

Course structure

Numerical Methods for Physicists

Introduction

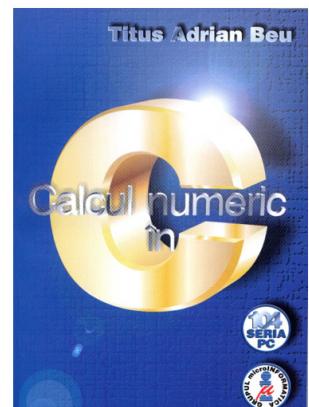
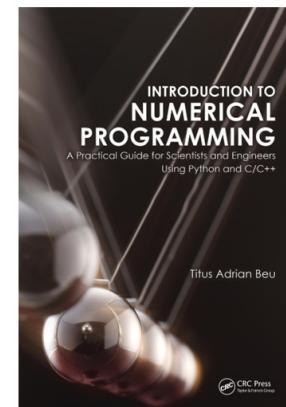
Titus Beu
 University "Babeş-Bolyai"
 Department of Biomolecular Physics
 Cluj-Napoca, Romania

1. Introduction
2. Basic programming techniques in Python
3. Elements of scientific graphics
4. Evaluation of functions
5. Systems of Linear Algebraic Equations
6. Eigenvalue problems
7. Approximation of tabulated functions – interpolation and regression
8. Integration of functions
9. Ordinary Differential Equations
10. Partial Differential Equations
11. Random Walks

Bibliography

- Beu, T. A., *Introduction to Numerical Programming: A Practical Guide for Scientists and Engineers Using Python and C/C++*, (CRC Press / Taylor & Francis, 2014).
- Beu, T. A., *Calcul numeric în C*, Ediția a III-a, (Editura Albastră, Microinformatica, Cluj-Napoca, 2004).
- <http://phys.ubbcluj.ro/~tbeu/INP/index.html>
- <https://www.python.org/>
- Kernighan, B. and P. J. Plauger (1978). *The Elements of Programming Style*, 2nd Edition. New York: McGraw-Hill.
- Knuth, D. E. (1998). *The Art of Computer Programming* (3rd Ed.), vol. 1: *Fundamental Algorithms*. Reading, MA: Addison-Wesley Professional.
- E. Yourdon, *Techniques of Program Structure and Design* (Prentice-Hall, Englewood Cliffs, NJ, 1975).

Introduction to Numerical Programming

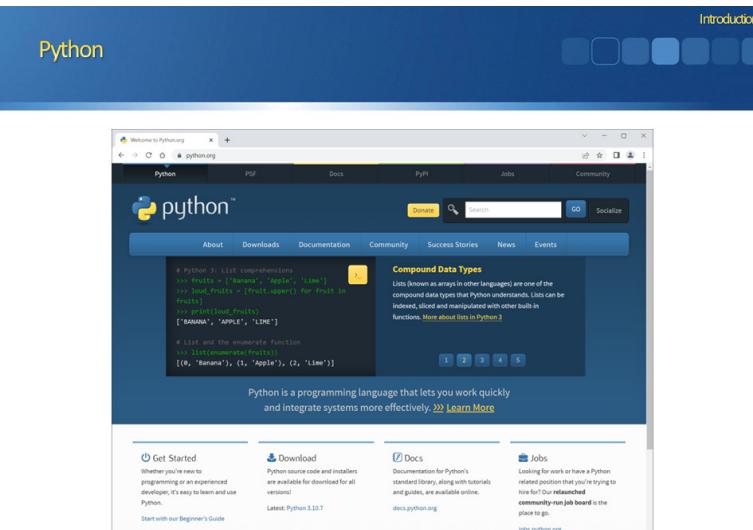


Introduction to Numerical Programming

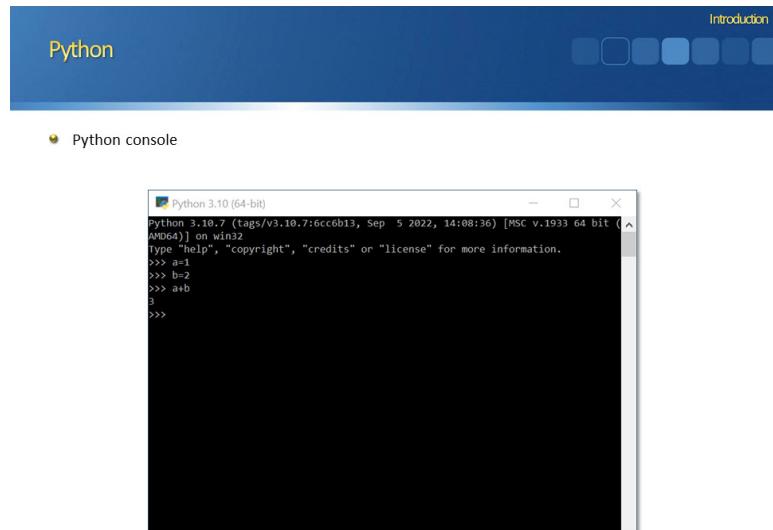
The screenshot shows the book's page on the Taylor & Francis website. It includes the book cover, title, author, and publication details. Below the book info, there are sections for 'About Us', 'Subjects', 'Browse', 'Products', 'Request a trial', 'Librarian Resources', and 'What's New'. There are also links for 'GET ACCESS', 'PREVIEW PDF', and 'GO TO ROUTLEDGE.COM'.

Introduction to Numerical Programming

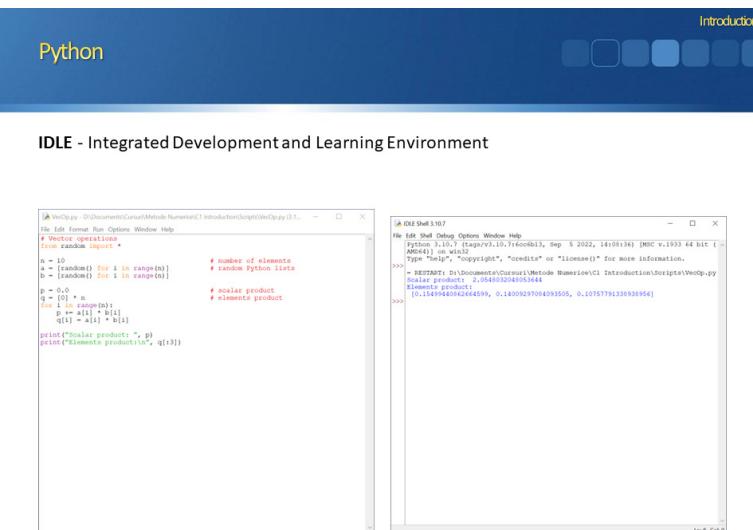
The screenshot shows the book's website. It features the book cover, title, and author. The page includes navigation links for 'HOME', 'AUTHOR', 'CONTENTS', 'LIBRARIES', 'GRAPHICS', and 'PROGRAMS'. There is also a section for 'RELATED BOOKS' and a copyright notice at the bottom.



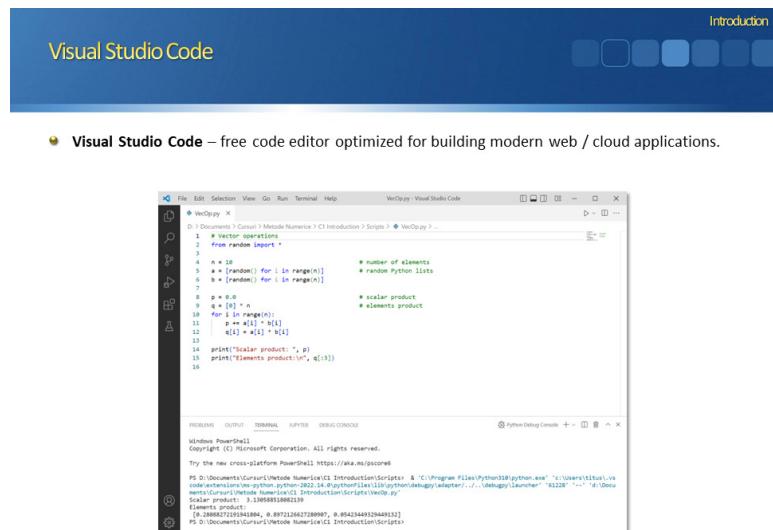
Titus Beu 2022



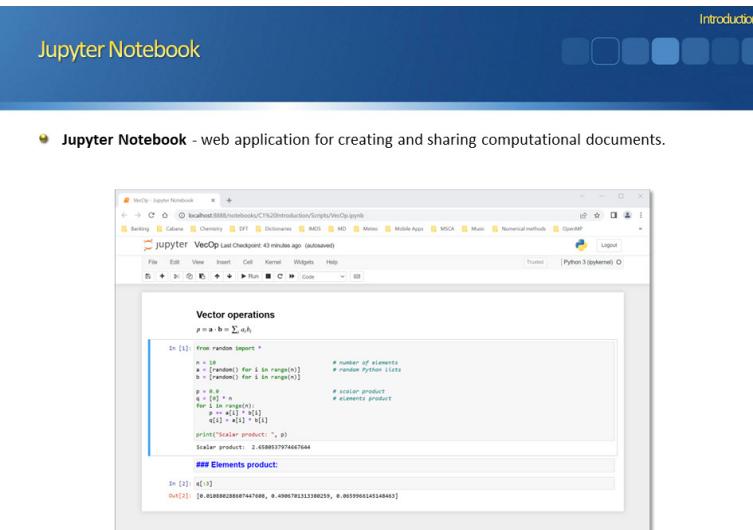
7 Titus Beu 2022



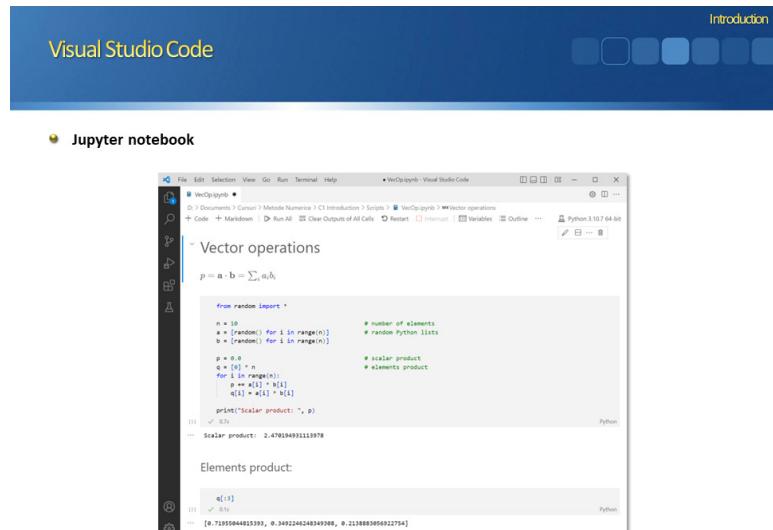
Titus Bey 2022



9 Titus Bey 2023



Titus Beu 2022



11 Titus Beu 2022

The NumPy website homepage features a dark header with the NumPy logo and navigation links. Below the header, a banner announces "NumPy 1.23.0 released". The main content area is divided into several sections: "POWERFUL N-DIMENSIONAL ARRAYS", "NUMERICAL COMPUTING TOOLS", "INTEROPERABLE", "PERFORMANT", "EASY TO USE", and "OPEN SOURCE". Each section contains a brief description and a small image.

Titus Beu 2022

What is NumPy?<https://numpy.org/doc/stable/user/whatisnumpy.html>

- NumPy is the **fundamental package/library for scientific computing** in Python.
- At the center of NumPy is the ndarray object - encapsulates n-dimensional contiguous arrays of homogeneous data types.
- For performance, many operations with ndarrays are performed in compiled code, using efficient vectorization (low-level parallelization).
- Differences between NumPy arrays and standard Python sequences:
 - NumPy arrays have a fixed size (set at creation), unlike Python lists (which can grow dynamically). Changing the size of an ndarray will create a new array and delete the original.
 - The elements in a NumPy array are all of the same data type and occupy the same size in memory. Exception: arrays of Python or NumPy objects.
 - NumPy arrays facilitate advanced operations on large data sets, which are executed more efficiently than is possible using Python's built-in sequences.

13 Titus Beu 2022

14

```

# VecOpNumpy.py - E:\Python\NumPy\VecOpNumpy.py (3.10.7)
# Vector operations
import numpy as np
from random import *
from time import time

n = 5000000000 # number of elements
a = [random() for i in range(n)] # random Python lists
b = [random() for i in range(n)]
time0 = time()

q = 0.0
q = [0] * n # scalar product
for i in range(n):
    p := a[i] * b[i]
    q[i] += a[i] * b[i]
print("Python0 : ", time() - time0, "sec")
print("NumPy : ", time() - time0, "sec")

# convert to NumPy arrays
an = np.array(a)
bn = np.array(b)
time0 = time()

pn = an * bn # scalar product
qn = an * bn # elements product
print("NumPy : ", time() - time0, "sec")
print("NumPy : ", time() - time0, "sec")

```

Titus Beu 2022

The NumPy ecosystem page features a dark header with the NumPy logo and navigation links. Below the header, a banner says "ECOSYSTEM". The main content area is divided into several sections: SCIENTIFIC DOMAINS, ARRAY LIBRARIES, DATA SCIENCE, MACHINE LEARNING, and VISUALIZATION. Each section contains a grid of icons representing different fields and their associated Python packages.

15 Titus Beu 2022

16

The Matplotlib website homepage features a dark header with the Matplotlib logo and navigation links. Below the header, a banner says "Matplotlib: Visualization with Python". The main content area includes a "pcolormesh(X, Y, Z)" heatmap example, a list of features, and links to "Try Matplotlib (on Binder)". At the bottom, there are links to "Getting Started", "Examples", "Reference", "Cheat Sheets", and "Documentation".

Titus Beu 2022

```

# SinePlot.py - E:\Python\Matplotlib\LinePlot.py (3.10.7)
# Plot several data sets
import matplotlib.pyplot as plt
x = np.linspace(0, np.pi, 60)
plt.figure(figsize=(6, 3), layout="constrained") # pyplot-style
plt.plot(x, np.sin(x), label="#sin(x)") # Plot several data sets
plt.plot(x, np.sin(2*x), label="#sin(2x)")
plt.plot(x, np.sin(3*x), label="#sin(3x)")

plt.xlabel("x") # Add x-label
plt.ylabel("y") # Add y-label
plt.title("Plot") # Add title
plt.legend() # Add legend
plt.show()

```

17 Titus Beu 2022

18

Matplotlib

