

# Python 语言程序设计: 流程控制之分支结构

李宽

likuan@dgut.edu.cn

东莞理工学院

2019.10

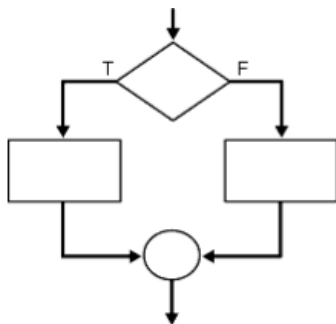


## 1 程序的分支结构

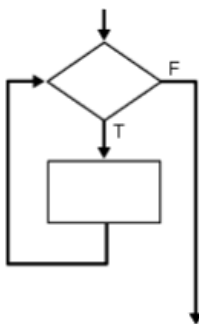
- 单分支结构
- 二分支结构
- 多分支结构
- 分支嵌套
- 条件判断及组合

# 程序的流程控制

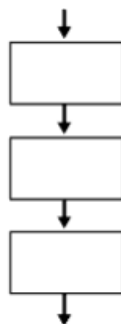
三种基本流程：**顺序**, **分支**, **循环**



**Selection**



**Iteration**



**Sequence**

# 程序的分支结构

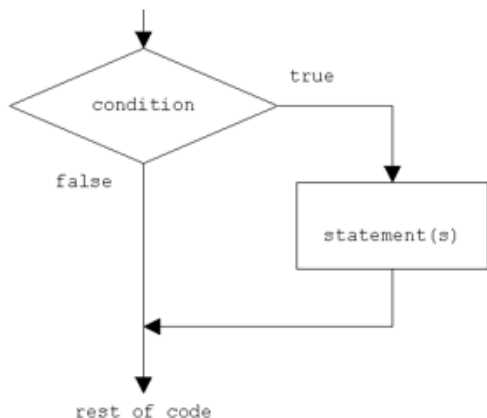
**分支**: 根据**判断条件的结果**选择不同向前路径

- 单分支结构
- 二分支结构
- 多分支结构
- 条件判断及组合

## 1 程序的分支结构

- 单分支结构
- 二分支结构
- 多分支结构
- 分支嵌套
- 条件判断及组合

# 单分支结构 1/2



```
1 if <条件>:  
2   <语句块>
```

条件和语句块的补充说明：

- 条件:  
非 0 和非空  $\text{null} \leftrightarrow \text{true}$   
0 或者  $\text{null} \leftrightarrow \text{false}$
- 语句块: 邻接同级缩进

语义：“如果，那么”

## 单分支结构 2/2

代码举例 (if-ex.py) :

---

```
1 x=20
2 y=0
3 if x > y :
4     print(" X is bigger 1st")
5     print(" X is bigger 2nd")
6     print(" X is bigger 3rd")
7
8 if x:
9     print("non-zero equals true")
```

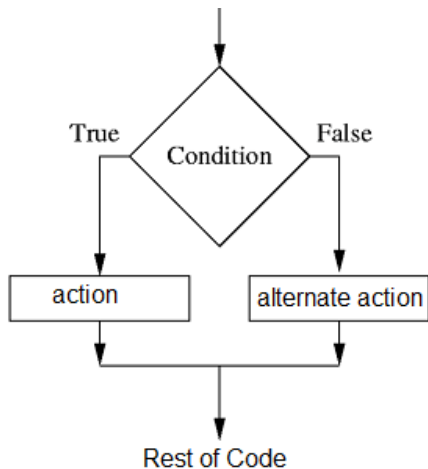
---

## 1 程序的分支结构

- 单分支结构
- 二分支结构
- 多分支结构
- 分支嵌套
- 条件判断及组合



## 二分支结构 1/2



---

```
1 if <条件>:  
2     <语句块>  
3 else:  
4     <语句块>
```

---

**语义:** 如果... 那么..., 否则...

## 二分支结构 2/2

代码举例 (ifelse-ex.py) :

---

```
1 x=10
2 y=20
3 if x > y:
4     print("x is bigger")
5 else :
6     print("y is bigger")
```

---

## 1 程序的分支结构

- 单分支结构
- 二分支结构
- 多分支结构
- 分支嵌套
- 条件判断及组合

# 多分支结构 1/4

```
1 if <条件1>:  
2     <语句块1>  
3 elif <条件2>:  
4     <语句块2>  
5     ...  
6 else:  
7     <语句块else>
```

注：else 部分允许缺失

## 多分支结构 2/4

多分支结构范例(纠错,multi\_if\_err.py):

```
1 #python 多分支结构纠错
2 mark = int(input("请输入取得的分数: "))
3
4 if mark >= 60:
5     print ("You got C Grade !!")
6 elif mark >=70:
7     print ("You got B Grade !!")
8 elif mark >=85:
9     print ("You got A Grade !!")
10 else:
11     print ("You failed !!")
```

- 多个条件之间的包含关系
- 变量取值范围的覆盖

## 多分支结构 3/4

修改 1: 限定范围 (multi\_if1.py)

---

```
1 #python 多分支结构纠错 1
2 mark = int(input("请输入取得的分数: "))
3
4 # 下句等价于 "if 60 <= mark and mark < 70"
5 if 60 <= mark < 70:
6     print ("You got C Grade !!")
7 elif 70 <= mark < 85:
8     print ("You got B Grade !!")
9 elif 85 < mark:
10    print ("You got A Grade !!")
11 else:
12    print ("You failed !!")
```

---

## 多分支结构 4/4

### 修改 2: 隐含排除 (multi\_if2.py)

---

```
1 #python 多分支结构纠错 2
2 mark = int(input("请输入取得的分数: "))
3
4 if mark >= 85:
5     print("You got A Grade !!")
6 elif mark >= 70:
7     print("You got B Grade !!")
8 elif mark >= 60:
9     print("You got C Grade !!")
10 else:
11     print("You failed !!")
```

---

## 1 程序的分支结构

- 单分支结构
- 二分支结构
- 多分支结构
- **分支嵌套**
- 条件判断及组合



## 补充: 分支嵌套 1/2

回顾 < 代码块 > 的概念

代码块中可以嵌套其他分支/顺序/循环结构

简单逻辑组合形成解决问题的复杂逻辑

---

```
1 if <条件1>:  
2     if <条件1.1>:  
3         <语句块1.1>  
4     else:  
5         <语句块1.2>  
6 else:  
7     <语句块else>
```

---

## 补充: 分支嵌套 2/2

代码举例 (nested-if.py):

---

```
1 #python 分支嵌套
2 mark = int(input("请输入取得的分数: "))
3
4 if mark >= 60:
5     if mark >= 85:
6         print ("You got A Grade !!")
7     elif mark >= 70:
8         print ("You got B Grade !!")
9     else:
10        print ("You got C Grade !!")
11 else:
12    print ("You failed !!")
```

---

## 1 程序的分支结构

- 单分支结构
- 二分支结构
- 多分支结构
- 分支嵌套
- 条件判断及组合

## 条件判断的操作符

操作符	数学函数	描述
<	<	小于
<=	≤	小于等于
>	>	大于
>=	≥	大于等于
==	=	等于
!=	≠	不等于

# 条件判断及组合 2/4

数据之间比较大小 vs 字符串之间比较大小 (字典顺序)

代码举例 (str-compare.py)

```
1 # 字符串的比较是按照顺序依次比较字母编码
2 #a 0x31; 1 0x61
3 str1="abc"
4 str2="1234"
5 str3="abc1234"
6
7 if str1 > str2:
8     print ("1")
9 else:
10    print ("2")
11
12 if str1 > str3:
13    print ("3")
14 else:
15    print ("4")
```

## 用于条件组合的三个保留字

操作符及使用	描述
<code>x and y</code>	两个条件 <code>x</code> 和 <code>y</code> 的 <b>逻辑与</b>
<code>x or y</code>	两个条件 <code>x</code> 和 <code>y</code> 的 <b>逻辑或</b>
<code>not x</code>	条件 <code>x</code> 的 <b>逻辑非</b>

# 条件判断及组合 4/4

代码举例 (multi\_if\_combine.py) :

```
1 #python 条件组合
2 mark = int(input("请输入取得的分数: "))
3
4 if not mark < 60:
5     # 下句中的 60 <= mark 可省略
6     if 60 <= mark and mark < 70:
7         print ("You got C Grade !!")
8     elif 70 <= mark and mark < 85:
9         print ("You got B Grade !!")
10    else:
11        print ("You got A Grade !!")
12 else:
13    print ("You failed !!")
```

## 补充: 条件判断之关键字 in

- 某项是否在列表中
- 字符串 A 是否在字符串 B 中

代码举例 (if-in.py)

```
1 #in and not in
2 color = ['Red', 'Blue', 'Green']
3
4 selColor = "Red"
5 if selColor in color:
6     print("Red is in the list")
7 else:
8     print("Red is not in the list")
9
10 if "Yellow" not in color:
11     print("Yellow is not in the list")
12 else:
13     print("Yellow is in the list")
```