PHY 2200: Computational Physics
Final Project Description

Requirements

For the final project, you are asked to complete a fairly substantial simulation of some physical system. You will need to write a Jupyter notebook and make use of various algorithms and procedures developed in class. The project itself consists of four components described below. Final submissions are due by **8 AM, Friday, April 28** (notebook + video). No late submissions accepted.

- **Create a working program that accurately performs the desired simulation** (40%) You should use a Markdown cell at the top of your notebook to describe what you plan to calculate. Use equations as needed. Beyond this, the notebook should run without errors and compute what you claim it does.

  As part of this task, you need to explain how you used material we have developed or *why* you did something different. Writing programs/scripts in the real world usually involves borrowing templates from other sources. I don't care how you complete your notebook, but I do ask that you understand what you submit. It is unlikely ChatGPT will spit out a working script to do exactly what you want. If you use other sources, explain how they do things differently from what has been presented in your course materials. If you make use of course material, explain where it came from and why you used that material.

  You have total freedom to copy and paste from previous course material (modifying as needed) or doing this from an entirely different approach. But you must explain (a) where you received assistance and (b) how your approach compares to what was used in course examples. Regardless of whether or not your approach works, you will receive no credit for a wildly different computational scheme without any explanation for how it works and why you chose that method over what was presented in class.

  The scope/difficulty of your topic will factor in heavily to the grade. Your project work should be of comparable difficulty to midterm project topics. More difficult topics will be graded more gently, while easier topics will be graded severely by a robot which shows no compassion.

- **Assess accuracy of your simulation** (25%) You should give some explanation of why you believe your notebook correctly simulates your chosen phenomenon. There is no single way to do this, but you might consider things like:

  - Why did you choose a particular method (e.g., forward Euler, RK2/4, etc.)?
  - Why did you choose a particular number of steps or size for dt?

  - Is there any special limit in which an analytic solution exists which you can compare your work to (or can you recreate some published/well-known result)?

- **Find something *interesting* to do with your simulation; for some of these topics suggestions are given for this (**25%**)** This is fairly open-ended. The fun only begins once a program *works* for some test case. The point is to use a working program to explore interesting questions. Once you have something that you feel confident works, explore. I define "interesting" fairly loosely. Maybe some parameter choice makes an interesting looking plot. Play with the result and explain what you find (and why it's interesting to you).

- **Presentation (**10%**)** You will need to make a short (less than five minutes in length) video screencast explaining what you did and highlighting interesting findings. This is an advertisement for your project rather than a formal talk. We will all gather in the classroom during the final exam period to watch the videos.

  Making a video like this is a legitimate life skill. Plus, with a fairly large class this is going to streamline our final exam period and allow everyone to showcase their work. There are no strict formatting requirements, but you should be as clear, concise, complete, and correct as possible.

Format

Your Jupyter notebook should be organized (roughly) into the following sections:

- Overview (explain what you're going to do and how you do it; include *all* references to assistance you received)

- The calculations (if you use any algorithms or techniques not discussed in class, explain them as needed)

- Reflections (describe what you did with the computation once it worked; use a Markdown cell at the end to format this neatly)

Possible topics

For this project, you get to find your own topic. Some rough suggestions are shown below, but it is expected that you will have to perform some level of research and refine a specific, interesting question. The instructor will need to approve all project topics by **Friday, April 21**.

- Use `scipy` the package `integrate` to investigate a three-body scenario. RK4 was not robust enough to give accurate simulations for most scenarios considered on the midterm project, but this method should do better.

- Investigate one-dimensional cellular automata (https://mathworld.wolfram.com/ElementaryCellularAutomaton.html)
- Explore other types of neuron models ("random walk" in voltage, integrate-and-fire, "leaky" integrate-and-fire, ...)
- Use the method of relaxation to solve the Poisson equation for electric potential V

  $$\nabla^2 V = -\frac{\rho}{\epsilon_0}$$

  Here ρ is charge density which is specified.
- Design a model (or implement someone else's model) for the spread of a virus.
- Investigate thoroughly some interesting physical system (swing spring, double pendulum, tumbling baseball, etc.)
- Modify some simulation we have used in a particularly motivated way (e.g., come up with a way to add "water" to the forest fire simulation—thanks to Parker for suggesting this).
- The world is littered with models based on differential equations. Find something nontrivial, and explore it.