

Sheet 1

Topic: Setup

General Notice

Exercise sheets will be published on Mondays and will be discussed in class. We will be using Python for the programming exercises. Python can be used both interactively in command line or by executing scripts, e.g. from within an IDE, for solving numerical computations. Python is freely available for Linux, Mac OS, and Windows at <https://www.python.org/downloads/>. Please check the box 'Add python 3.7 to PATH' during the installation for Windows users. After installing python 3.7.0 version. Install PyCharm IDE to write and edit your Python codes.

A quick guide on Python is given in the Python cheat sheet, which is available on the website of this lecture.

Exercise 1: Defining functions

Functions in Python are usually defined inside a file. Create a file named `myfirstscript.py` and implement the following function:

$$f(x) = \cos(x)\exp(x)$$

Next, launch your script as `python myfirstscript.py` in the command line. In Python multiple functions can be defined in the same file and the filename is independent of the function names used in the file.

Note: While you create a project and then a file in it using PyCharm, be sure that you use the correct interpreter as Python 3.7.0 etc..

Exercise 2: Plotting data

Every python file is a script which can be evaluated later. It can contain multiple functions and other numerical computations all in one file. The `matplotlib.pyplot` module can be used for plotting.

- a) In the same python script write commands which plot the graph of the function f in the interval $[-2\pi, 2\pi]$. Hint: python's `numpy` module has as a special variable for π : `numpy.pi`
- b) Save the resulting plot as a PNG-file to your hard disk.

Note: import `numpy` and `matplotlib.pyplot` modules into your file using the following commands:

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

If you get an error “ModuleNotFoundError” then you have to first install these modules using “pip3 install numpy” and “pip3 install matplotlib” commands on the command prompt.

Exercise 3: Generating random numbers

Random numbers are important in probabilistic robotics so it is preferable to know what kind of random variables are provided by Python and how to use them. Hint: use numpy.

- a) Create a vector with 100000 random variables which are normally distributed with a mean of 5.0 and a standard deviation of 2.0.
- b) Create a vector with 100000 uniformly distributed random variables between 0 and 10.
- c) Compute the mean and standard deviation of the two vectors with random variables. Are the results what you would expect?
- d) Modify your script so that the generated distributions are exactly the same each time you call it.