

# CONTENTS

<b>1 INTRODUCTION</b> .....	7
<b>2 TRENDS IN CONTROL STRUCTURES</b> .....	9
2.1 Abstraction in Control Structures .....	9
2.2 Parameter Passing .....	10
2.2.1 One-Way Flow into the Function .....	11
2.2.2 One-Way Flow out of the Function .....	11
2.2.3 Two-Way Flow into and out of the Function .....	11
<b>3 ABSTRACT DATA TYPES</b> .....	15
3.1 Data Abstraction via Classes .....	15
3.2 Classes in C++ .....	15
3.3 Class Specification and Implementation .....	16
3.4 Modules and Information Hiding .....	18
<b>4 SPECIFICATION OF DATA STRUCTURES</b> .....	24
4.1 Data Types and Data Structures .....	24
4.2 Hierarchy of Data Structures .....	24
4.3 Lists .....	24
4.4 Stacks .....	25
4.5 Queues .....	26
4.6 Binary Trees .....	28
4.7 General Trees .....	29
4.8 Graphs .....	30
<b>5 DESIGN AND IMPLEMENTATION OF DATA STRUCTURES</b> .....	32
5.1 Vector Implementation of a Stack .....	32
5.2 Stack as a Dynamic List .....	33
5.3 Circular Queue Representation via Vectors .....	35
5.4 Queue Representation via Linked Lists .....	37
5.5 Binary Tree Representation via Linked Lists .....	39
5.5.1 Binary Tree Traversals .....	41
5.6 Binary Tree Representation via Vectors .....	42
5.7 General Tree Representation via Linked Lists .....	43
5.8 Graph Representation via Linked Lists .....	45
5.9 A Short Overview of Other Data Structures .....	46

5.9.1	Arrays .....	46
5.9.2	Records .....	47
5.9.3	Sets.....	47
5.9.4	Summary of Data Structures.....	48
<b>6</b>	<b>SEARCHING AND SORTING BIG DATA.....</b>	<b>49</b>
6.1	Program Speed and Efficiency.....	49
6.2	Big-O Notation .....	50
6.3	Linear Lookup.....	54
6.4	Binary Search.....	54
6.5	Hash Table Lookup .....	57
6.6	N <sup>2</sup> Sorting.....	59
6.6.1	Straight Selection Sort .....	59
6.6.2	Bubble Sort .....	60
6.6.3	Quick Sort .....	61
6.7	Heap Sort .....	64
<b>7</b>	<b>SECURITY IN PROGRAMMING .....</b>	<b>67</b>
7.1	Introduction .....	68
7.2	Information Hiding.....	69
7.3	Defensive Programming.....	69
7.4	Assuming the Impossible .....	70
7.5	Programming Examples.....	71
7.5.1	Buffer Overflows .....	71
7.5.2	Missing Initialization of Variables .....	72
7.5.3	Cryptic Code.....	72
7.5.4	Program Ignoring Error Messages.....	72
7.5.5	Missing Null-Condition Restrictions.....	73
7.5.6	Dangling Pointers.....	73
7.5.7	Assertions .....	74
7.5.8	Work with Library Functions .....	74
7.5.9	Paradoxes – Testing the Impossible (Variant of “Y2K Bug”).....	75
7.6	Concluding Remarks .....	75
	<b>BIBLIOGRAPHY.....</b>	<b>77</b>