

Contents

<i>List of Algorithms</i>	<i>page</i> xiii
<i>Foreword</i>	xvii
<i>Acknowledgments</i>	xix
1 Introduction	1
1.1 Why Compact Data Structures?	1
1.2 Why This Book?	3
1.3 Organization	4
1.4 Software Resources	6
1.5 Mathematics and Notation	7
1.6 Bibliographic Notes	10
2 Entropy and Coding	14
2.1 Worst-Case Entropy	14
2.2 Shannon Entropy	16
2.3 Empirical Entropy	17
2.3.1 Bit Sequences	18
2.3.2 Sequences of Symbols	20
2.4 High-Order Entropy	21
2.5 Coding	22
2.6 Huffman Codes	25
2.6.1 Construction	25
2.6.2 Encoding and Decoding	26
2.6.3 Canonical Huffman Codes	27
2.6.4 Better than Huffman	30
2.7 Variable-Length Codes for Integers	30
2.8 Jensen's Inequality	33
2.9 Application: Positional Inverted Indexes	35
2.10 Summary	36
2.11 Bibliographic Notes	36

3	Arrays	39
3.1	Elements of Fixed Size	40
3.2	Elements of Variable Size	45
3.2.1	Sampled Pointers	46
3.2.2	Dense Pointers	47
3.3	Partial Sums	48
3.4	Applications	49
3.4.1	Constant-Time Array Initialization	49
3.4.2	Direct Access Codes	53
3.4.3	Elias-Fano Codes	57
3.4.4	Differential Encodings and Inverted Indexes	59
3.4.5	Compressed Text Collections	59
3.5	Summary	61
3.6	Bibliographic Notes	61
4	Bitvectors	64
4.1	Access	65
4.1.1	Zero-Order Compression	65
4.1.2	High-Order Compression	71
4.2	Rank	73
4.2.1	Sparse Sampling	73
4.2.2	Constant Time	74
4.2.3	Rank on Compressed Bitvectors	76
4.3	Select	78
4.3.1	A Simple Heuristic	78
4.3.2	An $\mathcal{O}(\log \log n)$ Time Solution	80
4.3.3	Constant Time	81
4.4	Very Sparse Bitvectors	82
4.4.1	Constant-Time Select	83
4.4.2	Solving Rank	83
4.4.3	Bitvectors with Runs	86
4.5	Applications	87
4.5.1	Partial Sums Revisited	87
4.5.2	Predecessors and Successors	89
4.5.3	Dictionaries, Sets, and Hashing	91
4.6	Summary	98
4.7	Bibliographic Notes	98
5	Permutations	103
5.1	Inverse Permutations	103
5.2	Powers of Permutations	106
5.3	Compressible Permutations	108
5.4	Applications	115
5.4.1	Two-Dimensional Points	115
5.4.2	Inverted Indexes Revisited	116
5.5	Summary	117
5.6	Bibliographic Notes	117

6 Sequences	120
6.1 Using Permutations	121
6.1.1 Chunk-Level Granularity	121
6.1.2 Operations within a Chunk	123
6.1.3 Construction	126
6.1.4 Space and Time	127
6.2 Wavelet Trees	128
6.2.1 Structure	128
6.2.2 Solving Rank and Select	132
6.2.3 Construction	134
6.2.4 Compressed Wavelet Trees	136
6.2.5 Wavelet Matrices	139
6.3 Alphabet Partitioning	150
6.4 Applications	155
6.4.1 Compressible Permutations Again	155
6.4.2 Compressed Text Collections Revisited	157
6.4.3 Non-positional Inverted Indexes	157
6.4.4 Range Quantile Queries	159
6.4.5 Revisiting Arrays of Variable-Length Cells	160
6.5 Summary	161
6.6 Bibliographic Notes	162
7 Parentheses	167
7.1 A Simple Implementation	170
7.1.1 Range Min-Max Trees	170
7.1.2 Forward and Backward Searching	175
7.1.3 Range Minima and Maxima	180
7.1.4 Rank and Select Operations	188
7.2 Improving the Complexity	188
7.2.1 Queries inside Buckets	190
7.2.2 Forward and Backward Searching	191
7.2.3 Range Minima and Maxima	196
7.2.4 Rank and Select Operations	200
7.3 Multi-Parenthesis Sequences	200
7.3.1 Nearest Marked Ancestors	201
7.4 Applications	202
7.4.1 Succinct Range Minimum Queries	202
7.4.2 XML Documents	204
7.5 Summary	207
7.6 Bibliographic Notes	207
8 Trees	211
8.1 LOUDS: A Simple Representation	212
8.1.1 Binary and Cardinal Trees	219
8.2 Balanced Parentheses	222
8.2.1 Binary Trees Revisited	228

8.3	DFUDS Representation	233
8.3.1	Cardinal Trees Revisited	240
8.4	Labeled Trees	241
8.5	Applications	245
8.5.1	Routing in Minimum Spanning Trees	246
8.5.2	Grammar Compression	248
8.5.3	Tries	252
8.5.4	LZ78 Compression	259
8.5.5	XML and XPath	262
8.5.6	Treaps	264
8.5.7	Integer Functions	266
8.6	Summary	272
8.7	Bibliographic Notes	272
9	Graphs	279
9.1	General Graphs	281
9.1.1	Using Bitvectors	281
9.1.2	Using Sequences	281
9.1.3	Undirected Graphs	284
9.1.4	Labeled Graphs	285
9.1.5	Construction	289
9.2	Clustered Graphs	291
9.2.1	K^2 -Tree Structure	291
9.2.2	Queries	292
9.2.3	Reducing Space	294
9.2.4	Construction	296
9.3	K -Page Graphs	296
9.3.1	One-Page Graphs	297
9.3.2	K -Page Graphs	299
9.3.3	Construction	307
9.4	Planar Graphs	307
9.4.1	Orderly Spanning Trees	308
9.4.2	Triangulations	315
9.4.3	Construction	317
9.5	Applications	327
9.5.1	Binary Relations	327
9.5.2	RDF Datasets	328
9.5.3	Planar Routing	330
9.5.4	Planar Drawings	336
9.6	Summary	338
9.7	Bibliographic Notes	338
10	Grids	347
10.1	Wavelet Trees	348
10.1.1	Counting	350
10.1.2	Reporting	353
10.1.3	Sorted Reporting	355

10.2	K^2 -Trees	357
	10.2.1 Reporting	359
10.3	Weighted Points	362
	10.3.1 Wavelet Trees	362
	10.3.2 K^2 -Trees	365
10.4	Higher Dimensions	371
10.5	Applications	372
	10.5.1 Dominating Points	372
	10.5.2 Geographic Information Systems	373
	10.5.3 Object Visibility	377
	10.5.4 Position-Restricted Searches on Suffix Arrays	379
	10.5.5 Searching for Fuzzy Patterns	380
	10.5.6 Indexed Searching in Grammar-Compressed Text	382
10.6	Summary	388
10.7	Bibliographic Notes	388
11	Texts	395
11.1	Compressed Suffix Arrays	397
	11.1.1 Replacing A with Ψ	398
	11.1.2 Compressing Ψ	399
	11.1.3 Backward Search	401
	11.1.4 Locating and Displaying	403
11.2	The FM-Index	406
11.3	High-Order Compression	409
	11.3.1 The Burrows-Wheeler Transform	409
	11.3.2 High-Order Entropy	410
	11.3.3 Partitioning L into Uniform Chunks	413
	11.3.4 High-Order Compression of Ψ	414
11.4	Construction	415
	11.4.1 Suffix Array Construction	415
	11.4.2 Building the BWT	416
	11.4.3 Building Ψ	418
11.5	Suffix Trees	419
	11.5.1 Longest Common Prefixes	419
	11.5.2 Suffix Tree Operations	420
	11.5.3 A Compact Representation	424
	11.5.4 Construction	426
11.6	Applications	429
	11.6.1 Finding Maximal Substrings of a Pattern	429
	11.6.2 Labeled Trees Revisited	432
	11.6.3 Document Retrieval	438
	11.6.4 XML Retrieval Revisited	441
11.7	Summary	442
11.8	Bibliographic Notes	442

12	Dynamic Structures	450
12.1	Bitvectors	450
12.1.1	Solving Queries	452
12.1.2	Handling Updates	452
12.1.3	Compressed Bitvectors	461
12.2	Arrays and Partial Sums	463
12.3	Sequences	465
12.4	Trees	467
12.4.1	LOUDS Representation	469
12.4.2	BP Representation	472
12.4.3	DFUDS Representation	474
12.4.4	Dynamic Range Min-Max Trees	476
12.4.5	Labeled Trees	479
12.5	Graphs and Grids	480
12.5.1	Dynamic Wavelet Matrices	480
12.5.2	Dynamic k^2 -Trees	482
12.6	Texts	485
12.6.1	Insertions	485
12.6.2	Document Identifiers	486
12.6.3	Samplings	486
12.6.4	Deletions	490
12.7	Memory Allocation	492
12.8	Summary	494
12.9	Bibliographic Notes	494
13	Recent Trends	501
13.1	Encoding Data Structures	502
13.1.1	Effective Entropy	502
13.1.2	The Entropy of RMQs	503
13.1.3	Expected Effective Entropy	504
13.1.4	Other Encoding Problems	504
13.2	Repetitive Text Collections	508
13.2.1	Lempel-Ziv Compression	509
13.2.2	Lempel-Ziv Indexing	513
13.2.3	Faster and Larger Indexes	516
13.2.4	Compressed Suffix Arrays and Trees	519
13.3	Secondary Memory	523
13.3.1	Bitvectors	524
13.3.2	Sequences	527
13.3.3	Trees	528
13.3.4	Grids and Graphs	530
13.3.5	Texts	534
	<i>Index</i>	549