

Contents

<i>Preface</i>	<i>page xi</i>
1 Elementary Structures	1
1.1 Stack	1
1.2 Queue	8
1.3 Double-Ended Queue	16
1.4 Dynamical Allocation of Nodes	16
1.5 Shadow Copies of Array-Based Structures	18
2 Search Trees	23
2.1 Two Models of Search Trees	23
2.2 General Properties and Transformations	26
2.3 Height of a Search Tree	29
2.4 Basic Find, Insert, and Delete	31
2.5 Returning from Leaf to Root	35
2.6 Dealing with Nonunique Keys	37
2.7 Queries for the Keys in an Interval	38
2.8 Building Optimal Search Trees	40
2.9 Converting Trees into Lists	47
2.10 Removing a Tree	48
3 Balanced Search Trees	50
3.1 Height-Balanced Trees	50
3.2 Weight-Balanced Trees	61
3.3 (a, b)- and B-Trees	72
3.4 Red-Black Trees and Trees of Almost Optimal Height	89
3.5 Top-Down Rebalancing for Red-Black Trees	101
3.6 Trees with Constant Update Time at a Known Location	111
3.7 Finger Trees and Level Linking	114

3.8	Trees with Partial Rebuilding: Amortized Analysis	119
3.9	Splay Trees: Adaptive Data Structures	122
3.10	Skip Lists: Randomized Data Structures	135
3.11	Joining and Splitting Balanced Search Trees	143
4	Tree Structures for Sets of Intervals	148
4.1	Interval Trees	148
4.2	Segment Trees	154
4.3	Trees for the Union of Intervals	162
4.4	Trees for Sums of Weighted Intervals	169
4.5	Trees for Interval-Restricted Maximum Sum Queries	174
4.6	Orthogonal Range Trees	182
4.7	Higher-Dimensional Segment Trees	196
4.8	Other Systems of Building Blocks	199
4.9	Range-Counting and the Semigroup Model	202
4.10	kd-Trees and Related Structures	204
5	Heaps	209
5.1	Balanced Search Trees as Heaps	210
5.2	Array-Based Heaps	214
5.3	Heap-Ordered Trees and Half-Ordered Trees	221
5.4	Leftist Heaps	227
5.5	Skew Heaps	235
5.6	Binomial Heaps	239
5.7	Changing Keys in Heaps	248
5.8	Fibonacci Heaps	250
5.9	Heaps of Optimal Complexity	262
5.10	Double-Ended Heap Structures and Multidimensional Heaps	267
5.11	Heap-Related Structures with Constant-Time Updates	271
6	Union-Find and Related Structures	278
6.1	Union-Find: Merging Classes of a Partition	279
6.2	Union-Find with Copies and Dynamic Segment Trees	293
6.3	List Splitting	303
6.4	Problems on Root-Directed Trees	306
6.5	Maintaining a Linear Order	317
7	Data Structure Transformations	321
7.1	Making Structures Dynamic	321
7.2	Making Structures Persistent	330

Contents

ix

8	Data Structures for Strings	335
8.1	Tries and Compressed Tries	336
8.2	Dictionaries Allowing Errors in Queries	356
8.3	Suffix Trees	360
8.4	Suffix Arrays	367
9	Hash Tables	374
9.1	Basic Hash Tables and Collision Resolution	374
9.2	Universal Families of Hash Functions	380
9.3	Perfect Hash Functions	391
9.4	Hash Trees	397
9.5	Extendible Hashing	398
9.6	Membership Testers and Bloom Filters	402
10	Appendix	406
10.1	The Pointer Machine and Alternative Computation Models	406
10.2	External Memory Models and Cache-Oblivious Algorithms	408
10.3	Naming of Data Structures	409
10.4	Solving Linear Recurrences	410
10.5	Very Slowly Growing Functions	412
11	References	415
	<i>Author Index</i>	441
	<i>Subject Index</i>	455