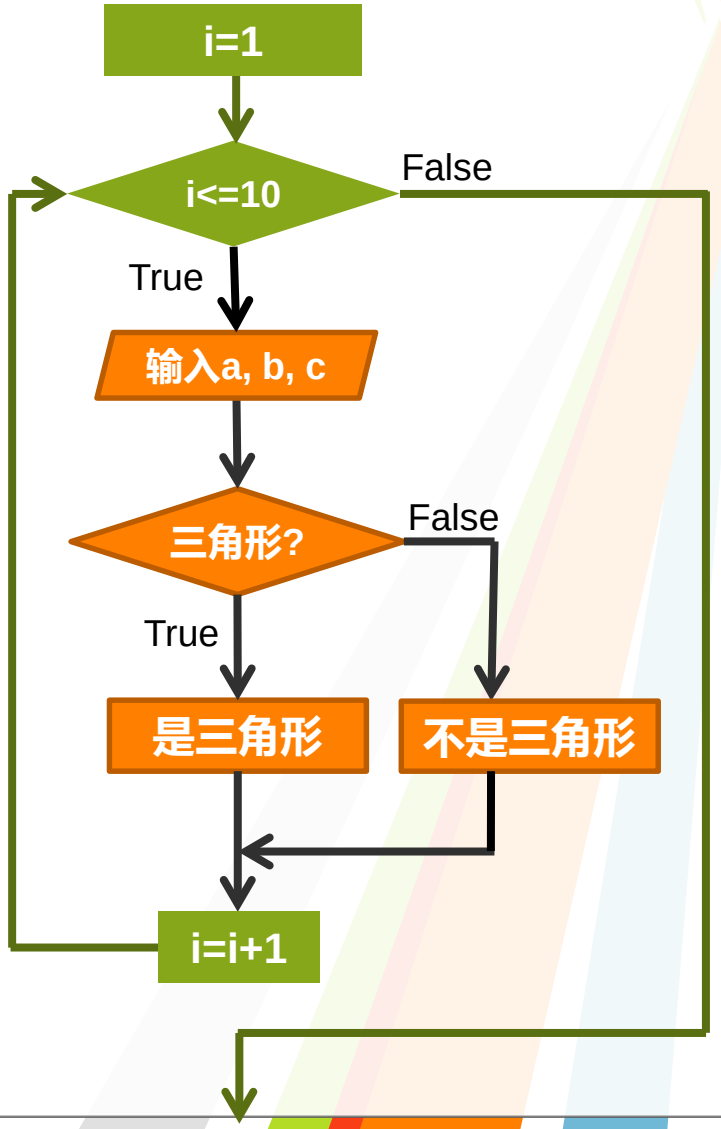
# 引例——流程图

- **同一件事情，有条件地反复做**

① 计数器i，初始设i=1

② 桔色的部分反复做10次，每完成一次都给i增加1

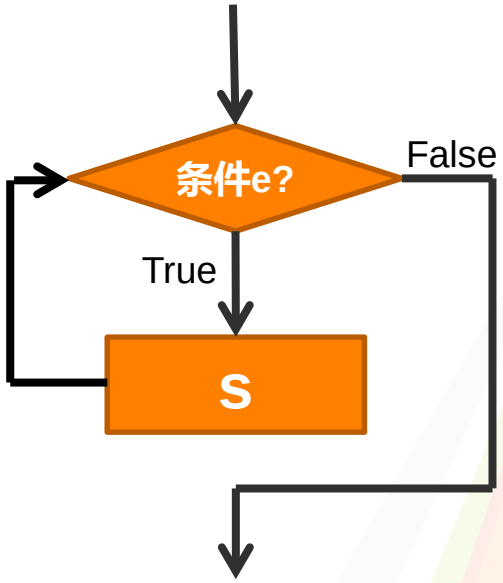
③ 直到“i<=10”不再成立



# 循环结构

- **while语句**

```
While e:
    语句s
```




– 注意：

① 条件与if语句中的的判断条件一样，结果为True或False

② While语义：当条件判断为True时，循环体重复执行语句块中语句；当条件为False时，循环终止，执行与while同级别缩进的后续语句。

# 引例1实现

- 10次输入，10次三边判断：

```
i = 1
while i<=10:
    a = float(input("请输入三角形的第一条边:"))
    b = float(input("请输入三角形的第二条边:"))
    c = float(input("请输入三角形的第三条边:"))
    if a+b>c and a+c>b and b+c>a:
        print("可以组成三角形")
    else:
        print("此三边不能组成三角形")
    i = i + 1
```

# 循环是否正常结束

- while循环的else扩展模式

```
while e:
    语句s
else:
    print(“循环正常结束”)
```

- else语句只在循环正常执行结束后才执行

```
i = 1
while i<=2:
    a = float(input("请输入三角形的第一条边:"))
    b = float(input("请输入三角形的第二条边:"))
    c = float(input("请输入三角形的第三条边:"))
    if a+b>c and a+c>b and b+c>a:
        print("可以组成三角形")
    else:
        print("此三边不能组成三角形")
    i = i + 1
else:
    print("循环正常执行")
```

```
请输入三角形的第一条边:1
请输入三角形的第二条边:2
请输入三角形的第三条边:3
此三边不能组成三角形
请输入三角形的第一条边:6
请输入三角形的第二条边:6
请输入三角形的第三条边:6
可以组成三角形
循环正常执行
```

# 引例2

- 打印如下矩阵：

| | | |
|---|---|---|
| 1 | 2 | 3 |
| 2 | 4 | 6 |
| 3 | 6 | 9 |
| 4 | 8 | 12 |
| 5 | 10 | 15 |
| 6 | 12 | 18 |

- 如果扩展成更大的矩阵，怎么实现打印？

# 循环嵌套

- **while嵌套**

```
While e:
    语句s(包含循环结构)
```

  - 注意：
    - 每层循环都要符合结构要求，理清缩进关系
    - 分析各循环所控制的功能，包含准确

# 引例2实现

- 分析行与行之间的关系，总结规律
- print()函数输出不换行，语法：
  - print(“输出内容”，end=“间隔符”)
  - “\t” 制表符
    - “输出内容”共占8个字节
    - 如果“输出内容”大于等于8个字节，则“输出内容”共占16个字节

```
i=1
while i<=6:
    j = 1
    while j <= 5:
        print(i*j, end="\t")
        j = j + 1
    i = i + 1
    print()
```

# 引例3

- 打印如下矩阵：

| | | |
|---|---|---|
| 1 | 2 | 3 |
| 2 | 4 | 6 |
| 3 | 6 | 9 |
| | | |
| 5 | 10 | 15 |
| 6 | 12 | 18 |

- 分析与引例2有什么不同？缺了什么？

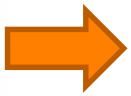
# 改变循环的语句1

- **break语句**
  - 语法：break
  - 用来跳出包含此break语句的那层循环
  - 脱离该循环后程序从此循环代码后继续执行
  - 如果跳出的包含break语句的循环是内层循环，则仍然继续执行外层的循环
  - 每个break语句只有能力跳出当前层次循环

# 引例3实现

- 满足条件后，才跳出内层循环

```
i=1
while i<=6:
    j = 1
    while j <= 3:
        if i==4:
            break
        else:
            print(i*j, end="\t")
        j = j + 1
    i = i + 1
    print()
```



```
1       2       3
2       4       6
3       6       9

5       10      15
6       12      18
```

# 引例4

- 打印如下矩阵：

| | | |
|---|---|---|
| 1 | 2 | 3 |
| 2 | 4 | 6 |
| 6 | 9 | |
| 4 | 8 | 12 |
| 5 | 10 | 15 |
| 6 | 12 | 18 |

- 分析与引例2有什么不同？缺了什么？

# 改变循环的语句2

- **continue语句**
  - 语法： continue
  - 用来结束当前当次循环
  - 即跳出循环体中下面尚未执行的语句，但不跳出当前循环
  - 与break的区别：
    - Continue只结束本次循环，而不终止整个循环的执行
    - Break语句则结束整个循环过程，不再判断执行循环的条件是否成立

# 引例4实现

- 忽略第3行第1列的输出，但数据仍需要继续计算：

```
i=1
while i<=6:
    j = 1
    while j <= 3:
        if i==3 and j==1:
            j = j + 1
            continue
        else:
            print(i*j, end="\t")
        j = j + 1
    i = i + 1
    print()
```

```
1	2	3
2	4	6
6	9
4	8	12
5	10	15
6	12	18
```

# 应用举例1

- **打印50以内的偶数。**
  - 偶数：可以被2整除
  - 一层循环，取到1~50中的每一个整数
  - 数据打印5个数据一行，如下所示：

| | | | | |
|---|---|---|---|---|
| 2 | 4 | 6 | 8 | 10 |
| 12 | 14 | 16 | 18 | 20 |
| 22 | 24 | 26 | 28 | 30 |
| 32 | 34 | 36 | 38 | 40 |
| 42 | 44 | 46 | 48 | 50 |

# 应用举例2

- **猜数字**：程序内先设定幸运数字n的值，然后编写程序，让用户去不断地猜测幸运数字。

(1) 若猜的数比n大，打印“猜大了！”，继续猜；

(2) 若猜的数比n小，打印“猜小了！”，继续猜；

(3) 若猜的正好是n的值，打印“Bingo！”，并退出。

- 循环结构和分支结构结合使用
- 使用break退出循环

```
幸运数字是：6
猜小了！
幸运数字是：100
猜大了！
幸运数字是：53
猜大了！
幸运数字是：40
猜小了！
幸运数字是：46
猜小了！
幸运数字是：50
猜小了！
幸运数字是：52
猜大了！
幸运数字是：51
Bingo!
```

# 应用举例3

- 判断一个正整数n是否为素数。**素数**：只能被1和其本身整除的自然数。

  ① 判断n是否有除了1和n以外的其它因子

  ② 如果有说明不是素数，如果没找到说明是素数

# 练习1

- 输出1~100之间所有能被7整除，但不能同时被5整除的所有整数。
  - 首先想清楚结构
  - 循环和分支的判断表达式如何写
  - 层次缩进关系要清晰，不能错位

# 练习2

- 输入若干个成绩，求所有成绩的平均分
  - 每输入一个成绩后询问是否继续输入下一个成绩，回答yes继续输入下一个成绩，回答no就停止输入成绩

# 练习3

- 打印1~100之间所有的素数。**素数**：只能被1和其本身整除的自然数。

① 外层循环取到1~100之间的每一个数n

② 内层循环判断n是否有除了1和n以外的其它因子，如果有说明不是素数，如果没找到说明是素数

```
素数:
2       3       5       7       11      13      17      19      23
29      31      37      41      43      47      53      59      61
67      71      73      79      83      89      97
```

# Thank You!