SYLLABUS
PHZ 5156, Computational Physics
Fall 2005
Michael D. Johnson

Lecture: PL 101 TR 4:30 – 5:45 PM
Office: CAS 190P Phone: 407-823-3491
Email: mjohnson@ucf.edu

Office hours: Mo 3 –5 PM Tu 9 – 10 AM Th 3 – 4 PM
Final exam: Thursday, December 8, 4:00 – 6:50 PM in PL 101

Required textbooks:

Susan M. Lea, *Mathematics for Physicists*, Brooks-Cole 2004, ISBN 0-534-37997-4.

A.L. Garcia, *Numerical Methods for Physics, 2nd ed.*, Prentice-Hall 2000, ISBN 0-13-906744-2.

Recommended:

W.J. Chun, *Core Python Programming*, Prentice Hall, 2001, ISBN 0-13-026036-3.

I.S. Gradshteyn and M. Ryzhik, ed. by A. Jeffrey and D. Zwillinger, *Table of Integrals, Series, and Products* (6th Edition), Academic Press, 2000, ISBN 0-12-294757-6.

This course introduces computational tools needed in physics. Physicists need both mathematical and numerical skills, and the judgement to decide which is appropriate for a given task. This class's goal is for you to learn these computational tools at the level needed for the graduate core and for research. About 65% of the time will be spent on numerical (computer) methods and the remainder on mathematical (analytical) techniques.

I intend to use the computer language Python for the introduction to numerical techniques. Python is powerful, modern, and free. Python code is similar to C, C++ or Fortran, but is usually simpler. If we do use Python, I recommend that you install it on a computer of your own if at all possible. I will introduce the use of Python, and assignments will be written in Python. However, you may use other computer languages if you prefer. The textbook by Garcia, for example, presents code snippets in both Matlab and C++, and Fortran versions are available on-line.

This course has a decidedly physical orientation. We're far more concerned with calculations than proofs, and with a practical understanding of numerical accuracy rather than formal error analysis. You will solve examples of important physics problems rather than learning generalities.

This is a practical, problem-solving subject, and an important part of the course will be the homework assignments. These will be assigned every week. You will sometimes work in groups, and hand in one solution per group.

I will make every effort to be available during my scheduled office hours, but my administrative duties may occasionally interfere. Please phone ahead to make sure that I'm in my office. You are also very welcome to contact me by email at any time. Usually I can answer quickly. Please talk with one another as well.

Besides being useful, this material is great fun. I hope you have a good time learning it.

Homework solutions:

- Solutions must be neat and organized. Do not hand in rough drafts.
- Include brief comments explaining your math or code. (What does this block of code do? What are you trying to do with the following math?)
- Computer-based assignments should usually include all of the following:
  1. An introduction containing a summary of the formulas your code will calculate and an outline of the planned code.
  2. A printout of the code.
  3. A printout of any output, including graphs. Label any graphs. When outputting values, label them (*e.g.*, '`print "a=",a`' rather than '`print a`').
  4. A conclusion containing any required analysis of your results.

Grades: The final grade will be determined by:

Homework	50%
Mid-Term Exam	25%
Final Exam	25%

I will use plus and minus grades (A, A-,B+,B, ...).

Planned Topics

- Introduction:
  - Resources to know about (GAMS, *Numerical Recipes*, integral tables,...).
  - Introduction to Python. Numerical accuracy, stability.
  - Elementary numerical methods: integration, differentiation, root finding.
- Discrete Quantum Mechanics: vector spaces, operators, matrices, determinants, inverses, linear equations, eigenvalues & eigenvectors. Least squares.
- Orbits and trajectories: Euler, Runge-Kutta, adaptive methods.
- Complex variables: residue theorem
- PDE's of physics: separation of variables, Green's functions.
- Continuum Quantum Mechanics: Sturm-Liouville theory, orthogonal functions.
- Numerical Quantum Mechanics: spectral method, Fourier series, FFT. Data fits.
- Electrostatics: relaxation method, finite elements.
- Quantum and EM wave propagation: implicit schemes.
- Statistical mechanics: Monte Carlo.

Useful Books

M. Abramotwitz & I. Stegun, *Handbook of Mathematical Functions* (the best way to use special functions is to look them up).

W.H. Press *et al.*, *Numerical Recipes: The Art of Scientific Computing.* (Available in Fortran 77, Fortran 90, and C. This is the first place to turn for a real project.)

T. Pang, *An Introduction to Computational Physics* (graduate).

J.M. Thijssen, *Computational Physics* (graduate).

G. Arfken, *Mathematical Methods for Physicists* (advanced undergraduate).

J. Matthews & R.L. Walker, *Mathematical Methods of Physics* (graduate).

R. Courant & D. Hilbert, *Methods of Mathematical Physics* (advanced classic).

P.M. Morse & H. Feschbach, *Methods of Theoretical Physics* (advanced classic).

E.T. Whittaker & G.N. Watson, *A Course of Modern Analysis* (a thing of beauty and a joy forever).

Useful Web Sites

www.physics.ucf.edu/~mdj : my page with class information and some Python links.

www.physics.ucf.edu/Academics/Python/ : some old information on getting Python to work on Windows machines (DISLIN is optional; Visual Python isn't needed for this class).