Numerical Methods for ODEs: Exercise Week 1

Problem 1: Installation and getting started

For implementing mathematical techniques you learn in the lecture Python will be used.

For writing Python programs you can take Spyder: `https://www.spyder-ide.org/`

Download: `https://docs.spyder-ide.org/current/installation.html`

For Windows and Mac scroll down to "Downloading and installing" and click on "Windows Installer" or "macOS Installer".

Intro-videos: `https://docs.spyder-ide.org/current/videos/first-steps-with-spyder.html`

Your main task for the first week is to become comfortable with basic concepts like variables, functions, matrices, arrays, loops and so on in Python.

For that look at videos 1 to 6 of the following YouTube Playlist:

`https://www.youtube.com/playlist?list=PLdb-TcK6Aqj2l_H1mtPqlOo-Yki5UPzp4`

(also available here `https://www.halvorsen.blog/documents/programming/python/python.php`)

Additionally or alternatively you may also use the following Python tutorials:

`https://www.learnpython.org/`

`https://learn.microsoft.com/en-us/training/modules/intro-to-python/`

Problem 2: "Hello world!" program

Implement, save, and run the following program `hello.py` and explain what happens.

```
x = 1
y = 2
sum = x + y
print(sum)
for i in range(sum):
    print(i,'Hello world!')
while sum > 0:
  print(sum,'Goodbye!')
  sum -= 1
```

Problem 3: Plot function graphs

(a) Implement the following program `plots.py`.

```
import numpy as np
import matplotlib.pyplot as plt

def f(a,b,x):
  return a*np.sin(b*x)

g = lambda c, d , x : np.cos(c*x) + d
x = np.linspace(0,20*np.pi,1000)
a = 1
b = 0.5
c = 0.9
d = 1
plt.figure(1)
plt.plot(x, f(a,b,x),'b--',x,g(c,d,x),'g')
plt.xlabel('x')
plt.ylabel('y')
plt.title('Function plots')
plt.legend(['f','g'],loc = 'best')
plt.show()
```

(b) Include the third function

$$h(\sigma, \mu, x) := \frac{20}{\sigma\sqrt{2\pi}}\, e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$

in the plot and adapt the legend accordingly (choose $\sigma = 5$ and $\mu = 30$).

(c) Create a second plot which, for fixed a, b, c, d, displays the curve of all points $(f(a, b, x), g(c, d, x))$, $x \in [0, 20\pi]$.

Pyplot tutorial: `https://matplotlib.org/stable/tutorials/introductory/pyplot.html#`
NumPy tutorial: `https://numpy.org/doc/stable/user/absolute_beginners.html`