Physics 3274—Computational Methods Fall Term 2020-2021
300 Old Engineering Hall (OEH) Tuesday & Thursday 2:50-4:05
Instructor: Joseph Boudreau (boudreau@pitt.edu)
Office 418 Allen Hall Phone +1 412 512 8335

Zoom meeting room: 956 1739 4220

Office hours Wed. 1:00-5:00 PM (use 478 805 8218)

Physics 3274 is a graduate course on computational physics. It aims to develop or reinforce programming skills, numerical analysis skills, familiarity with some important problems in computational physics and their methods of solution. Additionally, I hope it will build the student's self- confidence when confronted with computational challenges likely to arise during his/her career.

Special provisions for fall semester 2020: I plan to start the class online-only (as with all classes at the University of Pittsburgh) during the first week, then in-person with live+recorded video when allowed, which is nominally in the second week. Should the University require a retreat from in-person instruction at a later date, we will continue with online classes only. Also, if I personally sense an unacceptable level of risk, either to students or to myself, we will move to online-only classes. In the classroom, face coverings will be worn at all times; this is true for any of the university's three operational postures. Occupancy in the classroom is limited to 8 students, but current course enrollment exceeds that limit. Thus students may elect or be required to attend some sessions remotely. *Despite all of these restrictions I am confident that you will find this course useful, enjoyable, and rewarding.*

Learning objectives:

- Program with confidence in C++ within the paradigms of structured, object-oriented and generic programming.
- Write programs that harness the resources of multiple CPUs concurrently.
- Write algorithms to solve a variety of problems in classical, quantum, and statistical physics on a computer.
- Simulate physics processes and experimental data.
- Visualize and model data.

The following is a tentative schedule for the course. The qualification in the second column describes whether the lecture is mostly devoted to computing [C], to numerical analysis [N], or to physics [P]. As you can see the course is actually a mixture of these.

20 Aug	C	Building programs in a Unix environment
25	C	Assembling larger programs. The SVN system
27	C,	Encapsulation. Built-in datatypes and classes
1 Sep	C,P	A tour of some useful class libraries

3	P	Solutions to selected undergraduate problems.
8	N	Root finding. Interpolation.
10	N	Integration with classical quadrature formulae
15	N	Gaussian Integration
17	C	How to write a class I
22	C	How to write a class II
24	N	Uniform and nonuniform variate generation.
29	N	Monte Carlo Integration & Markov Chain Monte Carlo
1 Oct	P	Percolation Theory I
6	P	Percolation Theory II
8	C	Parallel computing
13	C	Graphics for Physicists
15	N,P	Graphics for Physicists/Tutorial session
20	P	Ordinary differential equations:
22	P	Solutions to problems in classical mechanics
27	C	Classical Chaos
27	P	Polymorphism
29	P	Rotations and Lorentz Transformations
3 Nov	P	Simulation I
5	P	Simulation II
10	P	Statistical analysis and data modeling. Basic notions, χ^2 and binned likelihood fits.
12	P	Unbinned maximum likelihood fits. Bayesian inference with Markov Chain Monte Carlo
17	C	Templates and the Standard Template Library
19	P	Quantum Mechanics I: few body systems
24	P	Quantum Mechanics II: many body systems
1 Dec	P,C	Hackathon
3 Dec	P,C	Hackathon

The textbook for the course is

- *Applied Computational Physics*, J. Boudreau and E. Swanson, Oxford University Press, 2018. This textbook was jointly developed specifically for this course by two Pitt professors who are frequent instructors of Physics 3274. Copies are available in the physics mini-library (321 Allen Hall), in the Engineering Library; online copies are available through the PittCat library system.

Another suggested reference which you can use for the rest of your life is

- *Numerical Recipes, the Art of Scientific Computing, 3rd Edition*, William H. Press, Saul A. Teukolsky, William T. Vetterling, and Brian P Flannery, 2007 (Cambridge University Press). You will handy not only in this course but (I believe) often throughout your career.

The grading is based entirely on weekly assignments. For each assignment, you will be asked to write up solutions containing the numerical answers and/or plots where requested, and/or source code which you should put in an SVN or GIT repository, which you will learn about in the first week of class.

We do not emphasize the use of computer algebra systems like Mathematica, Maple, or MathCad, but focus more on code development in the C++ language, the programming language in which this course is conducted.

Prior programming experience: It is highly desirable to possess prior programming experience with one or more languages in the "C" family: the most familiar of these are C, C++, Python, Java, PHP, and Perl. If you know C++ already, great, if not we develop the key concepts in object-based C++ programming throughout the course, particularly in the four lecture called *Encapsulation*, *How to write a class,* P*olymorphism and Object Oriented Programming*, and *Templates and the Standard Template Library*.

Students wishing a more elementary primer on more basic elements of C++ (such as datatypes, arrays, structures, control statements, functions) can come to two special lectures outside of the normal meeting time, which we will arrange early in the semester, and/or consult some of the reference materials on reserve in the Engineering library.

Resources: You will obviously need a computer to carry out the assignments in this course. In past years most students have used their personal laptop computer, but if necessary I will put other resources at your disposal. The acceptable computing platforms are linux (which can be installed on a personal computer) or Mac OS machines. During the first week we will strive to make sure every student is set up and able to function on some type of computer. Shortly into the class we will load additional external libraries onto each machine. This "courseware" consists of freely available program libraries plus some custom libraries that I have developed for this course.

This course is in many ways like a laboratory course: while in a laboratory course you build instrumentation and carry out measurements, in this course you will build programs and carry out calculations. *I expect that you will need to interact with me outside of class* for advice on how to do this, particularly because your computing backgrounds vary greatly from student to student. I can provide guidance and advice in program design, debugging, and physics interpretation. My office hours are 1:00-5:00 on Wednesdays but you are welcome solicit my assistance at any time (email, skype, phone during the pandemic).

Provisional list of reserve material:. During COVID, the library will not maintain reserve materials. However, the University Library system has electronic copies that may be accessed either directly or indirectly via PittCat (library.pitt.edu) Many of these books are on my shelf if you prefer to stop by and consult them in my office. This list may expand as the semester progresses.

The C++ Programming Language for scientists, engineers, and mathematicians, D.M. Capper (1994) [available online through archive.org]

C++ for scientists and engineers, G.J. Bronson (2013) [available online through archive.org]

The C programming language, Brian W. Kernighan and Dennis M. Ritchie (1988)

Object-Oriented Analysis and Design with Applications, Grady Booch (1994)

Effective C++ (2005), *More Effective C++,* (1996), and *Effective STL* (2001), *Effective Modern C++* (2014) Scott Myers.

Methods of Numerical Integration, Philip J. Davis and Philip Rabinowitz (1975)

Orthogonal Polynomials, Gabor Szego (1939), available online through PittCat

Numerical Methods for Ordinary Differential Equations, J.C. Butcher (2008

Numerical Recipes: the art of scientific computing, 3rd Edition, William H. Press et al. (2007)

Introduction to Percolation Theory, Dietrich Stauffer (1985)

Students with diabilities: If you have a disability for which you are or may be requesting an accommodation, you are encouraged to contact both your instructor and Disability Resources and Services, 140 William Pitt Union, (412) 648-7890/(412) 383-7355 (lTY), as early as possible in the term. DRS will verify your disability and determine reasonable accommodations for this course