Foundations of Combinatorics with Applications

Edward A. Bender
S. Gill Williamson

Contents

Contents iii

Preface ix

Part I Counting and Listing 1

Preliminary Reading 2

1 Basic Counting 5

Introduction 5

1.1 Lists with Repetitions Allowed 6

Using the Rules of Sum and Product 9

Exercises 10

1.2 Lists with Repetitions Forbidden 11

Exercises 17

1.3 Sets 19

*Error Correcting Codes 27

Exercises 30

1.4 Recursions 32

Exercises 35

1.5 Multisets 36

Exercises 38

Notes and References 39

2 Functions 41

Introduction 41

2.1 Some Basic Terminology 41

Terminology for Sets 41

What are Functions? 42

Exercises 44

2.2 Permutations 45

Exercises 50

2.3 Other Combinatorial Aspects of Functions 51

Monotonic Functions and Unordered Lists 51

Image and Coimage 54

The Pigeonhole Principle 55

Exercises 57

***2.4 Boolean Functions** 59

Exercises 63

Notes and References 64

3 Decision Trees 65

Introduction 65

3.1 Basic Concepts of Decision Trees 65
Exercises 73

*3.2 Ranking and Unranking 75
Calculating RANK 76
Calculating UNRANK 79
*Gray Codes 81
Exercises 83

*3.3 Backtracking 84
Exercises 90

Notes and References 91

4 Sieving Methods 93

Introduction 93
Structures Lacking Things 93
Structures with Symmetries 94

4.1 The Principle of Inclusion and Exclusion 94
Exercises 100
*Bonferroni's Inequalities 102
*Partially Ordered Sets 103
Exercises 104

4.2 Listing Structures with Symmetries 106
Exercises 109

4.3 Counting Structures with Symmetries 111
*Proofs 114
Exercises 116

Notes and References 117

Part II Graphs 119

5 Basic Concepts in Graph Theory 121

Introduction 121

5.1 What is a Graph? 121
Exercises 124

5.2 Equivalence Relations and Unlabeled Graphs 126
Exercises 130

5.3 Paths and Subgraphs 131
Exercises 134

5.4 Trees 136
Exercises 139

5.5 Directed Graphs (Digraphs) 141
Exercises 144

***5.6 Computer Representations of Graphs** 146
Exercises 146

Notes and References 147

6 A Sampler of Graph Topics 149

Introduction 149

6.1 Spanning Trees 150
Minimum Weight Spanning Trees 150
Lineal Spanning Trees 153
Exercises 155

6.2 Coloring Graphs 157
Exercises 161

6.3 Planar Graphs 162
Euler's Relation 163
Exercises 164
The Five Color Theorem 165
Exercises 166
*Algorithmic Questions 167
Exercises 169

6.4 Flows in Networks 170
The Concepts 170
An Algorithm for Constructing a Maximum Flow 173
Exercises 176
*Cut Partitions and Cut Sets 176
Exercises 179

***6.5 Probability and Simple Graphs** 180
Exercises 186

6.6 Finite State Machines 188
Turing Machines 188
Finite State Machines and Digraphs 189
Exercises 192

Notes and References 194

Part III Recursion 195

7 Induction and Recursion 197

Introduction 197

7.1 Inductive Proofs and Recursive Equations 198
Exercises 203

7.2 Thinking Recursively 204
Exercises 209

7.3 Recursive Algorithms 210
Obtaining Information: Merge Sorting 210
Local Descriptions 212
*Computer Implementation 215
Exercises 217

7.4 Divide and Conquer 220
Exercises 223

Notes and References 225

8 Sorting Theory 227

Introduction 227

8.1 Limits on Speed 228
Motivation and Proof of the Theorem 229
Exercises 231

8.2 Software Sorts 232
Binary Insertion Sort 233
Bucket Sort 234
Merge Sorts 235
Quicksort 235
Heapsort 236
Exercises 237

8.3 Sorting Networks 238
8.3.1 Speed and Cost 238
Parallelism 239
How Fast Can a Network Be? 240
How Cheap Can a Network Be? 240
Exercises 240
8.3.2 Proving That a Network Sorts 241
The Batcher Sort 243
Exercises 244

Notes and References 245

9 Rooted Plane Trees 247

Introduction 247

9.1 Traversing Trees 248
Depth First Traversals 249
Exercises 251

9.2 Grammars and RP-Trees 253
Exercises 258

***9.3 Unlabeled Full Binary RP-Trees** 259
Exercises 265

Notes and References 266

Part IV Generating Functions 267

10 Ordinary Generating Functions 269

Introduction 269

10.1 What are Generating Functions? 269
Exercises 273

10.2 Solving a Single Recursion 275
Exercises 280

10.3 Manipulating Generating Functions 282
Obtaining Recursions 282
Derivatives, Averages and Probability 284
Exercises 291

10.4 The Rules of Sum and Product 291
Exercises 298

Notes and References 304

*11 Generating Function Topics 307

Introduction 307

11.1 Systems of Recursions 308
Exercises 313

11.2 Exponential Generating Functions 315
The Exponential Formula 320
Exercises 326

11.3 Symmetries and Pólya's Theorem 330
Exercises 338

11.4 Asymptotic Estimates 339
Recursions 341
Sums of Positive Terms 344
Generating Functions 349
Exercises 355

Notes and References 359

Appendix A Induction 361
Exercises 365

Appendix B Rates of Growth and Analysis of Algorithms 367

B.1 The Basic Functions 368
Exercises 374

B.2 Doing Arithmetic 376

B.3 NP-Complete Problems 377
Exercises 379

Notes and References 380

Appendix C Basic Probability 381

C.1 Probability Spaces and Random Variables 381

C.2 Expectation and Variance 384

Appendix D Partial Fractions 387
Theory 387
Computations 388

Solutions to Odd Exercises and Most Appendix Exercises 393

Index 461

Preface

Combinatorics, the mathematics of the discrete, has blossomed in this generation. On the theoretical side, a variety of tools, concepts and insights have been developed that allow us to solve previously intractable problems, formulate new problems and connect previously unrelated topics. On the applied side, scientists from physicists to biologists have found combinatorics essential in their research. In all of this, the interaction between computer science and mathematics stands out as a major impetus for theoretical developments and for applications of combinatorics. This text provides an introduction to the mathematical foundations of this interaction and to some of its results.

Advice to Students

This book does not assume any previous knowledge of combinatorics or discrete mathematics. Except for a few items which can easily be skipped over and some of the material on "generating functions" in Part IV, calculus is not required. What is required is a certain level of ability or "sophistication" in dealing with mathematical concepts. The level of mathematical sophistication that is needed is about the same as that required in a solid beginning calculus course.

You may have noticed similarities and differences in how you think about various fields of mathematics such as algebra and geometry. In fact, you may have found some areas more interesting or more difficult than others partially because of the different thought patterns required. The field of combinatorics will also require you to develop some new thought patterns. This can sometimes be a difficult and frustrating process. Here is where patience, mathematical sophistication and a willingness to ask "stupid questions" can all be helpful.

Combinatorics differs as much from mathematics you are likely to have studied previously as algebra differs from geometry. Some people find this disorienting and others find it fascinating. The introductions to the parts and to the chapters can help you orient yourself as you learn about combinatorics. Don't skip them.

Because of the newness of much of combinatorics, a significant portion of the material in this text was only discovered in this generation. Some of the material is closely related to current research. In contrast, the other mathematics courses you have had so far probably contained little if anything that was not known in the Nineteenth Century. Welcome to the frontiers!

The Material in this Book

Combinatorics is too big a subject to be done justice in a single text. The selection of material in this text is based on the need to provide a solid introductory course for our students in pure mathematics and in mathematical computer science. Naturally, the material is also heavily influenced by our own interests and prejudices.

Parts I and II deal with two fundamental aspects of combinatorics: enumeration and graph theory. "Enumeration" can mean either counting or listing things. Mathematicians have generally limited their attention to counting, but listing plays an important role in computer science, so we discuss both aspects. After introducing the basic concepts of "graph theory" in Part II, we present a variety of applications of interest in computer science and mathematics. Induction and recursion play a fundamental role in mathematics. The usefulness of recursion in computer science and in its interaction with combinatorics is the subject of Part III. In Part IV we look at "generating functions," a powerful tool for studying counting problems. We have included a variety of material not usually found in introductory texts:

- Trees play an important role. Chapter 3 discusses decision trees with emphasis on ranking and unranking. Chapter 9 is devoted to the theory and application of rooted plane trees. Trees have many practical applications, have an interesting and accessible theory and provide solid examples of inductive proofs and recursive algorithms.
- Software and network sorts are discussed in Chapter 8. We have attempted to provide the overview and theory that is often lacking elsewhere.
- Part IV is devoted to the important topic of generating functions. We could not, in good conscience, deny our students access to the more combinatorial approaches to generating functions that have emerged in recent years. This necessitated a longer treatment than a quick ad hoc treatment would require. Asymptotic analysis of generating functions presented a dilemma. On the one hand, it is very useful; while on the other hand, it cannot be done justice without an introductory course in complex analysis. We chose a somewhat uneasy course: In the last section we presented some rules for analysis that usually work and can be understood without a knowledge of complex variables.

Planning a Course

A variety of courses can be based on this text. Depending on the material covered, the pace at which it is done and the level of rigor required of the students, this book could be used in a challenging lower division course, in an upper division course for engineering, science or mathematics students, or in a beginning graduate course. There are a number of possibilities for choosing material suitable for each of these classes. A graduate course could cover the entire text at a leisurely pace in a year or at a very fast pace in a semester. Here are some possibilities for courses with a length of one semester to two quarters, depending on how much parenthesized optional material is included. Parts of an optional chapter can also be used instead of the entire chapter.

- A lower division course: 1, 2.1–2.3, (2.4), 3.1, (4.1), 5.1, (5.2), 5.3–5.5, (6), 7.1, 7.2, (7.3), (8), 9.1, (9.2).
- An upper division or beginning graduate course emphasizing mathematics: 1–3, 4.1, (4.2), 4.3, 5, 6.1, (6.2–6.4), 7, (8) 9.1, (9.2–9.3), 10, (11).
- An upper division or beginning graduate course emphasizing computer science: 1–3, 4.1, 5, 6.1, 6.3, (6.4), (6.5), 7, 8, (9.1), 9.2, 9.3, 10, (11.4).

Asterisks, or stars, (*) appear before various parts of the text to help in course design. Starred exercises are either more difficult than other exercises in that section or depend on starred material. Starred examples are generally more difficult than other material in the chapter. A section or chapter that is not as central as the rest of the material is also starred. The material in Part IV, especially parts of Chapter 11, is more difficult than the rest of the text.

Special thanks are due Fred Kochman whose many helpful comments have enhanced the readability of this manuscript and reduced its errors. This manuscript was developed using TEX, Donald E. Knuth's impressive gift to technical writing.