Conditional Statements

CS 8: Introduction to Computer Science, Winter 2018
Lecture #4

Ziad Matni
Dept. of Computer Science, UCSB

Administrative

Turn in your Homework #1 now

- No more adds to this class: list is closed

Lecture Outline

- Reviewing basic functions
- Flow Control: If-Else Statements
- Loops: For Statements

Yellow Band = Class Demonstration! ☺

Recall: Defining Your Own Function

- To define a function in Python, the syntax is:

```
def functionName (list of parameters):
    # a block of statements appear here
    # all of them must be indented (with tabs)
```

Recall: Example Definition

```
# This function calculates the distance between (a,b) and (0,0)
def distance(a, b):
      x = a**2    # Note the tab indent!!!
      y = b**2    # Recall ** means “to the power of”
      z = (x + y) ** 0.5
      return z    # I need to “return” the result
```

!!! Alternatively !!!

```
def distance(a, b):
      return ( (a**2) + (b**2) ) ** 0.5
```

A Function To Draw A Square

- Part of listing 1.2 from the text (p. 30)

```
def drawSquare(myTurtle, sideLength):
    myTurtle.forward(sideLength)
    myTurtle.right(90)    # side 1
    myTurtle.forward(sideLength)
    myTurtle.right(90)    # side 2
    #…etc… do this again for sides 3 and 4
```

- Then to invoke it for drawing a square that has 20 pixels on each side using a turtle named **t**:

```
>>> drawSquare(t, 20)
```

What might happen if we invoked `drawSquare(20, t)`?

Let’s try it out!

Controlling the Flow of a Program

- Programs will often need to make decisions on what to continue doing
  - Like coming to a fork in the road…

- We present the algorithm/program with
  a *conditional statement* (a.k.a *if-then-else*)

If-Else Syntax in Python

- The syntax in Python is:

```
if conditional_statement :
    statement 1
    statement 2
    …
elif:
     else-statements
else:
    default else-statements
```

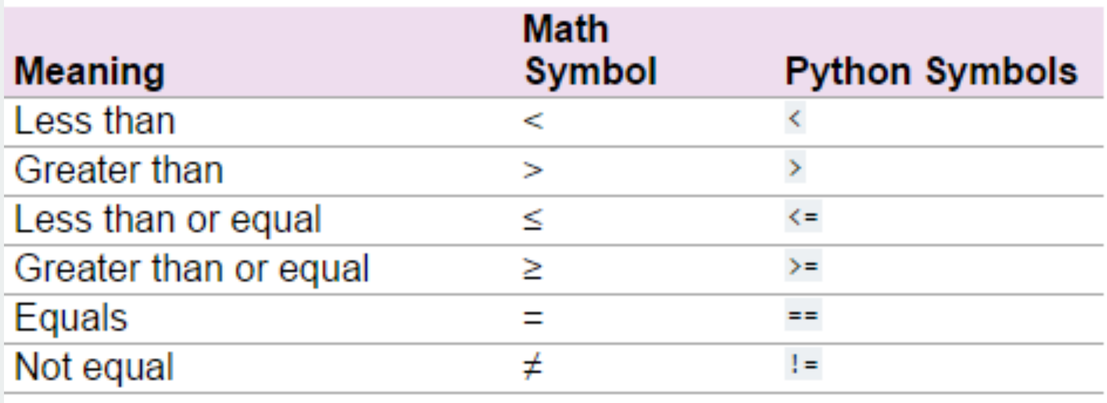
```
a = int(input("Enter a number: "))
# The above line makes the program
# ask the user for a direct input
# into an integer. More on this later.

if (a < 5):
    print("It’s less than five!")

elif (a > 5):
    print("It’s more than five!!!")

else:
    print("It’s equal to five!!!!!")
```

Let’s try it out!

More on Conditional Statements

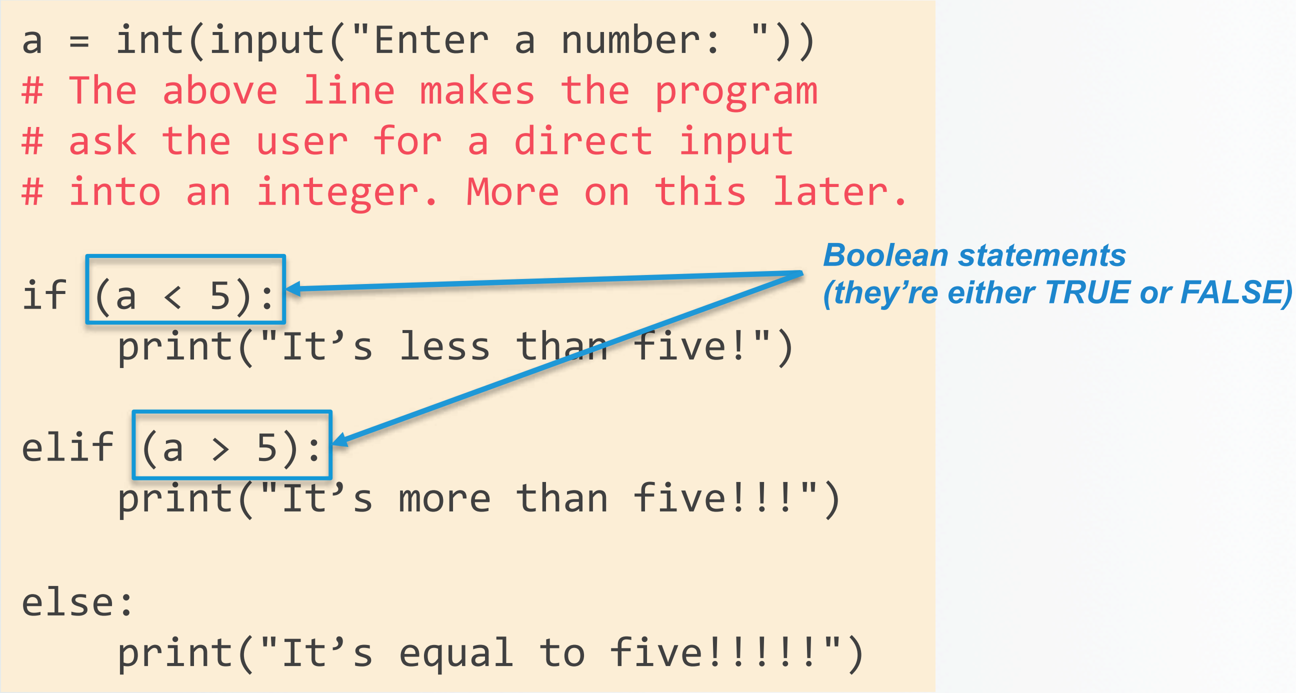
- Conditional statements follow **Boolean logic**
  - That is, they are either TRUE or FALSE

- Often we use comparisons, like “equal to” or “greater than or equal to”
  - Like in math…
  - But the symbols are not *exactly* the same



Meaning	Math Symbol	Python Symbols
Less than	<	<
Greater than	>	>
Less than or equal	≤	<=
Greater than or equal	≥	>=
Equals	=	==
Not equal	≠	!=

Conditional Statements ARE Boolean Values



```
a = int(input("Enter a number: "))
# The above line makes the program
# ask the user for a direct input
# into an integer. More on this later.

if (a < 5):
    print("It’s less than five!")

elif (a > 5):
    print("It’s more than five!!!")

else:
    print("It’s equal to five!!!!!")
```

Boolean statements (they’re either TRUE or FALSE)

Boolean Logic Operators

- Other than **comparison operations**, we can perform **Boolean logic operations**, like AND, OR, or NOT

- Logic AND (and)
  - True if *all* of the conditions are True
- Logic OR (or)
  - True if *any* of the conditions is True
- Logic NOT (not)
  - True if the condition is False
  - False if the condition is True

```
Exercise:
Given that a = 5, b = -5
What is (pay attention to detail):

1.  a > b
2.  a <= b
3.  not(b > a)
4.  not(not(a => b))
5.  (a > b) and (b == a)
6.  (a < b) or (-b == a)
```

```
1.  True
2.  False
3.  True
4. SYNTAX ERR!!!
5.  False
6.  True
```

Let's try it out!

Nested If-Else Statements

Think of If-Else as a way to describe logical branching

What does this do?

```
a = int(input("What is the cost of item X? "))
b = int(input("Enter (0) for not available, (1) for available "))

if (b == 0):
    print("It doesn’t matter what it costs: it’s not available!")
else:
    if (a >= 100):
        print("That’s expensive!")
    else:
        print("That’s not too expensive!")
```

Exercise:

What happens if I enter:

1. 100 for a and 0 for b?
2. 200 for a and 1 for b?
3. 20 for a and 0 for b?
4. 99 for a and 1 for b?

Let’s try it out!

YOUR TO-DOs

- ❑ Start reading **Chapter 3**
- ❑ Start **Homework2** (due next **Monday!**)
- ❑ Prepare for **Lab1** this week

- ❑ Be nice to others

</LECTURE>