CSC 111: Intro to Computer Science through Programming

Spring 2017
Prof. Sara Mathieson
Admin

+ Resources page is up

+ Please make sure you can access all websites (Moodle, Piazza, course website)

+ If you will be missing lab, please email Dave: dmarshall@smith.edu

+ Add deadline is TODAY! Talk to me ASAP if you are not in the course yet

+ I merged both sections on Moodle (it will look like you didn’t submit things before the merge, but don’t worry, I have your work)
Outline: 2/8

- Recap last time
- Boolean variables
- Conditionals
- Triangle practice
- Lab 2 and HW2 notes
- Indexing into a list
Recap
Recap Monday (great work on the quiz!)

+ Data types: **str**, **int**, **float** (so far)
  + \( x = "7.34" \) ........ type(x)? -> **str**

+ Use **type(...)** to query the type of a variable

+ Python **math** module (square root, trig function, pi)

+ Creating a triangle using the space (" ") character
I have been putting a lot of comments to reinforce concepts

It takes a while to develop intuition about when to comment

For now, a few “rules of thumb”:

1) Comment before each “block” of code (roughly 3-4 lines)
2) Comment before each for loop to explain what it does
3) Don’t need to comment things about Python (i.e. what main does, or why we indent using a tab)
4) Do need to comment things about your particular program
5) If your variable names are well chosen, they can often “self-comment”
# CSC 111, Day 5
# Author: Sara Mathieson and CSC 111 class
# Based on triangle program by Joe O'Rourke
# A program to draw a right triangle

```python
def main():
    n = int(input("Enter n = "))

    # Print one # to start
    print("#")

    # Within the loop, print two #s, one on the left, one spaced over
    for i in range(n):
        print("#" + " "*i + "#")

    # Final horizontal line of #s
    print("#"*(n+2))

main()
```
Booleans and Conditionals
New Type! Called **Boolean**

+ Idea: if something is thing is **true**, we want one thing to happen

+ If something is **false**, we want another thing to happen
New Type! Called **Boolean**

- Idea: if something is thing is **true**, we want one thing to happen
- If something is **false**, we want another thing to happen
- Example:

  **If it is raining:**
  
  I will wear rain boots

  **If it is not raining:**
  
  I will wear sandals
Third control statement: if-statement

- Control statements:
  1) Functions (keyword: `def`, then indent afterwards)
  2) For-loops (keyword: `for`, then indent afterwards)
  3) If-statements (keyword: `if`, then indent afterwards)

- If-statement syntax:

```python
if <condition>:
    <statements>  # Executed if <condition> is True
```
Boolean demo recap

1) Boolean: `type = 'bool'`

2) Comes from George Boole

3) Only two Boolean values: **True, False**

4) Boolean operators: `<, <=, ==, >=, >, !=`

5) Comparing two numbers

6) Comparing two strings

```python
>>> type( 12 == 13 )
<class 'bool'>
>>> 12 == 13
False
>>> type( 12 == 13 )
<class 'bool'>
>>> 12 < 13
True
>>> type( 12 < 13 )
<class 'bool'>
>>> if 12 == 13:
    print( 'Huh???' )

>>> if 12 < 13:
    print( 'This makes sense' )

This makes sense
```

Credit: Joe O’Rourke
Two-way decisions: if/else

+ Example: based on class year, has a student graduated or not?

```python
if <condition>:
    <statements_1> # Executed if <condition> is True
else:
    <statements_2> # Executed if <condition> is False
```
More triangle practice
How can we create a triangle shifted over?

Try to produce this output!
How can we create a triangle shifted over?

Try to produce this output!

- Work with a partner (introduce yourself)
- Create a file `triangle_shift.py`
- Create a loop that just prints the far right `#`
- Modify to add the intermediate `#`
- Fix any spacing issues
# CSC 111, Day 6
# Author: Sara Mathieson and CSC 111 class
# Based on triangle program by Joe O'Rourke
# A program to draw a right triangle (shifted right)

def main():
    n = int(input("Enter n = "))

    # Print one # all the way to the right
    print(" "+(n+1) + "#")

    # Within the loop, print two #s (one shifted and one on the right)
    for i in range(n):
        print(" "+(n-i) + "#" + " "+i + "#")

    # Final horizontal line of #s
    print("#")(n+2))

main()
Lab 2 and Homework 2 Notes
Lab 2 and Homework 2 notes

Round to a certain number of decimal places (try it!)

```python
>>> round(8.456, 1)
>>> round(8.456, 2)
>>> round(8.5)
>>> round(8.55, 1)
```

Q: If we don’t have to specify a parameter, its value is the ____.
Lab 2 and Homework 2 notes

**Q:** If we don’t have to specify a parameter, its value is the default.
Lab 2 and Homework 2 notes

+ Round to a certain number of decimal places (try it!)

```python
>>> round(8.456, 1)
>>> round(8.456, 2)
>>> round(8.5)
>>> round(8.55, 1)
```

Q: If we don’t have to specify a parameter, its value is the **default**.

+ Printing quotes
  1) Use single quotes for a string
  2) Use special (escape character)
Lab 2 and Homework 2 notes

+ Round to a certain number of decimal places (try it!)
  
  ```
  >>> round(8.456, 1)
  >>> round(8.456, 2)
  >>> round(8.5)
  >>> round(8.55, 1)
  ```

Q: If we don’t have to specify a parameter, its value is the default.

+ Printing quotes
  1) Use single quotes for a string  ```
  >>> print('She said, "Hello"')
  ```
  2) Use special (escape character)  ```
  >>> print("She said, "Hello"")
  ```
Lab 2 and Homework 2 notes

+ Round to a certain number of decimal places (try it!)

```python
>>> round(8.456, 1)
>>> round(8.456, 2)
>>> round(8.5)
>>> round(8.55, 1)
```

Q: If we don’t have to specify a parameter, its value is the default.

+ Printing quotes
  1) Use single quotes for a string  >>> print('She said, "Hello"')
  2) Use special (escape character)  >>> print("She said, \"Hello\"")

+ Ctrl-p to repeat a command
Indexing into a list
(IDLE demo)