Pearls of Algorithm Engineering

There are many textbooks on algorithms that focus on big-O notation and basic design principles. This book offers a unique approach to taking the design and analyses to the level of predictable practical efficiency, discussing core and classic algorithmic problems that arise in the development of big data applications, and presenting elegant solutions of increasing sophistication and efficiency. Solutions are analyzed within the classic RAM model, and the more practically significant external-memory model that allows one to perform I/O complexity evaluations. Chapters cover various data types, including integers, strings, trees, and graphs; algorithmic tools such as sampling, sorting, data compression, and searching in dictionaries and texts; and lastly, recent developments regarding compressed data structures. Algorithmic solutions are accompanied by detailed pseudocode and many running examples, thus enriching the toolboxes of students, researchers, and professionals interested in effective and efficient processing of big data.

Paolo Ferragina is Professor of Algorithms at the University of Pisa, with a post-doc at the Max-Planck Institute for Informatics. He served his university as Vice Rector for ICT (2019–22) and for Applied Research and Innovation (2010–16) and as the Director of the PhD program in Computer Science (2018–20). His research focuses on designing algorithms and data structures for compressing, mining, and retrieving information from big data. The joint recipient of the prestigious 2022 ACM Paris Kanellakis Theory and Practice Award and numerous other international awards, Ferragina has previously collaborated with AT&T, Bloomberg, Google, ST micro-electronics, Tiscali, and Yahoo. His research has produced several patents and has featured in over 170 papers published in renowned conferences and journals. He has spent research periods at the Max Planck Institute for Informatics, the University of North Texas, the Courant Institute at New York University, the MGH/Harvard Medical School, AT&T, Google, IBM Research, and Yahoo.

When I joined Google in 2000, algorithmic problems came up every day. Even strong engineers didn't have all the background they needed to design efficient algorithms. Paolo Ferragina's well-written and concise book helps fill that void. A strong software engineer who masters this material will be an asset.

— *Martin Farach-Colton, Rutgers University*

There are plenty of books on Algorithm Design, but few about Algorithm Engineering. This is one of those rare books on algorithms that pays the necessary attention to the more practical aspects of the process, which become crucial when actual performance matters and render some theoretically appealing algorithms useless in real life. The author is an authority on this challenging path between theory and practice of algorithms, which aims at both conceptually nontrivial and practically relevant solutions. I hope the readers will find the reading as pleasant and inspiring as I did.

— *Gonzalo Navarro, University of Chile*

Ferragina combines his skills as a coding engineer, an algorithmic mathematician, and a pedagogic innovator to engineer a string of pearls made up of beautiful algorithms. In this, beauty dovetails with computational efficiency. His data structures of Stringomics hold the promise for a better understanding of population of genomes and the history of humanity. It belongs in the library of anyone interested in the beauty of code and the code of beauty.

— *Bud Mishra, Courant Institute, New York University*

There are many textbooks on algorithms focusing on big-O notation and general design principles. This book offers a completely unique aspect of taking the design and analyses to the level of predictable practical efficiency. No sacrifices in generality are made, but rather a convenient formalism is developed around external memory efficiency and parallelism provided by modern computers. The benefits of randomization are elegantly used for obtaining simple algorithms whose insightful analyses provide the reader with useful tools to be applied to other settings. This book will be invaluable in broadening the computer science curriculum with a course on algorithm engineering.

— *Veli Mäkinen, University of Helsinki*

Pearls of Algorithm Engineering

PAOLO FERRAGINA
University of Pisa

CAMBRIDGE
UNIVERSITY PRESS

CAMBRIDGE
UNIVERSITY PRESS

Shaftesbury Road, Cambridge CB2 8EA, United Kingdom

One Liberty Plaza, 20th Floor, New York, NY 10006, USA

477 Williamstown Road, Port Melbourne, VIC 3207, Australia

314–321, 3rd Floor, Plot 3, Splendor Forum, Jasola District Centre, New Delhi – 110025, India

103 Penang Road, #05–06/07, Visioncrest Commercial, Singapore 238467

Cambridge University Press is part of Cambridge University Press & Assessment,
a department of the University of Cambridge.

We share the University's mission to contribute to society through the pursuit of
education, learning and research at the highest international levels of excellence.

www.cambridge.org
Information on this title: www.cambridge.org/9781009123280
DOI: 10.1017/9781009128933

First published 2023

A catalogue record for this publication is available from the British Library.

A Cataloging-in-Publication data record for this book is available from the Library of Congress.

ISBN 978-1-009-12328-0 Hardback

to Bebe and Franci

Contents

Preface

This book offers advice for programmers and software engineers: no matter how smart you are, in the field of algorithm engineering, the proverbial five-minutes thinking will not be enough to get a reasonable solution for real-life problems. Real-life problems have reached such a large size, machines have become so complicated, users so demanding, applications so resource-hungry, and algorithmic tools so sophisticated that you cannot improvise being an algorithm engineer: you need to be trained to be one.

The following chapters bear witness to this situation by introducing challenging problems together with elegant and efficient algorithmic techniques to solve them. In selecting their topics I was driven by a twofold goal: on the one hand, to provide readers with an *algorithm engineering toolbox* that will help them tackle programming problems involving massive datasets; and, on the other hand, to collect together the scientific material that I would have liked to have been taught when I was a master's/PhD student. Some of the following sections, typically (though not always) at the ends of chapters, have their titles completed by the superscripted symbol ∞; this indicates more advanced contents that the reader can skip without jeopardizing the reading of the book. As a final note for the reader passionate of programming, I point out another specialty of this book related to the fact that array indexes start from 1, instead of the classic 0 usually adopted in coding, because in this way algorithms may be paraphrased more easily and formulas do not become complicated by the presence of ± 1.

The style and content of these chapters is the result of many hours of illuminating, and sometimes hard and fatiguing, discussions with many fellow researchers and students. Some of these lectures comprised the courses on information retrieval and advanced algorithms that I have been teaching at the University of Pisa and in various international PhD schools since 2004. In particular, a preliminary draft of these notes was prepared by the students of the Algorithm Engineering course in the master's degree of Computer Science and Networking in September–December 2009, in a collaboration between the University of Pisa and the Sant'Anna School of Advanced Studies. Some other notes were prepared by the PhD students attending the course on Advanced Algorithms for Massive DataSets that I taught at the Bertinoro International Spring School (BISS), held in March 2010 (Bertinoro, Italy). I used these draft notes as a seed for some of the following chapters. Of course, many changes have been made to these notes in the following years, thanks to the corrections and suggestions made by the many students who attended the Algorithm Engineering courses since 2010.

Special thanks go to Antonio Boffa, Andrea Guerra, Francesco Tosoni, and Giorgio Vinciguerra for carefully reading the latest version of this book, to Gemma Martini for contributing to Chapter 15, and to Riccardo Manetti for the figures in tikz. I also thank my PhD students and colleagues: Jyrki Alakuijala, Ricardo Baeza-Yates, Lorenzo Bellomo, Massi Ciaramita, Marco Cornolti, Martin Farach-Colton, Andrea Farruggia, Raffaele Giancarlo, Roberto Grossi, Antonio Gullì, Luigi Laura, Veli Makinen, Giovanni Manzini, Kurt Mehlhorn, Ulli Meyer, Bud Mishra, S. Muthukrishnan, Gonzalo Navarro, Igor Nitto, Linda Pagli, Francesco Piccinno, Luca Pinello, Marco Ponza, Prabhakar Raghavan, Peter Sanders, and Rossano Venturini, Jeff S. Vitter, for the many hours of fascinating and challenging discussions about these topics over the years. My final and warmest thanks go to Fabrizio Luccio, my mentor, who has continuously stimulated my research passion and instilled in me the desire for and pleasure in teaching and writing as simply and clearly as possible… but not too much simply. You will be the judge of whether I have succeeded in achieving this goal with the present book.

My ultimate hope is that in reading the following pages you will be pervaded by the same pleasure and excitement that filled me when I met these algorithmic solutions for the first time. If this is the case, please read more about algorithms to find inspiration for your academic and professional work. It is still the case that *computer programming is an art*, but you need good tools to express it at the highest level of beauty.

P. F.