

# Porting code from Matlab to Python

## Introduction to the course and motivation

October, 2022 | Sandra Diaz Pier, Aaron Perez Martin

# Agenda

- Introduction to the course
- Scientific computing
- Motivation to use Python
- Review of the course schedule and topics

# Introduction to the course

- Who are we?
  - SimLab Neuroscience
  - Link between neuroscientists and HPC
  - Help port code to the supercomputers
  - Help in the development of scalable and maintainable code
  - Support best software development practices



# Introduction to the course



# Our course

Day 1	
Time	Topic
09:00 – 09:15	Motivation
09:15 – 10:00	Introduction to Python from the Matlab perspective
10:00 – 10:15	Example Python I/O
10:15 – 10:30	Introduction to Matplotlib
10:30 – 10:45	Coffee Break
10:45 – 11:15	Introduction to Numpy
11:15 – 11:30	Example of Numpy and Matplotlib
11:30 – 12:00	Introduction to unit testing, version control and debugging
12:00 – 13:00	Lunch
13:00 – 13:45	Image processing
13:45 – 14:15	Statistical analysis
14:15 – 14:30	Workflow to port scripts from Matlab to Python
14:30 – 14:45	Coffee Break
14:45 – 16:00	Hands on Session I

# Our course

Day 2	
Time	Topic
09:00 – 09:30	Introduction to classes and iterators
09:30 – 10:15	Introduction to MPI in Python
10:15 – 10:30	Example with MPI
10:30 – 10:45	Coffee Break
10:45 – 11:15	Numerical integration
11:15 – 12:00	Machine learning
12:00 – 13:00	Lunch
13:00 – 14:00	Questions pannel
14:00 – 14:15	Coffee Break
14:15 – 16:00	Hands on Session II

# Logistics

- Git repository
- Coffee Breaks
- Lunch

# What is Python?

- Python is a modern, general-purpose, object-oriented, high-level programming language.
  - clean and simple
  - expressive
- Technical details:
  - dynamically typed
  - automatic memory management
  - interpreted

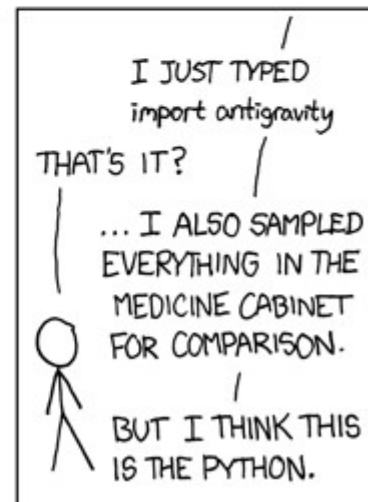
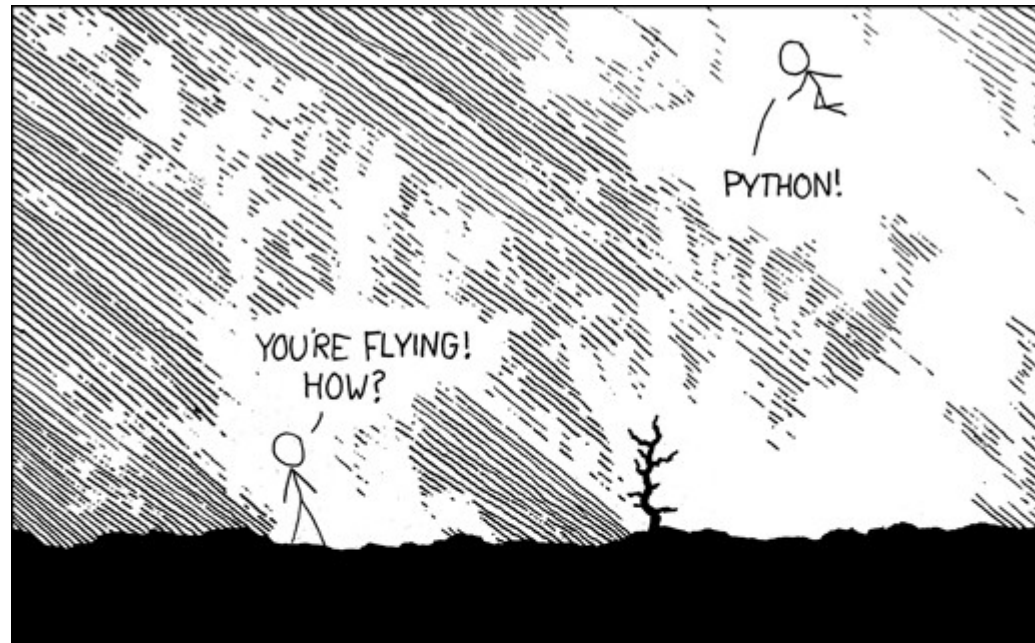


# What is Python?

- Advantages:
- ease of programming, minimizing the time required to develop, debug and maintain the code.
- modular and object-oriented programming
- large standard library
- portable
- easy to extend and link to optimized code
- “Python is powerful... and fast; plays well with others; runs everywhere; is friendly & easy to learn; is Open”. ([www.python.org](http://www.python.org))
- Disadvantages:
- interpreted and dynamically typed programming language
- decentralized

# Why Python for scientific computing?

- Large community of users
- Scientific libraries and environments
  - numpy: <http://numpy.scipy.org> - Numerical Python
  - scipy: <http://www.scipy.org> - Scientific Python
  - matplotlib: <http://www.matplotlib.org> - graphics library
- Support for
  - Parallelism with threads and interprocess communication (MPI)
  - GPU computing (OpenCL and CUDA)
- No license costs



# When not to port from Matlab to Python

Heavily rely on a complex, specialized and well tested toolbox like Simulink and no Python alternatives are available.

# Acknowledgments

Special thanks to Wouter Klijn, Lekshmi Deepu, Alper Yegenoglu and Lena Oden who helped in the preparation of the course material.



# References

(1) Based on the work by J.R. Johansson <http://jrjohansson.github.io>