

# 第 3 章 选 择 结 构

选择结构是程序设计中常用的基本结构,其功能是对给定的条件进行比较和判断,并根据判断结果采取不同的操作。Python 中常见的选择结构有 3 种:单分支选择结构、双分支选择结构和多分支选择结构。



## 3.1 单分支选择结构



单分支选择结构是最基本的选择结构语句,其语法格式如下:

```
if 表达式:
    语句块
```

功能: 如果表达式的值为真,则执行其后的语句块,否则不做任何操作。其结构如图 3-1 所示。

说明: if 语句中,表达式表示一个判断条件,可以是任何能够产生 True 或 False 的表达式或函数。表达式中一般包含关系运算符、成员运算符或逻辑运算符。

在 Python 中,整数 1 代表真(True),0 代表假(False)。非零数字或非空对象也被视为真,任意的空数据结构或空对象以及特殊对象 None 都被视为假。

例如,选拔身高 T 超过 1.7 米且年龄 age 小于 25 岁的人,则该条件的逻辑表达式如下:

```
T>1.7 and age<25
```

例如,在某个游戏过程中,如果本关积分达到 1000,则提示“进入下一关”。用选择结构来实现此功能,部分代码如下:

```
if fen==1000:
    fen=0
    print('进入下一关')
```

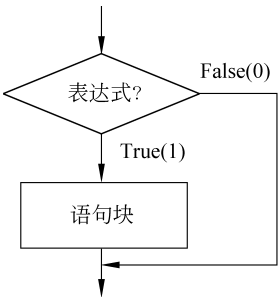


图 3-1 单分支选择结构

**注意：**这里的两条语句“fen=0”和“print('进入下一关')”是 if 条件成立时要执行的同一个语句块,所以语句前的缩进要一致。Python 最具特色的功能就是使用缩进表示语句块,不需要使用大括号。缩进的字符数是可变的,但是同一个语句块的语句必须包含相同的缩进字符数,如果缩进不一致会导致逻辑错误。



## 3.2 双分支选择结构

双分支选择结构的语句用于区分条件的两种执行结果,语法格式如下:

```
if 表达式:
    语句块 1
else:
    语句块 2
```

功能: 当表达式的值为真时,执行 if 后面的语句块 1,否则执行 else 后面的语句块 2,其结构如图 3-2 所示。

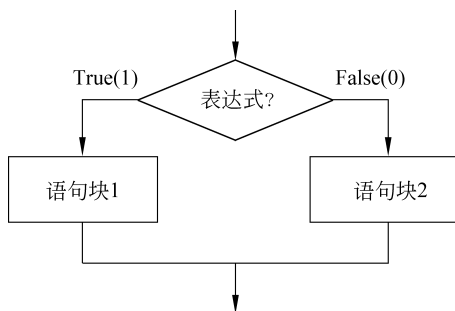


图 3-2 双分支选择结构

**注意：**双分支选择结构中,if 和 else 后面的冒号不能缺失,而且其后语句块 1 和语句块 2 都必须要有缩进,否则会产生语法错误。

**【例 3-1】** 已知某书店图书均九折销售,一次购书 100 元以上(包括 100 元)打八折。编写程序,要求根据输入的购书金额计算并输出应付款。

分析: 设购书金额为  $x$  元,应付款为  $y$  元。由题意可知:

$$y = \begin{cases} 0.9x, & x < 100 \\ 0.8x, & x \geq 100 \end{cases}$$

计算  $y$  的值可利用双分支选择结构的语句来实现。

程序代码如下:

```
x=eval(input("请输入购书金额: "))
if x<100:
    y=0.9 * x
else:
    y=0.8 * x
print("优惠后应付金额是{:.2f}".format(y))
```

**【例 3-2】** 输入选拔赛分数,如果达到 60 分,则输出“合格”,否则输出“淘汰”。

程序代码如下:

```
x=input('输入分数: ')
fen=int(x)
if fen>=60:
    print('合格')
else:
    print('淘汰')
print('评价完成')
```

在 Python 中,双分支选择结构还有一种更简洁的表达式,适合在判断后直接返回特定的结果值,其语法格式如下:

```
表达式 1 if 条件 else 表达式 2
```

其中,表达式 1 和表达式 2 一般是一个数字类型或字符串类型的值。例如,例 3-2 中的双分支选择语句块可以改为一条语句:

```
print('合格' if fen>=60 else '淘汰')
```

程序代码如下:

```
fen=int(input('输入分数: '))
print('合格' if fen>=60 else '淘汰')
print('评价完成')
```

运行结果如下:

```
输入分数: 78
合格
评价完成
>>>
```

### 3.3 多分支选择结构



Python 的多分支选择结构使用 if...elif...else 语句表达,其语法格式如下:

```
if 表达式 1:
    语句块 1
elif 表达式 2:
    语句块 2
...
else:
    语句块 n
```

说明：该语句适用于多个变量、多种条件的情况，判断的顺序为条件表达式 1、条件表达式 2、条件表达式 3……一旦与某一条件匹配就执行该条件下的语句块，随即跳出整个 if 结构，即使下面仍有条件匹配也不再执行，因此要注意语句中条件表达式的排列次序，以免某些情况不被处理。如果没有任何条件成立，else 下面的语句块将被执行，else 子句是可选的。

if...elif...else 语句的多分支选择结构如图 3-3 所示。

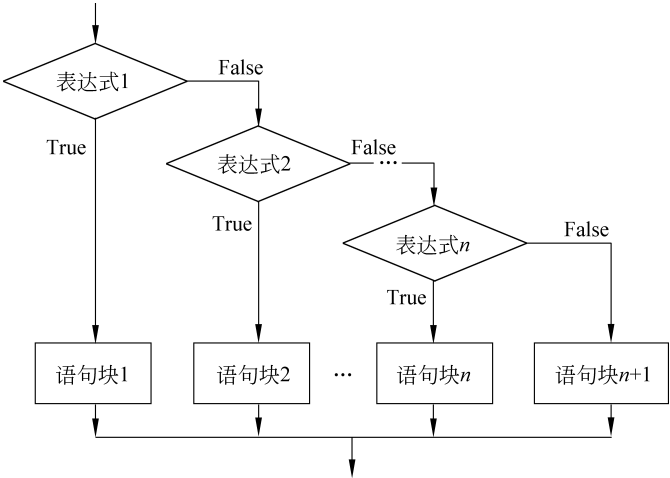


图 3-3 if...elif...else 语句的多分支选择结构

**【例 3-3】** 根据输入的学生百分制成绩，判定成绩的等级，输出相应的等级评价。等级评价标准如表 3-1 所示。

表 3-1 等级评价标准

| 等级 | 优            | 良                        | 中                        | 及格                       | 不及格       |
|----|--------------|--------------------------|--------------------------|--------------------------|-----------|
| 分数 | 分数 $\geq 90$ | $90 > \text{分数} \geq 80$ | $80 > \text{分数} \geq 70$ | $70 > \text{分数} \geq 60$ | 分数 $< 60$ |

该程序是一个多分支选择的问题，可以使用多分支选择结构 if...elif...else 语句实现。程序代码如下：

```
stumark=int(input("输入学生的成绩："))
if stumark<0 or stumark>100:
    print("学生成绩输入有误!")
else:
    if stumark >= 90 :
        grade="优"
    elif stumark >= 80 :
        grade="良"
    elif stumark >= 70:
        grade="中"
    elif stumark >= 60 :
        grade="及格"
```

```
else:
    grade="不及格"
print("这位学生的等级评价是:", grade)
```

**注意：**多分支选择结构的一系列条件判断会从上到下依次执行,如果某个条件判断为 True,则执行该条件判断对应的语句块后,后面的条件判断就直接忽略,不再执行了。

**思考：**例 3-3 中 if...elif...else 部分代码如果写成下面的形式,结果会如何?

```
if stumark>=60:
    grade="及格"
elif stumark>=70:
    grade="中"
elif stumark>=80:
    grade="良"
elif stumark>=90:
    grade="优"
else:
    grade="不及格"
```

## 3.4 选择结构的嵌套



如果 if 语句中又包含一个或多个 if 语句,则称为 if 语句的嵌套。例如:

```
if 表达式 1:
    if 表达式 2:
        语句块 1
    else:
        语句块 2
else:
    if 表达式 2:
        语句块 3
    else:
        语句块 4
```

**【例 3-4】** 输入平面坐标系中某点的坐标值( $x, y$ ),判定输出该点在第几象限。

分析: 在直角坐标系中,点所在的象限有以下几种情况。

- (1) 当  $x=0$  或  $y=0$  时,点在坐标轴上,不在任何象限。
- (2) 当  $x>0$  时,有两种情况:若  $y>0$ ,点在第一象限;若  $y<0$ ,点在第四象限。
- (3) 当  $x<0$  时,有两种情况:若  $y>0$ ,点在第二象限;若  $y<0$ ,点在第三象限。

该程序可使用 if 语句的嵌套实现。程序代码如下:

```
import random
x=random.randint(-100,100)
```

```

y=random.randint(-100,100)
print("随机生成的坐标点为: ",x," ",y)
if x==0 or y==0:
    print("该点在坐标轴上,不在任何象限")
elif x>0:
    if y>0:
        print("该点在第一象限")
    elif y<0:
        print("该点在第四象限")
elif x<0:
    if y>0:
        print("该点在第二象限")
    elif y<0:
        print("该点在第三象限")
print("判断完成!")

```

程序运行结果如下：

```

随机生成的坐标点为: -93 ,-38
该点在第三象限
判断完成!
>>>
随机生成的坐标点为: 22 ,-58
该点在第四象限
判断完成!
>>>

```

## 3.5 应用实例

**【例 3-5】** 设计一个简易四则运算计算器,输入任意两个整数及运算符后,输出计算结果。如果输入的运算符不是+、-、\*、/之一,则提示“运算符错误,无法计算!”。

分析: 本例利用了选择结构的嵌套模式,先判断输入的运算符是否是['+', '-', '\*', '/']符号,如果不是,则输出提示语并结束程序;如果是,则进入一个多分支选择语句,根据不同的运算符求解并输出数值。

程序代码如下:

```

x,y=eval(input("请输入两个整数(用逗号分隔): "))
op=input("输入算术运算符: ")
if op!='+' and op!='-' and op!='*' and op!='/' :
    #此句也可以利用列表功能来做判断,即 if op not in[ '+', '-', '*', '/' ]
    print('运算符错误,无法计算!')
elif op=='/' and y==0:

```

```

        print('除数为 0')
    else:
        if op=='+':
            z = x + y
        elif op=='-':
            z = x - y
        elif op=='*':
            z = x * y
        elif op=='/':
            z = x/y
    print('%d %c %d ='%(x,op,y), z)

```

程序运行结果如下：

```

请输入两个整数(用逗号分隔): 23,6
输入算术运算符: *
23 * 6=138
>>>
请输入两个整数(用逗号分隔): 78,12
输入算术运算符: /
78 / 12=6.5
>>>
请输入两个整数(用逗号分隔): 45,100
输入算术运算符: $
运算符错误,无法计算!
>>>

```

**【例 3-6】** 求  $ax^2+bx+c=0$  方程的根。系数  $a$ 、 $b$ 、 $c$  由键盘输入。

分析：根据系数  $a$ 、 $b$ 、 $c$  的值，可得出方程有以下几种可能。

- (1)  $a=0$ ，不构成一元二次方程。
- (2) 如果  $b^2-4ac=0$ ，一元二次方程有两个相等实根。
- (3) 如果  $b^2-4ac>0$ ，一元二次方程有两个不等实根。
- (4) 如果  $b^2-4ac<0$ ，一元二次方程有两个共轭复根，形式为  $m+nj$  和  $m-ni$ ，其中

$$m = -\frac{b}{2a}, n = \frac{\sqrt{-(b^2-4ac)}}{2a}。$$

本例中，在双分支选择结构中嵌套多分支选择来区分上面几种方程根的情况。程序开始，先判断是否是一元二次方程，如果是，再根据  $p$  的值即  $b^2-4ac$  的情况来分类判断属于哪种根的求解。如果是复根，先分别计算出实部  $m$  和虚部  $n$ ，再利用 `print()` 函数的格式输出功能输出“ $m+nj$ ”这样的复数形式。

程序代码如下，注意这里的分支结构及其嵌套的含义和缩进格式。

```

import math
a,b,c=eval(input("输入方程式 ax^2+bx+c=0 的系数 a,b,c: "))
print("方程式为 %dx^2+%dx+%d=0 " %(a,b,c))
if a==0:
    print("这不是一元二次方程!")
else:
    print("是一元二次方程:")
    p=pow(b,2)-4*a*c
    if p==0:
        print("有两个相等实根,x1=x2=%f " %(-b / (2*a)), "\n")
    elif p>0:
        x1=(-b+math.sqrt(p))/(2*a)
        x2=(-b-math.sqrt(p))/(2*a)
        print("有两个不同的根:x1=%f , x2=%f " %(x1,x2))
    else:
        m=-b / (2*a)
        n=pow(-p,0.5) / (2*a)
        x1=complex(m,n)
        x2=complex(m,-n)
        print("有两个复根: x1={:.6f},x2={:.6f}".format(x1,x2))

```

程序运行结果如下:

```

输入方程式 ax^2+bx+c=0 的系数 a,b,c: 1,2,1
方程式为 1x^2+2x+1=0
是一元二次方程:
有两个相等实根,x1=x2=-1.000000
>>>
输入方程式 ax^2+bx+c=0 的系数 a,b,c: 2,6,1
方程式为 2x^2+6x+1=0
是一元二次方程:
有两个不同的根:x1=-0.177124 , x2=-2.822876
>>>
输入方程式 ax^2+bx+c=0 的系数 a,b,c: 3,2,4
方程式为 3x^2+2x+4=0
是一元二次方程:
有两个复根:x1=-0.333333+1.105542j, x2=-0.333333-1.105542j
>>>
输入方程式 ax^2+bx+c=0 的系数 a,b,c: 0,2,5
方程式为 0x^2+2x+5=0
这不是一元二次方程!
>>>

```



## 习题 3



扫码答题

### 一、简答题

1. 简述选择结构的种类。
2. 简述 if、if...else 和 if...elif...else 语句的语法格式和使用时的区别。
3. 什么情况下条件表达式会认为结果是 False?
4. Python 的多分支选择结构中,多个 elif 分支随意调整位置会有什么后果?
5. 当多分支选择结构中有多个表达式条件同时满足时,则每个与之匹配的语句块都被执行,这种说法是否正确?

### 二、选择题

1. 可以用来判断某语句是否在分支结构的语句块内的是( )。  
A. 缩进                      B. 括号                      C. 逗号                      D. 分号
2. 以下选项中,不是选择结构里的保留字是( )。  
A. if                          B. elif                      C. else                      D. elseif
3. 以下条件选项中,合法且在 if 中判断是 False 的是( )。  
A.  $24 \leq 28 \& \& 28 > 25$                       B.  $24 < 28 > 25$   
C.  $35 < = 45 < 75$                               D.  $24 < = 28 < 25$
4. 以下针对选择结构的描述中,错误的是( )。  
A. 每个 if 条件后或 else 后都要使用冒号  
B. 在 Python 中,没有 select...case 语句  
C. Python 中的 pass 是空语句,一般用作占位语句  
D. elif 可以单独使用,也可以写为 elseif
5. Python 中,( )是一种更简洁的双分支选择结构。  
A. '合格' if fen  $\geq$  60 else '淘汰'  
B. if fen  $\geq$  60 '合格' else '淘汰'  
C. if fen  $\geq$  60: '合格': '淘汰'  
D. if fen  $\geq$  60: '合格' elseif '淘汰'

### 三、填空题

1. Python 中,用于表示逻辑与、逻辑或、逻辑非运算的关键字分别是 \_\_\_\_\_、\_\_\_\_\_ 和 \_\_\_\_\_。
2. 表达式  $1 \text{ if } 2 > 3 \text{ else } (4 \text{ if } 5 > 6 \text{ else } 7)$  的值为 \_\_\_\_\_。
3. if...else 双分支选择结构中,else 与 if 语句后面必须有 \_\_\_\_\_ 符号。
4. Python 中,表示条件真或假的两个关键字是 \_\_\_\_\_ 和 \_\_\_\_\_。
5. 算术运算符、关系运算符、逻辑运算符中优先级最高的是 \_\_\_\_\_。

### 四、编程题

1. 由键盘输入 3 个整数,请利用分支选择结构语句编程,输出其中最大的数。
2. 编程查询某日汽车限行车牌尾号。限行规则为工作日每天限行两个号:车牌尾数

为 1 和 6 的机动车周一限行;车牌尾数为 2 和 7 的周二限行;车牌尾数为 3 和 8 的周三限行;车牌尾数为 4 和 9 的周四限行;车牌尾数为 5 和 0 的周五限行。请输入星期几的代号(1~7),输出相应的限行车辆尾号。

3. 输入年份和月份,求该月的天数。

**提示:**当月份为 1、3、5、7、8、10、12 时,该月的天数为 31 天;当月份为 4、6、9、11 时,该月的天数为 30 天;当月份为 2 时,如果是闰年,则该月的天数为 29 天,否则为 28 天。某年为闰年的条件是,年份能被 4 整除但不能被 100 整除,或年份能被 400 整除。

4. 编写程序,输入三角形的三条边长,如果能构成三角形,则判断是等腰、等边、直角三角形,还是一般三角形。

5. 某汽车运输公司开展整车货运优惠活动,货运收费根据各车型货运里程的不同而定,其中一款货车的收费标准如下,编程实现自动计算运费问题。

(1) 距离在 100km 以内:只收基本运费 1000 元。

(2) 距离为 100~500km:除基本运费外,超过 100km 的部分,运费为 3.5 元/千米。

(3) 距离超过 500km:除基本运费外,超过 100km 的部分,运费为 5 元/千米。