Python II

“Modulo Operator”

```
Get remainder between two numbers: a % b

     Example:  12 % 6 == 0
               12 % 5 == 2
               10 % 3 == 1
               10 % 4 == ???
```

Lists

```
my_first_list = [1,2,3,4]

one = list[0]
two = list[1]
four = list[3]

print(one)
print(two)
```

- You can add numbers to a list!

```
      my_first_list.append(5)
      print(my_first_list)
```

- Or remove numbers from a list!

```
      my_first_list.remove(3)
      print(my_first_list)
```

Challenge: Prime Numbers

1. Review: what is a prime number?
2. How do we know if a number is prime?

3. Task:
   - Create a function that tells the user if a number is prime or not

Functions within Functions

- We can use functions inside of other functions!
- Example:

```
def add_one(x):
      y = x + 1
      return y

def add_two(a):
      b = add_one(a)
      c = add_one(b)
      return c
```

Challenge: Functions within Functions

Task: Get all prime numbers within a range

- Input: A number greater than 0
- Output: All prime numbers from 1 to that number
- Example:
  - Input = 10
  - Output = 1, 2, 3, 5, 7

How do we approach this?

- Ideas?
- Recall, we already wrote a function that checks if a number is prime!

- Solutions are on the Github website!

Libraries

- Libraries are files of code that other people have already written, that we can use in our own code!
- For example: to get random numbers:

```
import random

x = random.random()      # creates a random float number
y = random.randint()     # creates a random integer number
```

Some basic libraries

- Random
- Math

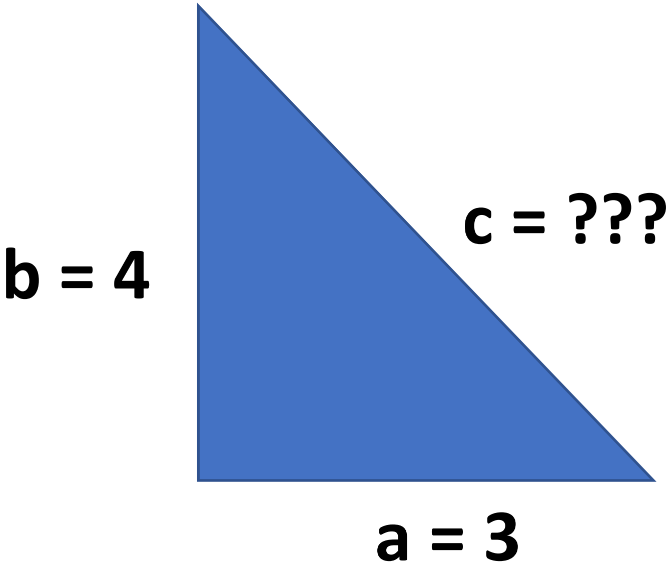
```
import math


a = 16
b = math.sqrt(a)       # b = 4
c= math.pi             # c = 3.14159265359
```

Challenge: Pythagorean Theorem

- Given two sides of a triangle, create a function to calculate the hypotenuse of that triangle

```
def find_hypotenuse(side1, side2):
      pass
```



Solution

```
import math

def find_hypotenuse(a, b):
    x = a**2 + b**2
    c = math.sqrt(x)
    return c
```

Dictionaries

Creates a dictionary, containing key words and their values

```
Grades = {}
Grades[“test1”] = 85
Grades[“test2”] = 97
Grades[“test3”] = 92

print(Grades[“test2])              ← “97”
```

Resources

All class materials:

https://github.com/jlwgong/hangman

Instructor Email:

jlwgong@mit.edu