Phys 50733
Homework 6: Synthesis

Due: April 17, 2025

For your final assignment, I would like you to combine the computational physics knowledge gained during this course with the Python specific wisdom I've done my best to impart. For this assignment, you will create a Python script, i.e. a `.py` file, that imports code from your own custom Python package, in order to solve a simple computational problem.
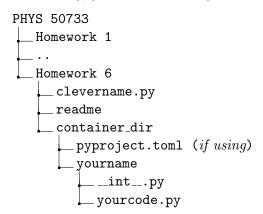
During this course we have written code to do numerical integration, numerical differentiation, interpolation, solve systems of simultaneous equations and eigenvalue problems, minimization problems, model fitting, and surely some things I've forgotten. At each step, I've encouraged you write flexible code that can be re-used later. Later has arrived.

Choose a type of problem we've solved during this course, find the code you wrote, and package it into an importable Python package. We discussed at least 3 ways to do this in class: relative imports, the PYTHONPATH, and pip installing. You may use any of these methods to make your package importable by your script, just make sure both the package, the `.py` file importing your script, and a `readme` file telling me what you did are included in your `Homework 6` GitHub directory. If you choose to make it pip installable, make sure it installs as your name to ensure I can install everyone's packages. Remember: for full credit your code must run on my machine.

Once your package is ready, move on to your `.py` script. In your script, use your package to solve 3 problems. This might be integrating 3 functions, or minimizing 3 functions, or fitting 3 functions, etc. You may wish to do something fancier like compute the electric fields for 3 different charge distributions. The sky is the limit! But not a requirement. I would prefer this assignment be a fun synthesis of all the computational physics and Python you've learned this semester. Ideally it will point the way towards using your own computational Python code in future research. So the requirements for this assignment are very loose. Full credit assignments will:

- Have a `clevername.py` script that

- Imports functionality from another user created file to solve

- Three problems of any level of complexity

- Resulting in three nicely formatted plots

For clarity, your final directory structure might looks something like:

```
PHYS 50733
├── Homework 1
├── ..
├── Homework 6
    ├── clevername.py
    ├── readme
    ├── container_dir
        ├── pyproject.toml (if using)
        ├── yourname
            ├── __int__.py
            ├── yourcode.py
```

Or it might look very different. The only requirement is your Python script import functions from another file, which using relative imports can be accomplished by just having another `.py` file in the same directory. (In fact this structure is by far the easiest for me to grade, but it is the least professional.) You may wish to try committing that structure to GitHub prior to day before it is due just to make sure it works for you.