Topics for Today and Wednesday

- Reading for this week...
  Chapter 4: Computing with Strings
- Reminder: Course notes available at the course website
  http://secant.cs.purdue.edu/cs190c:notes
- Taking your files with you from the lab
- Tips for Problem Set 1
- Dealing with numbers and the math library
- Strings
- Lists
- Arrays (NumPy library)

Working Outside the Lab

- Easy way: USB drive
- Map your ITaP shared drive onto CS lab machine
- Map your CS directory onto home machine…
  \verbatim.cs.purdue.edu\homes\YOURLOGINID
- Must be on campus or connected using a VPN
- VPN: Virtual Private Network
  http://www.itap.purdue.edu/connections/vpn/
Clicker Question

In Python, which expression does not equal 8?

A. 2 ^ 3
B. 2 * 4
C. 2 ** 3
D. 4 + 4
E. 10 - 2

Problem Set 1 Tips

- Create helper functions (e.g., fahrenToCelsius)
- Use variables with descriptive unit names (celsiusTemp)
- Use appropriately named conversion variables (metersPerMile)
- Compute sign using modular arithmetic
  \( i \% 2 \) evaluates to 0, 1, 0, 1, ... for \( i \) in range(5)
- Learn by playing
- Don’t try to solve the entire problem on paper first
- Build and test small pieces
- Understand and internalize the tools
- Problems?
  - Start early, take breaks, seek help (TA, office hours).
Computing with Numbers

Two fundamental number representations

- Integer
  - Whole number (+, 0, -)
  - Represented precisely, within limits
  - Integer division truncates (3/2 is 1)
- Floating point
  - Real numbers (i.e., with decimals)
  - Approximate representation, but with high accuracy
  - Small errors can accumulate

```python
for i in range(30000):
    x = x + 1.0/3
```

An Aside on Data Types

- Python tracks both the value and type of data
  - int with value 3
  - str with value “hello”
  - float with value 1.5
  - list with value [0, 2, 5]
- The type of a value affects the way operators behave
  - int + int
  - str + str
- Variables and other “containers” can hold a value of any type

Segue to binary...
Clicker Quiz

Which binary value below represents the decimal number 5?

(Participation points only; no penalty for wrong answer or “E”)

A. 01001
B. 0101
C. 1010
D. 000111
E. I don’t know

A Quick Primer on Binary

• In decimal representation, each digit is a factor times a power of 10…
  \[123 = 1 \times 10^2 + 2 \times 10^1 + 3 \times 10^0\]

• In binary representation, each digit (“bit”) is a factor times a power of two…
  \[101 = 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0\]

• What about negative numbers?
  • Decimal convention: special symbol (+ or -)
  • Binary computer convention…
    • Numbers stored in fixed-length “word” of binary bits (typically 32)
    • High order (“leftmost”) bit indicates sign (0 for +, 1 for -)
Binary Representation

- Integers stored in binary format
- Fixed length: 32 bits used for “standard” integers
  - \(2^{32} = 4,294,967,296\) combinations
    - Range: 0 to 4,294,967,295 (\(2^{32}-1\))
- But, need sign bit to represent + and –; so only 31 bits available
  - Range: 0 to 2,147,483,647 (\(2^{31}-1\))
  - Negative range: -1 to 2,147,483,648 (\(2^{31}\))
- Aside: Sound files use 16-bit integers (Project 1)

A Computer with 3-Bit Words

<table>
<thead>
<tr>
<th>Bit 2</th>
<th>Bit 1</th>
<th>Bit 0</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>= 0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>= 1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>= 2</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>= 3</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>= -4</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>= -3</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>= -2</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>= -1</td>
</tr>
</tbody>
</table>

- Range: 0 to 3 (\(2^{2}-1\))
- Negative range: -1 to -4 (\(2^{2}\))
- Bit 2 (“high order bit”) is the “sign bit”
- Negative values are “2’s complement”: invert bits and add 1
Long Integers

- Python supports integers that use more than 32 bits
- Literal representation uses “L" suffix…
  - 4L
  - 4294967296L
- Longs, like floats, are “contagious” in expressions, for example…
  - int + long has a long result (even if it “fits” in an integer)
  - int + float has float result (even if an exact integer)
  - \(2^{**30}\)
  - \(2^{**31}\)
  - \(2^{**30} - 1 + 2^{**30}\)
  - \(-(2^{**30} - 1 + 2^{**30}) - 1\)

Type Conversions

- Implicit: 3.0/2, 4L+2
- Explicit:
  - float(3)/2
  - float(x)/y
- Other explicit conversions…
  - int(4.5)
  - long(4)
  - str(25)
  - round(4.5)
  - round(-0.5)
Math Library

- Not available in Python by default, must import...
  - import math
  - math.sqrt(12)
- Or...
  - from math import sqrt
  - sqrt(12)

Useful Math Library Functions

- sin(x)
- cos(x)
- tan(x)
- asin(x)
- acos(x)
- atan(x)
- log(x)
- log10(x)
- exp(x)
- ceil(x)
- floor(x)

And constants...
- pi
- e

Demo: ceil, floor, round, ...
Clicker Question

Which value is not equal to the others?
A. ceil(3.5)
B. floor(4.8)
C. round(3.6)
D. ceil(3.1)
E. floor(3.9)

Strings

- Sequence of characters
- Literals delimited in programs by " or single '
  - Use " to include '
  - Use ' to include "
  - Use backslash (\) to “escape” either
    print "Alice said, \"Hello, 'Bob'\"
- Multiline string literals delimited by '''
  x = '''line one
  line two
  last line'''
Input: The Ugly

- \( x = \text{input}(\text{"Enter value: "}) \)
- Value returned by input is evaluated before storing in \( x \)
- \( \text{So, user input must be a valid Python expression (!)} \)
- Use \texttt{raw_input} for string input, which doesn’t evaluate the input
- \( x = \text{raw_input}(\text{"Your name: "}) \)