Computer Science has become indispensable to scientific inquiry and is permeating science in a transformative manner. The course introduces science majors to computational thinking via basic programming concepts, data and data management concepts, simulation, and visual interaction. The course will use Python and various Python libraries including VPython, NumPy, and NetworkX.

The course is team taught by faculty in Computer Science, Physics, and Chemistry. Lectures are MW 11:30-12:20, 2-hour labs on Friday.

Course Syllabus

Part I: Basic Tools (6 weeks)
Unit 1
   Introduction to Python. Elementary values and data types. Straight line programs, assignments to variables, type conversion, math library.
Unit 2
   Strings, sequences, lists, vectors, arrays. VPython and NumPy. Functions and parameters.
Unit 3
   Conditionals and loop structures. Recursion.
Exam 1

Part II: Introduction to Computational Tools, Methods, and Structures (6 weeks)
Unit 4
   Simulation, Monte Carlo methods, and optimization.
   Introductions to Computational Physics and Computational Chemistry
Unit 5
   Use and role of data structures: trees, traversal and exploration.
Unit 6
Exam 2

Part III: Looking under the hood at computer science (3 weeks)
Unit 7
   Object-oriented design.
Unit 8
Algorithm design, limits of computing, intractability, computability. Future models of computation: DNA computing, quantum computing

**Prerequisites**
General interest in science. No prior programming experience is expected. Calculus background is required (Math 161 or equivalent).

**Course Workload**
- Students will complete 4-5 problem sets (which will include small programming assignments) and 5 larger Python projects focused on science applications.
- Students will also complete small assignments in lab sessions in teams.
- There will be two midterm exams and one comprehensive final exam.

**Text Books**
- Python in a Nutshell, Alex Martelli, O'Reilly, 2006
- Python Tutorial, Guido van Rossum, 2006; [http://docs.python.org/tut/](http://docs.python.org/tut/)

**Course Grading**
- Exam 1 and Exam 2 20% (10% each)
- Final Exam 20%
- Lab assignments 5%
- Problem Sets 15%
- Projects 25%
- Participation 5% (based on clicker responses)

**Course Website**
[http://secant.cs.purdue.edu/cs190c](http://secant.cs.purdue.edu/cs190c)

The course was developed as part an NSF funded project on “Science Education in Computational Thinking (SECANT)” described at [http://secant.cs.purdue.edu/](http://secant.cs.purdue.edu/). Follow up research opportunities exist for interested and qualified students.

For questions, please contact Professor Susanne Hambrusch, seh@cs.purdue.edu.