# PHYS 501
# Mathematical Physics I

*Fall 2011*


| | |
|---|---|
| **Instructor:** | Prof. S. McMillan (12-610, x2723) |
| **Time and place:** | MWF 10:00–10:50, Disque 919 |
| **e-mail:** | `steve (at) physics.drexel.edu` |
| **URL:** | `http://www.physics.drexel.edu/students/courses/physics-501` |


## Course Outline

The goal of this course is to integrate classical analytical methods with modern computational techniques. Emphasis will be placed on application of the methods studied to problems in physics and elsewhere. Topics covered will include:

1. Linear vector spaces and matrices
2. Complex analysis
3. Ordinary differential equations
4. Numerical integration of functions
5. Numerical solution of ODEs
6. Random numbers and Monte-Carlo methods
7. Fourier series
8. Integral transforms
9. Discrete and Fast Fourier transforms

Topics may change or be rearranged at short notice, depending on circumstances.

## Texts

*Essential Mathematical Methods for the Physical Sciences* (K. F. Riley & M. P. Hobson, Cambridge University Press, 2011). Discussion of computational material will be based on *Numerical Recipes in C* (W. Press, S. Teukolsky, W. Vetterling, & B. Flannery 1992, Cambridge University Press; 2nd edition). This book contain a lot of useful explanatory text, along with numerous practical implementations of the algorithms discussed. The C and Fortran versions of this book are available free online, and all programs can be found on newton.

The recommended programming language for the course are C (or C++) or Python. Fortran will probably do (versions of Numerical Recipes exist for Fortran 77 and Fortran 90—see the main course web page); Basic and Pascal will not. Java might be acceptable, but is liable to be rather slow for the problems of interest here. Python benefits from the large number of high-performance modules that have been written for computationally intensive applications. You may find it more convenient to use Maple or Matlab instead for some of the assigned problems (but note that solutions will only be provided in C, and using a canned routine really doesn't count as programming).

## Evaluation

The final grade for the course will be based on (1) a mid-term examination (25% of the total), tentatively scheduled for Monday, October 24 (start of week 6), (2) a final examination (35%), to

be held during finals week, and (3) 7 homework assignments completed during the quarter (40%). Assignments will be due one week after they are distributed. Late homeworks will receive reduced credit, at a rate of -15% per week late. Homeworks turned in after they are discussed in class (about 1 week after they are due) or after the final examination will receive zero credit.