Welcome.

Everyone:

- Pull the updates from the course GitHub repo:
  - `cd <46120-PiWE repo>`
  - `git pull upstream main` ← you might have “upstream2” instead

Physical students:

- Sit WHEREVER you want. 😎
- Turn off laptop volume (mute). ←**IMPORTANT!**
- Log into the Zoom meeting.
  - Microphone muted. Camera off.

46120: Scientific Programming for Wind Energy

Final project introduction

Ju Feng

Agenda for today

- Pull new course material ✔
- Final project introduction
- Begin homework

Final project introduction

Important deadlines

- **Form group (2-3)**: by Tuesday, April 1st, at 23:59
  - **Petition for solo team**: by Sunday, March 30th, at 23:59 (email Jenni)
  - **Ask for custom project**: by Wednesday, April 9, at 23:59 (email Ju)
- **Hand in final project**: by Sunday, May 4th, at 23:59

Grading

Your individual final grade (pass/fail) will be determined as a cumulative evaluation of your contributions to the final project and an individual written quiz.

Selection of projects

We provide three pre-defined final projects

- **Wind resource assessment based on reanalysis data**
- **Wind turbine modeling based on Blade Element Momentum (BEM) theory**
- **Wind power forecasting using machine learning**

You may also submit a request for a **custom project** if your group have an idea you would like to pursue. Your custom project will need to fulfill a list of requirements to get approval.

Requirements to pass

1. Your final project should have the folder structure and required files as shown in the diagram:
2. Your Python package inside the `src` folder should include at least one class (we will cover class and object oriented programming in week 9).
3. Your Python package should implement the required functions, either as listed in functional requirements in the pre-defined projects, or as we agreed on and documented if you work on a custom project.

```
[your_final_project]
├── inputs/
│   ├── data_files_we_provide
│   └── data_files_you_found (optional)
├── outputs/
│   └── data_files_you_generate (no need to push to Github)
├── src/
│   └── your_python_package_codes
├── tests/
│   └── python_scripts_you_write_for_tests
├── examples/
│   ├── main.py (will run in evaluation)
│   └── other_example_scripts_you_write (optional)
├── .gitignore
├── LICENSE
├── Collaboration.md
├── README.md
├── pyproject.toml
└── any_other_files_you_may_want (optional)
```

Requirements to pass (continued)

4. The README file should contains:
   - A brief overview of the package objective.
   - Installation instructions.
   - A description of the package architecture, with at least one diagram.
   - A description of the class(es) you have implemented in your package, with clear reference to the file name of the code inside `src`.
   - A description of what peer review (if any) you have implemented.
5. We should be able to install your package successfully following the Installation instruction in your README.md.
6. Your `main.py` script inside the examples folder should demonstrate, in a clear and structured manner, how the required functions are called and executed.
7. Your `main.py` script should run successfully and generate the expected results (like plots) in a reasonable time (less than 10 mins). If needed, define a "demo" mode and/or use saved intermediate results.

Requirements to pass (continued)

8. Test coverage on your package should be higher than 80%, as evaluated using `pytest-cov` on your `src` folder, i.e., by running:

   ```
   pytest --cov=src tests/
   ```

9. pylint score on your package should be higher than 8.0, as evaluated using `pylint` on your `src` folder, i.e., by running:

   ```
   pylint src/
   ```

Questions?

Pre-defined projects

- Go to GitHub

Homework

- **PART 0: Sign up for a final project team (ASAP)**
- **PART 1: Explore one of the pre-defined projects**
  - Discuss with your group, choose one pre-defined project to explore.
  - Write a function to load, parse and plot the provided dataset in inputs, generate at least one figure. Ideally also with test in tests.
  - Think about the structure/architecture of your package.

  Note: the project you choose in this homework doesn't mean you have to stick to that project. Your group can make decision later on which project to work on. If you have a better project idea that your group want to pursue, first check the section on Custom project, and then come talk to us.

  But remember to make a decision early, so that your group has enough time to work on it.