

PREFACE

[Return to Table of Contents](#) [Next Chapter](#)

This book provides a comprehensive introduction to the modern study of computer algorithms. It presents many algorithms and covers them in considerable depth, yet makes their design and analysis accessible to all levels of readers. We have tried to keep explanations elementary without sacrificing depth of coverage or mathematical rigor.

Each chapter presents an algorithm, a design technique, an application area, or a related topic. Algorithms are described in English and in a "pseudocode" designed to be readable by anyone who has done a little programming. The book contains over 260 figures illustrating how the algorithms work. Since we emphasize *efficiency* as a design criterion, we include careful analyses of the running times of all our algorithms.

The text is intended primarily for use in undergraduate or graduate courses in algorithms or data structures. Because it discusses engineering issues in algorithm design, as well as mathematical aspects, it is equally well suited for self-study by technical professionals.

To the teacher

This book is designed to be both versatile and complete. You will find it useful for a variety of courses, from an undergraduate course in data structures up through a graduate course in algorithms. Because we have provided considerably more material than can fit in a typical one-term course, you should think of the book as a "buffet" or "smorgasbord" from which you can pick and choose the material that best supports the course you wish to teach.

You should find it easy to organize your course around just the chapters you need. We have made chapters relatively self-contained, so that you need not worry about an unexpected and unnecessary dependence of one chapter on another. Each chapter presents the easier material first and the more difficult material later, with section boundaries marking natural stopping points. In an undergraduate course, you might use only the earlier sections from a chapter; in a graduate course, you might cover the entire chapter.

We have included over 900 exercises and over 120 problems. Each section ends with exercises, and each chapter ends with problems. The exercises are generally short questions that test basic mastery of the material. Some are simple self-check thought exercises, whereas others are suitable as assigned homework. The problems are more elaborate case studies that often introduce new material; they typically consist of several questions that lead the student through the steps required to arrive at a solution.

We have starred (*) the sections and exercises that are more suitable for graduate students than for undergraduates. A starred section is not necessarily more difficult than an unstarred one, but it may require an understanding of more advanced mathematics.

Likewise, starred exercises may require an advanced background or more than average creativity.

To the student

We hope that this textbook provides you with an enjoyable introduction to the field of algorithms. We have attempted to make every algorithm accessible and interesting. To help you when you encounter unfamiliar or difficult algorithms, we describe each one in a step-by-step manner. We also provide careful explanations of the mathematics needed to understand the analysis of the algorithms. If you already have some familiarity with a topic, you will find the chapters organized so that you can skim introductory sections and proceed quickly to the more advanced material.

This is a large book, and your class will probably cover only a portion of its material. We have tried, however, to make this a book that will be useful to you now as a course textbook and also later in your career as a mathematical desk reference or an engineering handbook.

What are the prerequisites for reading this book?

- You should have some programming experience. In particular, you should understand recursive procedures and simple data structures such as arrays and linked lists.
- You should have some facility with proofs by mathematical induction. A few portions of the book rely on some knowledge of elementary calculus. Beyond that, Part I of this book teaches you all the mathematical techniques you will need.

To the professional

The wide range of topics in this book makes it an excellent handbook on algorithms. Because each chapter is relatively self-contained, you can focus in on the topics that most interest you.

Most of the algorithms we discuss have great practical utility. We therefore address implementation concerns and other engineering issues. We generally provide practical alternatives to the few algorithms that are primarily of theoretical interest.

If you wish to implement any of the algorithms, you will find the translation of our pseudocode into your favorite programming language a fairly straightforward task. The pseudocode is designed to present each algorithm clearly and succinctly. Consequently, we do not address error-handling and other software-engineering issues that require specific assumptions about your programming environment. We attempt to present each algorithm simply and directly without allowing the idiosyncrasies of a particular programming language to obscure its essence.

Errors

A book of this length is certain to contain errors and omissions. If you find any errors or have other constructive suggestions, we would appreciate hearing from you. We particularly welcome suggestions for new exercises and problems, but please include solutions. You can mail your comments to

Introduction to Algorithms

MIT Laboratory for Computer Science

545 Technology Square

Cambridge, Massachusetts 02139

Alternatively, you can use Internet electronic mail to submit bug reports, request a list of known errors, or make constructive suggestions. To receive instructions, send electronic mail to algorithms@theory.lcs.mit.edu with "Subject: help" in the message header. We regret that we cannot personally respond to all mail.

Acknowledgments

Many friends and colleagues have contributed greatly to the quality of this book. We thank all of you for your help and constructive criticisms.

MIT's Laboratory for Computer Science has provided an ideal working environment. Our colleagues in the laboratory's Theory of Computation Group have been particularly supportive and tolerant of our incessant requests for critical appraisal of chapters. We specifically thank Baruch Awerbuch, Shafi Goldwasser, Leo Guibas, Tom Leighton, Albert Meyer, David Shmoys, and Eva Tardos. Thanks to William Ang, Sally Bemus, Ray Hirschfeld, and Mark Reinhold for keeping our machines (DEC Microvaxes, Apple Macintoshes, and Sun Sparcstations) running and for recompiling T_EX whenever we exceeded a compile-time limit. Thinking Machines Corporation provided partial support for Charles Leiserson to work on this book during a leave of absence from MIT.

Many colleagues have used drafts of this text in courses at other schools. They have suggested numerous corrections and revisions. We particularly wish to thank Richard Beigel (Yale), Andrew Goldberg (Stanford), Joan Lucas (Rutgers), Mark Overmars (Utrecht), Alan Sherman (Tufts and Maryland), and Diane Souvaine (Rutgers).

Many teaching assistants in our courses have made significant contributions to the development of this material. We especially thank Alan Baratz, Bonnie Berger, Aditi Dhagat, Burt Kaliski, Arthur Lent, Andrew Moulton, Marios Papaefthymiou, Cindy Phillips, Mark Reinhold, Phil Rogaway, Flavio Rose, Arie Rudich, Alan Sherman, Cliff Stein, Susmita Sur, Gregory Troxel, and Margaret Tuttle.

Additional valuable technical assistance was provided by many individuals. Denise

Sergent spent many hours in the MIT libraries researching bibliographic references. Maria Sensale, the librarian of our reading room, was always cheerful and helpful. Access to Albert Meyer's personal library saved many hours of library time in preparing the chapter notes. Shlomo Kipnis, Bill Niehaus, and David Wilson proofread old exercises, developed new ones, and wrote notes on their solutions. Marios Papaefthymiou and Gregory Troxel contributed to the indexing. Over the years, our secretaries Inna Radzihovsky, Denise Sergent, Gayle Sherman, and especially Be Hubbard provided endless support in this project, for which we thank them.

Many errors in the early drafts were reported by students. We particularly thank Bobby Blumofe, Bonnie Eisenberg, Raymond Johnson, John Keen, Richard Lethin, Mark Lillibridge, John Pezaris, Steve Ponzio, and Margaret Tuttle for their careful readings.

Colleagues have also provided critical reviews of specific chapters, or information on specific algorithms, for which we are grateful. We especially thank Bill Aiello, Alok Aggarwal, Eric Bach, Vaek Chvtal, Richard Cole, Johan Hastad, Alex Ishii, David Johnson, Joe Kilian, Dina Kravets, Bruce Maggs, Jim Orlin, James Park, Thane Plambeck, Herschel Safer, Jeff Shallit, Cliff Stein, Gil Strang, Bob Tarjan, and Paul Wang. Several of our colleagues also graciously supplied us with problems; we particularly thank Andrew Goldberg, Danny Sleator, and Umesh Vazirani.

This book was typeset with $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$, a macro package for $\text{T}_{\text{E}}\text{X}$. The figures were drawn on an Apple Macintosh using MacDraw II; thanks to Joanna Terry of Claris Corporation and Michael Mahoney of Advanced Computer Graphics for timely customer support. The index was compiled using Windex, a C program written by the authors. The bibliography was prepared using $\text{BIBT}_{\text{E}}\text{X}$. The book was typeset at the American Mathematical Society with an Autologic phototypesetter; thanks to Ralph Youngen of AMS for his help. The cover for the book was designed by Jeannet Leendertse. The book design was created by Rebecca Daw, and Amy Hendrickson implemented the design in $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$.

It has been a pleasure working with The MIT Press and McGraw-Hill in the development of this text. We especially thank Frank Satlow, Terry Ehling, Larry Cohen, and Lorrie Lejeune of The MIT Press and David Shapiro of McGraw-Hill for their encouragement, support, and patience. We are particularly grateful to Larry Cohen for his outstanding copyediting.

Finally, we thank our wives--Nicole Cormen, Linda Lue Leiserson, and Gail Rivest--and our children--Ricky, William, and Debby Leiserson and Alex and Christopher Rivest--for their love and support during the writing of this book. (Alex Rivest also helped with the "Martian birthday paradox.") The love, patience, and encouragement of our families made this project possible. We affectionately dedicate this book to them.

Cambridge, Massachusetts

March 1990

THOMAS H. CORMEN

CHARLES E. LEISERSON

RONALD L. RIVEST

Go to [Chapter 1](#) Back to [Table of Contents](#)