

# Contents at a Glance

---

<b><i>Introduction</i></b> .....	<b>1</b>
<b><i>Part I: Introduction to C++ Programming</i></b> .....	<b>7</b>
Chapter 1: Writing Your First C++ Program.....	9
Chapter 2: Declaring Variables Constantly.....	25
Chapter 3: Performing Mathematical Operations.....	41
Chapter 4: Performing Logical Operations .....	49
Chapter 5: Controlling Program Flow.....	63
<b><i>Part II: Becoming a Functional C++ Programmer</i></b> .....	<b>79</b>
Chapter 6: Creating Functions.....	81
Chapter 7: Storing Sequences in Arrays .....	95
Chapter 8: Taking a First Look at C++ Pointers.....	111
Chapter 9: Taking a Second Look at C++ Pointers .....	127
Chapter 10: The C++ Preprocessor.....	145
<b><i>Part III: Introduction to Classes</i></b> .....	<b>159</b>
Chapter 11: Examining Object-Oriented Programming.....	161
Chapter 12: Adding Class to C++ .....	165
Chapter 13: Point and Stare at Objects .....	181
Chapter 14: Protecting Members: Do Not Disturb.....	197
Chapter 15: Why Do You Build Me Up, Just to Tear Me Down Baby?.....	205
Chapter 16: Making Constructive Arguments .....	217
Chapter 17: Copying the Copy Copy Copy Constructor .....	235
Chapter 18: Static Members: Can Fabric Softener Help? .....	247
<b><i>Part IV: Inheritance</i></b> .....	<b>255</b>
Chapter 19: Inheriting a Class .....	257
Chapter 20: Examining Virtual Member Functions: Are They for Real? .....	265
Chapter 21: Factoring Classes .....	273

<b><i>Part V: Optional Features</i></b> .....	<b>281</b>
Chapter 22: A New Assignment Operator, Should You Decide to Accept It .....	283
Chapter 23: Using Stream I/O .....	293
Chapter 24: Handling Errors — Exceptions.....	315
Chapter 25: Inheriting Multiple Inheritance .....	325
Chapter 26: Tempting C++ Templates .....	337
Chapter 27: Standardizing on the Standard Template Library .....	345
<b><i>Part VI: The Part of Tens</i></b> .....	<b>357</b>
Chapter 28: Ten Ways to Avoid Adding Bugs to Your Program .....	359
Chapter 29: Ten Major Recent Additions to C++ .....	367
<b><i>Appendix: About the CD</i></b> .....	<b>377</b>
<b><i>Index</i></b> .....	<b>383</b>

# Table of Contents

---

<b><i>Introduction</i></b> .....	<b>1</b>
What's in This Book .....	1
What's on the CD .....	2
What Is C++? .....	2
Conventions Used in This Book .....	3
How This Book Is Organized .....	3
And There's More .....	4
Part I: Introduction to C++ Programming .....	4
Part II: Becoming a Functional C++ Programmer .....	4
Part III: Introduction to Classes .....	5
Part IV: Inheritance .....	5
Part V: Optional Features .....	5
Part VI: The Part of Tens .....	5
Icons Used in This Book .....	6
Where to Go from Here .....	6

## ***Part I: Introduction to C++ Programming*** .....

<b>Chapter 1: Writing Your First C++ Program</b> .....	<b>9</b>
Grasping C++ Concepts .....	9
Installing Code::Blocks .....	11
Creating Your First C++ Program .....	13
Creating a project .....	14
Entering the C++ code .....	15
Cheating .....	16
Building your program .....	17
Executing Your Program .....	18
Reviewing the Annotated Program .....	19
Examining the framework for all C++ programs .....	19
Clarifying source code with comments .....	20
Basing programs on C++ statements .....	21
Writing declarations .....	22
Generating output .....	22
Calculating Expressions .....	23
Storing the results of an expression .....	23
Examining the remainder of Conversion .....	24



<b>Chapter 2: Declaring Variables Constantly</b> .....	<b>25</b>
Declaring Variables .....	25
Declaring Different Types of Variables .....	26
Reviewing the limitations of integers in C++ .....	27
Solving the truncation problem .....	28
Looking at the limits of floating-point numbers.....	29
Declaring Variable Types .....	31
Types of constants .....	32
Range of Numeric Types .....	33
Special characters .....	35
Wide Loads on Char Highway .....	36
Are These Calculations Really Logical? .....	37
Mixed Mode Expressions.....	37
Automatic Declarations .....	38
<b>Chapter 3: Performing Mathematical Operations</b> .....	<b>41</b>
Performing Simple Binary Arithmetic .....	41
Decomposing Expressions .....	43
Determining the Order of Operations.....	43
Performing Unary Operations.....	44
Using Assignment Operators .....	46
<b>Chapter 4: Performing Logical Operations</b> .....	<b>49</b>
Why Mess with Logical Operations? .....	49
Using the Simple Logical Operators.....	50
Storing logical values .....	51
Using logical int variables.....	53
Be careful performing logical operations on floating-point variables .....	53
Expressing Binary Numbers.....	55
The decimal number system .....	55
Other number systems.....	56
The binary number system .....	56
Performing Bitwise Logical Operations .....	58
The single-bit operators.....	58
Using the bitwise operators .....	60
A simple test.....	60
Do something logical with logical calculations.....	62
<b>Chapter 5: Controlling Program Flow</b> .....	<b>63</b>
Controlling Program Flow with the Branch Commands.....	63
Executing Loops in a Program .....	65
Looping while a condition is true .....	66
Using the autoincrement/autodecrement feature.....	67
Using the for loop .....	69
Avoiding the dreaded infinite loop.....	72
Applying special loop controls .....	73

Nesting Control Commands .....	75
Switching to a Different Subject? .....	77

## ***Part II: Becoming a Functional C++ Programmer..... 79***

### **Chapter 6: Creating Functions ..... 81**

Writing and Using a Function.....	81
Defining our first function.....	84
Defining the sumSequence() function.....	84
Calling the function sumSequence() .....	85
Divide and conquer .....	85
Understanding the Details of Functions .....	85
Understanding simple functions.....	86
Understanding functions with arguments .....	87
Overloading Function Names.....	90
Defining Function Prototypes .....	91
Variable Storage Types.....	93

### **Chapter 7: Storing Sequences in Arrays ..... 95**

Arraying the Arguments for Arrays.....	95
Using an array .....	97
Initializing an array.....	100
Accessing too far into an array.....	101
Using arrays.....	101
Defining and using arrays of arrays.....	101
Using Arrays of Characters .....	102
Creating an array of characters .....	102
Creating a string of characters.....	103
Manipulating Strings with Character .....	105
Adding Some Library Functions .....	107
Making Room for Wide Strings .....	109

### **Chapter 8: Taking a First Look at C++ Pointers ..... 111**

Variable Size.....	111
What's in an Address?.....	113
Address Operators .....	113
Using Pointer Variables .....	115
Using different types of pointers .....	116
Passing Pointers to Functions.....	117
Passing by value.....	118
Passing pointer values .....	118
Passing by reference .....	119
Constant const Irritation .....	120
Making Use of a Block of Memory Called the Heap.....	122
Limited scope.....	122
Examining the scope problem.....	123
Providing a solution using the heap.....	124

<b>Chapter 9: Taking a Second Look at C++ Pointers . . . . .</b>	<b>127</b>
Defining Operations on Pointer Variables.....	127
Reexamining arrays in light of pointer variables.....	128
Applying operators to the address of an array.....	130
Expanding pointer operations to a string.....	131
Justifying pointer-based string manipulation .....	133
Applying operators to pointer types other than char .....	134
Contrasting a pointer with an array .....	134
When Is a Pointer Not?.....	136
Declaring and Using Arrays of Pointers.....	137
Utilizing arrays of character strings.....	138
Accessing the arguments to main().....	140
<b>Chapter 10: The C++ Preprocessor . . . . .</b>	<b>145</b>
What Is a Preprocessor? .....	145
Including Files .....	146
#Defining Things .....	149
Okay, how about not #defining things?.....	152
Enumerating other options.....	153
Including Things #if I Say So .....	154
Intrinsically Defined Objects .....	156
Typedef .....	158
<b>Part III: Introduction to Classes.....</b>	<b>159</b>
<b>Chapter 11: Examining Object-Oriented Programming . . . . .</b>	<b>161</b>
Abstracting Microwave Ovens.....	161
Preparing functional nachos .....	162
Preparing object-oriented nachos .....	163
Classifying Microwave Ovens .....	163
Why Classify? .....	164
<b>Chapter 12: Adding Class to C++ . . . . .</b>	<b>165</b>
Introducing the Class .....	165
The Format of a Class .....	166
Accessing the Members of a Class .....	167
Activating Our Objects .....	167
Simulating real-world objects.....	168
Why bother with member functions?.....	168
Adding a Member Function .....	169
Calling a Member Function.....	170
Accessing other members from a member function.....	172
Scope Resolution (And I Don't Mean How Well Your Microscope Works).....	173
Defining a Member Function in the Class.....	175

Keeping a Member Function After Class .....	176
Overloading Member Functions .....	178
<b>Chapter 13: Point and Stare at Objects. ....</b>	<b>181</b>
Declaring Arrays of Objects .....	181
Declaring Pointers to Objects .....	182
Dereferencing an object pointer .....	183
Pointing toward arrow pointers.....	184
Passing Objects to Functions.....	184
Calling a function with an object value.....	185
Calling a function with an object pointer .....	186
Calling a function by using the reference operator.....	188
Why Bother with Pointers or References? .....	189
Returning to the Heap.....	189
Allocating heaps of objects .....	190
Comparing Pointers to References.....	191
Linking Up with Linked Lists .....	191
Performing other operations on a linked list .....	192
Hooking up with a LinkedListData program.....	193
A Ray of Hope: A List of Containers Linked to the C++ Library.....	196
<b>Chapter 14: Protecting Members: Do Not Disturb .....</b>	<b>197</b>
Protecting Members.....	197
Why you need protected members .....	197
Discovering how protected members work.....	198
Making an Argument for Using Protected Members.....	200
Protecting the internal state of the class.....	200
Using a class with a limited interface.....	201
Giving Nonmember Functions Access to Protected Members.....	201
<b>Chapter 15: Why Do You Build Me Up, Just to Tear Me Down Baby? .....</b>	<b>205</b>
Creating Objects .....	205
Using Constructors.....	206
Constructing a single object.....	207
Constructing multiple objects.....	208
Constructing a duplex .....	209
Dissecting a Destructor .....	211
Why you need the destructor .....	211
Working with destructors.....	212
<b>Chapter 16: Making Constructive Arguments .....</b>	<b>217</b>
Outfitting Constructors with Arguments.....	217
Using a constructor .....	218
Placing Too Many Demands on the Carpenter:	
Overloading the Constructor .....	220
Defaulting Default Constructors .....	222

Constructing Class Members .....	224
Constructing a complex data member .....	224
Constructing a constant data member .....	228
Reconstructing the Order of Construction .....	229
Local objects construct in order .....	229
Static objects construct only once .....	230
All global objects construct before main() .....	231
Global objects construct in no particular order .....	231
Members construct in the order in which they are declared .....	233
Destructors destruct in the reverse order of the constructors .....	233
Constructors as a Form of Conversion .....	233

### **Chapter 17: Copying the Copy Copy Copy Constructor . . . . . 235**

Copying an Object .....	235
Why you need the copy constructor .....	236
Using the copy constructor .....	236
The Automatic Copy Constructor .....	238
Creating Shallow Copies versus Deep Copies .....	240
It's a Long Way to Temporaries .....	244
Avoiding temporaries, permanently .....	246

### **Chapter 18: Static Members: Can Fabric Softener Help? . . . . . 247**

Defining a Static Member .....	247
Why you need static members .....	247
Using static members .....	248
Referencing static data members .....	249
Uses for static data members .....	250
Declaring Static Member Functions .....	251
What Is This About Anyway? .....	253

## ***Part IV: Inheritance* .....** 255

### **Chapter 19: Inheriting a Class . . . . . 257**

Do I Need My Inheritance? .....	258
How Does a Class Inherit? .....	259
Using a subclass .....	261
Constructing a subclass .....	262
Destructing a subclass .....	263
Having a HAS_A Relationship .....	263

### **Chapter 20: Examining Virtual Member Functions: Are They for Real? . . . . . 265**

Why You Need Polymorphism .....	268
How Polymorphism Works .....	269
When Is a Virtual Function Not? .....	270
Considering Virtual Considerations .....	271

**Chapter 21: Factoring Classes . . . . . 273**  
 Factoring..... 273  
 Implementing Abstract Classes ..... 277  
     Describing the abstract class concept..... 278  
     Making an honest class out of an abstract class ..... 279  
     Passing abstract classes ..... 280

***Part V: Optional Features..... 281***

**Chapter 22: A New Assignment Operator, Should You Decide to Accept It. . . . . 283**  
 Comparing Operators with Functions ..... 283  
 Inserting a New Operator ..... 284  
 Creating Shallow Copies Is a Deep Problem ..... 285  
 Overloading the Assignment Operator..... 286  
 Overloading the Subscript Operator ..... 291

**Chapter 23: Using Stream I/O . . . . . 293**  
 How Stream I/O Works..... 293  
     Default stream objects ..... 294  
 Stream Input/Output ..... 295  
     Open modes..... 297  
     Hey, file, what state are you in? ..... 298  
     Can you show me an example? ..... 298  
 Other Methods of the Stream Classes ..... 301  
     Reading and writing streams directly ..... 303  
     Controlling format ..... 305  
     What’s up with endl? ..... 307  
     Positioning the pointer within a file ..... 307  
 Using the stringstream Subclasses ..... 308  
 Manipulating Manipulators ..... 311

**Chapter 24: Handling Errors — Exceptions. . . . . 315**  
 Justifying a New Error Mechanism? ..... 317  
 Examining the Exception Mechanism ..... 318  
 What Kinds of Things Can I Throw? ..... 321  
 Just Passing Through..... 324

**Chapter 25: Inheriting Multiple Inheritance . . . . . 325**  
 Describing the Multiple Inheritance Mechanism ..... 325  
 Straightening Out Inheritance Ambiguities ..... 327  
 Adding Virtual Inheritance ..... 328  
 Constructing the Objects of Multiple Inheritance..... 335  
 Voicing a Contrary Opinion..... 335

<b>Chapter 26: Tempting C++ Templates . . . . .</b>	<b>337</b>
Generalizing a Function into a Template.....	338
Template Classes.....	340
Tips for Using Templates.....	343
<b>Chapter 27: Standardizing on the Standard Template Library . . . . .</b>	<b>345</b>
The string Container .....	346
Iterating through Lists .....	351
Making your way through a list .....	353
Operations on an entire list.....	354
Can you show me an example? .....	355
 <b>Part VI: The Part of Tens.....</b>	 <b>357</b>
<b>Chapter 28: Ten Ways to Avoid Adding Bugs to Your Program . . . . .</b>	<b>359</b>
Enable All Warnings and Error Messages.....	359
Adopt a Clear and Consistent Coding Style .....	360
Limit the Visibility .....	360
Comment Your Code While You Write It.....	362
Single-Step Every Path at Least Once .....	362
Avoid Overloading Operators.....	363
Manage the Heap Systematically.....	363
Use Exceptions to Handle Errors.....	363
Declare Destructors Virtual .....	364
Avoid Multiple Inheritance.....	365
<b>Chapter 29: Ten Major Recent Additions to C++. . . . .</b>	<b>367</b>
Use Smart Pointers.....	367
Initialize Variables with a Variable-Length List .....	368
Initialize Data Members Inline .....	369
Instantiate an Extern Template.....	370
Implement Thread Local Storage .....	371
Use Rvalue References.....	371
Implement Concepts .....	372
Define Lamda Expressions .....	373
Define Variadic Templates .....	374
Use typeid().....	374

---

<b><i>Appendix: About the CD</i></b> .....	<b>377</b>
System Requirements .....	377
Using the CD .....	378
What You'll Find on the CD .....	378
CPP programs .....	379
Code::Blocks development environment .....	379
Troubleshooting .....	380
Customer Care .....	381
 <b><i>Index</i></b> .....	 <b>383</b>

