Computational Motor Control - 2025

February 18, 2025

Contents

Student names: . . . (please update)

Instructions: This document contains the instructions to install and get familiarized with Python programming. **This lab is not graded.** *This file does not need to be submitted and is provided for your own benefit.*

1   Introduction to Python

1.1   Basic Python Concepts

In this section we will quickly go over the list of topics given below. You can open and run the individual files marked with the same topic name using Spyder. We suggest you to go through each section individually and spend time exploring each of the concepts by making changes to the code and observing the outputs.

0. HelloWorld

1. Imports

2. Data Types

3. Math

4. Conditional Statements

5. Data Containers : Lists, Tuples and Dictionaries

6. Functions

7. Loops

8. Numpy

9. Matplotlib

10. Classes

While you are executing each of the small exercises, try to learn how to use different features of Spyder. Especially the help and debugging feature. When you are unsure of any command, use the help service either the one built into Python or Spyder. After familiarizing yourself with the above concepts try to solve the following python exercises.

1.2   Exercise 1

Check if the following matrix M is a magic square or not?

Hint : *A magic square is a square matrix which contains distinct integers and whose sum along any of its individual rows or columns or diagonal is a constant. The constant is called as a magic constant or magic sum or magic square*

$$M = \begin{bmatrix} 16 & 3 & 2 & 13 \\ 5 & 10 & 11 & 8 \\ 9 & 6 & 7 & 12 \\ 4 & 15 & 14 & 1 \end{bmatrix}$$

Further Step : *Try if you can generalize your script to have a function to check any arbitrary matrix if it is a magic square or not. Import the function as a module in another script and use it to check the matrix M*

1.3   Exercise 2 - Plotting a function

Plot the following function $f(x)$ over an interval [0, 2] with proper labels and title

$$f(x) = sin(x-2)e^{-x^2}$$

2   References and additional useful links

2.1   Python

- NumPy for MATLAB users
- A byte of python
- A Crash Course in Python for Scientists
- The official Python documentation should be your first stop when looking for information
- numpy
- scipy
- matplotlib

2.2   Git

- Try Git!
- A Visual Git Reference

2.3   Spyder

- Getting started with Spyder