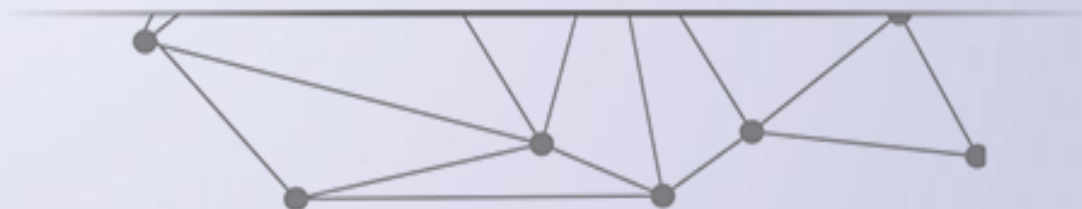




多分支



礼 欣

北京理工大学



多分支决策

```
>>> quadratic3.main()  
This program finds the real solutions to a quadratic  
  
Please enter the coefficients (a, b, c): 4,4,1  
  
The solutions are: -0.5 -0.5  
>>>
```

- 当产生双根结果时，打印两次容易让人误解。



多分支决策

- 要解决双根问题，就需要对 Δ 等于0的情况进行处理。语句的结构上要引入嵌套结构。
 - 当 $\Delta < 0$ ，处理无实根情况；
 - 当 $\Delta = 0$ ，处理实根情况；
 - 当 $\Delta > 0$ ，处理“双根情况”；
- 一种解决方案是在程序中使用两个if-else语句。



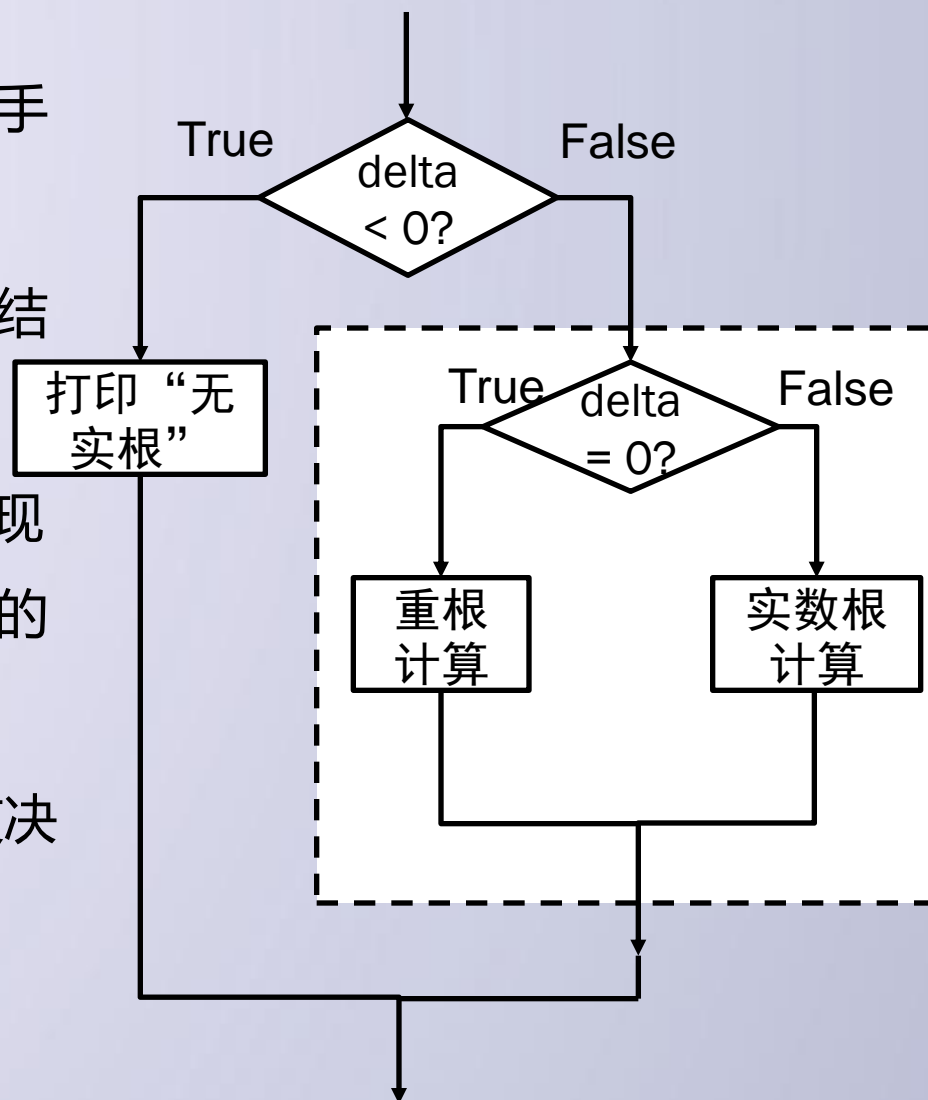
多分支决策

- 把一个复合语句放到另一个语句的结构之中称为嵌套。下面是使用嵌套实现了三分支决策的一个代码片段：

```
if delta < 0:
    print("Equation has no real roots")
else:
    if delta == 0:
        x = -b / (2 * a)
        print("There is a double root at", x)
    else:
        #计算两个实根
```

多分支决策

- 多分支决策是解决复杂问题的重要手段之一
- 一个三分支决策可以由两个二分支结构嵌套实现
- 使用if-else描述多分支决策时，实现更多分支需要更多嵌套，影响程序的易读性
- Python使用if-elif-else描述多分支决策，简化分支结构的嵌套问题





多分支决策

- 使用if-elif-else描述多分支决策：

```
if <condition1>:  
    <case1 statements>  
elif <condition2>:  
    <case2 statements>  
elif <condition3>:  
    <case3 statements>  
...  
else:  
    <default statements>
```

- Python轮流评估每个条件，来寻找条件为True的分支，并执行该分支下的语句；如果没有任何条件成立，else下面的语句被进行，else子句是可选的。



多分支决策

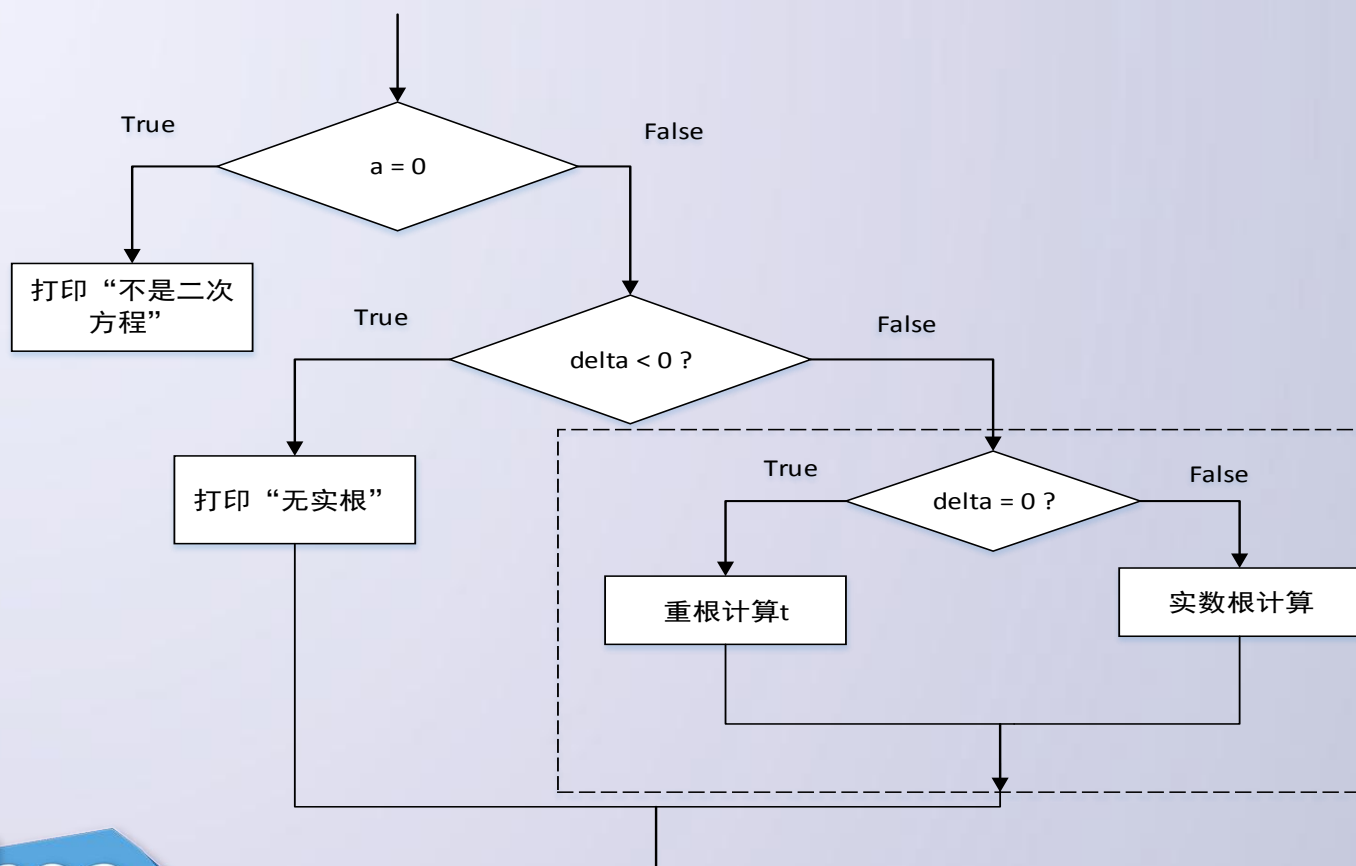
- 当a的输入为0时，运行quad3.py 系统报错

```
>>>
This program finds the real solutions to a quadratic

Please enter the coefficients (a, b, c): 0, 2, 3
Traceback (most recent call last):
  File "C:\Python34\quadratic3.py", line 14, in <module>
    main()
  File "C:\Python34\quadratic3.py", line 11, in main
    root1 = (-b + delta) / (2 * a)
ZeroDivisionError: float division by zero
```



二次方程求解（版本4）



二次方程求解（版本4）

```
# quadratic4.py
import math
def main():
    print("Let us finds the solutions to a quadratic\n")
    a, b, c = eval(input("Do enter the coefficients (a, b, c): "))
    delta = b * b - 4 * a * c
    if a == 0:
        x = -b / c
        print("\nThere is an solution", x)
    elif delta < 0:
        print("\nThe equation has no real roots!")
    elif delta == 0:
        x = -b / (2 * a)
        print("\nThere is a double root at", x)
    else:
        disc = math.sqrt(delta)
        x1 = (-b + disc) / (2 * a)
        x2 = (-b - disc) / (2 * a)
        print("\nThe solutions are:", x1, x2)
```

PM2.5指数分级例子（版本升级）

■ 使用if-elif-else结构改进之前的程序

```
# newpm25.py
def main():
    PM = eval(input("What is today's PM2.5?"))
    # 打印相应提醒
    if PM > 250:
        print("Hazardous. REMAIN INDOORS!")
    elif PM > 150:
        print("Very unhealthy. Avoid prolonged exertion!")
    elif PM > 115:
        print("Unhealthy. Limit prolonged exertion!")
    elif PM > 75:
        print("Unhealthy for sensitive group!")
    elif PM > 35:
        print("Moderate. Go walking!")
    else:
        print("Good. Go running!")
main()
```

PM2.5	空气质量等级
0-35	优
35-75	良
75-115	轻度污染
115-150	中度污染
150-250	重度污染
250-500	严重污染

PM2.5指数分级例子（采用小于条件）

```
# newpm25.py
def main():
    PM = eval(input("What is today's PM2.5?"))
    # 打印相应提醒
    if PM < 35:
        print("Good. Go running!")
    elif PM < 75:
        print("Moderate. Go walking!")
    elif PM < 115:
        print("Unhealthy for sensitive group!")
    elif PM < 150:
        print("Unhealthy. Limit prolonged exertion!")
    elif PM < 250:
        print("Very unhealthy. Avoid prolonged exertion!")
    else:
        print("Hazardous. REMAIN INDOORS!")
main()
```

PM2.5	空气质量等级
0-35	优
35-75	良
75-115	轻度污染
115-150	中度污染
150-250	重度污染
250-500	严重污染



请思考这两个程序是否等价，如果不等价又该如何修改呢？