CSC 280 Introduction to Computer Science: Programming with Python

Lecture 2 : Data Types, Variables, Statements, Conditionals

Prof. Bei Xiao

Fall, 2014

American University

Review: kinds of primitive data

- Number: integer, floating number
- String: “abc”

- Boolean (discuss later): true, false
- a = true

Combination of number and strings

- What will happen for the following statement:

```
>>> 3*”abc"
```

- What will happen for the following statement

```
>>> 3 + ‘ab'
```

- Example: raw_input_exercise.py

Operations on String

```
>>> 'doesn\'t'
"doesn't"
>>> "doesn't"
“doesn't"
>>> '"Yes," he said.'
'"Yes," he said.'
>>> "\"Yes,\" he said."
'"Yes," he said.’
```

What is the output of the following statement?

```
>>>'"Isn\'t," she said.'
```

Order of Operations

- Rule of Precedence
- e.g What is the output of the following expression:

```
>>> 3+4*5
```

- Question: wow do we get this to give us *35*?
- Precedence: () > exponentiation > *,/ > -,+ s

Python & type checking

- What happens of the following statement?

```
>>> 'a' <3

>>> 4 < “3"
```

- Strong typed vs. weak types language. Python is mediocre in type checking.
- x = "5" + 6 different language assign different values to x.
- Further reading:

http://en.wikipedia.org/wiki/Strong_and_weak_typing

Assignment

- variable = value

```
>>> width = 20
>>> height  = 5*9
>>> width * height
```

- A value can be assigned to several variables simultaneously

```
>>> x = y = z = 0
>>> x
0
>>> y
0
>>> z
0
```

Dynamical Typing

- Python has Dynamic types.

- x = ‘abc’
- x = *3*
- Rules: Don’t change types arbitrarily

The Modulus operator

```
>>> quotient = 7/3
>>> print quotient
2
>>> remainder = 7%3
>>> print remainder
1
```

Floor division

```
>>> 9.0/2.0

4.5

>>> 9.0//2.0

4.0
```

The digits after the decimal points are removed.

Drill

- Shout out answer of the following statements?

```
print 2+3*4
print (2+3)*4
print 2**10
print 2*2**2
print 6/3
print 7/3
print 7//3
print 7%3
print 3/6
print 3//6
print 3%6
print 2**100
print 250.05 % 100
print 250.05 % 10
```

Boolean Operators

- True or false

```
>>> a = 6
>>> b = 7
>>> c = 42
>>> print 1, a == 6
1 True
>>> print 2, a == 7
2 False
>>>
print 3, a == 6 and b == 7
3 True
```

Logical Operators

```
>>> x = 5
>>> x and 1
1
>>> y = 0
>>> y and 1
0
>>>
```

Statements

- legal commands that Python can interpreter
- print, assignment

- Straight line programming: program in which we execute the sequences of instructions one by one.

Branching programs: Conditionals

- An ability of check conditions and change the behavior of the program accordingly, e.g. change ordering of instructions based on some test.

```
if <some test>:
    block of instructions
else:
    block of instructions
```

A word on indentation

- Leading whitespace (spaces and tabs) at the beginning of a logical line is used to compute the indentation level of the line, which in turn is used to determine the grouping of statements.
- Use <tab> to add indent and <del> remove indent of single line of code.
- Example: indentation.py
- We will return to talk more about indentation when we introduce Functions

Code Demos

- Lecture2Handout.py

Review

- What is the difference between straight line programming and branching programming?
- What composes a conditional?

Exercises

- Prompt (using raw_input) to ask how many course credits a student have, if it has received more than 120 credits, print out that they can graduate.
- Write an alternative code of checking whether a number is even or odd. If it is even, print “x is even”, if it is odd, “print x is odd”.

Hint: using remainder.

Exercise: Dream Job

The purpose of this exercise is to understand conditionals. Tiberius is looking for his dream job, but has some restrictions. He loves California and would take a job there if it paid over 40,000 a year. He hates Massachusetts and demands at least 100,000 to work there. Any other place he's content to work for 60,000 a year, unless he can work in space in which case he would work for free. The following code shows his basic strategy for evaluating a job offer.

Exercise: dream job

```
pay = _____
location = _____
if  location == "U.S.S. Enterprise":"
    print "So long, suckers! I’ll take it!"
elif  location == "Massachusetts":
    if  pay < 100000:
         print "No way"
     else:
         print "I’ll take it!"
elif  location == "California" and pay > 40000:
      print "I’ll take it!"
elif  pay > 60000:
      print "I’ll take it!"
else:
     print "No thanks, I can find something better."
```

Quiz: Types

Python automatically infers the type from the value you assign to the variable. Write down the type of the values stored in each of the variables below. Pay special attention to punctuation: values are not always the type they seem!

1. a = False
2. b = 3.7
3. c = ‘Alex’
4. d = 7
5. e = ‘True’
6. f = 17
7 g = ’17’
8. h = True
9. i = ‘3.14.159’

Take-home exercises

- Ask the user to input their names, if it is your name, tell them that is a a nice name, if it is one of the stored name: Albert Einstein, George W Bush, Peter Pan, tell them what you feel about them, if it is not one of the stored names, tell them they have a nice name. Use raw_input and elif