

Contents

Acknowledgments	xi
Why This Book?	xiii
1 Introduction	1
1.1 The Need for Efficiency	2
1.2 Interfaces	4
1.2.1 The Queue, Stack, and Deque Interfaces	5
1.2.2 The List Interface: Linear Sequences	6
1.2.3 The USet Interface: Unordered Sets	8
1.2.4 The SSet Interface: Sorted Sets	9
1.3 Mathematical Background	9
1.3.1 Exponentials and Logarithms	10
1.3.2 Factorials	11
1.3.3 Asymptotic Notation	12
1.3.4 Randomization and Probability	15
1.4 The Model of Computation	18
1.5 Correctness, Time Complexity, and Space Complexity	19
1.6 Code Samples	22
1.7 List of Data Structures	22
1.8 Discussion and Exercises	26
2 Array-Based Lists	29
2.1 ArrayStack: Fast Stack Operations Using an Array	30
2.1.1 The Basics	30
2.1.2 Growing and Shrinking	33
2.1.3 Summary	35

2.2	FastArrayStack: An Optimized ArrayStack	35
2.3	ArrayQueue: An Array-Based Queue	36
2.3.1	Summary	40
2.4	ArrayDeque: Fast Deque Operations Using an Array	40
2.4.1	Summary	43
2.5	DualArrayDeque: Building a Deque from Two Stacks	43
2.5.1	Balancing	47
2.5.2	Summary	49
2.6	RootishArrayStack: A Space-Efficient Array Stack	49
2.6.1	Analysis of Growing and Shrinking	54
2.6.2	Space Usage	54
2.6.3	Summary	55
2.6.4	Computing Square Roots	56
2.7	Discussion and Exercises	59
3	Linked Lists	63
3.1	SLList: A Singly-Linked List	63
3.1.1	Queue Operations	65
3.1.2	Summary	66
3.2	DLList: A Doubly-Linked List	67
3.2.1	Adding and Removing	69
3.2.2	Summary	70
3.3	SEList: A Space-Efficient Linked List	71
3.3.1	Space Requirements	72
3.3.2	Finding Elements	73
3.3.3	Adding an Element	74
3.3.4	Removing an Element	77
3.3.5	Amortized Analysis of Spreading and Gathering	79
3.3.6	Summary	81
3.4	Discussion and Exercises	82
4	Skiplists	87
4.1	The Basic Structure	87
4.2	SkiplistSSet: An Efficient SSet	90
4.2.1	Summary	93
4.3	SkiplistList: An Efficient Random-Access List	93

4.3.1	Summary	98
4.4	Analysis of Skiplists	98
4.5	Discussion and Exercises	102
5	Hash Tables	107
5.1	ChainedHashTable: Hashing with Chaining	107
5.1.1	Multiplicative Hashing	110
5.1.2	Summary	114
5.2	LinearHashTable: Linear Probing	114
5.2.1	Analysis of Linear Probing	118
5.2.2	Summary	121
5.2.3	Tabulation Hashing	121
5.3	Hash Codes	122
5.3.1	Hash Codes for Primitive Data Types	123
5.3.2	Hash Codes for Compound Objects	123
5.3.3	Hash Codes for Arrays and Strings	125
5.4	Discussion and Exercises	128
6	Binary Trees	133
6.1	BinaryTree: A Basic Binary Tree	135
6.1.1	Recursive Algorithms	136
6.1.2	Traversing Binary Trees	136
6.2	BinarySearchTree: An Unbalanced Binary Search Tree	140
6.2.1	Searching	140
6.2.2	Addition	142
6.2.3	Removal	144
6.2.4	Summary	146
6.3	Discussion and Exercises	147
7	Random Binary Search Trees	153
7.1	Random Binary Search Trees	153
7.1.1	Proof of Lemma 7.1	156
7.1.2	Summary	158
7.2	Treap: A Randomized Binary Search Tree	159
7.2.1	Summary	166
7.3	Discussion and Exercises	168

8	Scapegoat Trees	173
8.1	Scapegoat Tree: A Binary Search Tree with Partial Rebuilding	174
8.1.1	Analysis of Correctness and Running-Time	178
8.1.2	Summary	180
8.2	Discussion and Exercises	181
9	Red-Black Trees	185
9.1	2-4 Trees	186
9.1.1	Adding a Leaf	187
9.1.2	Removing a Leaf	187
9.2	RedBlackTree: A Simulated 2-4 Tree	190
9.2.1	Red-Black Trees and 2-4 Trees	190
9.2.2	Left-Leaning Red-Black Trees	194
9.2.3	Addition	196
9.2.4	Removal	199
9.3	Summary	205
9.4	Discussion and Exercises	206
10	Heaps	211
10.1	BinaryHeap: An Implicit Binary Tree	211
10.1.1	Summary	217
10.2	MeldableHeap: A Randomized Meldable Heap	217
10.2.1	Analysis of merge(h_1, h_2)	220
10.2.2	Summary	221
10.3	Discussion and Exercises	222
11	Sorting Algorithms	225
11.1	Comparison-Based Sorting	226
11.1.1	Merge-Sort	226
11.1.2	Quicksort	230
11.1.3	Heap-sort	233
11.1.4	A Lower-Bound for Comparison-Based Sorting	235
11.2	Counting Sort and Radix Sort	238
11.2.1	Counting Sort	239
11.2.2	Radix-Sort	241

11.3 Discussion and Exercises	243
12 Graphs	247
12.1 AdjacencyMatrix: Representing a Graph by a Matrix	249
12.2 AdjacencyLists: A Graph as a Collection of Lists	252
12.3 Graph Traversal	256
12.3.1 Breadth-First Search	256
12.3.2 Depth-First Search	258
12.4 Discussion and Exercises	261
13 Data Structures for Integers	265
13.1 BinaryTrie: A digital search tree	266
13.2 XFastTrie: Searching in Doubly-Logarithmic Time	272
13.3 YFastTrie: A Doubly-Logarithmic Time SSet	275
13.4 Discussion and Exercises	280
14 External Memory Searching	283
14.1 The Block Store	285
14.2 B-Trees	285
14.2.1 Searching	288
14.2.2 Addition	290
14.2.3 Removal	295
14.2.4 Amortized Analysis of <i>B</i> -Trees	301
14.3 Discussion and Exercises	304
Bibliography	309
Index	317