

Lab 2 Feedback

- Getting a little tougher in grading
 - Paying more attention to style (e.g., variable names), efficiency, readability, good output
 - High-level descriptions
 - More strict on adhering to problem specification
 - Constants
 - Demonstrate program **more than once** if gets input from user or outcome changes when run again
 - Find errors before I do!

Feb 2, 2016

Sprenkle - CSC111

1

Testing Discussion

- Consider what inputs could allow you to see different behaviors
- Consider how easily you can validate
- What are good test cases for:
 - Greatest hits problem
 - F → C conversion

Feb 2, 2016

Sprenkle - CSC111

2

Lab 2 Feedback: Common Issues

Which solution is more efficient?

```
i=6
for j in range(1, 9):
    result = i % j
    print( i, "%", j, "=", result)
```

vs

```
for j in range(1, 9):
    i=6
    result = i % j
    print(i, "%", j, "=", result)
```

Feb 2, 2016

Sprenkle - CSC111

3

Lab 2 Feedback: Common Issues

Which solution is more efficient?

```
i=6
for j in range(1, 9):
    result = i % j
    print( i, "%", j, "=", result)
```

vs

```
for j in range(1, 9):
    i=6 ← Additional assignment each time through loop
    result = i % j
    print(i, "%", j, "=", result)
```

Feb 2, 2016

Sprenkle - CSC111

4

Lab 2 Feedback: Common Issues

Which solution is simpler?

```
i=6
for j in range(1, 9):
    result = i % j
    print( i, "%", j, "=", result)
```

vs

```
i=6
j=0
for x in range(8):
    j = x + 1
    result = i % j
    print( i, "%", j, "=", result)
```

Feb 2, 2016

Sprenkle - CSC111

5

Lab 2 Feedback: Common Issues

Which solution is simpler?

```
i=6
for j in range(1, 9):
    result = i % j
    print( i, "%", j, "=", result)
```

vs

```
i=6
j=0
for x in range(8):
    j = x + 1
    result = i % j
    print( i, "%", j, "=", result)
```

More code makes solution more difficult to understand

Feb 2, 2016

Sprenkle - CSC111

6

Recommendations

- Review the slides, example programs, and/or text book every day to review what we discussed
 - This problem made sense in class... Does it still make sense?
- Practice a problem every day
 - I rarely use problems from the text book so they're good practice
- Ask questions
- "sense of accomplishment after lab"

Feb 2, 2016

Sprenkle - CSC111

7

More Hints

- Debugging practices
 - Trace through the program as if you are the computer
 - Similar to some exam problems
 - Use print statements to display variables' values
 - Or, use Python visualizer to show how variables' values change

Feb 2, 2016

Sprenkle - CSC111

8

Review

- How do we call functions from a module?

Feb 2, 2016

Sprenkle - CSC111

9

Review: Object-Oriented Programming

- How do we create a new object?
- How do we give commands to/do operations on objects?
- What is the syntax for calling a method on an object?
- What are two types of methods we talked about?
 - How do they work differently?

Feb 2, 2016

Sprenkle - CSC111

10

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

Feb 2, 2016

Sprenkle - CSC111

circleShiftAnim.py

11

Lab 3 Overview

- Practice Python programming
 - Constants
 - Functions
 - Practice with Graphics API and animation
 - Screen capture using **xv** tool
 - Create a Web page to display artwork

Feb 2, 2016

Sprenkle - CSC111

12