Output to the Screen, Variables, and Arithmetic Operators

CS303E: Elements of Computers and Programming

June 8, 2012

Unregistered iClickers

- 22FF13CE

As of Thursday, 6/7, 10a.

If you can't write it down in English, you can't code it.

-- Peter Halpern

Computers Helping People: Hurricanes

- Hurricanes used to be predicted by observation
  - 1940s: prediction entirely based on airplane reconnaissance
- Now using computer modeling (simulation)
  - Global models
  - Can predict intensity (since 90s, not as well as the path)

Output: Print Statement, Part 1

`print` statement:

- Quotes are used to identify the text to be printed
  - double or single
- `print "Hello, World!"`
  - Prints, then goes to the next line
- `print "Hello, World!",`
  - Prints, then prints a space and stays on the same line
  - Comma says "Something else is coming"

Output: Escape Sequences

Two character sequences that represent other characters:

- `\n` : the new line escape sequence (like hitting return)
- `\t` : the tab escape sequence (like hitting tab)
- `\"` : double quote (print " instead of beginning or ending a string)
- `\'` : single quote
- `\\` : backslash

Output: Examples

```
set i to 5   #1
print        #2
print i      #3
print i,     #4
print i,i    #5
print i,i,   #6
print "i"    #7
```

Output: Examples

```
print "Hello, World!"          #1
print "Hello", "World!"        #2
print "Hello,\nWorld!"         #3
print "\"Hello, World!\""      #4
print '"Hello, World!"'        #5
```

Example

- Camus.py

Variables: What are they?

- A word/letter/phrase that represents a value
- Denotes memory location where value is stored
- Examples:
  - `i`, placeholders in our algorithms

Variables: How are they used?

- Gives us the ability to refer to a value without knowing what it is
- Values are *assigned* to variables
  - `i = 5`
- Variables *store* a value
- A variable can hold different types of values (but only one at a time)
  - Python is *untyped*

Variables: What can we name them?

- Variable names are *identifiers*
- Must start with a letter or an underscore (_)
- Can be followed by any number of letters, underscores, digits
- Are case-sensitive (`score` is different from `Score`)
- Must avoid *reserved words*
- Follow naming conventions

Variables: Reserved Words

- Special identifiers that are reserved for a special purpose in Python
  - Such as `def`, `for`, `print`, `main`
  - Turn colors in IDLE

Variables: Naming Conventions

- Choose meaningful names
  - `max` rather than `m`
  - `currentItem` or `current_item` rather than `c`
- Variables should begin with a lowercase letter

iClicker Question: Variable Names

What is a good and legal name for a variable?

A. m
B. Maximum
C. 2max
D. max

Arithmetic Operators: What are they?

- Operators that do arithmetic! (ha.)
- `+, -, *, /, %`
- Perform mathematical operations using:
  - Values referenced by variables (`sum1+sum2`)
  - Numerical literals (`2+3`)
  - Both (`sum1+2`)

Arithmetic Operators: How do they work?

- Precedence same as math:
  - `()`
  - `**`
  - `*, /, %`
  - `+, -`
  - Evaluated from left to right
  - When in doubt, use parentheses
- `%` indicates *remainder* from division

Arithmetic Operators: Example Math

- `15+2`
- `3-1`
- `15%2`
- `25%5`
- `2**3`

- `result = 2*((16-4)/2)`
- `result = 13%2-1`
- `result = 4`
- `result = result + 1`
- `result = 17%3*4`
- `result = result - 6`

Arithmetic Operators: Example Programs

- Write a program that calculates the area of a rectangle.
- Write a program that calculates the average of three test scores.

Testing the Program

- ALWAYS test your code
- Run by hand
- Print statements
  - Check the values of variables to see if they are as expected
  - Check that execution is reaching parts of the code
- Comment out portions

iClicker Question: Arithmetic

What is the final value of `i`?

```
i = 3
i = i + 5
```

A. 3
B. 5
C. 8
D. 12

Reminders

Assignment 1 posted today, due Thursday, 6/14, 11pm