Book Review
Python Scripting for Computational Science

Jason Papathanasiou, University of Macedonia, Greece

Python Scripting for Computational Science
(3rd ed., 758 pp.)
New York: Springer.
ISBN: 978-3540739159

The Python programming language (Van Rossum & Drake, 2009, www.python.org) is a modern, powerful, object oriented and open source tool, running on most available platforms. It is still under heavy development and continually refined; it has a strong supporter base and some even claim that its popularity has surpassed that of FORTRAN. For engineering applications though, environments like FORTRAN and MATLAB are still dominant, despite the specific advantages claimed by the Python community and the undeniable argument that the language is free to obtain and use.

The author of the book, Langtangen, clearly has the experience required to build his case; this is a book not for the newcomers to the programming world, quite the opposite actually; it is directed to already skilled researchers wishing to broaden their perspectives and grasp the use of new and promising tools. Knowledge of Python is indeed considered a prerequisite, but the author offers an introductory chapter anyway and he recently released another more introductory title by the same publisher (Langtangen, 2009). This is not however the only available book in the domain, as J. Kiusalaas (2010) offers another excellent title, already in the second edition, more focused on Python facilities for numerical computations.

There is a plethora of editions for other languages like C/C++ and Java; Python still lags behind. There have been published a lot of introductory titles, but not many for complicated topics as those in this book. This third edition provides an update to the original in 2004 and is divided in 12 chapters, plus a couple of annexes. A website with the code available for download is mentioned in the preface, where interestingly enough a pdf version is available as well, with the printing capabilities deactivated.

Chapter 1 explains the difference in working with a scripting language like Python compared with other more traditional languages like FORTRAN, C/C++ and Java. Scripting as a concept is explained versus traditional programming techniques and some remarks on the efficiency of Python are made. A list of

DOI: 10.4018/joris.2010100105
specific arguments of why to choose, at the end of the day, Python is presented and the final part of the chapter is about the preparations needed for working with this book, the more important being the installation of the source code and other tools.

Chapter 2 begins with the familiar to most programmers ‘Hello World’ script and moves on to file and data manipulation. Code efficiency is discussed to some extent, making a special point that Python, using certain packages, can run at a speed well comparable to that of C++! The Gnuplot package, a command-line driven graphing utility known to the science community, is discussed as regards its integration with Python, offering on the same time plenty of visualization examples. Issues like generating HTML and LATEX reports come next and the chapter ends with a primer on conducting numerical experiments.

Chapter 3 deals with the basics of Python and spans around 55 pages. My first impression would be that this chapter is really not necessary, as the book requires prior knowledge of the language anyway; however it can serve as a quick reference guide. Exercises and hints are scattered around the text and some tricky points like the ‘call by reference’ procedure regarding variable manipulation in functions are cleared.

Chapter 4, titled ‘Numerical Computing in Python’ is where the real fun begins. The NumPy package is presented and the array manipulation modules plus the array computational efficiency of the package are thoroughly tested. Chapter 5, followed at a later stage by chapters 9 and 10, are all three about mixed language programming, one of the main assets, according to the author, of Python. Python integration techniques with FORTRAN, C and C++ and relevant issues like the generation of wrapper code and the calling and combining of functions among different languages are discussed extensively. The later 2 chapters are about FORTRAN and C/C++ programming with NumPy arrays; they can be considered one the main parts of the book.

Chapter 6 discusses GUI programming; Tkinter, the main Python library for graphical user interface programming, is thoroughly described and with plenty of examples. Binding events to widgets, layout managers and an extension of Tkinter called Pmw are explained; this could be a point of further discussion though, as Python possesses other GUI libraries as well more capable than Tkinter/Pmw, like wxPython. Chapter 11 introduces more advanced GUI programming, like the BLT graph widget for adding plot areas in GUIs. Coordinate systems, animation techniques, simulation and visualization scripts and even CGI GUI scripts occupy the rest of the text in this chapter.

Chapter 7 focuses on Web interfaces and CGI programming. For years, Perl was, more or less, the de facto choice for writing CGI scripts and some may claim that in any case, PHP has replaced CGI to a large extent. However, Python possesses similar to Perl capabilities and large scale sophisticated Web applications like Plone, a very active CMS (Content Management System), are proof of that. Langangen deals with Web forms, some debugging issues and an important aspect in Web development, security.

Chapter 8 actually continues where chapter 3 run out; it extends the overview of Python functionalities; it can be considered as an introduction to advanced topics like run-time generation of code, regular expressions and text processing, iterators, generators, tools for handling data in files and object oriented programming. Perhaps the most interesting parts of the chapter are the comparison along the various techniques for storing data in files with respect to CPU time storage and the final part investigating efficiency and the optimization of Python code. Finally, chapter 12 is about running computer experiments and the available tools for this task. Interactive drawing of functions and tips for solving partial differential equations are among the topics presented.

Overall, this is a must have to anyone interested in scientific programming and Python. The text is packed with examples and exercises; code is everywhere. Langangen apparently has years of developing experience and wrote a book offering dense knowledge; a book requiring hard work to comprehend. The text does have its (minor) flaws though; more
performance tips would be very welcomed, especially as regards the implementation of algorithms. Code efficiency is discussed in the text, but a chapter devoted to the topic would be very welcome. The flow of the chapters is also somewhat confusing and not everything is about computational science; the GUI development chapters for instance. Although Python is great for gluing code written in various tools, as the author often points out, the text seems sometimes as a number of autonomous recipes wrapped together without any such glue. I wish also he would elaborate more on specific debugging techniques, as he allocates a mere couple of pages in one of the appendices and tips scattered in the main text; the same goes for unit testing. Again, a devoted chapter would be preferable. Java integration and especially Jython deserves a chapter as well; surely C/C++, MATLAB and FORTRAN are the main choices for scientific programming, but currently Java is the number one platform and is used for all kinds of purposes. I have also personally used the excellent Eclipse editor with the PyDev extension enjoying superb debugging capabili-
ties and all the other virtues of a very good (and free) editor available for both Windows and Linux; there is not any special preference to an editor in the text. Some remarks about Python version 3 and (not fully available at the time of the writing of the 3rd edition) on should be included in a 4th edition of the book when and if published; especially as the 3.xx versions have no background compatibility with the 2.xx versions. As a final remark, I fully recommend this book to anyone interested; Langtangen has done a very good and creative job.

REFERENCES


Jason Papathanasiou is a full time lecturer at the department of Marketing and Operations Management, University of Macedonia, Greece. He holds a Ph.D. in Operational Research and Informatics and a degree in Physics, both from the Aristotle University of Thessaloniki, Greece. He has worked for a number of years as an external lecturer at the Technical Institute of Technology in Thessaloniki and on the Universities of Macedonia and Western Macedonia teaching courses like ‘Computer Applications Programming’, ‘Decision Support Systems’, ‘Web Programming’ and ‘Operational Research’. He has participated in a number of national and EU funded (under the FP6 and FP7 schemes) projects and has published his work in international scientific peer referred journals.