

Python-Programming

Gregor Lingl

Zielgruppe

- Lehrende, die eine fundierte Grundausbildung im Programmieren anbieten wollen.
- Lehrende, die wissen wollen, wie man mit Python als „erster Programmiersprache“ arbeitet.
- Lehrende, die Musterbeispiele und Übungsmaterial für Einführungskurse ins Programmieren suchen.
- Schüler und Studenten, die ernsthaft einen gründlichen aber vergnüglichen Einstieg ins Programmieren suchen.

Bücher, die eine Einführung ins Programmieren mit einem didaktischen Anspruch bieten, sind heute nicht wie Sand am Meer zu finden. Bei vielen Programmierbüchern, die heute in EDV-Buchhandlungen zu finden sind, steht doch meistens das Erlernen einer bestimmten Implementation einer bestimmten Programmiersprache, d. h. ihrer Sprachelemente, ihrer Syntax, eventuell ihrer Bibliotheken und ihrer Entwicklungsumgebung im Vordergrund, wobei sich naturgemäß bei jeder neuen Versionsnummer die Neuauflagen bzw. die auf wenige Kapitelteile und den Buchtitel beschränkten „Neubearbeitungen“ jagen.

Ungeachtet seines Titels ist John Zelles Buch erfreulicherweise ein Buch der anderen Art: eine didaktisch hervorragend aufgebaute Einführung in die grundlegenden Konzepte der Programmierung. Am Anfang einer solchen Arbeit steht für den Autor die Wahl der Programmiersprache. Für dieses Buch ist die Wahl auf Python gefallen. Als Grund dafür gibt der Autor an, dass ihn seine jahrelange Erfahrung mit „CSI-Kursen“, Einführungskursen in die „computer science“, beobachten ließen, welche Schwierigkeiten viele Studenten mit den Grundlagen der Programmierung haben. Er führt das auf die Komplexität der Sprachen und Werkzeuge zurück, die dabei oft verwendet werden. Sein Ziel war daher ein Buch zu schreiben, das diese Grundlagen so einfach wie möglich, aber ohne unzulässig zu vereinfachen, darstellt. Die Verwendung von Python ist für die Erreichung dieses Zieles ein zentraler Punkt, denn Python ist einfach, elegant und unterstützt gut das Experimentieren mit Ideen.

Der Schwerpunkt des Buches liegt in der Darstellung der grundlegenden Prinzipien des Programmierens, wie sie in jeder Programmiersprache angewendet werden müssen. Der Autor vermeidet deshalb sogar an manchen Stellen, spezielle Eigenschaften von Python zu benutzen, die in anderen Sprachen selten zu finden sind.

Der Geist und auch die Gliederung der Darstellung ist ein wenig an den Büchern von Cay Horstmann orientiert, etwa an „Computing Concepts with Java Essentials“ (wie der Autor auch in der Einleitung verrät).

Gliederung

Kapitel 1-2: Computer, Programme und wie man sie schreibt.

Kapitel 3-4: Die Datentypen Zahlen und Strings, Dateien

Kapitel 5: Objekte und Grafik

Kapitel 6: Definition von Funktionen

Kapitel 7-8: Programmkontrolle: Verzweigungen und Schleifen

Kapitel 9: Programmentwurf. top-down und bottom-up (am Beispiel von Simulationen)

Kapitel 10: Definition von Klassen

Kapitel 11: Kollektionen: Listen, Dictionaries

Kapitel 12: OOD, Objektorientierter Entwurf

Kapitel 13: Entwurf von Algorithmen und Rekursion

Schon an dieser Gliederung erkennt man, dass die Lernkurve einen ziemlich flachen Verlauf hat. Die grundlegenden **Kapitel 1-8** werden auf 260 Seiten abgehandelt. Wie es bei einem Lehrbuch, das Python verwendet, nahe liegend ist, wird ausführlicher Gebrauch vom interaktiven Pythoninterpreter gemacht, um die Arbeitsweise der neu eingeführten Sprachelemente direkt zu untersuchen.

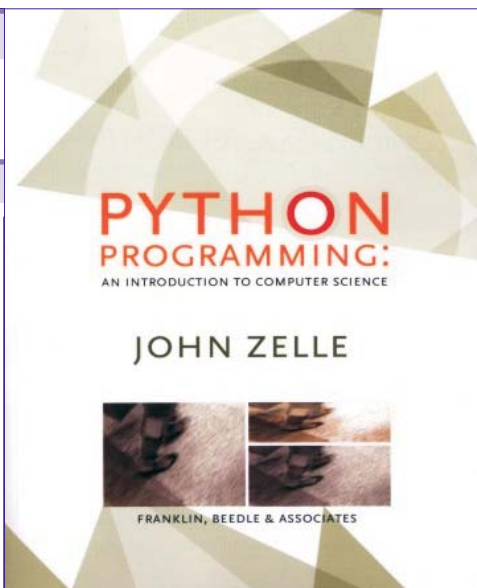
Weiters zeichnet sich die Darstellung dadurch aus, dass sie eine große Anzahl von vollständigen Beispielprogrammen enthält, deren Entwicklung – manchmal in mehreren Varianten – sehr gründlich diskutiert wird. Ein kleines Gustostück ist beispielsweise die Entwicklung von Programmen zur Bestimmung des Maximums dreier Zahlen in **Kapitel 7**. Dafür werden vier verschiedene Entwurfs-Strategien angewendet mit vier verschiedenen Ergebnissen, wobei auch die Vor- und Nachteile jedes Ergebnisprogramms diskutiert werden.

Für das **Kapitel 5**, das in das Programmieren mit Objekten einführt, hat der Autor eigens eine kleine sehr übersichtliche Bibliothek von Graphik-Objekten erstellt: `graphics.py`. (Sie ist auf der mit dem Buch gelieferten CD zu finden, aber auch aus dem Internet herunterzuladen – vgl. den Kasten am Ende der Buchbesprechung.) Sie basiert auf Tkinter, dem in der Python-Distribution enthaltenen GUI-Toolkit. Sie ermöglicht auf einfache Weise interaktive Grafik-Programme zu entwickeln und entlastet den Lernenden von den technischen Details von Tkinter. Diese Grafik-Bibliothek wird in den folgenden Kapiteln häufig eingesetzt um die jeweiligen Inhalte anschaulich und vergnüglich darzustellen.

Die grundlegenden Kapitel erhalten eine Ergänzung in **Kapitel 11** über Listen und Dictionaries. Höhepunkt dieses Kapitels ist die Entwicklung eines Taschenrechner-Programms mit graphischer Benutzeroberfläche, das bereits Klassen und Objekte, insbesondere eine in **Kapitel 10** entwickelte Button-Klasse verwendet. Die Implementation der Grafik findet auch hier mit `graphics.py` ihr Auslangen.

Die **Kapitel 9, 10, 12** und auch **13** sind verschiedenen Verfahren des Programmentwurfs gewidmet. **Kapitel 10** ist vornehmlich der Programmierung von Klassen und Objekten gewidmet. Alle Verfahren werden an Beispielen vorgeführt und ihre Vor- und Nachteile sowie Anwendungsbereiche werden diskutiert. In **Kapitel 12** steht als Anwendungsbeispiel die Entwicklung eines Würfelpoker Programms im Zentrum (Code: `guipoker.py`). Dabei steht die Sachlogik der Anwendung – das „Fachkonzept“ – in einem eigenen Modul, für das davon getrennt ein textbasiertes und ein graphikorientiertes Benutzerinterface erstellt werden, die wahlweise eingesetzt werden können. Zuletzt werden die für die objektorientierte Programmierung wichtigen Konzepte der Datenkapselung, des Polymorphismus und der Vererbung im Rückblick auf die entwickelten Programme diskutiert.

Kapitel 13 steht etwas einzeln da und widmet sich klassischen Themen wie Suchen und Sortieren, aber auch dem Thema Rekursion mit den Klassikern Türme von Hanoi, Fraktal (Koch-Kurve) und auch der Fibonacci-Folge mit



John Zelle, *Python Programming, An Introduction to Computer Science*: Verlag: Franklin, Beedle and Associates, ISBN 1-887902-99-6, Preis: \$ 35.00

einer Analyse der bekannten Probleme bei der rekursiven Programmierung.

Damit bietet das Buch meines Erachtens einen Grundstock an Themen zur „Einführung ins Programmieren“, der alles (und mehr) abdeckt, was diesbezüglich in einem Grundkurs an AHS/BHS gebraucht wird. Es bietet darüber hinaus dies in einer didaktisch wohl durchdachten Gliederung an. Aus langjähriger Unterrichtserfahrung schöpfend behandelt der Autor alle Themen in kleinen Lernschritten mit ausführlichen Besprechungen und Erläuterungen verschiedenster Stolpersteine die dem Lernenden bei der Erarbeitung des Stoffes im Wege stehen könnten.

Einen weiteren überaus brauchbaren Pluspunkt stellen die umfangreichen Übungsabschnitte am Ende jedes Kapitels dar. Gliederung:

- Wiederholungsfragen, die ja/nein - Antworten erwarten oder Multiple Choice Fragen sind,
- Diskussionsfragen, das Nachdenken über spezielle Abschnitte des Kapitels fördern und längere verbale Antworten verlangen, und
- 10 bis 20 Programmieraufgaben (je nach Umfang derselben).

Meines Erachtens kann damit dieses in einem sehr lesbaren Englisch geschriebene Buch in vielfacher Weise Anregung und sehr wirkungsvolle Hilfe bei der Planung und Vorbereitung von Programmier-Grundkursen sein.

Einziger Wermutstropfen mag sein, dass es in Europa nur mit relativ langen Lieferzeiten zu beschaffen ist. Vor einer Bestellung empfehle ich jedenfalls, die im untenstehenden Kasten angeführten Quellen zu begutachten, eventuell die Probekapitel herunter zu laden und durch zu schmökern, mit `graphics.py` zu experimentieren und/oder ein paar Beispielprogramme anzusehen. Von den angegebenen Webadressen aus ist auch eine direkter Kontakt mit dem Verlag möglich.

Informationen und Material

<http://mcs.wartburg.edu/zelle/python/> ist die „Teaching with Python“ Abteilung auf der Website des Autors. Dort steht auch die

- Graphik-Bibliothek `graphics.py` inklusive der Dokumentation im HTML- und PDF-Format und
- der Code aller Beispielprogramme aus dem Buch zum Download bereit.

<http://www.fbeedle.com/99-6.html> ist die zum Buch gehörige Website des Verlags. Sie bietet zum Kennenlernen und Beurteilen die **Kapitel 2: Writing Simple Programs**, **5: Objects and Graphics** und **13: Algorithm Design and Recursion** als PDF-Dateien zum Download an.