Lecture 10: Python Programing 2

Objectives

- *The student should learn more about writing more advanced Python code.*
- *The student should learn the precedence rules for writing equations in Python.*
- *The student should learn decisions in a Python program using different forms of if..else statement in Python.*

Simple Decisions

- *Control structures allow us to alter the sequential program flow.*
- *In this Lecture, we'll learn about decision structures, which are statements that allow a program to execute different sequences of instructions for different cases, allowing the program to "choose" an appropriate action.*

If….Statement

- ***The Python `if` statement is used to implement the decision.***

```
if <condition>:
    <body>
```

- ***The body is a sequence of one or more statements indented under the `if` heading.***

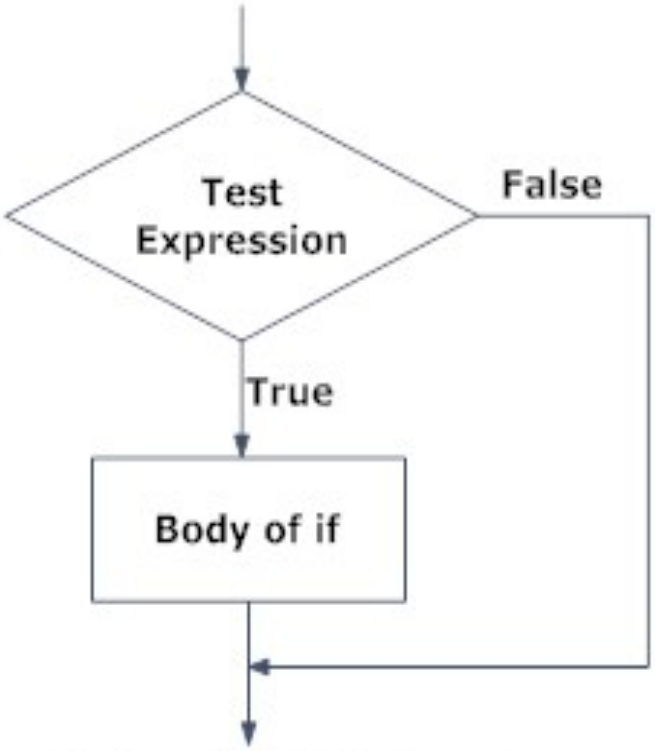


Fig: Operation of if statement

If….Statement

```
num = float(input("Enter a number: "))
if num >0 :
    print(num, "is a positive number.")
print("This is always printed.")
```

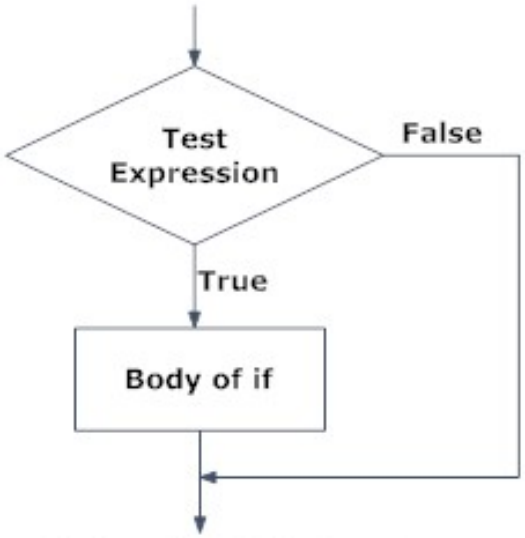


Fig: Operation of if statement

```
Enter a number: 3
3.0 is a positive number.
This is always printed.
```

Example:
Temperature Warnings

- Let's return to our Celsius to Fahrenheit temperature conversion program.

```
# convert.py
# A program to convert Celsius temps to Fahrenheit

celsius = float (input("What is the Celsius temperature? "))
fahrenheit = 9/5 * celsius + 32
print("The temperature is", fahrenheit, "degrees Fahrenheit.")
```

Example:

Temperature Warnings

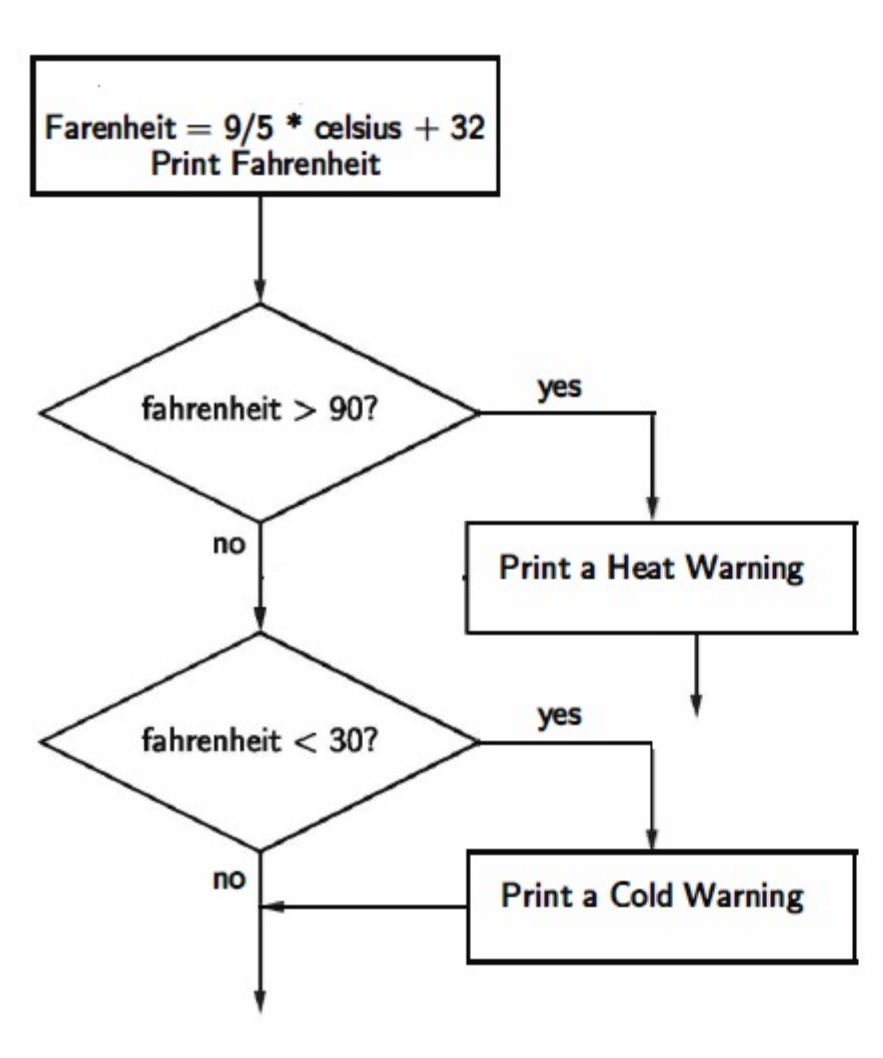
- *Let's say we want to modify that program to print a warning when the weather is extreme.*
- *Any temperature over 90 degrees Fahrenheit and lower than 30 degrees Fahrenheit will cause a hot and cold weather warning, respectively.*

Example:

Temperature Warnings

- Input the temperature in degrees Celsius (call it celsius)
- Calculate fahrenheit as 9/5 celsius + 32
- Output fahrenheit
- If fahrenheit > 90
  print a heat warning
- If fahrenheit < 30
  print a cold warning

Example:
Temperature Warnings



Example:
Temperature Warnings

```
# convert2.py
#       A program to convert Celsius temps to Fahrenheit.
#       This version issues heat and cold warnings.

celsius = float(input("What is the Celsius temperature? "))
fahrenheit = 9 / 5 * celsius + 32
print("The temperature is", fahrenheit, "degrees fahrenheit.")
if fahrenheit > 90:
    print("Heat Warning")
if fahrenheit < 30:
    print("cold warning")
```

Forming Simple Conditions

Python	Mathematics	Meaning
<	<	Less than
<=	≤	Less than or equal to
==	=	Equal to
>=	≥	Greater than or equal to
>	>	Greater than
!=	≠	Not equal to

Two-Way Decisions

- In Python, a two-way decision can be implemented by attaching an `else` clause onto an `if` clause.
- This is called an `if-else` statement:

```
if <condition>:
     <statements>
else:
     <statements>
```

If….else

- The if…..else statement evaluates test expression and will execute body of if only when test condition is True.
- If the condition is False, body of else is executed. Indentation is used to separate the blocks.

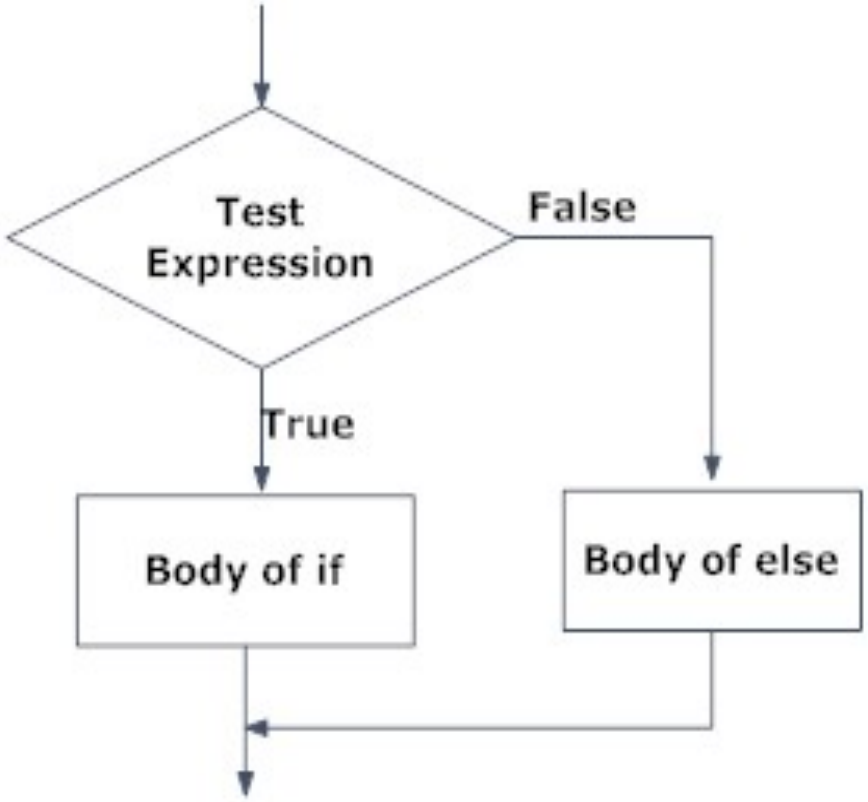


Fig: Operation of if...else statement

If….else

```
num = float(input("Enter a number: "))
if num >= 0:
    print("Positive or Zero")
else:
    print("Negative number")
```

```
Enter a number: -3
Negative number
Enter a number: 3
Positive or Zero
```

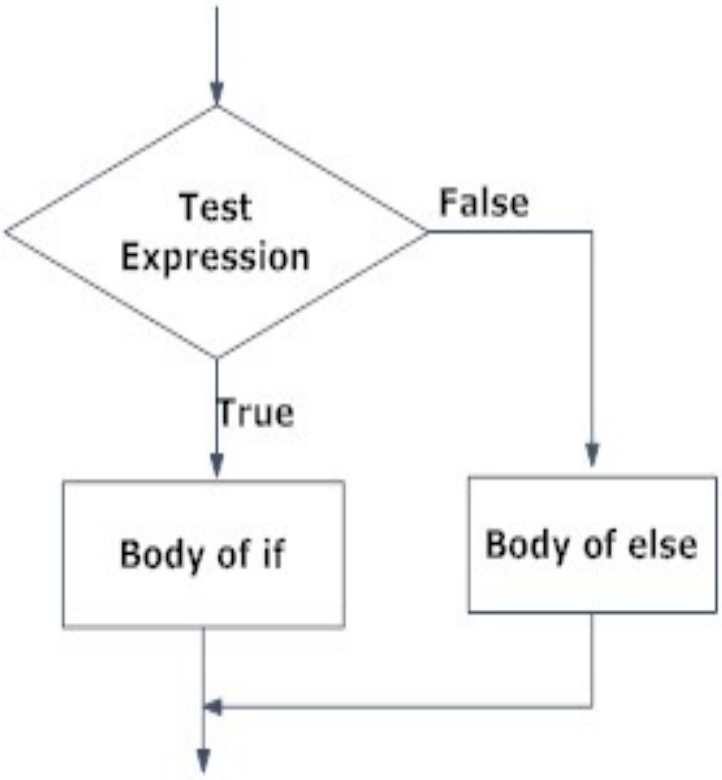


Fig: Operation of if...else statement

If….else

```
# quadratic3.py
#    A program that computes the real roots of a quadratic
  equation.
#    Illustrates use of a two-way decision
import math
print ("This program finds the real solutions to a quadratic\n")
a = float(input("Please enter the coefficients a: "))
b = float(input("Please enter the coefficients b: "))
c = float(input("Please enter the coefficients c: "))
discrim = b * b - 4 * a * c
if discrim < 0:
    print("\nThe equation has no real roots!")
else:
    discRoot = math.sqrt(b * b - 4 * a * c)
    root1 = (-b + discRoot) / (2 * a)
    root2 = (-b - discRoot) / (2 * a)
    print ("\nThe solutions are:", root1, root2 )
```

If....else

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

Please enter the coefficients a: 1
Please enter the coefficients b: 1
Please enter the coefficients c: 2

The equation has no real roots!
>>>
This program finds the real solutions to a quadratic

Please enter the coefficients a: 2
Please enter the coefficients b: 5
Please enter the coefficients c: 2

The solutions are: -0.5 -2.0
```

If….else

- The newest program is great, but it still has some quirks!

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

This program finds the real solutions to a
quadratic


Please enter the coefficients a: 1
Please enter the coefficients b: 2
Please enter the coefficients c: 1


The solutions are: -1.0 -1.0
```

Multi-Way Decisions

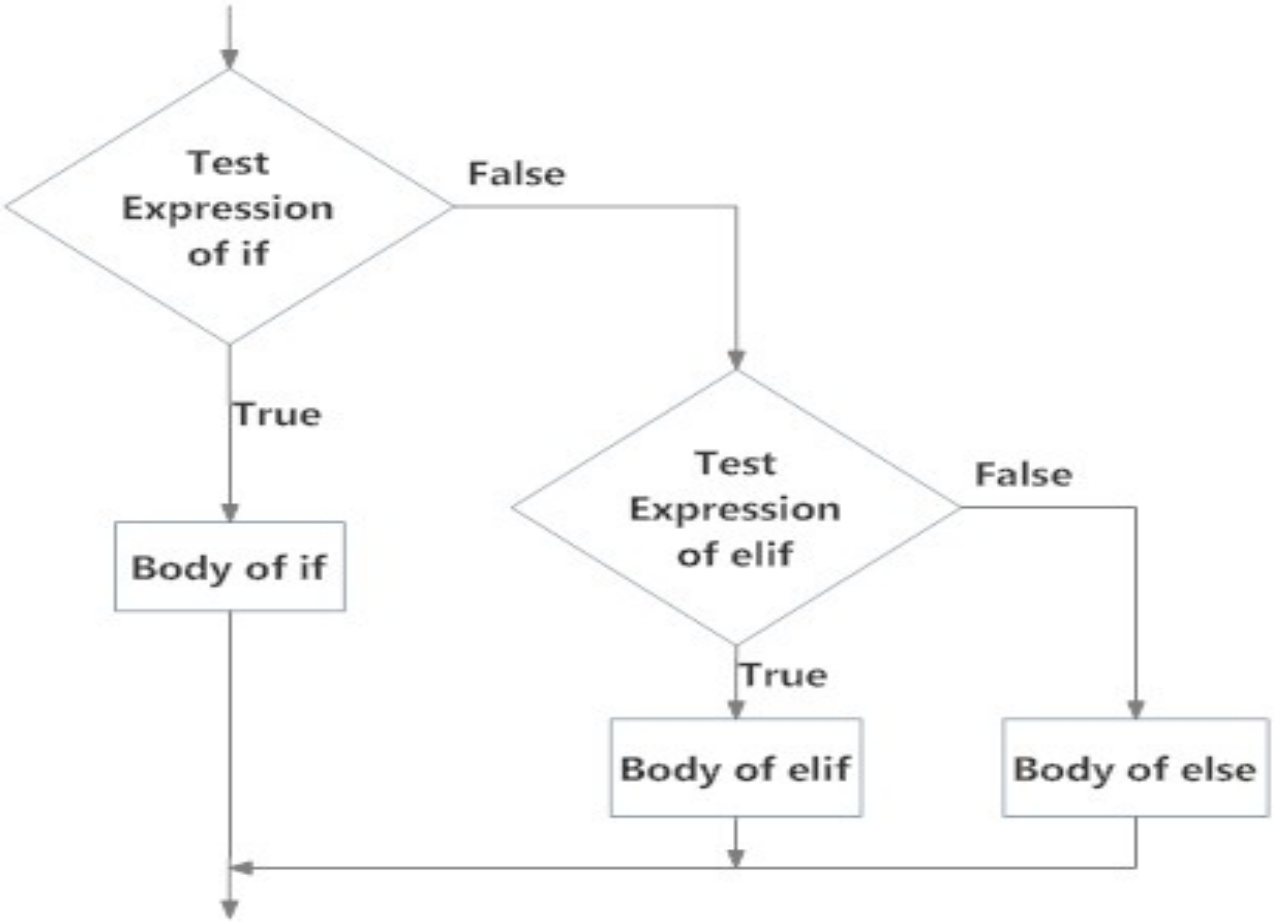
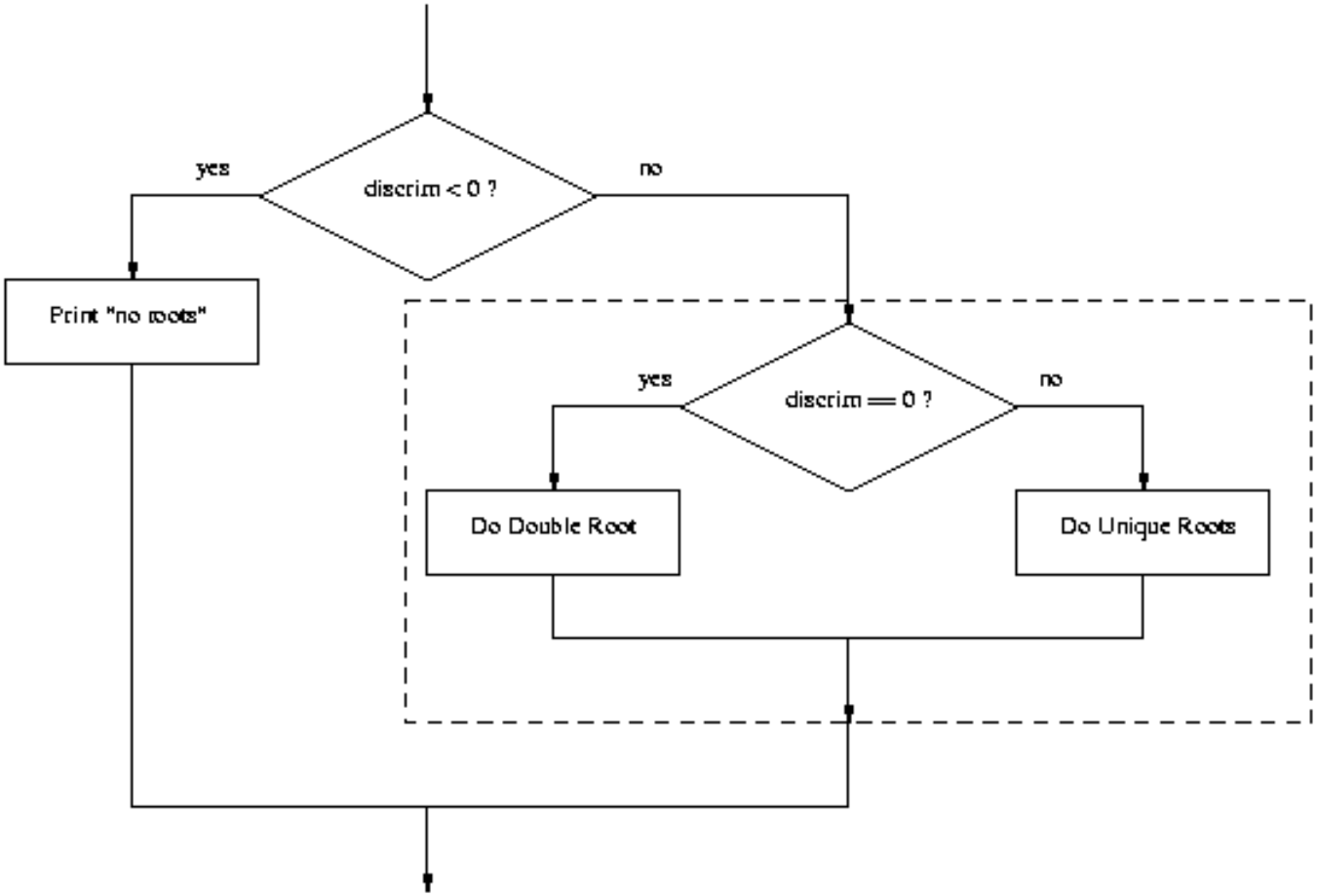


Fig: Operation of if...elif...else statement

Multi-Way Decisions



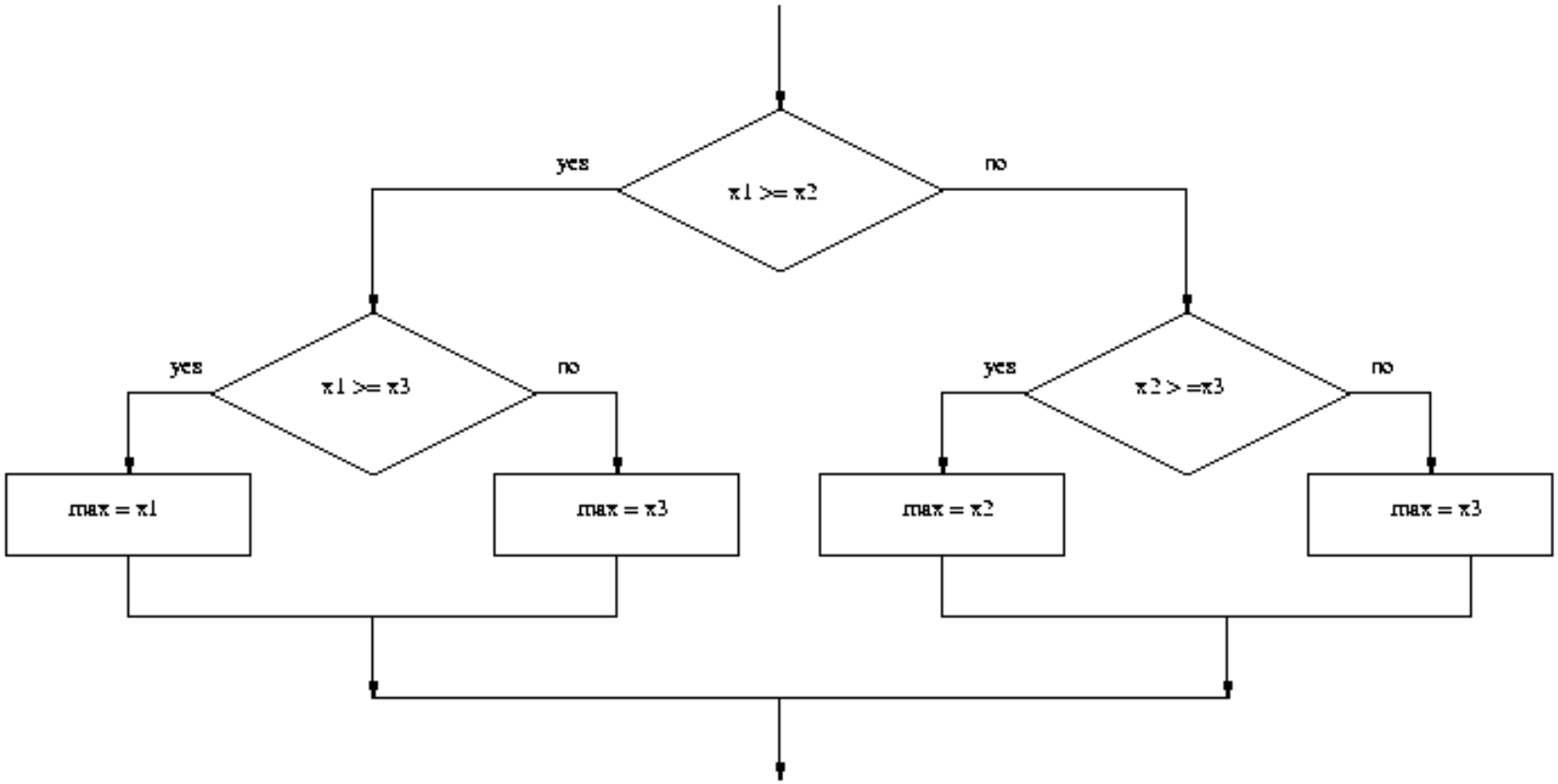
Multi-Way Decisions

- ```
  Check the value of discrim
      when < 0: handle the case of no roots
      when = 0: handle the case of a double root
      when > 0: handle the case of two distinct
                roots
  ```
- We can do this with two if-else statements, one inside the other.
- Putting one compound statement inside of another is called *nesting*.
  ```

Multi-Way Decisions

```
if discrim < 0:
    print("Equation has no real roots")
else:
    if discrim == 0:
        root = -b / (2 * a)
        print("There is a double root at", root)
    else:
        # Do stuff for two roots
```

Multi-Way Decisions

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

Multi-Way Decisions

```
# quadratic4.py
#    Illustrates use of a multi-way decision
print("This program finds the real solutions to a quadratic\n")
a, b, c = eval(input("Please enter the coefficients (a, b, c): "))
discrim = b * b - 4 * a * c
if discrim < 0:
    print("\nThe equation has no real roots!")
elif discrim == 0:
    root = -b / (2 * a)
    print("\nThere is a double root at", root)
else:
    discRoot = math.sqrt(b * b - 4 * a * c)
    root1 = (-b + discRoot) / (2 * a)
    root2 = (-b - discRoot) / (2 * a)
    print("\nThe solutions are:", root1, root2 )
```

Strategy 2: Decision Tree



Study in Design: Max of Three

```
x1, x2, x3 = eval(input("Please enter three values: "))
if x1 >= x2:
    if x1 >= x3:
        max = x1
    else:
        max = x3
else:
    if x2 >= x3:
        max = x2
    else:
        max = x3
print("The largest value is", max)
```