Python Programming

Introduction

Prof. Chang-Chieh Cheng

Information Technology Service Center

National Chiao Tung University

Syllabus

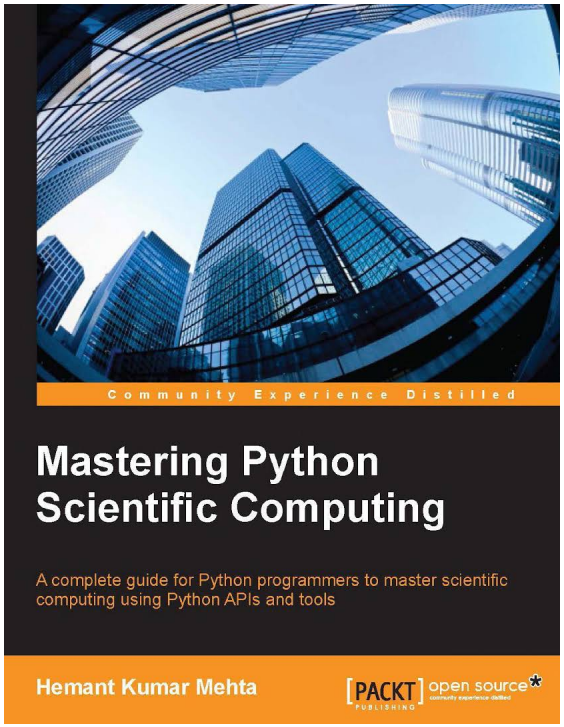
- Basic programming skills
- Built-in data containers
- File I/O
- Data visualization

Scoring

- At least one exercise in every class: 30%
- Two programming exams
  - Midterm: 35%
  - Final: 35%

Textbook

- Hemant Kumar Mehta, ***Mastering Python Scientific Computing***, Packt Publishing, Sep. 2015.



Introduction to Python

- Why Python?
  - The learning of Python is more easy than other programming language for a beginner.
  - You can quickly design a useful application.
  - You can have increased motivation when you feel some sense of accomplishment
  - Cross-platform and portable code
  - Many free resources can be found on the internet
  - Development of AI applications
    - Scikit-learn
    - TensorFlow
    - Keras

The History of Python

- The concept of Python was started in the late 1980s
- The implementation of Python began in Dec.1989 by **Guido van Rossum** at Centrum Wiskunde & Informatica (CWI) in the Netherlands.
  - 16 Oct. 2000, Python 2.0 released.
  - 3 Dec. 2008, Python 3.0 released.
  - 28 Mar. 2018, Python 3.6.5 released.

Guido van Rossum

Programming Environment

- Typical installation
  - Installing python 3.x
    - https://www.python.org/
  - Upgrading pip
    - python -m pip install --upgrade pip
  - Installing common used libraries
    - pip install --user numpy scipy matplotlib ipython jupyter pandas sympy nose
  - About Jupyter Notebook
    - A web-based python IDE
    - https://jupyter.org/

In Windows, you need to append the paths of Python and Jupyter to PATH, which is a system environment variable for default paths.

Programming Environment

- Easiest installation: Anaconda
  - https://www.anaconda.com/
  - Download the 64-bit installer of Python 3.x
    - Windows
    - Mac OS X
    - Linux

Basic programming skills

- Mathematical and logic expressions
- Conditional statements
- Iteration statements
- Functions
- Text processing
- Data containers
- Sorting and search
- Data type definition (designing a class)
- Numerical computing

File processing

- Excel files
  - **openpyxl**
    - A library for excel file reading and writing.
  - **pandas**
    - A library for data analysis.
- Text files
  - Simple file I/O

Data Visualization

- pyplot