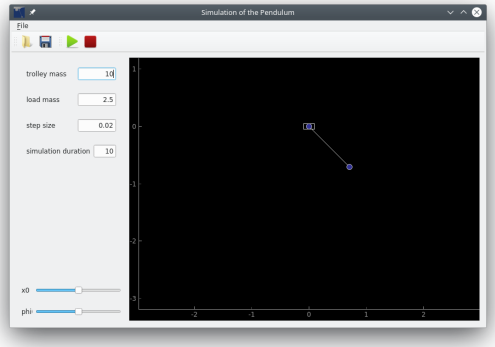
Exercise 12: GUI Programming with PyQt – Part 2

This exercise consolidates the concept of signals and slots and combines numerical simulation with visualization in a graphical user interface. The goal is an application with the following features:

- setting the parameters of the mechanical system and the initial values of the simulation via widgets
- saving and loading of these values from a file
- start and stop the simulation via buttons
- visualization directly coupled to simulation



For the example, already known elements from the course are reused (numerical simulation of the cart-pendulum-system, saving data as config file, `LineEdit` widgets etc.) and combined. By outsourcing functions / classes to modules a high reusability is aimed at. The actual program code of the `main.py` is more compact thereby. The functionalities are contained in the provided modules as follows:

`main.py`	main program with Gui class
`cart_pendulum_model.py`	math. model of the cart-pendulum-system `rhs(...)` – time derivative of the state
`customwidgets.py`	widgets (own classes), which are used by the main program `NumberInput` - Label + `TextEdit` for parameter intput `ParameterMask` - block of four `NumberInput`-widgets, realizes saving and loading of the parameters `IVSlider` - Label + `Slider` to determine of the initial values (x, φ)

Exercise 11.1:

1. Get an overview of the modules and classes and visualize the dependencies on paper.
2. Rewrite the program so that only one action is used for play and pause (suggested name `actn_toggle_anim`). This should change its icon depending on the state. Create a variable (e.g. `is_playing`), which stores the current state of the animation and is used when „triggering“ the action.
3. Following the example of the existing code, add a slider for setting pendulum length and make sure that its value is used both in the display and in the simulation.