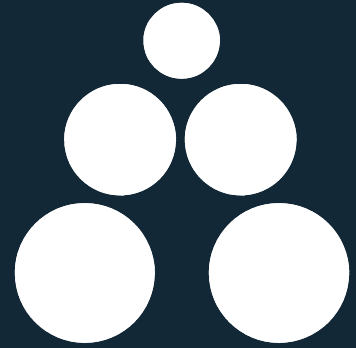


Data Structures and Algorithms

DSA



Annotated Reference with Examples

Data Structures and Algorithms:  
Annotated Reference with Examples

First Edition

Copyright © Granville Barnett, and Luca Del Tongo 2008.



This book is made exclusively available from DotNetSlackers  
(<http://dotnetslackers.com/>) *the* place for .NET articles, and news from  
some of the leading minds in the software industry.

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	What this book is, and what it isn't . . . . .	1
1.2	Assumed knowledge . . . . .	1
1.2.1	Big Oh notation . . . . .	1
1.2.2	Imperative programming language . . . . .	3
1.2.3	Object oriented concepts . . . . .	4
1.3	Pseudocode . . . . .	4
1.4	Tips for working through the examples . . . . .	6
1.5	Book outline . . . . .	6
1.6	Testing . . . . .	7
1.7	Where can I get the code? . . . . .	7
1.8	Final messages . . . . .	7
<b>I</b>	<b>Data Structures</b>	<b>8</b>
<b>2</b>	<b>Linked Lists</b>	<b>9</b>
2.1	Singly Linked List . . . . .	9
2.1.1	Insertion . . . . .	10
2.1.2	Searching . . . . .	10
2.1.3	Deletion . . . . .	11
2.1.4	Traversing the list . . . . .	12
2.1.5	Traversing the list in reverse order . . . . .	13
2.2	Doubly Linked List . . . . .	13
2.2.1	Insertion . . . . .	15
2.2.2	Deletion . . . . .	15
2.2.3	Reverse Traversal . . . . .	16
2.3	Summary . . . . .	17
<b>3</b>	<b>Binary Search Tree</b>	<b>19</b>
3.1	Insertion . . . . .	20
3.2	Searching . . . . .	21
3.3	Deletion . . . . .	22
3.4	Finding the parent of a given node . . . . .	24
3.5	Attaining a reference to a node . . . . .	24
3.6	Finding the smallest and largest values in the binary search tree	25
3.7	Tree Traversals . . . . .	26
3.7.1	Preorder . . . . .	26

3.7.2	Postorder . . . . .	26
3.7.3	Inorder . . . . .	29
3.7.4	Breadth First . . . . .	30
3.8	Summary . . . . .	31
<b>4</b>	<b>Heap</b>	<b>32</b>
4.1	Insertion . . . . .	33
4.2	Deletion . . . . .	37
4.3	Searching . . . . .	38
4.4	Traversal . . . . .	41
4.5	Summary . . . . .	42
<b>5</b>	<b>Sets</b>	<b>44</b>
5.1	Unordered . . . . .	46
5.1.1	Insertion . . . . .	46
5.2	Ordered . . . . .	47
5.3	Summary . . . . .	47
<b>6</b>	<b>Queues</b>	<b>48</b>
6.1	A standard queue . . . . .	49
6.2	Priority Queue . . . . .	49
6.3	Double Ended Queue . . . . .	49
6.4	Summary . . . . .	53
<b>7</b>	<b>AVL Tree</b>	<b>54</b>
7.1	Tree Rotations . . . . .	56
7.2	Tree Rebalancing . . . . .	57
7.3	Insertion . . . . .	58
7.4	Deletion . . . . .	59
7.5	Summary . . . . .	61
<b>II</b>	<b>Algorithms</b>	<b>62</b>
<b>8</b>	<b>Sorting</b>	<b>63</b>
8.1	Bubble Sort . . . . .	63
8.2	Merge Sort . . . . .	63
8.3	Quick Sort . . . . .	65
8.4	Insertion Sort . . . . .	67
8.5	Shell Sort . . . . .	68
8.6	Radix Sort . . . . .	68
8.7	Summary . . . . .	70
<b>9</b>	<b>Numeric</b>	<b>72</b>
9.1	Primality Test . . . . .	72
9.2	Base conversions . . . . .	72
9.3	Attaining the greatest common denominator of two numbers . .	73
9.4	Computing the maximum value for a number of a specific base consisting of N digits . . . . .	74
9.5	Factorial of a number . . . . .	74
9.6	Summary . . . . .	75