

Object-Oriented Programming in C++, Fourth Edition

Robert Lafore

SAMS

800 East 96th St., Indianapolis, Indiana 46240 USA

Copyright © 2002 by Sams Publishing

All rights reserved. No part of this book shall be reproduced, stored in a retrieval system, or transmitted by any means, electronic, mechanical, photocopying, recording, or otherwise, without written permission from the publisher. No patent liability is assumed with respect to the use of the information contained herein. Although every precaution has been taken in the preparation of this book, the publisher and author assume no responsibility for errors or omissions. Nor is any liability assumed for damages resulting from the use of the information contained herein.

International Standard Book Number: 0-672-32308-7

Library of Congress Catalog Card Number: 2001094813

Printed in the United States of America

First Printing: December 2001

04 03 02 01 4 3 2 1

Trademarks

All terms mentioned in this book that are known to be trademarks or service marks have been appropriately capitalized. Sams Publishing cannot attest to the accuracy of this information. Use of a term in this book should not be regarded as affecting the validity of any trademark or service mark.

Warning and Disclaimer

Every effort has been made to make this book as complete and as accurate as possible, but no warranty or fitness is implied. The information provided is on an “as is” basis. The author and the publisher shall have neither liability nor responsibility to any person or entity with respect to any loss or damages arising from the information contained in this book.

EXECUTIVE EDITOR

Michael Stephens

ACQUISITIONS EDITOR

Michael Stephens

MANAGING EDITOR

Matt Purcell

PROJECT EDITORS

Angela Boley

Christina Smith

INDEXER

Rebecca Salerno

PROOFREADER

Matt Wynałda

TECHNICAL EDITOR

Mark Cashman

TEAM COORDINATOR

Pamalee Nelson

MEDIA DEVELOPER

Dan Scherf

INTERIOR DESIGNER

Gary Adair

COVER DESIGNER

Alan Clements

PAGE LAYOUT

Ayanna Lacey

Overview

	Introduction	1
1	The Big Picture	9
2	C++ Programming Basics	29
3	Loops and Decisions	75
4	Structures	131
5	Functions	161
6	Objects and Classes	215
7	Arrays and Strings	263
8	Operator Overloading	319
9	Inheritance	371
10	Pointers	429
11	Virtual Functions	503
12	Streams and Files	567
13	Multifile Programs	633
14	Templates and Exceptions	681
15	The Standard Template Library	725
16	Object-Oriented Software Development	801
A	ASCII Chart	849
B	C++ Precedence Table and Keywords	859
C	Microsoft Visual C++	863
D	Borland C++Builder	871
E	Console Graphics Lite	881
F	STL Algorithms and Member Functions	895
G	Answers to Questions and Exercises	913
H	Bibliography	977
	Index	981

Contents

Introduction 1

1 The Big Picture 9

Why Do We Need Object-Oriented Programming?	10
Procedural Languages	10
The Object-Oriented Approach	13
Characteristics of Object-Oriented Languages	16
Objects	16
Classes	18
Inheritance	18
Reusability	21
Creating New Data Types	21
Polymorphism and Overloading	21
C++ and C	22
Laying the Groundwork	23
The Unified Modeling Language (UML)	23
Summary	25
Questions	25

2 C++ Programming Basics 29

Getting Started	30
Basic Program Construction	30
Functions	31
Program Statements	32
Whitespace	33
Output Using cout	33
String Constants	34
Directives	35
Preprocessor Directives	35
Header Files	35
The using Directive	36
Comments	36
Comment Syntax	36
When to Use Comments	37
Alternative Comment Syntax	37
Integer Variables	38
Defining Integer Variables	38
Declarations and Definitions	40
Variable Names	40
Assignment Statements	40

Integer Constants	41
Output Variations	41
The endl Manipulator	41
Other Integer Types	42
Character Variables	42
Character Constants	43
Initialization	44
Escape Sequences	44
Input with cin	45
Variables Defined at Point of Use	47
Cascading <<	47
Expressions	47
Precedence	47
Floating Point Types	48
Type float	48
Type double and long double	49
Floating-Point Constants	50
The const Qualifier	51
The #define Directive	51
Type bool	51
The setw Manipulator	52
Cascading the Insertion Operator	54
Multiple Definitions	54
The iomanip Header File	54
Variable Type Summary	54
unsigned Data Types	55
Type Conversion	56
Automatic Conversions	57
Casts	58
Arithmetic Operators	60
The Remainder Operator	61
Arithmetic Assignment Operators	61
Increment Operators	63
Library Functions	65
Header Files	66
Library Files	66
Header Files and Library Files	67
Two Ways to Use #include	67
Summary	68
Questions	69
Exercises	71

3	Loops and Decisions	75
	Relational Operators	76
	Loops.....	78
	The for Loop.....	78
	Debugging Animation	84
	for Loop Variations	84
	The while Loop	86
	Precedence: Arithmetic and Relational Operators	89
	The do Loop.....	91
	When to Use Which Loop	93
	Decisions.....	93
	The if Statement.....	94
	The if...else Statement	98
	The else...if Construction	106
	The switch Statement	107
	The Conditional Operator	111
	Logical Operators	114
	Logical AND Operator	115
	Logical OR Operator.....	116
	Logical NOT Operator	117
	Precedence Summary	118
	Other Control Statements	118
	The break Statement	119
	The continue Statement	121
	The goto Statement	123
	Summary.....	123
	Questions.....	124
	Exercises	126
4	Structures	131
	Structures	132
	A Simple Structure	132
	Defining the Structure	133
	Defining a Structure Variable	134
	Accessing Structure Members.....	136
	Other Structure Features	137
	A Measurement Example	139
	Structures Within Structures	141
	A Card Game Example	145
	Structures and Classes.....	148
	Enumerations	148
	Days of the Week.....	148
	One Thing or Another	151

Organizing the Cards	153
Specifying Integer Values	155
Not Perfect.....	155
Other Examples	155
Summary	156
Questions.....	156
Exercises	158
5 Functions 161	
Simple Functions	162
The Function Declaration	164
Calling the Function	164
The Function Definition	164
Comparison with Library Functions	166
Eliminating the Declaration.....	166
Passing Arguments to Functions.....	167
Passing Constants	167
Passing Variables	169
Passing by Value	170
Structures as Arguments	171
Names in the Declaration	176
Returning Values from Functions	176
The return Statement	177
Returning Structure Variables	180
Reference Arguments	182
Passing Simple Data Types by Reference	182
A More Complex Pass by Reference	185
Passing Structures by Reference	186
Notes on Passing by Reference	188
Overloaded Functions	188
Different Numbers of Arguments	189
Different Kinds of Arguments	191
Recursion	193
Inline Functions	195
Default Arguments	197
Scope and Storage Class.....	199
Local Variables	199
Global Variables.....	202
Static Local Variables	204
Storage.....	205
Returning by Reference	206
Function Calls on the Left of the Equal Sign	207
Don't Worry Yet.....	207

const Function Arguments	208
Summary	209
Questions.....	210
Exercises	212
6 Objects and Classes 215	
A Simple Class	216
Classes and Objects	217
Defining the Class	218
Using the Class	221
Calling Member Functions	221
C++ Objects as Physical Objects	223
Widget Parts as Objects.....	223
Circles as Objects	224
C++ Objects as Data Types	226
Constructors	227
A Counter Example	228
A Graphics Example	231
Destructors.....	232
Objects as Function Arguments	233
Overloaded Constructors	234
Member Functions Defined Outside the Class	236
Objects as Arguments	237
The Default Copy Constructor	238
Returning Objects from Functions	240
Arguments and Objects	241
A Card-Game Example.....	243
Structures and Classes	247
Classes, Objects, and Memory	247
Static Class Data	249
Uses of Static Class Data	249
An Example of Static Class Data	249
Separate Declaration and Definition	251
const and Classes	252
const Member Functions	252
const Objects	255
What Does It All Mean?	256
Summary	257
Questions.....	257
Exercises	259

7 Arrays and Strings 263

Array Fundamentals	264
Defining Arrays	265
Array Elements	265
Accessing Array Elements.....	267
Averaging Array Elements	267
Initializing Arrays	268
Multidimensional Arrays	270
Passing Arrays to Functions	274
Arrays of Structures.....	277
Arrays as Class Member Data	279
Arrays of Objects	283
Arrays of English Distances	283
Arrays of Cards	286
C-Strings	290
C-String Variables	290
Avoiding Buffer Overflow.....	292
String Constants.....	292
Reading Embedded Blanks	293
Reading Multiple Lines	294
Copying a String the Hard Way	295
Copying a String the Easy Way.....	296
Arrays of Strings	297
Strings as Class Members	298
A User-Defined String Type	300
The Standard C++ string Class.....	302
Defining and Assigning string Objects	302
Input/Output with string Objects.....	304
Finding string Objects	305
Modifying string Objects	306
Comparing string Objects	307
Accessing Characters in string Objects.....	309
Other string Functions.....	310
Summary	310
Questions.....	311
Exercises	313

8 Operator Overloading 319

Overloading Unary Operators.....	320
The operator Keyword	322
Operator Arguments	323

Operator Return Values	323
Nameless Temporary Objects	325
Postfix Notation	326
Overloading Binary Operators	328
Arithmetic Operators	328
Concatenating Strings	332
Multiple Overloading	334
Comparison Operators	334
Arithmetic Assignment Operators	337
The Subscript Operator ([])	340
Data Conversion	344
Conversions Between Basic Types	344
Conversions Between Objects and Basic Types	345
Conversions Between Objects of Different Classes	350
Conversions: When to Use What	357
UML Class Diagrams	357
Associations	357
Navigability	358
Pitfalls of Operator Overloading and Conversion	358
Use Similar Meanings	358
Use Similar Syntax	359
Show Restraint	359
Avoid Ambiguity	360
Not All Operators Can Be Overloaded	360
Keywords <code>explicit</code> and <code>mutable</code>	360
Preventing Conversions with <code>explicit</code>	360
Changing <code>const</code> Object Data Using <code>mutable</code>	362
Summary	364
Questions	364
Exercises	367
9 Inheritance 371	
Derived Class and Base Class	373
Specifying the Derived Class	375
Generalization in UML Class Diagrams	375
Accessing Base Class Members	376
The <code>protected</code> Access Specifier	377
Derived Class Constructors	380
Overriding Member Functions	382
Which Function Is Used?	383
Scope Resolution with Overridden Functions	384

Inheritance in the English Distance Class	384
Operation of ENGLN	387
Constructors in DistSign	387
Member Functions in DistSign	387
Abetting Inheritance	388
Class Hierarchies	388
“Abstract” Base Class	392
Constructors and Member Functions	393
Inheritance and Graphics Shapes	393
Public and Private Inheritance	396
Access Combinations	397
Access Specifiers: When to Use What	399
Levels of Inheritance	399
Multiple Inheritance	403
Member Functions in Multiple Inheritance.....	404
private Derivation in EMPMULT	409
Constructors in Multiple Inheritance.....	409
Ambiguity in Multiple Inheritance	413
Aggregation: Classes Within Classes	414
Aggregation in the EMPCONT Program.....	416
Composition: A Stronger Aggregation	420
Inheritance and Program Development	420
Summary	421
Questions.....	422
Exercises	424
 10 Pointers 429	
Addresses and Pointers	430
The Address-of Operator &	431
Pointer Variables	433
Syntax Quibbles.....	434
Accessing the Variable Pointed To	436
Pointer to void.....	439
Pointers and Arrays.....	440
Pointer Constants and Pointer Variables	442
Pointers and Functions	443
Passing Simple Variables.....	443
Passing Arrays	446
Sorting Array Elements	448
Pointers and C-Type Strings	452
Pointers to String Constants	452
Strings as Function Arguments	453

Copying a String Using Pointers	454
Library String Functions	456
The const Modifier and Pointers	456
Arrays of Pointers to Strings	456
Memory Management: new and delete	458
The new Operator	459
The delete Operator	461
A String Class Using new	462
Pointers to Objects	464
Referring to Members	465
Another Approach to new	465
An Array of Pointers to Objects	467
A Linked List Example	469
A Chain of Pointers	469
Adding an Item to the List	471
Displaying the List Contents	472
Self-Containing Classes	473
Augmenting LINKLIST	473
Pointers to Pointers	474
Sorting Pointers	476
The person** Data Type	476
Comparing Strings	478
A Parsing Example	479
Parsing Arithmetic Expressions	479
The PARSE Program	481
Simulation: A Horse Race	484
Designing the Horse Race	485
Multiplicity in the UML	489
UML State Diagrams	490
States	491
Transitions	491
Racing from State to State	492
Debugging Pointers	492
Summary	493
Questions	494
Exercises	497
11 Virtual Functions 503	
Virtual Functions	504
Normal Member Functions Accessed with Pointers	505
Virtual Member Functions Accessed with Pointers	507
Late Binding	509

Abstract Classes and Pure Virtual Functions	510
Virtual Functions and the person Class	511
Virtual Functions in a Graphics Example	514
Virtual Destructors.....	517
Virtual Base Classes	518
Friend Functions	520
Friends as Bridges	520
Breaching the Walls.....	522
English Distance Example.....	522
friends for Functional Notation	526
friend Classes.....	528
Static Functions.....	529
Accessing static Functions	531
Numbering the Objects	532
Investigating Destructors	532
Assignment and Copy Initialization	532
Overloading the Assignment Operator	533
The Copy Constructor	536
UML Object Diagrams	539
A Memory-Efficient String Class	540
The this Pointer.....	547
Accessing Member Data with this.....	547
Using this for Returning Values.....	548
Revised STRIMEM Program.....	550
Dynamic Type Information.....	553
Checking the Type of a Class with dynamic_cast	553
Changing Pointer Types with dynamic_cast.....	554
The typeid Operator	556
Summary.....	557
Questions.....	558
Exercises	561
12 Streams and Files 567	
Stream Classes	568
Advantages of Streams	568
The Stream Class Hierarchy	568
The ios Class	570
The istream Class.....	574
The ostream Class.....	575
The iostream and the _withassign Classes.....	576
Stream Errors	577
Error-Status Bits	577
Inputting Numbers.....	578

Too Many Characters	579
No-Input Input	579
Inputting Strings and Characters	580
Error-Free Distances	580
Disk File I/O with Streams	583
Formatted File I/O	583
Strings with Embedded Blanks	586
Character I/O	588
Binary I/O	589
The reinterpret_cast Operator	591
Closing Files	591
Object I/O	591
I/O with Multiple Objects	594
File Pointers	597
Specifying the Position	598
Specifying the Offset	598
The tellg() Function	601
Error Handling in File I/O	601
Reacting to Errors	601
Analyzing Errors	602
File I/O with Member Functions	604
Objects That Read and Write Themselves	604
Classes That Read and Write Themselves	607
Overloading the Extraction and Insertion Operators	616
Overloading for cout and cin	616
Overloading for Files	618
Memory as a Stream Object	620
Command-Line Arguments	622
Printer Output	624
Summary	626
Questions	627
Exercises	628
13 Multifile Programs 633	
Reasons for Multifile Programs	634
Class Libraries	634
Organization and Conceptualization	635
Creating a Multifile Program	637
Header Files	637
Directory	637
Projects	637

Inter-File Communication.....	638
Communication Among Source Files	638
Header Files.....	643
Namespaces	647
A Very Long Number Class	651
Numbers as Strings	652
The Class Specifier	652
The Member Functions	654
The Application Program	657
A High-Rise Elevator Simulation	658
Running the ELEV Program	658
Designing the System	660
Listings for ELEV	662
Elevator Strategy	674
State Diagram for the ELEV Program.....	675
Summary.....	676
Questions.....	677
Projects	679
 14 Templates and Exceptions 681	
Function Templates.....	682
A Simple Function Template.....	684
Function Templates with Multiple Arguments	686
Class Templates	690
Class Name Depends on Context	694
A Linked List Class Using Templates.....	696
Storing User-Defined Data Types	698
The UML and Templates.....	702
Exceptions.....	703
Why Do We Need Exceptions?	703
Exception Syntax.....	704
A Simple Exception Example	706
Multiple Exceptions.....	710
Exceptions with the Distance Class	712
Exceptions with Arguments.....	714
The bad_alloc Class	717
Exception Notes.....	718
Summary.....	720
Questions.....	720
Exercises	722

15 The Standard Template Library 725

Introduction to the STL	726
Containers	727
Algorithms	732
Iterators	733
Potential Problems with the STL	734
Algorithms	735
The find() Algorithm	735
The count() Algorithm	736
The sort() Algorithm	737
The search() Algorithm	737
The merge() Algorithm	738
Function Objects	739
The for_each() Algorithm	742
The transform() Algorithm	742
Sequence Containers.....	743
Vectors	743
Lists	747
Deque	750
Iterators	751
Iterators as Smart Pointers.....	752
Iterators as an Interface	753
Matching Algorithms with Containers	755
Iterators at Work	759
Specialized Iterators	763
Iterator Adapters	763
Stream Iterators	767
Associative Containers	771
Sets and Multisets	771
Maps and Multimaps	775
Storing User-Defined Objects.....	778
A Set of person Objects	778
A List of person Objects	782
Function Objects	786
Predefined Function Objects	786
Writing Your Own Function Objects.....	789
Function Objects Used to Modify Container Behavior	794
Summary	794
Questions.....	795
Exercises	797

16 Object-Oriented Software Development	801
Evolution of the Software Development Processes	802
The Seat-of-the-Pants Process	802
The Waterfall Process	802
Object-Oriented Programming	803
Modern Processes	803
Use Case Modeling	805
Actors	805
Use Cases	806
Scenarios	806
Use Case Diagrams	806
Use Case Descriptions	807
From Use Cases to Classes	808
The Programming Problem	809
Hand-Written Forms	809
Assumptions	811
The Elaboration Phase for the LANDLORD Program	812
Actors	812
Use Cases	812
Use Case Descriptions	813
Scenarios	815
UML Activity Diagrams	815
From Use Cases to Classes	816
Listing the Nouns	816
Refining the List	817
Discovering Attributes	818
From Verbs to Messages	818
Class Diagram	820
Sequence Diagrams	820
Writing the Code	824
The Header File	825
The .CPP Files	831
More Simplifications	841
Interacting with the Program	841
Final Thoughts	843
Summary	844
Questions	844
Projects	846
A ASCII Chart	849
B C++ Precedence Table and Keywords	859
Precedence Table	860
Keywords	860

C Microsoft Visual C++ 863

Screen Elements	864
Single-File Programs	864
Building an Existing File.....	864
Writing a New File	865
Errors	865
Run-Time Type Information (RTTI)	866
Multifile Programs	866
Projects and Workspaces	866
Developing the Project	867
Saving, Closing, and Opening Projects.....	868
Compiling and Linking	868
Building Console Graphics Lite Programs.....	868
Debugging.....	868
Single-Stepping	869
Watching Variables	869
Stepping Into Functions.....	869
Breakpoints	870

D Borland C++Builder 871

Running the Example Programs in C++Builder.....	872
Cleaning Up the Screen	873
Creating a New Project.....	873
Naming and Saving a Project	874
Starting with Existing Files	875
Compiling, Linking, and Executing	875
Executing from C++Builder	875
Executing from MS-DOS	875
Precompiled Header Files	876
Closing and Opening Projects.....	876
Adding a Header File to Your Project	876
Creating a New Header File	876
Editing an Existing Header File	876
Telling C++Builder the Header File's Location	877
Projects with Multiple Source Files	877
Creating Additional Source Files	877
Adding Existing Source Files to Your Project	877
The Project Manager	878
Console Graphics Lite Programs	878
Debugging.....	878
Single-Stepping	879
Watching Variables	879
Tracing into Functions.....	879
Breakpoints	879

E	Console Graphics Lite	881
	Using the Console Graphics Lite Routines	882
	The Console Graphics Lite Functions	883
	Implementations of the Console Graphics Lite Functions	884
	Microsoft Compilers	885
	Borland Compilers.....	885
	Source Code Listings	885
	Listing for MSOFTCON.H	886
	Listing for MSOFTCON.CPP	886
	Listing for BORLACON.H	890
	Listing for BORLACON.CPP	891
F	STL Algorithms and Member Functions	895
	Algorithms	896
	Member Functions	907
	Iterators	909
G	Answers to Questions and Exercises	913
	Chapter 1.....	914
	Answers to Questions	914
	Chapter 2.....	914
	Answers to Questions	914
	Solutions to Exercises	916
	Chapter 3.....	917
	Answers to Questions	917
	Solutions to Exercises	918
	Chapter 4.....	921
	Answers to Questions	921
	Solutions to Exercises	922
	Chapter 5.....	924
	Answers to Questions	924
	Solutions to Exercises	925
	Chapter 6.....	928
	Answers to Questions	928
	Solutions to Exercises	929
	Chapter 7.....	932
	Answers to Questions	932
	Solutions to Exercises	933
	Chapter 8.....	937
	Answers to Questions	937
	Solutions to Exercises	938
	Chapter 9.....	943
	Answers to Questions	943
	Solutions to Exercises	944

Chapter 10.....	949
Answers to Questions	949
Solutions to Exercises	950
Chapter 11.....	954
Answers to Questions	954
Solutions to Exercises	956
Chapter 12.....	960
Answers to Questions	960
Solutions to Exercises	961
Chapter 13.....	963
Answers to Questions	963
Chapter 14.....	964
Answers to Questions	964
Solutions to Exercises	965
Chapter 15.....	969
Answers to Questions	969
Solutions to Exercises	970
Chapter 16.....	974
Answers to Questions	974

H Bibliography 977

Advanced C++	978
Defining Documents	978
The Unified Modeling Language	978
The History of C++	979
Other Topics	979

Index 981