

## Objective

- Using modules
- Animation

## Review

- What is a benefit of functions?

## Python Libraries

- Beyond built-in functions, Python has a rich **library** of functions and definitions available
  - The library is broken into **modules**
  - A **module** is a file containing Python definitions and statements
- Example modules
  - **math** — math functions
  - **random** — functions for generating random numbers
  - **os** — operating system functions
  - **network** — networking functions

Sept 29, 2017

Sprenkle - CSCI111

3

## math Module

- Defines constants (variables) for **pi** (i.e.,  $\pi$ ) and **e**
  - These values never change, i.e., are **constants**
  - Recall: **we** name constants with all caps
- Defines functions such as

Function	What it Does
<code>ceil(x)</code>	Return the ceiling of X as a float
<code>exp(x)</code>	Return e raised to the power of X
<code>sqrt(x)</code>	Return the square root of X

Sept 29, 2017

Sprenkle - CSCI111

4

## Using Python Libraries

- To use the definitions in a module, you must first **import** the module
  - Example: to use the **math** module's definitions, use the the import statement: **import math**
  - Typically import statements are at **top** of program
- To find out what a module contains, use the **help** function
  - Example within Python interpreter

```
import math
help(math)
```

Sept 29, 2017

Sprenkle - CSCI111

5

## Using Definitions from Modules

- Prepend constant or function with **moduleName**.
  - Examples for constants:
    - **math.pi**
    - **math.e**
  - Examples for functions:
    - **math.sqrt**
- Practice
  - How would we write the expression  $e^{i\pi} + 1$  in Python?

Sept 29, 2017

Sprenkle - CSCI111 [module\\_example.py](#)

6

## Alternative Import Statements

```
from <module> import <defn_name>
```

- Examples:
  - `from math import pi`
    - Means “import pi from the math module”
  - `from math import *`
    - Means “import *everything* from the math module”
- With this **import** statement, don't need to prepend module name before using functions
  - Example: `e**(1j*pi) + 1`

Sept 29, 2017

Sprenkle - CSCI111

7

## Benefits of Using Python Libraries/Modules

- Don't need to rewrite code
- If it's in a module, it is very *efficient* (in terms of computation speed and memory usage)

Sept 29, 2017

Sprenkle - CSCI111

8

## Finding Modules To Use

- How do I know if functionality that I want already exists?
  - Python Library Reference:  
<http://docs.python.org/py3k/library/>
- For the most part, in the beginning you will write most of your code from scratch

## RANDOM MODULE

## random module

- Python provides the **random** module to generate pseudo-random numbers
- Why “pseudo-random”?
  - Generates a list of random numbers and grabs the next one off the list
  - A “seed” is used to initialize the random number generator, which decides which list to use
    - By default, the current time is used as the seed

Sept 29, 2017

Sprenkle - CSCI111

11

## List of Lists of Random Numbers

Seed	List of Random Numbers				
1	0.1343642441	0.8474337369	0.763774619	0.2550690257	...
2	0.9560342719	0.9478274871	0.0565513677	0.0848719952	...
3	0.2379646271	0.5442292253	0.3699551665	0.6039200386	...
4	0.2360480897	0.1031660342	0.3960582426	0.1549722708	...
...			...		...

Sept 29, 2017

Sprenkle - CSCI111

12

## Some **random** Functions

- **random()**
  - Returns the next random floating point number in the range [0.0, 1.0)
- **randint(a, b)**
  - Return a random integer N such that  $a \leq N \leq b$

```
import random

#random.seed(1)      # module.function()

for x in range(10):
    print(random.random())
```

Sept 29, 2017

Sprenkle - CSCI111

**random\_test.py** 13

## VA Lottery: Pick 4

- To play: pick 4 numbers between 0 and 9, inclusive
- To win: select the numbers that are selected by the magic ping-pong ball machine
- Your job: Simulate the magic ping-pong ball machines
  - Display the number on one line

Sept 29, 2017

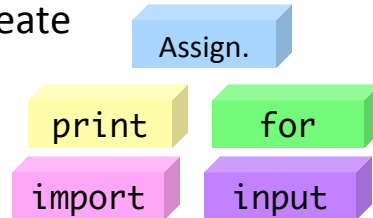
Sprenkle - CSCI111

**pick4.py**

14

## Programming Building Blocks

- Adding to your tool set
- We can combine them to create more complex programs
  - Solutions to problems



## ANIMATION



## Problem: Circle Shift

- Move a circle to the position clicked by the user
  - Repeat five times

Sept 29, 2017

Sprenkle - CSCI111

[circleShift.py](#) 17

## Animation

- Use combinations of the method **move** and the function **sleep**
  - Need to **sleep** so that humans can see the graphics moving
  - Computer would process the **moves** too fast!
- **sleep** is part of the **time** module
  - takes a float representing *seconds* and pauses for that amount of time

Sept 29, 2017

Sprenkle - CSCI111

[animate.py](#)

18

## Problem: Animate Moving to User Click

- Use combinations of the method **move** and the function **sleep**
  - Need to **sleep** so that humans can see the graphics moving
  - Computer would process the **moves** too fast!
- **sleep** is part of the **time** module
  - Takes a **float** parameter representing *seconds* and pauses for that amount of time

## Problem: Animate Moving to User Click

- In X steps, move from the circle's current location to the location clicked by user

## Broader Issues

Angel  
Gabe  
Katlin  
Pengrui  
Rinn

Abhi  
Drew  
Liam  
Tristan  
Turner

Amalia  
Daniel  
JD  
Sam

Chris  
Isaac  
Max  
Prakriti  
Utkrist

Aimee  
Alex  
Annie B.  
Pranam

Sept 29, 2017

Sprenkle - CSCI111

21

## Facebook in the News

- What is all the fuss?
- What is fake news?
  - Have you been tricked by it?
  - How can you/an algorithm recognize it?
- What is legislatable about fake news and ads?
- “Imagine if Facebook were to ask a different question. Instead of asking what someone *wants* to read, it could ask what someone *should* read.”  
<https://www.wired.com/story/facebook-zuckerberg-trump-election/>
- If you were a business, would you advertise on Facebook?

Sept 29, 2017

Sprenkle - CSCI111

22

## Looking Ahead

- Exam next Friday
  - Document posted this weekend about topics and how to prepare