

Preface

The excitement of discrete mathematics comes from discovery, the act of uncovering beautiful and important properties of graphs, partitions, permutations, and other combinatorial objects. The appeal of discrete mathematics is its concreteness, that you can draw these objects on a blackboard and get a feel for them in a way quite different from more abstract areas of mathematics.

Unfortunately, only very small structures can be built on a blackboard; computers are needed to experiment with larger ones. The goal of *Combinatorica* is to advance the study of combinatorics and graph theory by making a wide variety of functions available for active experimentation. Together, this book and *Combinatorica* provide a unique resource for discovering discrete mathematics.

■ About *Combinatorica*

Combinatorica has been perhaps the most widely used software for teaching and research in discrete mathematics since its initial release in 1990. *Combinatorica* is an extension to *Mathematica*, which has been used by researchers in mathematics, physics, computer science, economics, and anthropology. *Combinatorica* received a 1991 EDUCOM Higher Education Software Award for Distinguished Mathematics Software and has been employed in teaching from grade school to graduate school.

But times change, in this case for the good. Desktop computers (and *Mathematica*) are now more than 100 times faster than when *Combinatorica* was originally developed. Computational problems unimaginable on research machines then can now be done at home by high school students. *Mathematica* itself has gone through several versions, resulting in a significantly improved user interface, more functionality, better performance, and improved typesetting facilities.

This book presents the second-generation version of *Combinatorica*, which is a dramatic improvement over the original. Enhancements since the previous version include:

- *Improved Performance* – We have made incredible strides in efficiency since the original *Combinatorica*. All examples in the original edition of *Implementing Discrete Mathematics* involved graphs of fewer than 100 vertices, because larger examples were either impossible or hopelessly slow. Now we can work on interesting graphs with tens of thousands of vertices and hundreds of thousands of edges. Indeed, all the examples in this book run comfortably on a two-year-old laptop computer that has seen better days!
- *Improved Graph Representation and Visualization* – The *Combinatorica* graph structure has been completely revamped, enabling us to efficiently represent very large graphs, with edge/vertex weights, labels, and graphics. These improvements make *Combinatorica* a viable platform for developing significant graph applications in *Mathematica*. Color graphics displays and printers are now ubiquitous, although they were prohibitively expensive back in 1990. In response, we have brought a new degree of color and style to *Combinatorica* graphics. Graphs can now be drawn with a variety of edge and vertex presentations and give

complete freedom to highlight and position graph elements. We provide the power to animate interesting structures and algorithms on graphs, and to transform these videos to animated GIF files for Web pages.

- *New Functionality* – *Combinatorica* now adds more than 450 functions for combinatorics and graph theory to *Mathematica* – twice as many as the previous version. Existing functions have been made more sophisticated, with new options and better algorithms. The new *Combinatorica* provides extensive support for Pólya’s theory of counting and algebraic combinatorics, as well as new combinatorial objects such as set partitions and restricted growth functions. These changes reflect demands from our users, recent developments in combinatorics and graph theory, and the additional functionality of new releases of *Mathematica*.

Combinatorica is included with every copy of *Mathematica* as `DiscreteMath`Combinatorica``. The *Combinatorica* described in this book appears in *Mathematica* Version 4.2 and beyond, but our package is backwards compatible with earlier versions of *Mathematica*. The latest release of *Combinatorica* is available from our Web site <http://www.combinatorica.com>, where we maintain software, data, and other materials of interest to the *Combinatorica* community. Register at www.combinatorica.com to hear about updates to the package.

■ What’s Between the Covers

This book is a successor to *Implementing Discrete Mathematics* [Ski90], the original description of *Combinatorica*. This book, like the program itself, is a complete rewrite. Here, we present a general computational approach to discrete mathematics that will serve as the definitive user’s guide/manual to *Combinatorica*. What’s new?

- *Introductory User’s Guide* – *Combinatorica* has been widely used by groups of people who know relatively little graph theory, in interesting ways that we had never envisioned. We now begin with a substantial user’s guide containing numerous examples of how to use the package and surveying the range of what *Combinatorica* can do.
- *Conversion Guide* – We have tried our best to provide backwards compatibility, but in a small number of cases providing backwards compatibility conflicted with other goals such as increasing efficiency or providing better functionality. Thus a small number of *Combinatorica* functions will not work they way they used to in the old package. We include a section that enumerates functions that have changed and give new *Combinatorica* code that replicates the old behavior.
- *Selected Function Implementations* – The previous version of this book contained literally every line of *Mathematica* code that made up *Combinatorica*. With the new package three times larger than the old one, maintaining this policy is neither feasible nor desirable. Instead, we present the implementations of only the most enlightening and important algorithms. These have been kept away from the preliminary chapter to aid and comfort new *Combinatorica* users.
- *Expanded Tutorial Component* – We have significantly improved and expanded the tutorial component of this book. Now, in addition to being a reference, supplement, or guide to self-study,

it can serve as a textbook for *Mathematica*-based undergraduate discrete mathematics courses. In particular, the first author has taught such courses at both IIT Bombay and the University of Iowa. This is the only combinatorics/graph theory textbook we are aware of that is built around significant software.

There is more than enough material here to teach full-semester, experimentally enhanced courses in combinatorics and graph theory. *Mathematica* is now quite well established at many colleges and universities around the world. It is highly likely that your school already has a site license for *Mathematica*. Contact your campus computing center if you need more information on this. Students in such schools/departments will particularly benefit from our approach.

- *New Exercises* – Finally, we have included three interesting classes of exercises at the end of each chapter. Some are thought problems, typically requesting proofs of well-known or interesting theorems in combinatorics or graph theory that we illustrate as examples of *Combinatorica* in action. Some are programming exercises, where the reader is encouraged to extend or improve existing *Combinatorica* functions. The remainder suggest interesting discrete mathematics experiments to conduct using *Combinatorica*.

This book concentrates on two distinct areas in discrete mathematics. The first section deals with combinatorics, loosely defined as the study of counting. We provide functions for generating combinatorial objects such as permutations, partitions, and Young tableaux, as well as for studying various aspects of these structures.

The second section considers graph theory, which can be defined equally loosely as the study of binary relations. We consider a wide variety of graphs and provide functions to generate them. Although graphs are combinatorial structures, understanding them requires pictures or embeddings. Thus we provide functions to create a variety of graph embeddings, enabling a given structure to be viewed in several different ways. Algorithmic graph theory is an important interface between mathematics and computer science, and in this text we present a variety of polynomial- and exponential-time algorithms to solve computational problems in graph theory.

These two sections are relatively independent of each other, so feel free to jump in the middle if you are primarily interested in graph theory.

This book is designed as a guide to manipulating discrete structures. You will find no formal proofs in this book, but enough discussion to understand and appreciate the literally hundreds of algorithms and theorems contained within. Further, we provide extensive references as pointers to the appropriate results. Since the body of the text contains the most interesting of more than 450 *Combinatorica* functions, it is also an excellent guide for writing your own *Mathematica* programs. We include a brief guide to *Mathematica* for the uninitiated to help in this regard.

This book is also a complete reference manual for using *Combinatorica* to explore discrete mathematics. As you read the book we urge you to play with the package. Documentation for all *Combinatorica* functions appears in the *Glossary of Functions* at the end of the book, and cross-references to examples using a particular function in an interesting way appear in the index.

■ Why Mathematica?

At its initial release in 1990, *Combinatorica* was the largest package ever written for the then-recently released *Mathematica*. Today *Mathematica* has established itself as the mathematical tool of choice, with more than one million users.

Building a discrete mathematics package in *Mathematica* has several advantages. Arbitrary precision arithmetic means we are free of the burdens of computer word length. Where appropriate, we have access to portable PostScript graphics, bringing life to graphs and their embeddings. Working with symbolic formulas makes convenient such techniques as generating functions and chromatic polynomials. The freedom of a high-level language with so much mathematics already under the hood liberates us to explore a much larger fraction of what is known about discrete mathematics.

The chief drawback to using such a high-level language as *Mathematica* is that we lose tight control over the time complexity of our algorithms. The model of computation that it presents (the Wolf-RAM?) is dramatically and mysteriously different from the traditional random access machine. In this new version, we have tamed the Wolf-RAM – achieving efficiencies that allow interaction with large and interesting structures in real time.

■ Acknowledgments

We would like to thank all the people who helped make this book and the new *Combinatorica* a reality.

First, we thank the people at Wolfram Research Inc. (WRI). John Novak provided an amazing amount of support throughout the development of the package, guiding us through some of the more obscure aspects of *Mathematica* and providing the interface between us and various *Mathematica* gurus at WRI. Eric Weisstein played with several versions of *Combinatorica*, reported lots of bugs, and contributed code for some of the graph constructors. Arnoud Busing, Darren Glosemeyer, Shiral Devmal, Anna Pakin, and Andy Shiekh of the “bug testing department” bugged us regularly with bug reports. Daniel Lichtblau and Robby Villegas made suggestions that improved the package. Bill White patiently answered questions about WRI’s version of LaTeX. Stephen Wolfram’s interest in *Combinatorica* since its beginning has had much to do with its becoming a reality.

The insights of Joan Trias into graph drawings and representation significantly shaped the graph data structure we employ in the new *Combinatorica*. Lenny Heath commented extensively on initial drafts of the first two chapters. Kaushal Kurapati helped with perl scripts for integrating *Combinatorica* code and *Combinatorica* examples into the text. Eugene Curtin’s package on Pólya theory helped crystallize some of our initial ideas on functionality needed for Pólya-theoretic computations. Levon Lloyd built a nice Java-based editor for *Combinatorica* graphs and King Mak developed interesting algorithm animations – check these out. We especially thank Marty Golumbic for his efforts to bring us together to finish this book and Alberto Apostolico for providing a home where part of it was written.

Lauren Cowles, Alan Harvey, and Elise Oranges of Cambridge University Press helped us throughout the publication process.

Two batches of students at IIT Bombay and one at Iowa survived the first author's teaching experiments with *Combinatorica*. We thank all the people who have downloaded preliminary versions of the new *Combinatorica* and have sent us encouragement and bug reports.

We cannot neglect those who helped with the previous version of *Combinatorica* or the book. Anil Bhansali made a significant contribution to the original *Combinatorica* by writing many of the functions, managing the testing of the code, and performing sundry tasks. The original work on *Combinatorica* was done at Apple Computer in the summer of 1988. Allan Wylde, then of Addison-Wesley, provided encouragement and a means to communicate this to the world. Fred Buckley, Nora Hartsfield, Matthew Markert, Marko Petkovšek, Ilan Vardi, Jürgen Koslowski, and Rick Wilson all provided helpful pre-publication feedback on *Combinatorica* and/or *Implementing Discrete Mathematics*. A variety of other Stony Brook (Philip Hsu, Phil Lewis, Yaw-Ling Lin, Gene Stark, Brian Tria, Alan Tucker, Shipai Zhang), Addison-Wesley (Jan Benes, Laura Likely, Karl Matsumoto), and WRI (Dave Ballman, John Bonadies, Martin Buchholz, Joe Grohens, Igor Rivin, Monte Sayer, Lisa Shipley, Cameron Smith) people all helped out in one way or another.

On a personal level, we thank Amma, Nanna, Sachi, Geeta, and Rama Rao, who provided encouragement and support for the first author. Mom, Dad, Len, and Rob provided moral and emotional support for the second author.

■ Caveat

It is traditional for the author to magnanimously accept the blame for whatever deficiencies remain. We don't. Any errors, deficiencies, or problems in this book are somebody else's fault, but report them to bugs@combinatorica.com so we can determine who to blame.

Sriram Pemmaraju
Dept. of Computer Science
University of Iowa
Iowa City, IA 52240-1419
sriram@combinatorica.com

Steven S. Skiena
Dept. of Computer Science
State University of New York
Stony Brook, NY 11794-4400
skiena@combinatorica.com

■ Dedication

Combinatorialists know how to count their blessings. As a proof of correctness, both of us got the same answer of $2 + \epsilon$ when we did the counting.

We dedicate this book to our respective wives, Santhi and Renee. The writing of this book was a long and complicated process involving cross-country travel, hours at the keyboard, lengthy phone calls, and even fears of terrorist actions and thermonuclear war! We thank them for their love and patience during this period, before, and beyond.

We also dedicate this book to our daughters, Ela and Bonnie. We look forward to the day, not so long from now, when the pretty pictures in this book will help us explain to you the joy of doing and understanding discrete mathematics.

Finally, we dedicate this book to our respective future children, both well on the way to joining us. We will be meeting you both about the time this book is published. We don't know you yet, but certainly know that we love you.