PROJECT 1: DOUBLE PENDULUM

Creating a program which represents a system of two connected pendulums, and animating the system for visual presentation.

— Exponential decay, simple ODE system

— Single pendulum, setting up a single pendulum system

— Double pendulum, expanding the system to include two connected pendulums

— Double pendulum animation, animating the system

Topics from IN1910:

— ODE Solvers, for making calculations with initial value systems

— Classes, methods, and decorators, for building the classes

— Codestyle and docstrings, to make the code readable

PROJECT 1: HOW WE WORKED

— Most of the project was done over 3 days, except the docstrings and readme

— Pair programming, mostly taking turns per sub-task given and working on them in the order given

— Consulting lecture notes or the internet when unsure of how to solve a task

When the code didn't work:

— Running different parts of the code with print statements to find out what did work

— Googling the issue

PROJECT 1: CHALLENGES

General:

— Code style disagreements: whitespaces, newlines, variable naming.

Animation:

— Making the _next_frame() method work correctly, we had forgotten a comma at the end of the return statement.

— Certain parts of the pendulum would disappear in show() but showed up correctly in save, this was fixed by having blit = False in the show method.

— Getting the video to save in real-time with the right FPS, we spent quite a bit of time working out how to calculate this.

PROJECT 1: TESTING

Testing on the go:

— Running the program for each new method to check for syntax errors

— Printing in different parts of a method when it didn't work, or to check that conditions were triggered

— Seeing if plots and animations look sensible

Project tasked Pendulum unit tests (repeated for Double Pendulum):

— Check pendulum at rest stays at rest

— Check that properties raise exceptions if solve hasn't been called

— Check that the radius is the same as the length