Welcome.

Everyone:

- Pull the updates from the course GitHub repo:
  - `cd <46120-PiWE repo>`
  - `git pull origin main`

LIVE

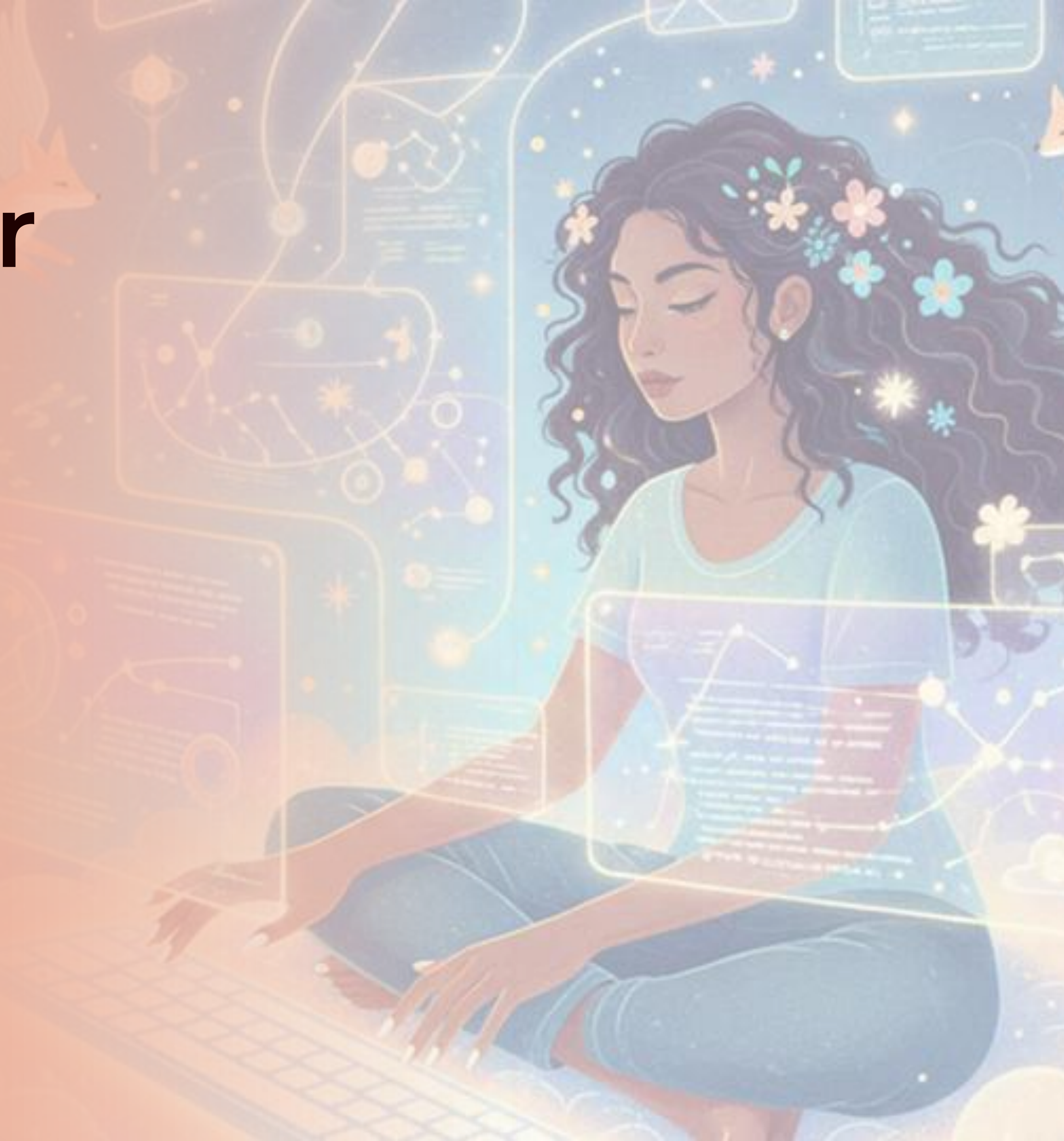
NB:

- By attending this class, you consent to being recorded. Recording will be shared to this class and possibly other DTU students for training purposes.

46120: Scientific Programming for Wind Energy

Turbie

Jenni Rinker



Agenda for today.

- Pull new course material. ✓
- Round robin.
- Introduction to CodeCamp projects.
- Begin teamwork on Week 3 homework.
  - Form CodeCamp teams! Deadline is **Monday Feb. 23 23.59**.
  - Random team if not signed up by deadline.

Round robin

Share solutions with your peers and give feedback.

20 minutes

Time to review and collaborate.

- 1 round of 20 minutes.
- 5 minutes: chaos.
- 15 minutes: present/discuss homework.
  - Functions AND tests! Discuss also the numpy/matplotlib tutorials/exercises.
  - Any issues with git/GitHub?
  - Team A screenshares & presents their solutions. Teams B & C provides feedback.
  - Switch which group presents/provides feedback.
- Afterwards: plenum discussion.
  - Be ready with questions!

Notes in plenum.

- Add here.

Agenda for today.

- Pull new course material. ✓
- Round robin. ✓
- Introduction to CodeCamp projects.
- Begin teamwork on Week 3 homework.
  - Form CodeCamp teams! Deadline is **Monday Feb. 23 23.59**.
  - Random team if not signed up by deadline.

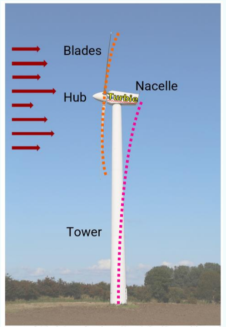
LIVE

CodeCamp project

Pass this to submit a final project.

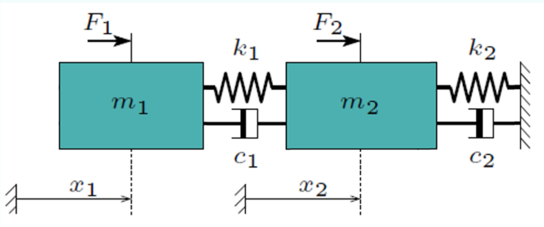
CodeCamp project overview.

- We provide:
  - Folders of txt files with turbulent wind time series at different mean wind speeds
    - Different turbulence intensities for “extra credit”
  - Parameters and modelling methodology of a simple wind-turbine model
- Your ultimate task:
  - Write code that
    1. Simulates the time-marching response of the turbine to different wind time series
    2. Plots statistics of the blade/tower deflections as a function of mean wind speed
  - Final code will be well organized, documented, tested, etc.



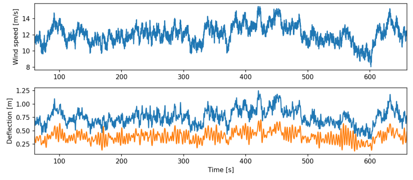
Original picture from https://en.wind-turbine-models.com/

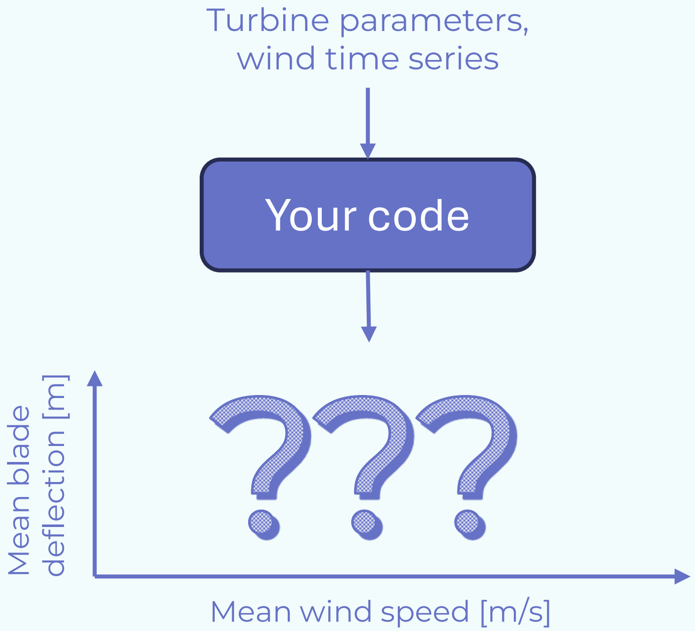
Turbie ❤



Original sketch from https://i.stack.imgur.com/fRMco.png

Details on model in Appendix





Turbine parameters, wind time series

Your code

Weekly development.

Weeks 3 and 4: Make functions for project.

- Specified explicitly in weekly assignment.
- After Week 4, your code can load a turbulent wind time series, simulate time-marching response of turbine, and save/plot.

Week 5: Design remaining code.

- Team needs to align on/develop remaining functions/scripts.
- Need to figure out how to handle the multiple wind time series, what to do with intermediate response time series, how to process/save statistics, etc.
- Other documentation also required.

Monday, March 9 at 23:59: Repos are locked for pushing, opened for viewing.

Before Week 6: Complete feedback of other teams.

Week 6: P2P presentations of code and feedback sessions.

List of functions in Week 3 and 4.

- For your information.
- Your team can start planning your attack for the final CodeCamp code if you have bandwidth.
  - NO CODE TO START. Make diagrams, discuss feature branches, etc.

Week 3

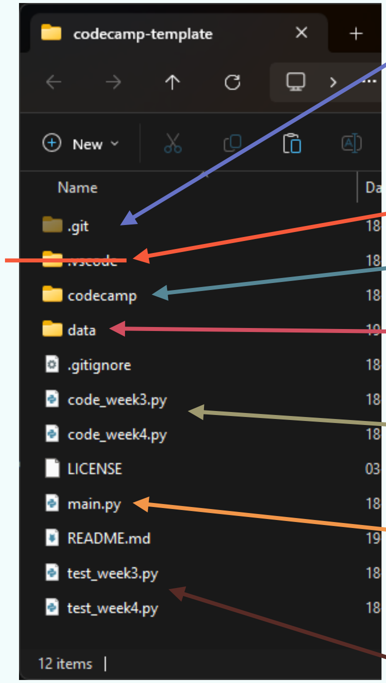
- load_resp(): load resp file
- load_wind(): load wind file
- plot_resp(): plot time series
- load_turbie_parameters(): load parameters from file
- get_turbie_system_matrices(): create M, C, K matrices from parameters

Week 4

- calculate_ct(): calculate ct for wind time series
- calculate_dydt(): calculate dy/dt for Turbie
- simulate_turbie(): simulates time-marchine response to wind time series in file
- save_resp(): save response time series to file

Files in CodeCamp repo.

- You will join a new GitHub assignment today with your CodeCamp team.
- Your CodeCamp team repo will have these files:



```
.git            18
.vscode         18
codecamp        18
data            19
.gitignore      18
code_week3.py   18
code_week4.py   18
LICENSE         03
main.py         18
README.md       19
test_week3.py   18
test_week4.py   18

12 items
```

hidden folder with git history (may be hidden depending on your file viewer)

not tracked in repo but may appear if you set up tests. VS Code settings.

codecamp/__init__.py is where your will place functions you develop

wind time series, turbine parameters, etc.

code to call functions for week 3 and 4 homework

executable code to demonstrate completion of CodeCamp project

tests for functions from week 3 and week 4

Questions?

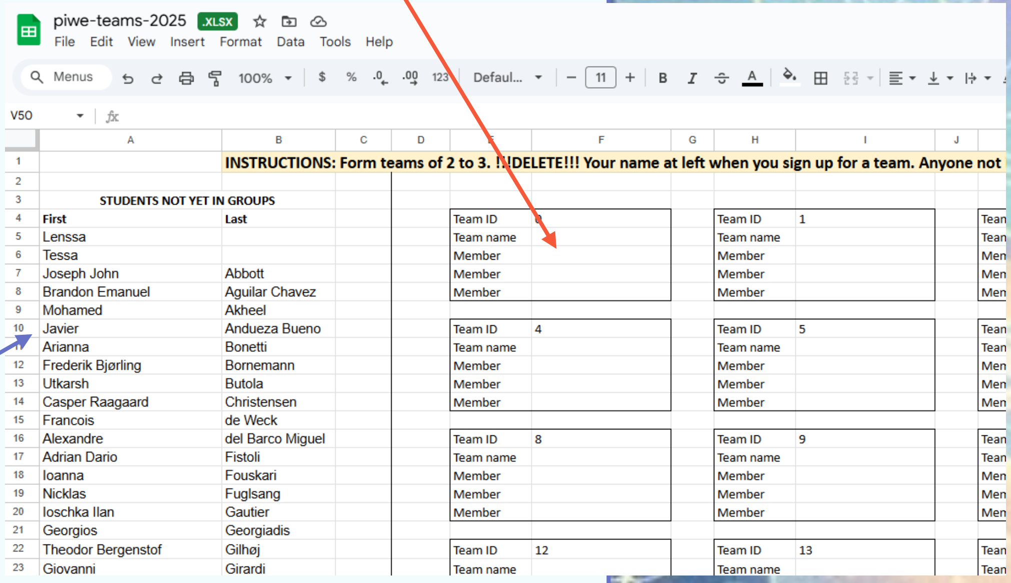
Homework for this week

Go forth and meet your destiny.

First things first: group formation.

- Groups of <u>2 to 3 students</u> for the CodeCamp project.
- Details on 46120 GitHub. Links on Learn.
- **VERY IMPORTANT for proper sign-up**:
  1. <u>DELETE NAME</u> in left column in sign-up sheet AND FULL name listed under a team.
  2. You must have joined your team in the CodeCamp GitHub assignment.
- Deadline to sign up is <u>**Monday Feb. 23, 23:59**</u>.
  - Any leftover students after this deadline will be placed into random groups. I.e., you could get placed with someone who drops the course.

add full name here!



	A	B	C	D	E	F	G	H	I	J
1		INSTRUCTIONS: Form teams of 2 to 3. !!!DELETE!!! Your name at left when you sign up for a team. Anyone not								
2										
3	STUDENTS NOT YET IN GROUPS									
4	First	Last			Team ID	0		Team ID	1	Tean
5	Lenssa				Team name			Team name		Tean
6	Tessa				Member			Member		Men
7	Joseph John	Abbott			Member			Member		Men
8	Brandon Emanuel	Aguilar Chavez			Member			Member		Men
9	Mohamed	Akheel								
10	Javier	Andueza Bueno			Team ID	4		Team ID	5	Tean
11	Arianna	Bonetti			Team name			Team name		Tean
12	Frederik Bjørling	Bornemann			Member			Member		Men
13	Utkarsh	Butola			Member			Member		Men
14	Casper Raagaard	Christensen			Member			Member		Men
15	Francois	de Weck								
16	Alexandre	del Barco Miguel			Team ID	8		Team ID	9	Tean
17	Adrian Dario	Fistoli			Team name			Team name		Tean
18	Ioanna	Fouskari			Member			Member		Men
19	Nicklas	Fuglsang			Member			Member		Men
20	Ioschka Ilan	Gautier			Member			Member		Men
21	Georgios	Georgiadis								
22	Theodor Bergenstof	Gilhøj			Team ID	12		Team ID	13	Tean
23	Giovanni	Girardi			Team name			Team name		Tean

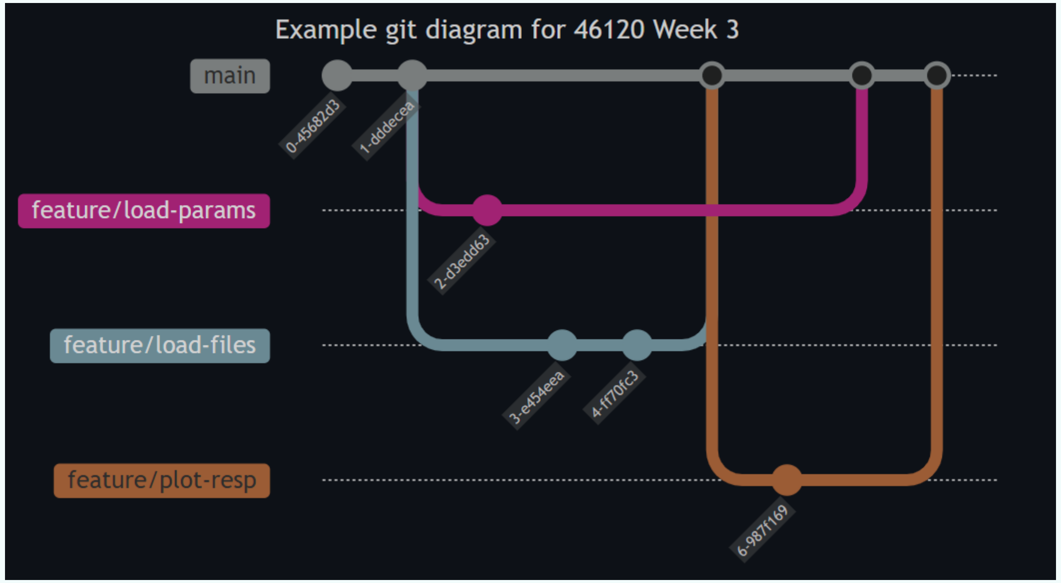
delete name from here!

Overview of homework.

- Objectives:
  1. Watch/read material on Turbie.
  2. Functions to load things from file.
  3. Function to plot time-series response.

- Example git workflow at right.

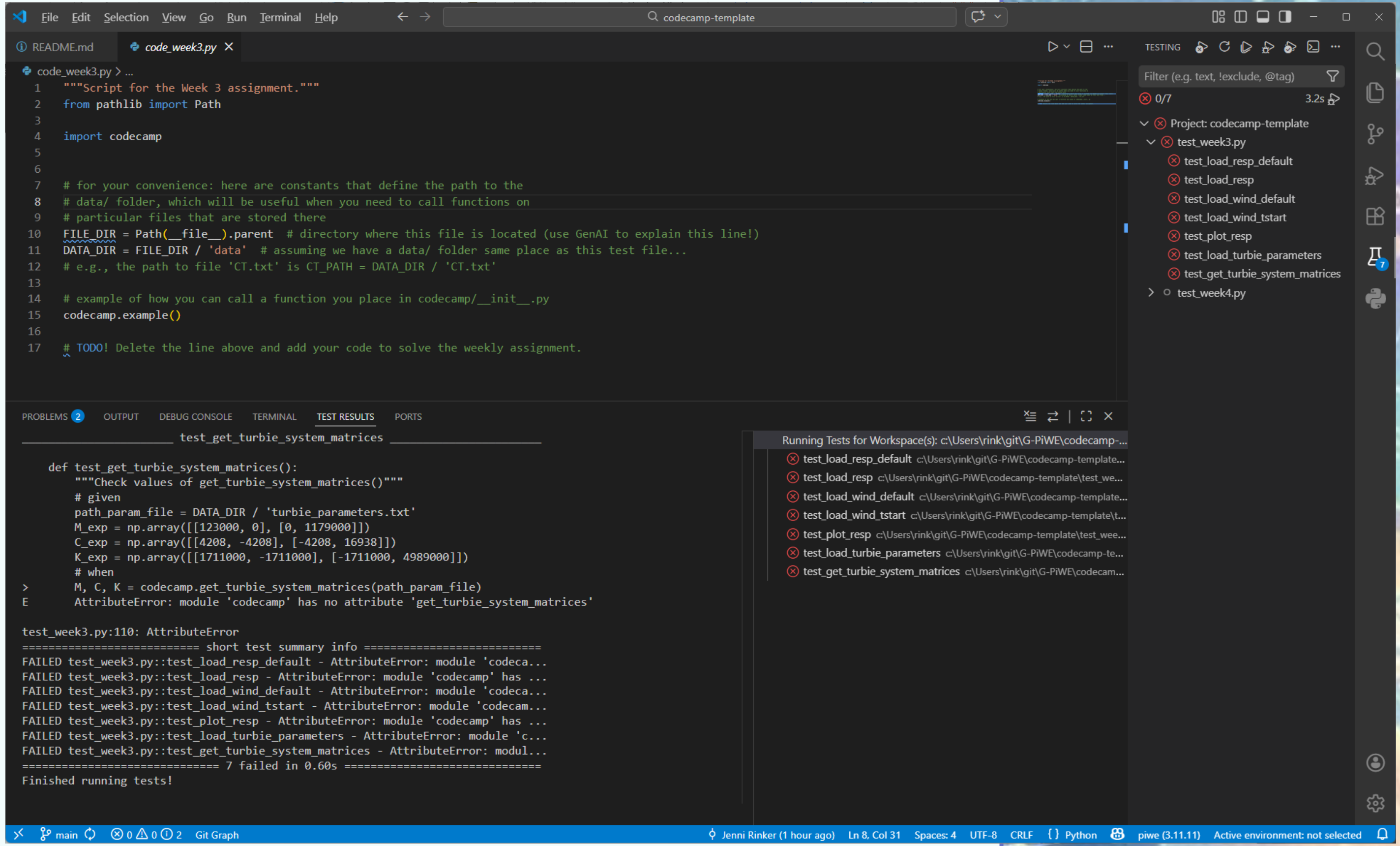
- More details on 46120 GitHub.



Notes, tips and expansions.

- You will write several functions this week. Suggested development steps:
  1. Write the code in code_week3.py first as a script.
  2. Turn code into a function in code_week3.py.
  3. Move function to codecamp/__init__.py.
- We changed naming of feature branches in the instructions.
  - I didn't like every branch starting with "add". New workflow more consistent with industry workflows.
  - You aren't required to exactly follow our suggested git workflow. If you want a different branch structure, go for it. But PRs and reviews are expected.

- Option to use integrated VS Code testing. Instructions in Appendix.
  - Can rerun individual tests, more.



```python
"""Script for the Week 3 assignment."""
from pathlib import Path

import codecamp


# for your convenience: here are constants that define the path to the
# data/ folder, which will be useful when you need to call functions on
# particular files that are stored there
FILE_DIR = Path(__file__).parent  # directory where this file is located (use GenAI to explain this line!)
DATA_DIR = FILE_DIR / 'data'  # assuming we have a data/ folder same place as this test file...
# e.g., the path to file 'CT.txt' is CT_PATH = DATA_DIR / 'CT.txt'

# example of how you can call a function you place in codecamp/__init__.py
codecamp.example()

# TODO! Delete the line above and add your code to solve the weekly assignment.
```

```
Project: codecamp-template
  test_week3.py
    test_load_resp_default
    test_load_resp
    test_load_wind_default
    test_load_wind_tstart
    test_plot_resp
    test_load_turbie_parameters
    test_get_turbie_system_matrices
  test_week4.py
```

```
______________________ test_get_turbie_system_matrices ______________________

    def test_get_turbie_system_matrices():
        """Check values of get_turbie_system_matrices()"""
        # given
        path_param_file = DATA_DIR / 'turbie_parameters.txt'
        M_exp = np.array([[123000, 0], [0, 1179000]])
        C_exp = np.array([[4208, -4208], [-4208, 16938]])
        K_exp = np.array([[1711000, -1711000], [-1711000, 4989000]])
        # when
>       M, C, K = codecamp.get_turbie_system_matrices(path_param_file)
E       AttributeError: module 'codecamp' has no attribute 'get_turbie_system_matrices'

test_week3.py:110: AttributeError
========================== short test summary info ==========================
FAILED test_week3.py::test_load_resp_default - AttributeError: module 'codeca...
FAILED test_week3.py::test_load_resp - AttributeError: module 'codecamp' has ...
FAILED test_week3.py::test_load_wind_default - AttributeError: module 'codeca...
FAILED test_week3.py::test_load_wind_tstart - AttributeError: module 'codecam...
FAILED test_week3.py::test_plot_resp - AttributeError: module 'codecamp' has ...
FAILED test_week3.py::test_load_turbie_parameters - AttributeError: module 'c...
FAILED test_week3.py::test_get_turbie_system_matrices - AttributeError: modul...
============================== 7 failed in 0.60s ==============================
Finished running tests!
```

```
Running Tests for Workspace(s): c:\Users\rink\git\G-PiWE\codecamp-...
  test_load_resp_default c:\Users\rink\git\G-PiWE\codecamp-template...
  test_load_resp c:\Users\rink\git\G-PiWE\codecamp-template\test_we...
  test_load_wind_default c:\Users\rink\git\G-PiWE\codecamp-template...
  test_load_wind_tstart c:\Users\rink\git\G-PiWE\codecamp-template\t...
  test_plot_resp c:\Users\rink\git\G-PiWE\codecamp-template\test_wee...
  test_load_turbie_parameters c:\Users\rink\git\G-PiWE\codecamp-te...
  test_get_turbie_system_matrices c:\Users\rink\git\G-PiWE\codecam...
```

Remember, you're expected to work ***about 6 hours outside of class****. Schedule accordingly.*

Before you are freed…

NB: Recall you're expected to work ***about 6 hours outside of class****.*

- Homework details on the course GitHub repo.

Complete **Part 1** in class, move on as agreed with your team.

Online: we will open self-navigable BORs.

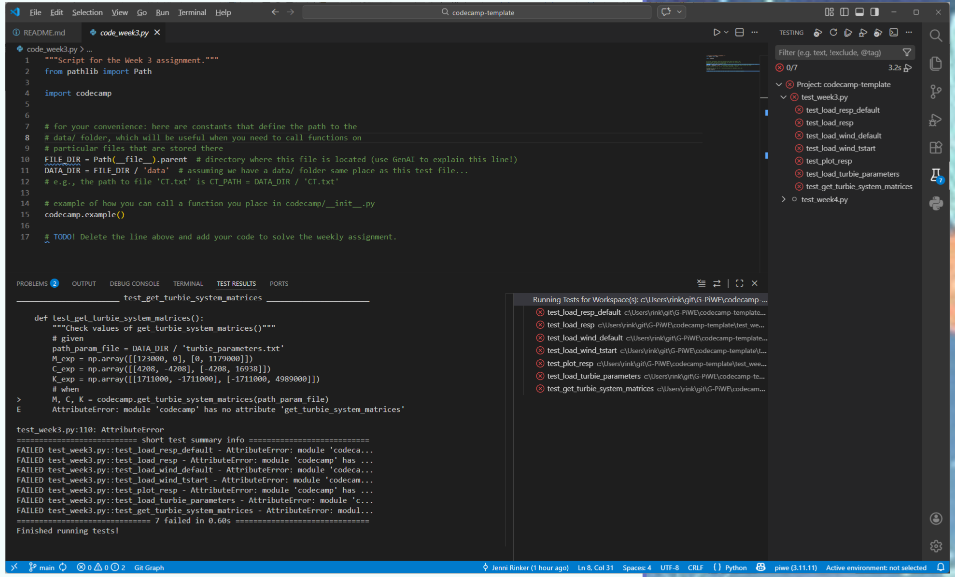
- **To get help during class**: Post in Slack / #debugging if you want a TA to enter your BOR or come find your group.
- NB: We may close the Zoom meeting without warning at 12:00. Be ready with a backup plan.

Any questions?

Appendix

Test integration in VS Code.

- Open folder where tests are.
- Open a random .py file (e.g., test_week3.py) and make sure correct Python environment is selected.
    - E.g., base for Anaconda/miniconda.
- Click beaker panel.
- Configure Python tests.
- Select a test framework: pytest.
- Select the directory: ". Root directory" (a.k.a., tests are run from current folder).
- If it says "pytest selected but not installed", **click "cancel"**.
    - This is a false warning. We have pytest.
- You should now be able to see/run tests in the beaker panel.



```python
"""Script for the Week 3 assignment."""
from pathlib import Path

import codecamp


# for your convenience: here are constants that define the path to the
# data/ folder, which will be useful when you need to call functions on
# particular files that are stored there
FILE_DIR = Path(__file__).parent  # directory where this file is located (use GenAI to explain this line!)
DATA_DIR = FILE_DIR / 'data'  # assuming we have a data/ folder same place as this test file...
# e.g., the path to file 'CT.txt' is CT_PATH = DATA_DIR / 'CT.txt'

# example of how you can call a function you place in codecamp/__init__.py
codecamp.example()

# TODO! Delete the line above and add your code to solve the weekly assignment.
```

```
_______________________ test_get_turbie_system_matrices _______________________

    def test_get_turbie_system_matrices():
        """Check values of get_turbie_system_matrices()"""
        # given
        path_param_file = DATA_DIR / 'turbie_parameters.txt'
        M_exp = np.array([[123000, 0], [0, 1179000]])
        C_exp = np.array([[4208, -4208], [-4208, 16938]])
        K_exp = np.array([[1711000, -1711000], [-1711000, 4989000]])
        # when
>       M, C, K = codecamp.get_turbie_system_matrices(path_param_file)
E       AttributeError: module 'codecamp' has no attribute 'get_turbie_system_matrices'

test_week3.py:110: AttributeError
=========================== short test summary info ===========================
FAILED test_week3.py::test_load_resp_default - AttributeError: module 'codeca...
FAILED test_week3.py::test_load_resp - AttributeError: module 'codecamp' has ...
FAILED test_week3.py::test_load_wind_default - AttributeError: module 'codeca...
FAILED test_week3.py::test_load_wind_tstart - AttributeError: module 'codecam...
FAILED test_week3.py::test_plot_resp - AttributeError: module 'codecamp' has ...
FAILED test_week3.py::test_load_turbie_parameters - AttributeError: module 'c...
FAILED test_week3.py::test_get_turbie_system_matrices - AttributeError: modul...
============================== 7 failed in 0.60s ==============================
Finished running tests!
```

```
Running Tests for Workspace(s): c:\Users\rink\git\G-PiWE\codecamp-...
  ⊗ test_load_resp_default c:\Users\rink\git\G-PiWE\codecamp-template...
  ⊗ test_load_resp c:\Users\rink\git\G-PiWE\codecamp-template\test_we...
  ⊗ test_load_wind_default c:\Users\rink\git\G-PiWE\codecamp-template...
  ⊗ test_load_wind_tstart c:\Users\rink\git\G-PiWE\codecamp-template\t...
  ⊗ test_plot_resp c:\Users\rink\git\G-PiWE\codecamp-template\test_wee...
  ⊗ test_load_turbie_parameters c:\Users\rink\git\G-PiWE\codecamp-te...
  ⊗ test_get_turbie_system_matrices c:\Users\rink\git\G-PiWE\codecam...
```

```
⊗ 0/7                                   3.2s
⌄ ⊗ Project: codecamp-template
  ⌄ ⊗ test_week3.py
      ⊗ test_load_resp_default
      ⊗ test_load_resp
      ⊗ test_load_wind_default
      ⊗ test_load_wind_tstart
      ⊗ test_plot_resp
      ⊗ test_load_turbie_parameters
      ⊗ test_get_turbie_system_matrices
  › ○ test_week4.py
```

- If you need to reset testing settings:
    - CTRL + SHIFT + P (Windows) or CMD + SHIFT + P (Mac/Linux)
    - Type "Preferences: Open settings (UI)"
    - Search for "Python testing".
    - Under Extensions / Python:
        - Uncheck "Auto test discover on save enabled"
        - Uncheck "pytest enabled"
        - Close VS Code and re-open the folder.
        - The beaker panel should prompt you to configure tests again.
- To remove test results when looking at test file, on "Test Results" panel click "Clear all results" (horizontal lines with x in upper left corner).

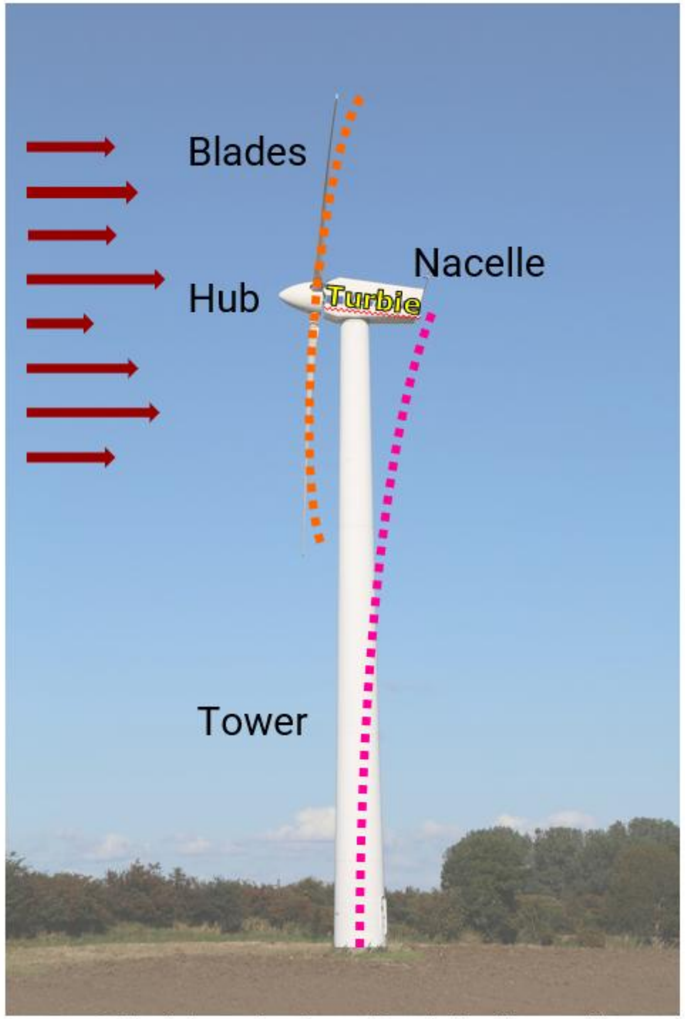
LIVE

Turbie

A beautiful windy girl.

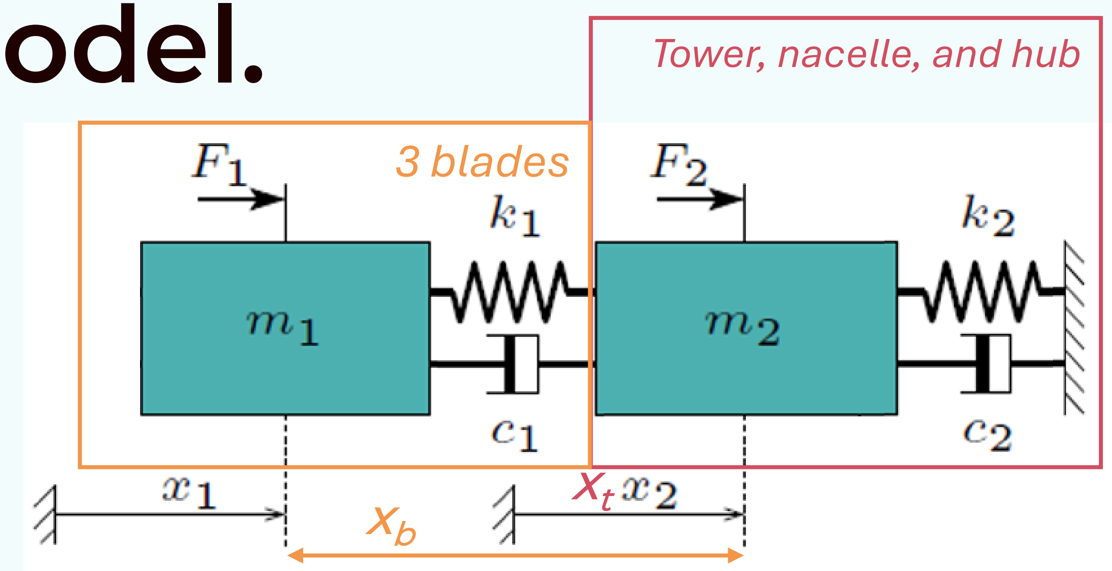
Turbie.

- Wind turbine model with 2 flexible DOF.
  - Blade collective flap deflection
  - Tower fore-aft deflection
- Time-varying wind loads applied on blades cause time-varying response in the 2 DOFs.
- Will model and simulate this dynamical system.
  - *(Using simplified physics.)*



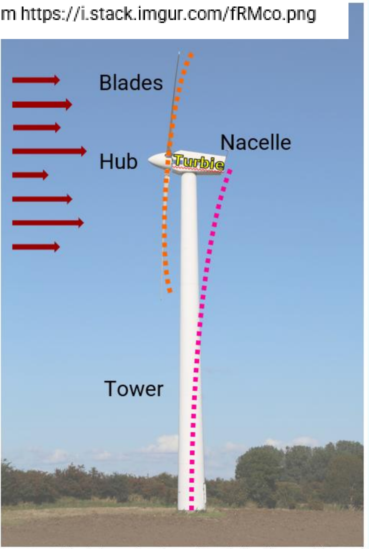
Original picture from https://en.wind-turbine-models.com/

Simple 2DOF model.



Original sketch from https://i.stack.imgur.com/fRMco.png

- 2 DoF mass-spring-damper



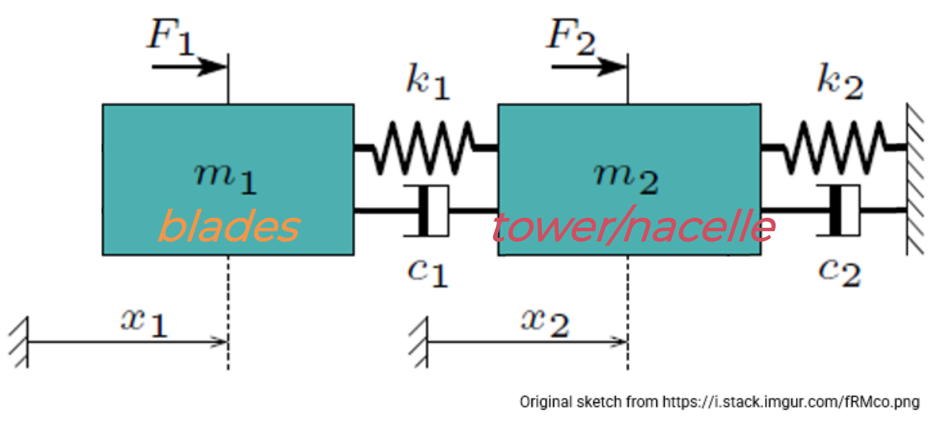
Original picture from https://en.wind-turbine-models.com/

- m_2 is the combined mass of hub, nacelle and tower
- x_2 (x_t) is the towertop deflection (relative to ground)

- m_1 is the combined mass of 3 blades
- x_1 is the absolute blade tip deflection relative to ground

- $x_b = x_1 - x_2$ is the blade tip deflection relative to tower

Equations of motion and parameters.



Original sketch from https://i.stack.imgur.com/fRMco.png

- Equations of motion for this 2DOF system:

$$\boldsymbol{M}\ddot{\boldsymbol{x}} + \boldsymbol{C}\dot{\boldsymbol{x}} + \boldsymbol{K}\boldsymbol{x} = \boldsymbol{F}$$

with

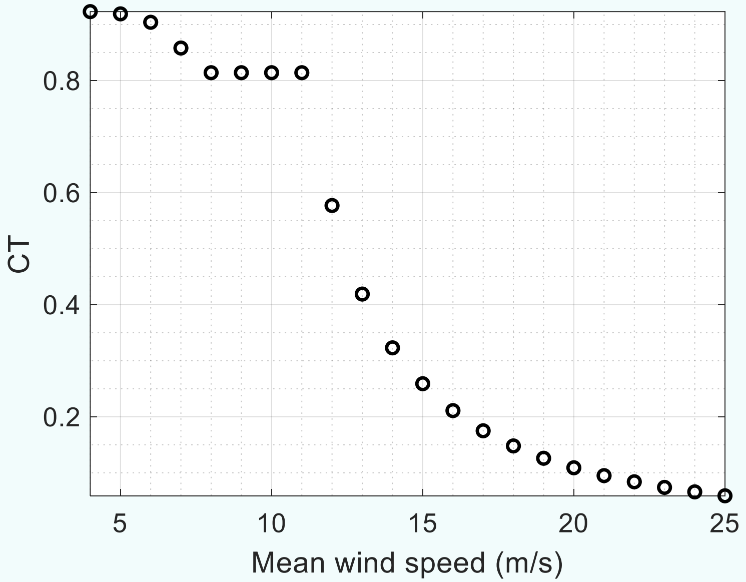
$$\boldsymbol{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \qquad \boldsymbol{M} = \begin{bmatrix} m_1 & 0 \\ 0 & m_2 \end{bmatrix} \qquad \boldsymbol{C} = \begin{bmatrix} c_1 & -c_1 \\ -c_1 & c_1 + c_2 \end{bmatrix} \qquad \boldsymbol{K} = \begin{bmatrix} k_1 & -k_1 \\ -k_1 & k_1 + k_2 \end{bmatrix} \qquad \boldsymbol{F} = \begin{bmatrix} F_1 \\ 0 \end{bmatrix}$$

aerodynamic forcing!

- All parameters given in turbie_parameters.txt.
  - File in codecamp team repo (will clone soon), under data/ folder.

(Overly) simple model of aerodynamic forcing.

- Assume rotor thrust coefficient (C_T) constant for a 10-minute simulation.
  - But it *is* a function of mean wind speed!



Wind force on the rotor is modelled as

$$f_{aero}(t) = \frac{1}{2}\rho A_r C_T [V(t) - \dot{x}_1(t)]|V(t) - \dot{x}_1(t)|$$

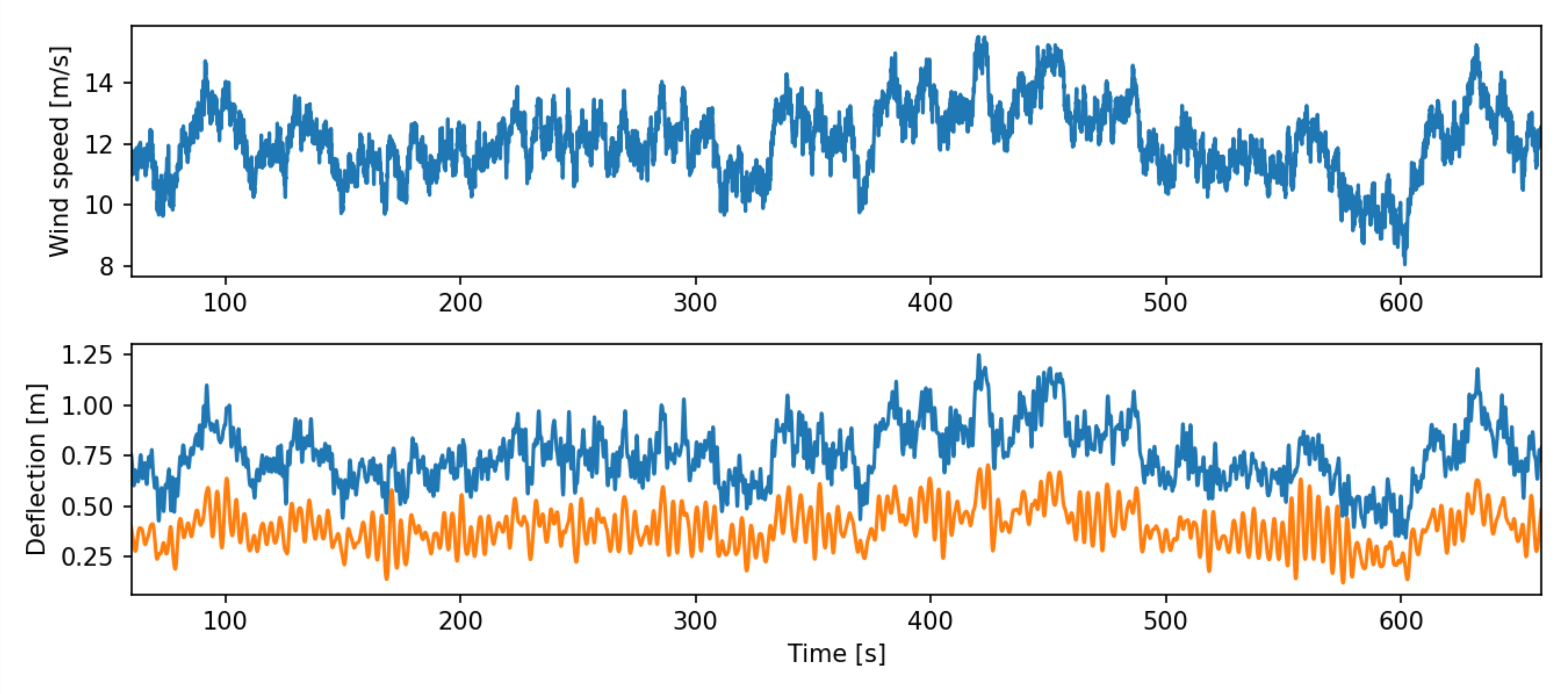
with

ρ	air density	$V(t)$	wind speed at hub
A_r	rotor area	$\dot{x}_1(t)$	blade velocity
$C_T(\bar{V})$	thrust coefficient		

- All parameters given in turbie_parameters.txt and CT.txt, in data/ folder on codecamp team repo.

What your code will do by the end.

- By the end of the CodeCamp module you will generate results like this



AND analyze statistics as a function of wind speed!