

Objectives

- Software Qualities
- Problems with print
- DARPA Urban Challenge

Feb 3, 2012

Sprenkle - CSCI111

1

Review

- String operators:
 - How do I concatenate two strings?
 - How can I concatenate a string x times?
- How can we find out the number of characters in a string?
- How can we “get” the last character of a string? (2 ways)
- How can we “get” the first 3 characters of a string?
- What are the two ways to iterate through a string?
- How can we tell if some string comes before another string, alphabetically?

Feb 3, 2012

Sprenkle - CSCI111

2

Review Lottery

Feb 3, 2012

Sprenkle - CSCI111

3

Software Qualities

- Beyond functionality, what qualities do you like in software?
 - Web included

Feb 3, 2012

Sprenkle - CSCI111

4

Software Qualities

- Correct
- Efficient
 - Fast
 - Uses little memory
- Reliability
- Stability
- Robustness
- Secure
- Low cost
- Good support
- Usable 
 - Get what I need to do quickly
 - Pretty
 - ...
- Compatibility
- Portable (use anywhere)
- Maintainability

Feb 3, 2012

Sprenkle - CSCI111

5

Usability

- Want users to *like* to use your software
 - More revenue
 - Develop even better software
- How Apple makes money: best user interfaces → user buys products

Feb 3, 2012

Sprenkle - CSCI111

6

Usability Goals

- Pretty output, display, layout
 - Easy to read, understand, interpret
- Clear navigation
- Easy to perform frequent tasks
- Undoability
- Difficult to make irrecoverable errors



Feb 3, 2012

Sprenkle - CSC1111

7

Escape Sequences

- Escape character: \
- Escape sequences
 - newline character (carriage return) → \n
 - tab → \t
 - quote → \" or \'
 - backslash → \\
- Example:
 - `print("To print a \\, you must use \\\"\\\\\\")`
 - What does this display?

Shell demonstration

Feb 3, 2012

Sprenkle - CSC1111

demo_str.py

8

Practice

- Display To print a tab, you must use '\t'.
- Display I said, "How are you?"

escape_sequence.py

Feb 3, 2012

Sprenkle - CSC1111

9

Problem with print

- By default, `print` puts spaces in the output whenever there is a comma
- Example:

```
x = 13.54
print("You owe $", x, ".")
```

Displays:

You owe \$ 13.54 .

Extra spaces

Feb 3, 2012

Sprenkle - CSC1111

10

Solution: using str()

- Recall: `str()` is constructor/converter function to convert other data types to strings
 - Example: `str(33) → '33'`
- Use constructor with the `+` (i.e., concatenation) operator when printing output
 - Gets rid of extra spaces you don't want.

```
print("You owe $" + str(x) + ".")
```

Feb 3, 2012

Sprenkle - CSC1111

11

Another problem with print

```
SALES_TAX=.05 # the sales tax in VA
value = eval(input("How much does your item cost? "))
with_tax = value * (1+SALES_TAX)
print("Your item that cost $", value, end=' ')
print("costs $", with_tax, "with tax.")
```

Feb 3, 2012

Sprenkle - CSC1111

sales_tax.py

12

FORMATTING STRINGS

Feb 3, 2012 Sprenkle - CSCI111 13

Solution: Format Operator & Specifiers

- Format operator: **%**
- Format specifiers allow us to control how output is displayed to user
 - Right, left justification
 - Number of decimals to display

Feb 3, 2012 Sprenkle - CSCI111 14

Formatting Strings

- Syntax is
 - `<templatestring> % (<value1>, <value2>, ..., <valuen>)`
- Semantics: creates a **formatted string**
 - Means “format the templatestring, using the format(s) specified by **format specifiers** on the corresponding replacement values”
 - Evaluates to a the `str` data type
- Typically used with print statements

Feb 3, 2012 Sprenkle - CSCI111 15

Formatting Strings

- **templatestring** is a template for the resulting string 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 ()

`("%.2f" % 3.14159)` Evaluates to "3.14"

Feb 3, 2012 Sprenkle - CSCI111 16

Example Format Specifiers

`"%5d" % 12`

				1	2
--	--	--	--	---	---

Field width is 5

Right-justified

`"%.2f" % 23.1999`

			2	3	.	2	0
--	--	--	---	---	---	---	---

Precision is 2

Field width is 9

- What if precision is bigger than the decimal places?
- What if field width is smaller than the length of the value?

Any guesses? Try out in interpreter.

Feb 3, 2012 Sprenkle - CSCI111 17

Example Format Specifiers

`"%5d" % 12`

			1	2
--	--	--	---	---

Field width is 5

Right-justified

`"%.2f" % 23.1999`

			2	3	.	2	0
--	--	--	---	---	---	---	---

Precision is 2

Field width is 9

- 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?
 - String contains entire value

Feb 3, 2012 Sprenkle - CSCI111 18

Format Specifiers [] mean optional

- General format: `%[flags][width][.precision]code`
 - flags:
 - 0: zero fills
 - +: adds a + sign before positive values
 - : left-justification (default is right-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

Feb 3, 2012 Sprengle - CSC1111 19

Format Specifiers

- General format: `%[flags][width][.precision]code`
 - precision:
 - Number of digits after the decimal point for **floating point** values
 - code:
 - Indicates the value's **type**/how to format

Code	Type
s	string
d or i	integer
f	float
e	floating point with exponent

Feb 3, 2012 Sprengle - CSC1111 20

Example using Format Operator

```
print("Your item that cost ($%.2f)" % value)
print("costs $%.2f with tax" % tax)
```

Alternative:

```
print("Your item that cost ($%.2f) costs $%.2f with tax" % (value, tax))
```

Feb 3, 2012 Sprengle - CSC1111 21

Exam 1: Next Wednesday

- Preparation document on course schedule page
 - All general topics listed there
- Questions similar to review questions, handouts, lab problems
 - Output from expressions, programs
- All on paper—no computers

Feb 3, 2012 Sprengle - CSC1111 22

Broader Issue: DARPA Urban Challenge

- Challenge: automated cars in an urban setting
 - Deal with human drivers, automated drivers
 - Correctly obey traffic laws
 - Winners: 1st - \$2Mill, 2nd - \$1Mill, 3rd - \$500K
 - Apply for \$1Million in "seed money"

Kari Gabi Josh Cory Jonathan	Sam Koven Mary Emily John K	Will Trang Colby Haley Phil	John G Luke Lijiang Hang Gaurav	Deirdre Connor Drew Shannon
--	---	---	---	--------------------------------------

Feb 3, 2012 Sprengle - CSC1111 23

DARPA Urban Challenge

- Will you feel safe (safer?) with an automated driver in the lane next to you?
- What guarantees about the cars would you want from the company/government?
- Are there situations that would be particularly difficult for software to handle that a person would be better equipped to handle?
- What algorithms might be used in a car? How are they implemented?
- What should the next DARPA Challenge be?
 - In a year?
 - In 5 years?

Feb 3, 2012 Sprengle - CSC1111 24

Relation To Our Class

- One IF statement from victory
 - Programming you're learning applied to major tasks
- Importance of testing
 - Need robustness, reliability of systems
 - Test the "small"
- Likely, infinite loop waiting for signals
- Imagine an API to a Car object
 - `move(direction, amount)`
 - `getSpeed()`
 - ...

Feb 3, 2012

Sprenkle - CSC1111

25

Formatting Practice

- `x = 10`
- `y = 3.5`
- `z = "apple"`
- `"%6.2f" % x`
- `"%6.2d" % x`
- `"%06.2f" % y`
- `"%+6.2f" % y`
- `"%-10s" % z`
- `"%5d %-7.3f" % (x,y)`

Feb 3, 2012

Sprenkle - CSC1111

26

Example: Printing Out Tables

- A table of temperature conversions

Temp F	Temp C	Temp K
-459.7	-273.1	0.0
0.0	-17.8	255.2
32.0	0.0	273.1

- If we want to print data in rows, what is the template for what a row looks like?
 - How do we make the column labels line up?

Feb 3, 2012

Sprenkle - CSC1111

`temp_table.py` 27