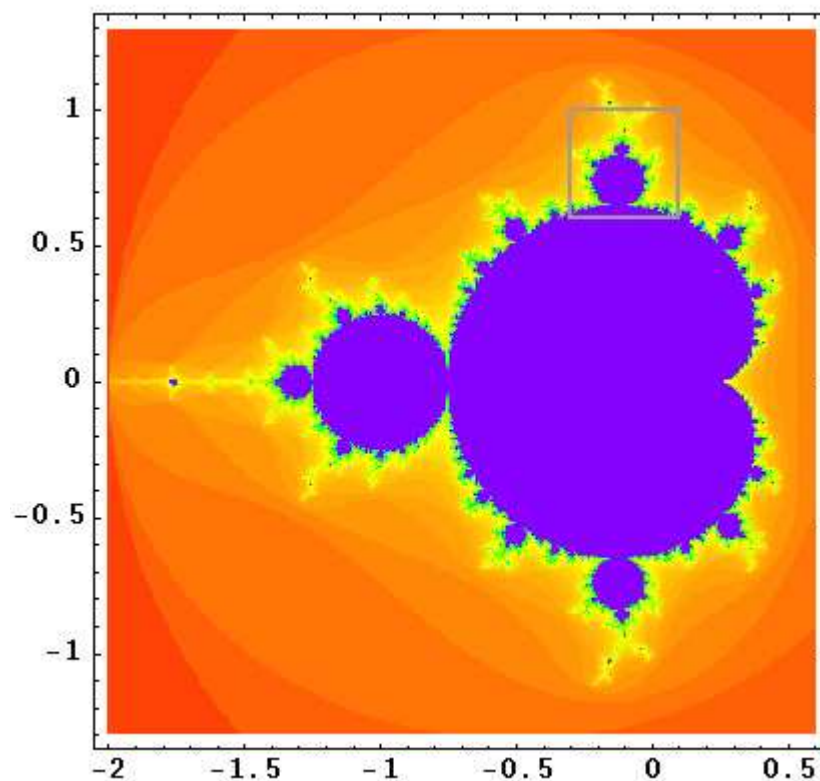


Physics 115/242, Computational Physics



Instructor: Peter Young, ISB 212, Tel: 9-4151

e-mail: peter@physics.ucsc.edu

Time and Place: TTh 12:00-1:45 pm, Nat Sci Annex 102

Office Hour: Will be decided at the first class.

Computational Physics is intended to be of interest to students in other science and engineering departments as well as physics. Two aspects of the course should be particularly noted:

1. In addition to requiring students to write code in one of the standard programming languages, C, C++, or fortran, to study such topics as errors, integration, and solution of differential equations, a substantial part of the course will involve using the powerful features of **MATHEMATICA**, including its graphics capabilities, to study some more advanced topics such as chaos, period doubling, fractals, and quantum mechanics problems with non-trivial potentials.
2. It will also be offered at the **GRADUATE** level as Physics 242. Students taking the course at the graduate level will be required to solve some additional and harder problems, and do some more advanced projects.

Prerequisites

This is NOT a course in programming, and it is important that you can write a simple program in one of the standard languages C, C++, or Fortran. If you are not sure whether you have sufficient fluency in programming, please see me. No previous experience with Mathematica, however, is required; I have prepared a 50 page introductory tutorial which we will go over in class, and which is available [on-line](#). You will also need to have a good undergraduate knowledge of classical mechanics, and also be familiar with basic topics in quantum mechanics, such as Schrödinger's Equation, matrix methods (242 students only), and the simple harmonic oscillator.

If you have trouble with the prerequisites, then either talk to me, or send me an e-mail at peter@physics.ucsc.edu or see me at the end of the first class.

Books

I will not follow any book very closely and will provide a lot of handouts, which will be available on my web site at <http://physics.ucsc.edu/~peter/115>

No books are required. An best (optional) text for the C/fortran part is:

- *Computational Physics: Problem Solving with Computers*, by R. H. Landau and M. J. Paez, Wiley. This has quite a bit of the early material but not some of the later material of the C/fortran part (e.g. Monte Carlo and Molecular Dynamics simulations). There is a web site: <http://www.physics.orst.edu/~rubin/Books/CPbook/index.html> which includes programs in C and fortran.

Other useful books are:

- *Computational Physics* by Tao Pang, Cambridge University Press. Has quite a bit on simulations, but doesn't have the early material such as sources of error.
- *Numerical Methods for Physics* by A. L. Garcia, Prentice Hall. Emphasis on differential equations, ordinary and partial.
- *Computational Physics* by N. J. Giardino and H. Nakanishi, Prentice Hall. Lots of material on differential equations and simulations.
- *Numerical Recipes in C* (also exists in versions for Fortran and C++) by Press et al. Cambridge University Press. This is the "bible" for numerical methods. It is far more thorough and detailed than the material to be covered in the course, but no serious student who use computational methods in science should be without it. There is a web site with all the routines available on-line at <http://www.nr.com>

The best book for the Mathematica part is:

- *A Physicists Guide to Mathematica* by P. T. Tam, Academic Press (2nd Edition). See the author's web site <http://www.humboldt.edu/~ptt1/PTHome.html>.

Other books about Mathematica include:

- *The Mathematica Book* by S. Wolfram, Cambridge University Press. A huge volume written by the

creator of Mathematica.

- *Mathematica for Physicists* by R.L. Zimmermann and F.L. Olness, Addison Wesley. Has a useful concise introduction followed by lots of examples of using Mathematica to solve problems in physics.
- *Mathematica for Scientists and Engineers* by R. Gass, Prentice Hall.
- *A crash course in Mathematica* by S. Kaufmann, Birkhauser. A useful concise introduction.
- *Mathematica for Calculus--Based Physics* by M. de Jong, Addison-Wesley.
- *Mastering Mathematica* by John W. Gray, Academic Press.

There are also two books which combine programming in C with use of Mathematica. Unfortunately, they are not at the right level for the course, in my view, but are useful for consultation.

- *Introduction to Scientific Programming* by J. L. Zachary, Telos. This book is too elementary for the course, but does provide some simple examples, and an introduction to Mathematica.
- *Physics by Computer* by W. Kinzel and G. Reents, Springer. A good source of ideas for numerical problems to work on. Requires a high degree of sophistication and independence from the student.

These books are all available on reserve in the library. It is probably not necessary to buy both a C-based and an Mathematica-based book, and you may wish to discuss with me before buying a book.

Software

The software, *Mathematica For Students*, will be available at the bookstore. The current version is Version 7. The price is quite high (was \$139.95 last year). However, it is not essential that you buy it since there are computer labs on campus with Mathematica. Nonetheless, some students may find it worthwhile to buy Mathematica. However, don't buy it yet; there is the possibility of a much better deal from the company (Wolfram).

Topics

(These may change a little as the course progresses.)

- Representation of numbers on the computer.
- Errors; roundoff and approximation.
- Numerical Differentiation; use of midpoint and error-extrapolation methods to improve accuracy.
- Numerical Integration; trapezoidal rule, Simpson's rule, Romberg integration, treatment of singularities at the endpoints, midpoint rule. Monte Carlo integration.
- Root finding; bisection, secant method, Newton-Raphson, and fixed point iteration.
- Numerical Solution of Ordinary Differential Equations; Euler method, Runge-Kutta. leapfrog, discussion of symplectic algorithms. Application to the Kepler problem. Molecular dynamics simulations.
- Least squares fitting.
- Introduction to sorting algorithms.
- Stochastic (i.e. random) processes; random numbers, random walks, Monte Carlo simulations in statistical physics.
- Introduction to Mathematica, including miscellaneous problems.

- Zeroes of the Riemann zeta function (an example of Mathematica's knowledge of Mathematical functions)
- Projectiles with air resistance.
- Logistic Map--Period Doubling.
- Chaos in differential equations; e.g. transition to chaos in the Duffing equation.
- Fractals--Mandelbrot set.
- Quantum Mechanics; energy levels in quantum wells--coordinate representation.
- Quantum Mechanics; energy levels in quantum wells--matrix formulation.
- Solitons in the sine-Gordon and Korteweg de Vries equations.

Evaluation of Performance

The class will be examined on the basis of homework assignments, two longer term projects, and a take home final.

Peter Young

Sat Mar 21 14:52:24 PST 2009