Note: The purpose of this exercise sheet is to give a brief review over some of the basic and necessary functions for this course in python. Please make sure you can solve all the problems with ease. If you are new to python you may get a very brief (and incomplete) introduction by looking up the intro sheet on the gitlab.
Note: This exercise sheet is not counted for the bonus. Therefore, you do not need to show or send the solutions.

Exercise 0.1: Plotting

In this exercise, you will get familiar with `numpy.array` and the `matplotlib` package.

Plotting the function $f(x) = x^2$ for $x \in [-1, 1]$.

a) Create a numpy array X of evenly distributed points in $[-1, 1]$. For example, $X = [-1., -0.9, -0.8, \ldots, 1.]$

b) Create a numpy array Y of $f(x_i)$ with point x_i from the array X we created.

c) Plot the function. *Hint:* `matplotlib.pyplot.plot`

Now, we repeat the same procedure but considering $f(x) = e^{-x}$ for $x \in [0, 10]$.

d) Repeat step (a)-(c)

e) Plot now the Y in log scale. *Hint:* `matplotlib.pyplot.semilogy`

Suppose in additional to the data Y, we also have the standard deviation ΔY to the data resulting from some random error,

f) Create a numpy array ΔY and put in some small random numbers. *Hint:* `numpy.random.rand`

g) Repeat the steps before, but plot the errorbar ΔY with the data Y.
Hint: `matplotlib.pyplot.errorbar`

Exercise 0.2: Numerical linear algebra methods

In this exercise, you will get familiar with `numpy.array` and `numpy.linalg` package.

Consider the Heisenberg model of **two** spins,

$$H = \sum_{\alpha=x,y,z} \sigma_1^\alpha \sigma_2^\alpha = \sigma_1^x \otimes \sigma_2^x + \sigma_1^y \otimes \sigma_2^y + \sigma_1^z \otimes \sigma_2^z \,,$$

where the Pauli matrices are given as

$$\sigma^x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \qquad \sigma^y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \qquad \sigma^z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}.$$

a) Construct the Pauli matrices as numpy arrays.

b) Construct the Hamiltonian of the Heisenberg model as a numpy array.

c) Diagonalize the Hamiltonian. What are the eigenvalues and eigenstates?

   Hint: `numpy.linalg.eigh`

d) Construct the operator for the total magnetization in z direction

$$M = \sigma_1^z \otimes \mathbb{1}_2 + \mathbb{1}_1 \otimes \sigma_2^z \,,$$

   and compute its expectation value and variance $\langle M^2 \rangle - \langle M \rangle^2$ for for the ground state of H as well as the state $\frac{1}{\sqrt{2}}(|\uparrow\uparrow\rangle + |\downarrow\downarrow\rangle)$.

e) **Optional:** How would you generalize your code from part b)-d) to more than two spins with the same nearest-neighbor Heisenberg interaction?