Exercise 1:  (Fail of Euler Methods)
The explicit and implicit Euler method are simple methods to integrate ODEs numerically but there are a lot of examples where these methods fail.

(a) Consider the ODE
\[ \dot{y}(t) = -3y(t), \quad y(0) = 1, \quad t \in [0, 5] \]
and implement the explicit Euler method in MATLAB. Use time step sizes \( h = 0.7, 0.5, 0.1 \) and see how the numerical solution behaves. Compare the numerical solution with the exact solution \( y(t) = e^{-3t} \).

(b) Consider the ODE
\[ \dot{y}(t) = Ay(t), \quad y(0) = [1, 0, \ldots, 0]^T \in \mathbb{R}^N, \quad t \in [0, 1] \]
with
\[ A = \begin{pmatrix} \cdots & 1 \\ -1 & 1 & \cdots & 1 \end{pmatrix} \]
and implement the implicit Euler method in MATLAB using time step size \( h = 0.01 \) and dimension \( N = 10, 100, 1000, 10000 \) of the system. What can you observe concerning the time MATLAB needs to compute the solution?

Exercise 2:  (Representation of the exact solution of a linear ODE)

(a) Show that for \( t > 0 \) the exact solution of a differential equation
\[ y'(t) = Ly(t), \quad y(0) = y_0, \quad L \in \mathbb{R}^{n \times n}, n > 0 \]
is given by
\[ y(t) = \exp(Lt)y_0. \quad (1) \]
Hint: Consider the Taylor expansion of \( y(t) \) and use that \( y'(t) = f(y(t)) := Ly(t) \). How does \( y''(t) \) look like? For the representation of \( \frac{d^k}{dt^k} y(t) \), \( k > 1 \) use an induction argument.

(b) Consider the differential equation
\[ z''(t) = -\omega^2 z(t), \quad t > 0, \quad z(0) = 1, \quad z'(0) = 0, \quad \omega \in \mathbb{R}, \quad (H) \]
which describes the movement of a harmonic oscillator. Rewrite equation \( (H) \) as a first order system of differential equations, i.e. set
\[ y(t) := [z(t), z'(t)]^T \]
and find a matrix \( L \in \mathbb{R}^{2 \times 2} \) such that
\[ y'(t) = Ly(t), \quad t > 0, \quad y(0) = y_0 := [z(0), z'(0)]^T. \quad (2) \]

(c) in MATLAB, plot the exact solution of \( (2) \) with \( \omega = 1, 5, 10 \) in the time interval \( t \in [0, 10] \) by applying \( (1) \). Here please make use of
- a for loop
- the MATLAB built-in Matrix exponential function \( \exp \)

Discussion in the problem class wednesday 3:45 pm, in room 1C-03 in building Allianzgebäude 5.20.