Phys 512 — Computational Physics with Applications — Fall 2020

***** This course outline is preliminary and is subject to change *****

1 Contact information

Lecturer:
Prof. Jonathan Sievers
Email: jonathan.sievers@mcgill.ca

Teaching assistant details: Matthew Lundy (matthew.lundy@mail.mcgill.ca)

Course materials will be available predominantly on github (`https://github.com/sievers/phys512-2020`).

2 Lecture schedule

Lectures (Sievers): Monday & Wednesday 16:05 – 17:25
Tutorial schedule TBA
Zoom link for lectures:
`https://mcgill.zoom.us/j/92342674635?pwd=cmFCeTFiOW13YkxFczEvdG5XYjUwZz09`

Lectures will be delivered live and online at the scheduled times and will consist of both derivations and coding examples. Lectures will be recorded and made available for later viewing. The lecturer will also have regularly scheduled office hours each week. During these times, the instructor will be online using a web conferencing tool (e.g. Zoom) and will be available to answer questions from anyone who chooses to call in.

3 Evaluation policies

- **Homework 40%:** there will be assignments every 1-2 weeks, all of which will be submitted electronically. The assignments will predominantly consist of coding problems that will implement techniques discussed in class.
- **Final Project 30%:** There will be a final project that will involve coding a larger scale problem than the homeworks. Suggested problems will be provided but students are free to come up with their own, subject to instructor approval.
- **Final exam 30%:** Take home exam that will be a mix of coding and analysis.

4 Course material and textbook

There is no required textbook, however *Numerical Recipes* covers many of the subjects that will be covered in class. Other useful resources include "What every computer scientist should know about floating point arithmetic." by David Goldberg, and *Introduction to Astrophysical Fluid Dynamics* by Mike Zingale, available at `http://bender.astro.sunysb.edu/hydro_by_example/CompHydroTutorial.pdf`. Material to be covered includes:

1. Brief introduction/review of python and coding practices.
2. Floating point math/roundoff error. How to optimize *e.g.* numerical derivatives.
3. Function interpolation and numerical integration. Error analysis of various techniques.
4. Integration of ODEs. Runge Kutta methods. Stiff equations, and implicit techniques.

5. Linear algebra and linear least-squares fitting. Singular value decomposition and its application to numerically unstable models. Legendre and Chebyshev polynomials. Iterative solutions to large problems using conjugate gradient.

6. Nonlienar least-squares fitting. Newton's method, Levenberg-Marquardt, and Markov Chain Monte Carlo.

7. Discrete Fourier transforms. Convolutions and applications to image processing. Aliasing and the Nyquist theorem. Stationary noise and matched filters.

8. Random variables. Transformation method, rejection method, and ratio-of-uniforms.

9. Partial differential equations. Eulerian and Lagrangian techniques. Advection equation, stability analysis, and the CFL condition. Inviscid fluid flow, numerical dissipation, and the Lax method.

10. Brief introduction to machine learning, including automated classification of handwritten numbers.

5 Language of submission

In accord with McGill University's Charter of Students' Rights, students in this course have the right to submit in English or in French any written work that is to be graded. This does not apply to courses in which acquiring proficiency in a language is one of the objectives.

6 Academic integrity

McGill University values academic integrity. Therefore, all students must understand the meaning and consequences of cheating, plagiarism and other academic offences under the Code of Student Conduct and Disciplinary Procedures" (see `www.mcgill.ca/students/srr/honest/` for more information).

In the event of extraordinary circumstances beyond the University's control, the content and/or evaluation scheme in this course is subject to change.