

## **Table of contents**

### **INTRODUCTORY PAGES**

Preface: Why choose this book? . . . . .	v
Acknowledgments . . . . .	vi
How to best study using this book? . . . . .	vii
The structure of this book . . . . .	ix
The structure of program descriptions . . . . .	x
Table of contents . . . . .	xii
C++ programs in this book . . . . .	xvi
Pages where exercises can be found . . . . .	xix

### **PART I: THE WORLD OF COMPUTERS. . . . . 1**

#### **1 COMPUTING: SOME CONCEPTS AND TERMINOLOGY . . . . . 3**

1.1 Hardware and software . . . . .	4
1.2 Operating systems, main memory, and memory devices . . . . .	4
1.3 Source programs, object programs, and executable programs . . . . .	8
1.4 Programs, applications, and systems . . . . .	9

#### **2 A FIRST LOOK AT C++ SOURCE PROGRAMS. . . . . 11**

2.1 A program that can print text to the screen. . . . .	12
2.2 A program that can read from the keyboard and calculate. . . . .	14
2.3 Getting a C++ compiler for your computer . . . . .	20
2.4 Installing the electronic material of this book . . . . .	23
2.5 Compiling with the Borland C++ compiler . . . . .	25
2.6 Getting a free editor for your computer . . . . .	26
2.7 Modifying and recompiling programs . . . . .	27

#### **3 HOW INFORMATION IS STORED IN THE MEMORY OF A COMPUTER . . . . . 29**

3.1 Numerical information: numbering systems . . . . .	30
3.2 Numerical information: the binary world of computers . . . . .	38
3.3 Textual information: the ASCII coding system . . . . .	43
3.4 More information: pictures, sound, and moving pictures. . . . .	46

#### **4 LOGICAL OPERATING PRINCIPLES OF COMPUTERS . . . . . 47**

4.1 How does the main memory operate? . . . . .	48
4.2 The components of an imaginary computer . . . . .	52
4.3 Inside the imaginary processor. . . . .	54
4.4 Machine instructions. . . . .	57
4.5 The steps and states of program execution. . . . .	64
4.6 Programs to print text "Hello!". . . . .	66
4.7 Programming language IML and compilation . . . . .	71
4.8 ICOM – simulated imaginary computer. . . . .	78
4.9 A program that contains a loop. . . . .	80
4.10 Subroutine calls and stack operations. . . . .	82
4.11 Programs that use the keyboard, memory area, and stack . . . . .	85
4.12 Chapter summary – towards high-level programming. . . . .	92

<b>PART II: FUNDAMENTALS OF PROGRAMMING</b> .....	<b>95</b>
<b>5 VARIABLES AND OTHER BASIC ELEMENTS IN C++ PROGRAMS</b> .....	<b>97</b>
5.1 Integer variables (int, short, long, char) .....	98
5.2 Keywords, names, spaces, and newlines .....	104
5.3 Floating-point variables .....	106
5.4 Operators, assignments, and numerical constants .....	109
5.5 Formatting the output on the screen .....	114
5.6 Chapter summary .....	116
<b>6 DECISIONS AND REPETITIONS: BASIC ACTIVITIES IN PROGRAMS</b> .....	<b>117</b>
6.1 Making decisions with keywords if and else .....	118
6.2 Making decisions with switch-case constructs .....	131
6.3 while loops enable repetition .....	135
6.4 for loops repeat a known number of times .....	140
6.5 do-while loops execute at least once .....	146
6.6 The block structure of C++ programs .....	149
6.7 The truth about truth values in C++ .....	153
6.8 Chapter summary .....	155
<b>7 ARRAYS AND STRINGS: SETS OF SIMILAR DATA ITEMS</b> .....	<b>157</b>
7.1 Declaring arrays and referring to array elements .....	158
7.2 Initialized and multidimensional arrays .....	169
7.3 Arrays of type char are called strings .....	174
7.4 String functions .....	184
7.5 Chapter summary .....	191
<b>8 POINTERS</b> .....	<b>193</b>
8.1 Operator & returns the address of a data item .....	194
8.2 An asterisk * declares pointers and refers to pointed-to data .....	195
8.3 Pointers and arrays .....	200
8.4 Pointers and strings .....	207
8.5 Arrays of pointers to string .....	211
8.6 Chapter summary .....	214
<b>9 FUNCTIONS: LOGICAL PERFORMING UNITS IN PROGRAMS</b> .....	<b>217</b>
9.1 Simple functions and the concept of calling .....	218
9.2 Functions that take arguments .....	222
9.3 Functions that return data to the caller .....	230
9.4 Function prototypes .....	238
9.5 The role of the stack in function calls .....	245
9.6 Scope of variables .....	251
9.7 Arguments for the function main() .....	256
9.8 Overloading function names .....	260
9.9 Chapter summary .....	264
<b>10 STRUCTURES AND FILES – LARGER PIECES OF DATA</b> .....	<b>269</b>
10.1 Structures make new data types .....	270
10.2 Tables based on structures .....	275
10.3 Pointers to structure types .....	284
10.4 A larger program that uses structures and files .....	290
10.5 Reading and writing text files .....	308
10.6 Chapter summary .....	319

<b>11</b>	<b>GOING CLOSER TO THE MACHINE</b> .....	<b>321</b>
11.1	Bit operators &,  , ^, ~, >>, and << .....	322
11.2	Getting the current time and date into your program .....	331
11.3	Playing with the time in programs .....	334
11.4	Memory inspection program for debugging .....	340
11.5	Chapter summary .....	346
 <b>PART III: OBJECT-ORIENTED PROGRAMMING</b> .....		<b>347</b>
<b>12</b>	<b>CLASSES AND OBJECTS</b> .....	<b>349</b>
12.1	Classes, data members, and member functions .....	350
12.2	Constructors and destructors .....	356
12.3	Several constructors in a class .....	362
12.4	Arrays of objects .....	366
12.5	Inline member functions .....	370
12.6	Dynamic memory: the operators new and delete .....	371
12.7	Objects on the free store .....	376
12.8	A stack that grows dynamically .....	380
12.9	Chapter summary .....	387
<b>13</b>	<b>MORE ADVANCED CLASSES</b> .....	<b>389</b>
13.1	Overloading of operators .....	390
13.2	The this pointer .....	398
13.3	Friends of a class .....	400
13.4	Class Date – an example of a larger class .....	402
13.5	operator=( ) functions and constructors generated by the compiler .....	422
13.6	Graphical UML class diagrams .....	425
13.7	Objects as members of a class .....	426
13.8	Templates for functions and classes .....	444
13.9	Chapter summary .....	450
<b>14</b>	<b>INHERITANCE AND CLASS HIERARCHIES</b> .....	<b>451</b>
14.1	Base classes and derived classes .....	452
14.2	Larger class hierarchies .....	468
14.3	Virtual functions .....	482
14.4	Chapter summary .....	496
<b>15</b>	<b>STANDARD C++ CLASSES</b> .....	<b>497</b>
15.1	The origins of cin and cout .....	498
15.2	Class string .....	500
15.3	Class stringstream .....	512
15.4	Classes to read and write files .....	519
15.5	Handling files as binary files .....	530
15.6	Class templates vector and list .....	540
15.7	Chapter summary .....	554
<b>16</b>	<b>LARGE OBJECT-ORIENTED APPLICATIONS</b> .....	<b>555</b>
16.1	An editor program – 3700 source program lines .....	556
16.2	Programs for travelers .....	564
16.3	Advice for developing large (and small) applications .....	568
 <b>EPILOGUE: WHY DID I WRITE THIS BOOK?</b> .....		<b>573</b>

<b>APPENDIX A: SUMMARY OF IMPORTANT C++ FEATURES</b> .....	<b>575</b>
A - 1: Literals (literal constants) .....	575
A - 2: Declarations (definitions) that reserve memory .....	576
A - 3: Declarations that do not reserve memory .....	577
A - 4: Expressions .....	578
A - 5: Assignments and l-values .....	578
A - 6: C++ operators in order of precedence .....	579
A - 7: Control structures to make decisions (selections) .....	580
A - 8: Control structures to perform repetitions (iterations) .....	581
A - 9: Various kinds of C++ functions .....	582
A - 10: Functions to manipulate C-style strings (defined in string.h) .....	583
A - 11: Keyboard input mechanisms .....	584
A - 12: Screen output mechanism .....	584
A - 13: Input/output from/to stringstream objects .....	584
A - 14: Input/output from/to files .....	585
A - 15: Data conversions .....	586
A - 16: C++ class declaration .....	587
A - 17: Operators used with classes and objects .....	588
<b>APPENDIX B: C++ KEYWORDS (RESERVED WORDS)</b> .....	<b>589</b>
<b>APPENDIX C: PRACTICAL ADVICE FOR PROGRAMMING EXERCISES</b> .....	<b>593</b>
C - 1: Starting an exercise .....	593
C - 2: Writing your program .....	593
C - 3: Compiling and executing your program .....	594
C - 4: Correcting compilation errors .....	595
C - 5: Searching for errors that compilers do not detect .....	596
C - 6: Programs that do not terminate .....	597
C - 7: Incremental program development .....	598
C - 8: Printing your program on paper .....	598
C - 9: Program versions and backups .....	598
<b>INDEX</b> .....	<b>599</b>
<b>USEFUL TABLES</b> .....	<b>613</b>