Physics 165: Intro to Programming for the Physical Sciences

MWF 10:00 – 10:50 am, PLS 1129

INSTRUCTOR: Professor Michelle Girvan

Email: girvan@umd.edu (Please include 'phys165' in the subject of all emails)
Phone: x5-1610
Office Hours: Wed 2:30 – 3:30 pm, or by appointment, A.V. Williams 3327
Course website:
Our course website is accessible through http://www.elms.umd.edu

COURSE DESCRIPTION:

Introduction to programming using examples in the physical sciences.

COURSE SOFTWARE:

MATLAB: We will be using Matlab for this class. Problem sets will require that you have access to a computer with Matlab capability. You may want to purchase a student edition of Matlab or simply use the computer labs on campus. Alternatively, you can use the university's virtual computer lab to use Matlab remotely. This allows you to run Matlab without having it installed on your local computer. For information on using the virtual computer lab go to the following page:
http://www.it.umd.edu/vcl
Also note that many of the problem sets can be done using FreeMat, a free software package designed to emulate Matlab. FreeMat can be downloaded at:
http://freemat.sourceforge.net/download.html

REQUIRED TEXTBOOK:

Matlab: A Practical Introduction to Programming and Problem Solving, S. Attaway
Note: While many of the topics covered in class are also presented in the textbook, some topics addressed in the lectures are not included in the textbook. You are responsible for all material covered in class and all assigned textbook readings.

ADDITIONAL REFERENCES:

- *Essential Matlab for Engineers and Scientists*, Hahn and Valentine, Third Edition
- *Graphics and GUIS with Matlab,* Marchand.
- *Guide to Matlab Object-Oriented Programming*, Register
- *An Introduction to Computer Simulation Methods,* H. Gould and J. Tobochnik
- *Physical Modeling in Matlab*, A. Downey. Can be purchased online or downloaded for free at:
  http://www.greenteapress.com/matlab/

EVALUATIONS:

1. Problem sets: about 6-8
2. In class exams: Wednesday February 29 (tentative), and Wednesday, April 18 (tentative)
3. Programming Project: Due Monday, May 7. In class presentations Monday, May 7 and Wednesday, May 9.
4. Final Exam: Friday, May 18, 8-10am.

GRADING:

Problem sets: 30%
In class exams (2): 15% each
Programming project: 20%
Final Exam: 20%
Participation and attendance will count toward borderline grade cases.

POLICY FOR LATE HOMEWORK SUBMISSIONS:

- Submissions received on time will be scored at the 100% level.
- Submissions received after the deadline, but with 24 hours thereof, will be scored at the 75% level. (*i.e.* you can receive at most 75% of the total possible points.)
- Between 1 & 2 days late: 60%
- Between 2 & 3 days late: 45%
- Between 3 & 4 days late: 30%
- Between 4 & 5 days late: 15%
- No submissions accepted after 5 days.
- Exceptions can be granted for serious situations, documentation necessary.
- The entire problem set will be scored at a single level as indicated above. *i.e.,* you can't submit part of it to be scored at the 100% level and the rest at the 75% level.

ACADEMIC HONESTY:

Working together on assignments is encouraged. However, each student is expected to do the assigned problems and write the assigned programs independently, and hand in his or her own work for grading. If you work with other students on a problem set, you must list their names on the first sheet of your submitted solutions. Examinations are to be worked completely independently.

MATERIAL (*Approximate dates indicated in italics*):

Basic concepts for scientific programming (~4 weeks)

- Intro to Matlab, variables: scalars and arrays (Chapter 1) – *W 1/25, F 1/27*
- Simple Matlab scripts and functions (Chapter 2) – *M 1/30, W 2/1*
- If statements, for and while loops (Chapters 3 and 4) – *F 2/3, M 2/6, W 2/8, F 2/10, M 2/13*
- Logical variables and masks – *W 2/15, F 2/17*
- Errors, pitfalls, and debugging (Chapter 5) – *M 2/20, W 2/22*

Application: Dynamical systems, chaos and fractals (~1 week)
F 2/24, M 2/27, F 3/2

Numerical methods for the physical sciences: Part I (~1 week)

- Root finding – *M 3/5, W 3/7*
- Curve fitting (Chapter 14) – *F 3/9, M 3/12*

More advanced programming concepts (~2 weeks)

- Strings (Chapter 6) – *W 3/14, F 3/16*
- Recursion (Chapter 9) – *M 3/26, W 3/28*
- Advanced data structures (Chapter 7) – *F 3/30, M 4/2*
- Basic statistics, searching and sorting (Chapter 12) –*W 4/4*

Monte Carlo simulations (~2 weeks)

- Monte Carlo integration – *F 4/6*
- Computational studies of error propagation – *M 4/9, W 4/11*
- Simple stochastic simulations: random walks, levy flights, physically inspired cellular automata
  F 4/13, M 4/16, F 4/20

Numerical methods, Part II: Integration and ODEs (Chapter 14) (~2 weeks)
M 4/23, W 4/25, F 4/27, M 4/30, W 5/2, F 5/4