SYLLABUS

PHY410/PHY505: Computational Physics 1

Hours: **MWF 2-2:50 PM**
Classroom: **TBD**
Instructor: Dr. Salvatore (Sal) Rappoccio
Office: 335 Fronczak
Phone: 645-6250
E-mail: srrappoc@buffalo.edu
Office Hours: Wed 3-5, and by appointment

This course is the first in a sequence of two courses in Computational Physics that integrates numerical analysis and computer programming in C++ and python (and their combination), to study a variety of problems in physics. An introduction to technicalities of scientific programming (including git, containers like docker, pip, etc), the basics of numerical computation, and a review of numerical best programming practices in C++ and python will be covered for several weeks in the beginning of the course. The course will then cover numerical algorithms for root finding, interpolation, matrix inversion, numerical differentiation, and quadrature, data analysis, Fourier transformations, linear and nonlinear differential equations, boundary-value and eigenvalue problems. The computational content of the course will be organized in the following topics: (0) Technicalities and Basics of Numeric Computing, (1) Data Analysis, (2) Basic Numerical Algorithms, (3) Linear Algebra, (4) Solving Nonlinear Equations, (5) Ordinary Differential Equations.

PREREQUISITES AND BASIC RESOURCES:
This course assumes familiarity with undergraduate physics at the junior/senior level. You should have passed PHY 301, PHY 401, and PHY 403, or equivalent courses, or be taking them concurrently. If you are not a physics major, a strong background in undergraduate mathematics or computer science should suffice if you spend extra time to learn the physics background required for each topic, although you should be familiar with ordinary and partial differential equations at the very least.

Familiarity with a modern programming language is required (C++/Java/Fortran/`python`/etc). Programming mainly with C++ and `python` will be covered in the first 4-8 weeks of lecture. If you are not familiar with C++ or `python` you should spend extra time very early in the course to bring yourself up to speed. Depending on experiences of the class, we will spend more or less time on introductions to programming. We will discuss how to compile and execute your code on your chosen platform. For instance, it will be helpful to have familiarity with `bash`, `tcsh`, or `zsh` for Linux/Unix/Macintosh, or `cygwin` for Windows. We will discuss how to combine C++ and `python` with existing tools such as SWIG.

REQUIRED MATERIALS:
There will be two supported platforms for the course. The first will be the `vidia` platform sponsored by UB's Center For Computational Physics (CCR). There will also be a docker container that is maintained. However, if you have a personal laptop, this may be used instead. All required software for this course can be downloaded for free. **There will be no class time devoted to configuration of private laptop software computing environments.**

The required textbooks are required (and free of charge). You are expected to have working knowledge of things covered in these books.

- Fundamentals of C++ Programming by Richard Halterman
    - Example code at https://github.com/halterman/CppBook-SourceCode
- https://www.tutorialspoint.com/python3/ : Introduction to python
- Numerical Recipes in C++ :
    - The latest version **does cost money** but is a worthwhile investment for your career, while **older versions of NR are free.**
    - Earlier online version of NR for free

The following are also helpful resources:

- http://www.physics.buffalo.edu/phy410-505/ Previous years' course site
- Programming - Principles and Practice Using C++ by Stroustrup
- http://www.python.org Python programming language official website
- http://www.swig.org : SWIG for combining C++ and python
- Numerical Methods for Physics by Alexander Garcia

The course website is at UBLearns :

- http://ublearns.buffalo.edu/ UBLearns course site

You will also be required to use the "piazza" software (free of charge):

- https://piazza.com/class/jl3tpcrqvde2pe

Editors :

- http://www.gnu.org/software/emacs/ : emacs
- http://www.vim.org : VIM
- https://developer.apple.com/xcode/ : XCode

Version Control Software :

- http://github.com : git

Containers:

- https://www.docker.com: docker

SCHEDULE:
The course is scheduled MWF 2-2:50 PM. Homework will be regularly assigned (~weekly). There is a take-home midterm and final exam.

EXPECTATION

To succeed in this course you should read the lecture notes and posted materials, attend class and participate actively in discussion and quizzes, complete the homework assignments on time, and take the midterm and final exams. Exceptions will be made for documented medical reasons or major emergencies.

If you are having difficulty with the course material, it is best to be proactive and contact me directly, either in office hours or by appointment. Discussing difficulty beforehand is encouraged, but asking for special consideration after the fact is not usually helpful.

GRADING:

Grades will be based on your scores on homework (50%), one in-class midterm (25%), and a take-home final exam (25%). Graduate students and undergraduates will be graded separately.
The lowest homework score will be dropped from consideration to accommodate personal situations such as illnesses or missed classes.

MIDTERM: Mid semester (take home).
FINAL: Last week of classes (take home).

ACADEMIC INTEGRITY

Academic integrity is a core value underlying all scholarly activity in the Department of Physics. Please review UB undergraduate policy at http://undergrad-catalog.buffalo.edu/policies/course/integrity.shtml or graduate policy in http://www.grad.buffalo.edu/policies/academic_integrity.pdf. You are encouraged to discuss class material and assignments with your colleagues (with acknowledgment of who you worked with on your assignment). However, you should code and run your simulations yourself, and your homework writeup must be entirely your own effort. If you copy and/or modify code from any source for your assignments you should acknowledge this with an appropriate citation in your writeup.

STUDENTS WITH DISABILITIES

If you have a disability, (physical or psychological) and require reasonable accommodations to enable you to participate in this course, such as note takers, readers, or extended time on exams and assignments, please contact the Office of Disability Services, 25 Capen Hall, 645-2608, http://www.student-affairs.buffalo.edu/ods/, and also see me me during the first two weeks of class. ODS will provide you with information and review appropriate arrangements for reasonable accommodations.

Learning Outcomes

TOPIC UNITS	LEARNING OUTCOMES	OUTCOME ASSESSMENT
Programming and Technical Computing	Introduction to UNIX environment, git, docker, compilation, programming in C++ and python, swig, debugging.[U:3][G:3]	Homework, midterm, exam
Data analysis	plotting, data fitting, analyzing large datasets, shell scripts and compilation [U: 1,2,3] [G:1,2,4]	Homework, midterm, final
Basic numerical algorithms	Derivatives, quadrature, interpolation, root-finding, special functions, the FFT algorithm [U:2,5] [G:2,4]	Homework, midterm
Linear algebra	Matrices, algorithms, solving linear algebraic equations, programming with objects [U: 2,5] [G:2,4]	Homework, midterm
Solving nonlinear equations	Minimization and maximization of functions, multi-dimensional root finding, nonlinear models of data [U:2,5] [G:2,4]	Homework, final
Ordinary differential equations	Initial value and boundary value problems, the Kepler and 3-body problems, chaotic dynamics in nonlinear systems, quantum eigenfunctions and eigenvalues [U:2,5] [G:2,4]	Homework, final

The “U” (undergraduate) bracketed numbers in the 2nd column give the correspondence to the Physics Department’s undergraduate curriculum goals: [1] The basic laws of physics; [2] Critical thinking and problem solving; [3] Laboratory skills; [4] General knowledge of the development of physics; [5] Contemporary areas of physics inquiry; [6] Written and oral communication skills. Note that not all courses emphasize all of the above goals.

The “G” (graduate”) bracketed numbers in the 2nd column give the correspondence to the Physics Department’s graduate curriculum goals: [1] The basic laws of physics; [2] Advanced knowledge in a specialty area; [3] Broad knowledge of physics topics outside the specialty area; [4] In-depth scientific research skills; [5] Teaching and communication skills. Note that not all courses emphasize all of the above goals.