CS 190C:  
Introduction to Computational Thinking

http://secant.cs.purdue.edu/cs190c:start-s09

Python Programming: An Introduction to Computer Science by John Zelle
A gentle introductory computing text used for a wide range of majors

Learning Python by M. Lutz
A complete reference book with examples; also available on Purdue’s Digital Library

Good on-line reference material is available – see webpage
What is Computer Science?

- It is not the study of computers
  “Computers are to computer science what telescopes are to astronomy.” – E. Dijkstra

- Wikipedia:
  Computer science is the study of the foundations of information and computation and their implementation and application in computer systems.

- Combines science, engineering, and mathematics into a unique and powerful field.

Computer Science versus Programming

- Not every programmer is a computer scientist
- A computer scientist knows the art of programming
- “Computer programming has the same relation to studying computer science as playing an instrument does to studying music or painting does to studying art. In each case, even a small amount of hands-on experience adds immensely to appreciation and understanding.”

  From ACM K–12 Curriculum Committee
What is Computational Thinking?

- Solve problems using computational processes
- Integrate computer science concepts into scientific discovery.
- Use of computer science “mental tools” like abstraction, decomposition, modularization, and recursion to
  - develop solutions and algorithms
  - model scientific processes
  - simulate complex systems

What scientific processes can be simulated/evaluated through computation?

To solve problems
- formulate the problem
- make abstractions and design solutions
- write an algorithm specifying all steps needed to solve the problem
- implement the algorithm in a programming language
Application/Science Part
- Conduct experiments through computational modeling and simulation
- Visualize the process and data generated

Computer Science Part
- Evaluate the impact of algorithms on performance and efficiency
- Argue the correctness of an algorithmic approach
- Understand what all can be computed

Programming Languages
- Programs are expressed in an unambiguous, precise way using a programming language.
- Every structure in programming language has a precise form, called its syntax
  - incorrect syntax → syntax errors
- Every structure in programming language has a precise meaning, called its semantics.
  - incorrect semantics → run-time errors or wrong results
Programming Languages

- High-level languages
  - Python, C, C++, Java, Pearl, etc.
- Low-level language
  - Computer hardware can only understand a very low level language known as *machine language*
  - A program in a high-level language is translated into a low-level language
  - Done by a compiler or an interpreter

- *Interpreters* simulate a computer that understands a high-level language.
- An interpreter analyzes and executes the source code instruction by instruction.
- More on compiling versus interpreting later in the course.
Why Python?

- Python provides a good balance between the practical and the conceptual
- Python allows beginners to start doing interesting things quickly
- Python comes with a large library of modules
- Python serves as an excellent foundation for introducing important computer science concepts

Download Python & VPython: [http://www.vpython.org](http://www.vpython.org)

What is the primary platform you plan to work on outside the lab?

A. Windows XP/Vista
B. Mac OS
C. Linux/Unix
Which of the following best describes your most relevant computing experience?

A. none
B. High School programming course
C. Physics 172
D. other programming experience

The Magic of Python

When you start a Python shell, you will see something like:

Python 2.5.2 (r252:60911, Feb 21 2008, 13:11:45) [MSC v.1310 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.

****************************************************
Personal firewall software may warn about the connection IDLE makes to its subprocess using this computer's internal loopback interface. This connection is not visible on any external interface and no data is sent to or received from the Internet.

IDLE 1.2.2
>>>

Monday, January 12, 2009
The Magic of Python

- The “>>>” is a Python prompt indicating that Python is ready for a command.

```python
>>> print "Hello, world"
Hello, world
>>> print 2+3
5
>>> print "2+3 =", 2+3
2+3 = 5
>>> 
```

Using the Python Shell

```python
>>> (12 + 34.25)/5
9.25
>>> >>> 12+34+56
102
>>> x=12.55
>>> x
12.550000000000001
>>> print x
12.55
>>> 
```
The Magic of Python

- Usually we want to execute several statements together that solve a common problem.
- One way to do this is to use a function.

```python
>>> def hello():
    print "Hello"
    print "Welcome to CS 190C"

>>> 
```

The first line tells Python we are defining a new function called hello.

- The following lines are (automatically) indented to show that they are part of the hello function.
- The blank line (hit enter twice) lets Python know the definition is finished.