ABSOLUTE C++

WALTER SAVITCH
University of California, San Diego

Boston San Francisco New York London
Toronto Sydney Tokyo Singapore Madrid Mexico City
Munich Paris Cape Town Hong Kong Montreal

Executive Editor	Susan Hartman Sullivan
Executive Marketing Manager	Michael Hirsch
Project Management	Argosy Publishing
Composition and Art	Argosy Publishing
Copyeditor	Cindy Kogut
Proofreader	Bob LaRoche
Indexer	Larry Sweazy
Text and Cover Design	Leslie Haimes
Cover Photo	Renee Lynn/Stone by Getty Images
Design Manager	Gina Hagen Kolenda
Prepress and Manufacturing	Caroline Fell

Access the latest information about Addison-Wesley titles from our World Wide Web site:
http://www.aw.com/cs

Library of Congress Cataloging-in-Publication Data
Savitch, Walter J., 1943-
Absolute C++ / Walter Savitch
p. cm.
ISBN 0-201-70927-9 (pbk.)
1. C++ (Computer program language) I. Title.
QA76.73.C153 S279 2002
005.13'3--dc212001046383

ISBN 0-201-70927-9
12345678910-DOC-04030201

Preface

This book is designed to be a text book and reference for programming in the C++ language. Although it does include programming techniques, it is organized around the features of the C+++ language, rather than any particular curriculum of programming techniques. The main audience I had in mind when writing this book were undergraduate students who have had some programming experience but who had not yet learned the C++ language. As such it would be a suitable C++ text or reference for a second or later computer science course that uses C++ and could even be used for a first programming course. This book is designed to accommodate a wide range of users. The beginning chapters are written at a level that is accessible to beginners, while the boxed sections of those chapters serve to quickly introduce more experienced programmers to basic C++ syntax. Later chapters are still designed to be accessible, but are written at a level suitable for students who have progressed to these more advanced topics. (For those who want a beginning textbook with more pedagogical material and more on very basic programming technique, I suggest another book I wrote that is more along these lines.)

This book is also suitable for somebody learning the C++ language on their own. It is accessible for selfstudy and complete as a reference.

The C++ coverage in this book is very complete. It goes well beyond what a beginner needs to know. In particular it has extensive coverage of inheritance, polymorphism, and exception handling in C++. It also has an extensive introduction to the Standard Template Library (STL).

This book also includes an introduction to Patterns and UML. Since many computer science curricula postpone recursions to a second computer science course, it includes a full chapter on recursion.

More on the versatility of the text and other special features are given in the remainder of this preface.

ANSI/ISO C++ STANDARD

This book was written to conform to the new ANSI/ISO C++ Standard.

STANDARD TEMPLATE LIBRARY

The Standard Template Library is an extensive library collection of preprogrammed data structure classes and important algorithms. The STL is perhaps as big a topic as the core C++ language. However, this book does contain a very substantial introduction to the STL. There is a full chapter on the general topic of templates, a full chapter on the particulars of the STL, as well as other material on or related to the STL at other points in the text.

OBJECT-ORIENTED PROGRAMMING

This book is organized around the structure of the C++ language. As such, the earlier chapters which cover aspects of C++ that are common to most all high level programming languages are not particularly oriented toward OOP programming. As a reference book and as a book for learning a second language this makes sense. However, I do consider C++ to be an OOP language. If you are really programming in C++ and not C, then you must be availing yourself of the OOP features of C++. This book gives extensive coverage of encapsulation, inheritance, and polymorphism as realized in the C++ language. The final Chapter on Patters and UML gives additional coverage of OOP related material.

FLEXIBILITY IN TOPIC ORDERING

This book allows instructors wide latitude in reordering the material. This is important if a book is to serve as a reference. This is also in keeping with my philosophy of writing books that accommodate themselves to an instructor's style, rather than tieing the instructor to an author's personal preference of topic ordering. With this in mind, each chapter introduction explains what material must be covered before doing each section of the chapter.

ACCESSIBLE TO STUDENTS

It is not enough for a book to present the right topics in the right order. It is not even enough for it be clear and correct when read by an instructor or other expert. The material needs to be presented in a way that is accessible to the person who does not yet know the material. Like my other text books that proved to be very popular with students, this book was written to be friendly and accessible to the student.

SUMMARY BOXES

Each major point is summarized in a boxed section. These boxed sections are spread throughout each chapter. They serve as summaries of the material, as a quick reference source, and as way to quickly learn the C++ syntax for feature you know about in general but for which you do not know the C++ particulars.

SELF-TEST EXERCISES

Each chapter contains numerous Self-Test Exercises at strategic points in the chapter. Complete answers for all the Self-Test Exercises are given at the end of each chapter.

OTHER FEATURES

Pitfall sections, programming technique sections, and examples of complete programs with sample I/O are given throughout each chapter. Each chapter ends with a summary section and a collection of programming projects suitable to assign to students.

SUPPORT MATERIAL

A full package of support material is available to instructors who adopt this book. They include: an instructor's manual, all the source code from the text, PowerPoint slides, a computerized testbank, and a free copy of the Microsoft Visual C++ 6.0 Introductory edition.

The source code and other information is available to all readers at the books companion website.

ACKNOWLEDGMENTS

Numerous individuals have contributed invaluable help and support in making this book happen. Frank Ruggirello and my editor Susan Hartman at Addison-Wesley are the ones who first conceived of the idea for this book. Susan Hartman, Galia Shokry, Lisa Kalner, and the other fine people at Addison-Wesley were a continuing source of support and encouragement in getting the book reviewed, revised, and out the door.

Cindy Kogut did an incredible thorough job of copy editing. Sally Boylan and others at Argosy Publishing did great work under rushed conditions in converting the manuscript to type set pages.

David Teague deserves special acknowledgment. I very much appreciate his hard work, good insights, and careful researching for this book.

I thank my good friend Mario Lopez for the many helpful conversations we had about C++.

The following list of reviewers provided corrections and suggestions that contributed greatly to the final product. I thank them all. In random order they are: Kenrick Mock, University of Alaska, Anchorage; Richard Albright, University of Delaware; H.E. Dunsmore, Purdue University; Christopher E. Cramer; Drue Coles, Boston University; Evan Golub, University of Maryland; Stephen Corbesero, Moravian College; Fredrick H. Colclough, Colorado Technical University; Joel Weinstein, Northeastern University; Stephen P. Leach, Florida State University; Alvin S. Lim, Auburn University; Martin Dulberg, North Carolina State University.

I again thank David Teague. This time for his excellent work in preparing the instructor's guide.

Finally, I thank Christina for putting up with my working late on the book and even offering encouragement instead of complaining.

W.S.
http://www-cse.ucsd.edu/users/savitch/
wsavitch@ucsd.edu

Contents