

**CS 190C:
Introduction to Computational Thinking**

<http://secant.cs.purdue.edu/cs190c:start>

**Python Programming: An Introduction to
Computer Science**

Zelle's book is a gentle introductory computing text used for a wide range of majors

Python in a Nutshell

A very good and complete reference book

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.

3

What is Computational Thinking?

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

4

Focus: What processes can be analyzed and evaluated through computation?

- Know how to solve problems
 - formulate the problem
 - make abstractions and design solutions
 - write an algorithm by specifying all steps needed to solve the problem
 - Implement the algorithm in a programming language
- Conduct experiments through computational modeling and simulation
- Realize the impact of algorithms on performance and efficiency
- Understand what can be computed and analyzed

5

Programming Languages

- Natural language has ambiguity and imprecision problems when used to describe algorithms.
- Programs are expressed in an unambiguous, precise way using *programming languages*.
- Every structure in programming language has a precise form, called its *syntax*
- Every structure in programming language has a precise meaning, called its *semantics*.

6

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

7

Programming Languages

- *Interpreters* simulate a computer that understands a high-level language.
- The source program is not translated into machine language all at once.
- An interpreter analyzes and executes the source code instruction by instruction.
- More on compiling versus interpreting in next lecture.

8

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>

9

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

- A. Windows XP/Vista**
- B. Mac OS**
- C. Linux/Unix**

CLICK
NOW

10

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

11

CLICK
NOW

The Magic of Python

When you start Python, you will see something like:

```
Python 2.5.1 (r251:54863, Apr 18 2007, 08:51:08) [MSC v.1310 32 bit (Intel)] on win32
```

```
Type "copyright", "credits" or "license()" for more information.
```

```
*****
```

```
Personal firewall software may warn about the connection IDL  
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 1.2.1
```

```
>>>
```

12

The Magic of Python

- The “>>>” is a Python *prompt* indicating that Python is ready for a command.
- ```
>>> print "Hello, world"
Hello, world
>>> print 2+3
5
>>> print "2+3=", 2+3
2+3= 5
>>>
```

13

## 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*.

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

>>>
```

14

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

15

## The Magic of Python

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

```
>>>
```

- Notice that nothing has happened yet!
- We've defined the function, but we haven't told Python to perform the function!
- A function needs to be *invoked*:

```
>>> hello()
Hello
Welcome to CS190C
```

16