

Objectives

- Review: format specifiers, for loops
- Nested for loops
- Code readability

Sept 24, 2007

Sprenkle - CS111

1

Exam-like Review Questions

- If you want to use functions or constants that are defined in a module, what type of statement do you use?
- In module_example.py, when we computed $e^{i\pi} + 1$, the answer wasn't 0. Why?

Sept 24, 2007

Sprenkle - CS111

2

Formatting Output

- Make the output from the program easy for user to read, understand
- Using str() constructor
- Format specifiers:
 - More formatting options for **print**
 - Control over how output is displayed to user
 - Right, center, left justification
 - Number of decimals to display

Sept 24, 2007

Sprenkle - CS111

3

Solution: using str()

- Recall: str() is **constructor/converter** function to convert other data types to strings
 - Example: str(33) → '33'
- Use when printing output with the + (i.e., concatenation) operator
 - print "You owe \$" + str(x) + "."

Sept 24, 2007

Sprenkle - CS111

4

Format Specifiers

- General format: `%[flags][width][.precision]code`
 - flags:
 - 0: zero fills
 - +: include a sign before positive integers
 - -: left-justification
 - width:
 - Minimum number of character spaces reserved to display the entire value
 - Includes decimal point, digits before and after the decimal point and the sign

The [] mean "optional"

Sept 24, 2007

Sprenkle - CS111

5

Format Specifiers

- General format: `%[flags][width][.precision]code`
 - precision:
 - Number of digits after the decimal point for **real** values
 - code:
 - For the value's **type**
 - s - string
 - d (or i) - integer
 - f - floating point
 - e - floating point with exponent

Sept 24, 2007

Sprenkle - CS111

6

Using Format Specifiers

- Basic format is
 - print <templatestring> % (<value1>, <value2>, ..., <valuen>)
- templatestring is a template for the print statement with format specifiers instead of the values
 - For each format specifier in templatestring, should have a replacement value
 - Throws **TypeError** if not enough replacements for specifiers in templatestring
 - If only one replacement value, don't need ()

Sept 24, 2007

Sprengle - CS111

7

Format Specifiers

print "%5d" % month print "%9.2f" % expense

Field width is 5 Right-justified Field width is 9 Precision is 2

- What if precision is bigger than the decimal places?
 - Fills decimal with 0s
- What if field width is smaller than the length of the value?
 - Prints entire value
- For more info:
 - <http://docs.python.org/lib/typesseq-strings.html>

Sept 24, 2007

Sprengle - CS111

format_specifiers.py 8

Practice

- Format output from xrange_analysis.py nicely

Sept 24, 2007

Sprengle - CS111

9

Practicing For Loops

- Print the following:

1 2 3 4 5

Sept 24, 2007

Sprengle - CS111

10

Practicing For Loops

- Print the following:

1 2 3 4 5
1 2 3 4 5
1 2 3 4 5

Sept 24, 2007

Sprengle - CS111

11

Improving Code Readability

- Comments
 - Don't affect Python's execution
 - Start with a '#' sign
- Constants
 - Change one value (at top of program) to change value everywhere in program
 - Flexible programs
 - Gets rid of "magic numbers"
 - Give a clear name/purpose to values

Sept 24, 2007

Sprengle - CS111

12

Variable Name Conventions

- Variables start with lowercase letter
- Constants (values that won't change) are in all capitals
- Example: Variable for the current year
 - `currentYear`
 - `current_year`
 - ~~`current year`~~
 - `CURRENT_YEAR`

Sept 24, 2007

Sprenkle - CS111

13

Improving Code Readability/Usability

- What does this program do?
 - How would you figure it out?
- What would you do to improve the program's readability and usability?

`program_before.py`
`program_after.py`

Sept 24, 2007

Sprenkle - CS111

14

Fence Post Problem

- Given some posts and some beams to connect the posts, build a fence that is X fenceposts long

Sept 24, 2007

Sprenkle - CS111

`fence_post.py`

15

Assigning Students to Groups

- Using a for loop and the modulo (%) operator

Sept 24, 2007

Sprenkle - CS111

16

TODO

- Read "Why You Can't Cite Wikipedia in My Class" for Friday

Sept 24, 2007

Sprenkle - CS111

17