# Physics 1321: Computational Methods in Physics
## Spring 2013

**Introduction:** Introductory physics courses are full of simplifications: projectiles fly without air resistance, pendulums swing only at small angles, no more than two particles move at any time, etc. These kinds of simplifications are necessary and appropriate when you're first trying to understand the basic laws of nature, but the real world is far more complex, and far more interesting. Fortunately, modern electronic computers make it possible to perform extremely lengthy calculations in a negligible amount of time. In this course we will present several computational techniques and apply them to different problems in physics. Examples will be drawn, as time permits, from several different areas of physics and or astronomy.

After completion of three credits of this seminar students will be able to:

1. Find approximate solutions of typical nonlinear equations using various numerical techniques.

2. Evaluate derivatives to an expected accuracy using finite difference methods.

3. Perform numerical integration using various algorithms.

4. Formulate numerical algorithms for the solution of single and multiple variable ordinary linear and non-linear differential equations in Physics, such as simple harmonic motion, large angular motion of a pendulum, Van de Pol oscillator etc.

5. Write computer programs to implement the formulated numerical algorithms and output the calculated values of selected physical quantities to suitable data files

6. Propose solutions to a problem and to assess the appropriateness of the solutions.

**Lecturer:** Prof. Brian R. D'Urso
E-Mail: dursobr@pitt.edu
113B Old Engineering Hall
Phone: 412-624-2756
Office Hours: Wednesday 12:00 - 2:00 pm in 210 Thaw Hall

**Text:** **A Survey of Computational Physics, Python Multimodal eTextBook**
by Rubin H. Landau, Manuel Jose Paez, Cristian C. Bordeianu.
Note that this is the eTextBook, **not the printed version** (which uses Java).
You can download a pdf of it from:
[http://physics.orst.edu/~rubin/Books/eBookWorking/index.html](http://physics.orst.edu/~rubin/Books/eBookWorking/index.html)
You should read the appropriate chapter in the book **before** the associated class.

Suggested book, especially if you are new to programming:
**A Primer on Scientific Programming with Python**
Hans Petter Langtangen

**Lectures and Labs:**
Monday and Friday 12:00 – 1:15 pm in Room 210 Thaw Hall.
Most classes will begin with an interactive lecture followed by time to work on assignments in the lab.

**Assignments:** The assignment for each week will be available before the week's class meetings**.** You should download the assignment and read it before the associated class meetings. There will be time for you to start the assignment in class. You may discuss the assignment with other students, but the work you turn in must be your own and you must understand everything you submit.

**CourseWeb:** There is a CourseWeb site associated with this course. It can be accessed through your http://my.pitt.edu account. This site has:

- Important announcements.
- Lab hand-outs and additional course materials.
- Grade information.
- A discussion board (good for comments and suggestions).

**Grading:**

- 80% in class assignments / homework.
- 20% final project and in-class presentation

For the final project, you are expected to calculate, using computational techniques discussed in class, properties of a physical system of your own choosing. Deliverables for the project will consist of:

- A paper discussing the physical problem to be examined, the algorithm used to perform the calculation, and the results of the calculation and any conclusions drawn from them.
- An oral presentation to the class summarizing the project.
- Working versions of the computer simulations described in the paper.

**Late Assignments:**

Assignments turned in late will lose points exponentially at a rate of 10% per day. So, the maximum score you can receive is (100%)*(0.90)^(number of days late). The number of days late will always be rounded up to the next integer value.

**Academic Integrity:**

Students in this course will be expected to comply with the University of Pittsburgh's Policy on Academic Integrity. Any student suspected of violating this obligation for any reason during the semester will be required to participate in the procedural process, initiated at the instructor level, as outlined in the University Guidelines on Academic Integrity. This may include, but is not limited to, the confiscation of the examination of any individual suspected of violating University Policy. Furthermore, no student may bring any unauthorized materials to an exam, including dictionaries.

**Disabilities:** If you have a disability that requires special testing accommodations or other classroom modifications, you need to notify both the instructor and the Disability Resources and Services no later than the 2nd week of the term. You may be asked to provide documentation of your disability to determine the appropriateness of accommodations. To notify Disability Resources and Services, call 648-7890 (Voice or TTD) to schedule an appointment. The Office is located in 216 William Pitt Union.

**Approximate outline and schedule for the course (subject to change):**

| **Week** | **Date** | **Topic** |
|---|---|---|
| 1 (Mon.) | 1/7 | installing Python and Pythics, introduction to Python |
| 1 (Fri.) | 1/11 | introduction to Python (basic syntax, math, loops, functions), introduction to Pythics (callback functions) |
| 1 (Sun.) | 1/13 | **Problem Set 1 due** |
| 2 (Mon.) | 1/14 | introduction to numpy (types, arrays), plotting |
| 3 (Fri.) | 1/18 | random numbers, Monte Carlo simulation, random walk |
| 2 (Sun.) | 1/20 | **Problem Set 2 due** |
| 3 (Fri.) | 1/25 | |
| 3 (Sun.) | 1/27 | **Problem Set 3 due** |
| 4 (Mon.) | 1/28 | numerical integration: trapezoidal, Simpson's rule, Monte Carlo |
| 4 (Fri.) | 2/1 | |
| 4 (Sun.) | 2/3 | **Problem Set 4 due** |
| 5 (Mon.) | 2/4 | finite difference, numerical differentiation (first and second derivatives) |
| 5 (Fri.) | 2/8 | solving equations, root finding (bisection, Newton-Raphson) |
| 5 (Sun.) | 2/10 | **Problem Set 5 due** |
| 6 (Mon.) | 2/11 | introduction to Numpy/SciPy, matrix methods, files |
| 6 (Fri.) | 2/13 | |
| 6 (Sun.) | 2/17 | **Problem Set 6 due** |
| 7 (Mon.) | 2/18 | solving ordinary differential equations (ODEs) (Euler’s rule, Runge-Kutta) |
| 7 (Fri.) | 2/22 | |
| 7 (Sun.) | 2/24 | **Problem Set 7 due** |
| 8 (Fri.) | 2/24 | solving systems of ODEs, two-dimensional motion (projectiles, orbits) |
| 8 (Mon.) | 2/27 | |
| 8 (Sun.) | 3/3 | **Problem Set 8 due** |
| 9 (Mon.) | 3/4 | Fourier transforms |
| 9 (Fri.) | 3/8 | |
| | 3/11 | SPRING BREAK |
| 9 (Sun.) | 3/17 | **Problem Set 9 due** |
| 10 (Mon.) | 3/18 | discrete and continuous nonlinear problems |
| 10 (Frid.) | 3/22 | |
| 10 (Sun.) | 3/24 | **Problem Set 10 due** |
| 11 (Mon.) | 3/25 | thermodynamic simulations, Metropolis algorithm |
| 11 (Fri.) | 3/29 | |
| 11 (Sun.) | 3/31 | **Problem Set 11 due** |
| 12 (Mon.) | 4/1 | Solving partial differential equations (PDEs) (electrostatics, heat flow) |
| 12 (Fri.) | 4/5 | |
| 12 (Sun.) | 4/7 | **Problem Set 12 and Final Project descriptions due** |
| 13 (Mon.) | 4/8 | PDEs continued (waves) |
| 13 (Fri.) | 4/12 | final project presentations |
| 13 (Sun.) | 4/14 | **No assignment due** |
| 14 (Mon.) | 4/15 | final project presentations |
| 14 (Fri.) | 4/19 | final project presentations |
| 14 (Sun.) | 4/21 | **Final Projects due** |