More Coding Basics with Some Turtle
Announcement

- Homework 2 will be posted this Thursday. Due dates will be adjusted.
- Missing quiz policy: no quiz submissions will be accepted late, except in the case of a documented medical emergency.
- Since we drop a quiz grade, there is NO make-up quiz unless there is a medical emergency.
- If you have situations where you suspect you will miss the quiz regularly (e.g. military service), please discuss with the instructor via writing, supporting documents are required.
- If you do miss a quiz, you will have other opportunities to earn points throughout the class.
Learn to program

= Learn to solve problems
Python

More Coding Basics

Credit: lecture notes modeled after http://www.openbookproject.net/thinkcs/python/english2e/index.html
What is a function?
Syntax

Def FUNCTIONNAME(parameters):

___ STATEMENTS

Indentation (4 empty spaces)

header

body
User-defined Functions
def new_line():
    print
print "The weirdest movie at Sundance this year is:"
new_line()
print "Swiss Army Man."
The weirdest movie at Sundance this year is:
Swiss Army Man.
def new_line():
    print
    print "The weirdest movie at Sundance this year is:"
    new_line()
    new_line()
    new_line()
    print "Swiss Army Man."
The weirdest movie at Sundance this year is:

Swiss Army Man.
print '  *  '
print ' *** '  
print ' ***** '
print '*******'

for h in range(0,5):
    print '   |   '
    print '-------'
def print_house(height):
    print '   *   '
    print '  ***  '
    print ' ***** '
    print '*******'
    for h in range(0,height):
        print '|     |'
        print '-------'
print_house(2)
A Function can be called repeatedly. Function can call another function.
def new_line():
    print

def print_house(height):
    print '   *   '
    print '  ***  '
    print ' ***** '
    print '*******'
    for h in range(0,height):
        print '|     |'
    new_line()
    print '-------'
print_house(2)
Advantages of using a function:
- Group complex computations
- Smaller code for repetitive ops
Flow of Execution

http://www.pythontutor.com/index.html
Forward bottom will show step by step execution of the program
Build-in Functions
<table>
<thead>
<tr>
<th>Function</th>
<th>Function</th>
<th>Function</th>
<th>Function</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>abs()</td>
<td>divmod()</td>
<td>input()</td>
<td>open()</td>
<td>staticmethod()</td>
</tr>
<tr>
<td>all()</td>
<td>enumerate()</td>
<td>int()</td>
<td>ord()</td>
<td>str()</td>
</tr>
<tr>
<td>any()</td>
<td>eval()</td>
<td>isinstance()</td>
<td>pow()</td>
<td>sum()</td>
</tr>
<tr>
<td>basestring()</td>
<td>execfile()</td>
<td>issubclass()</td>
<td>print()</td>
<td>super()</td>
</tr>
<tr>
<td>bin()</td>
<td>file()</td>
<td>iter()</td>
<td>property()</td>
<td>tuple()</td>
</tr>
<tr>
<td>bool()</td>
<td>filter()</td>
<td>len()</td>
<td>range()</td>
<td>type()</td>
</tr>
<tr>
<td>bytearray()</td>
<td>float()</td>
<td>list()</td>
<td>raw_input()</td>
<td>unichr()</td>
</tr>
<tr>
<td>callable()</td>
<td>format()</td>
<td>locals()</td>
<td>reduce()</td>
<td>unicode()</td>
</tr>
<tr>
<td>chr()</td>
<td>frozenset()</td>
<td>long()</td>
<td>reload()</td>
<td>vars()</td>
</tr>
<tr>
<td>classmethod()</td>
<td>getattr()</td>
<td>map()</td>
<td>repr()</td>
<td>xrange()</td>
</tr>
<tr>
<td>cmp()</td>
<td>globals()</td>
<td>max()</td>
<td>reversed()</td>
<td>zip()</td>
</tr>
<tr>
<td>compile()</td>
<td>hasattr()</td>
<td>memoryview()</td>
<td>round()</td>
<td><strong>import</strong>()</td>
</tr>
<tr>
<td>complex()</td>
<td>hash()</td>
<td>min()</td>
<td>set()</td>
<td></td>
</tr>
<tr>
<td>delattr()</td>
<td>help()</td>
<td>next()</td>
<td>setattr()</td>
<td></td>
</tr>
<tr>
<td>dict()</td>
<td>hex()</td>
<td>object()</td>
<td>slice()</td>
<td></td>
</tr>
<tr>
<td>dir()</td>
<td>id()</td>
<td>oct()</td>
<td>sorted()</td>
<td></td>
</tr>
</tbody>
</table>

https://docs.python.org/2/library/functions.html
print abs(5)
print abs(-5)
print pow(2,3)
print max(2,3,8,1)
Functions can be composed.
print abs(max(-1, -2, -5))
print max(abs(-1), abs(-2), abs(-5))
Local variables are only defined inside the function.
def print_house(height):
    print '   *   '  
    print '  ***  ' 
    print ' ***** ' 
    print '*******'
    for h in range(0, height):
        print '     |'
        print '-------'
    print type(height)
print_house(2)
def print_house(height):
    print '   *   '
    print '  ***  '
    print ' ***** '
    print '*******'
    for h in range(0,height):
        print '|     |'
        print '-------'
print_house(2)
print type(height)
Modulus operator: %
remainder = 11%2
print remainder
right_most_digit = 123%10
print right_most_digit
Conditionals
Boolean Values and Expressions
True, False
==, !=, >=, <=, >, <
print type(True)
print 5==5
print 4==5
<type 'bool'>
True
False
x=4
y=5
print x!=y
print x>y
print x<y
print x>=y
print x>=y
print x<=y
Boolean Values and Expressions
x=4
y=5
print (x!=y) and (x>y)
print (x!=y) or (x>=y)
print not(x==y)
If Statement
Syntax

IF BOOLEAN EXPRESSION:

STATEMENTS

Indentation (4 empty spaces)
x = 4
if x>0:
    print "x is positive."
TrumpIsPresident = True
if TrumpIsPresident
    print "What would you do?"
Syntax

if BOOLEAN EXPRESSION:
    STATEMENTS
else:
    STATEMENTS
x = -4
if x>0:
    print "x is positive."
else:
    print "x is not positive."
x = 4
if x%2==0:
    print x, "is even."
else:
    print x, "is odd."
if TrumpBecomesPresident
    print "The republicans win."
else
    print “The democrats might win.”
if BOOLEAN EXPRESSION:
    STATEMENTS
elif BOOLEAN EXPRESSION:
    STATEMENTS
else:
    STATEMENTS
x = 0
if x>0:
    print "x is positive."
elif x<0:
    print "x is negative."
else:
    print "x is zero."
Nested Conditionals
x = 2
y = 3
if x==y:
    print x," equals ", y
else:
    if x<y:
        print x," less than ", y
    else:
        print x," is bigger than ", y
TrumpBecomsePresident=False
HillaryBecomesPresident=False
if TrumpBecomsePresident:
    print "The republicans win."
else:
    if HillaryBecomesPresident:
        print "The Democrats win."
    else:
        print "Some independent wins."
Return Statement
Terminates the execution of a function before it reaches the end
def print_square_root(x):
    if x <= 0:
        print "Invalid input."
        return

    result = x**0.5
    print "The square root of", x, "is", result

print_square_root(-8)
Type Conversion
print int("123")
print int(1.25)
print float(2.5)
print float("2.6")
print str("12345")
Function that Returns
def absolute_value(x):
    if x < 0:
        return -x
    else:
        return x

print absolute_value(-5)

http://www.pythontutor.com/index.html
For Loop
Syntax

for iterating_var in sequence:
      ___STATEMENTS
Python Turtle
import turtle
johnny = turtle.Turtle()
for i in range(0,4):
    johnny.forward(100)
    johnny.right(90)

https://trinket.io/python
import turtle

johnny = turtle.Turtle()

for i in range(0,4):
    johnny.forward(100)
    johnny.right(90)
import turtle
def run_in_square(x, y):
    johnny = turtle.Turtle()
    johnny.hideturtle()
    johnny.setpos(x, y)
    print(johnny.position())
    for i in range(0, 4):
        johnny.forward(10)
        johnny.right(90)
run_in_square(10, 10)
run_in_square(20, 20)
for i in range(1, 5):
    run_in_square(i * 10, i * 10)
import turtle

def draw_polygon(sides, length):
    johnny = turtle.Turtle()
    for i in range(0, sides):
        johnny.forward(length)
        johnny.right(360/sides)

draw_polygon(4, 20)

draw_polygon(6, 20)

Credit: https://www.linuxvoice.com/issues/002/02drawing.pdf
import turtle

def draw_spiral(angle, length_start, length_increase, sides):
    for i in range(0, sides):
        johnny.forward(length_start+(i*length_increase))
        johnny.right(angle)

johnny = turtle.Turtle()
draw_spiral(30, 10, 2, 20)

Credit: https://www.linuxvoice.com/issues/002/02drawing.pdf
Play with Python labs on your own!
THANKS!

Any questions?

You can find me at beiwang@sci.utah.edu

http://www.sci.utah.edu/~beiwang/teaching/cs1060.html
Credits

Special thanks to all the people who made and released these awesome resources for free:

- Presentation template by SlidesCarnival
- Photographs by Unsplash