



PEARSON

DEITEL®

C++大学教程

——(第八版)(英文版)——

C++ How to Program, Eighth Edition



中国工信出版集团



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
<http://www.phei.com.cn>

[美]

Paul Deitel
Harvey Deitel

著

C++

HOW TO PROGRAM

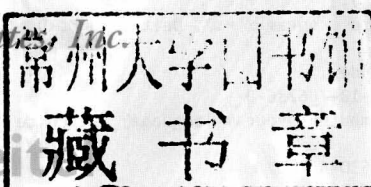
EIGHTH EDITION

Paul Deitel

Deitel & Associates, Inc.

Harvey Deitel

Deitel & Associates, Inc.



Prentice Hall

Boston Columbus Indianapolis New York San Francisco Upper Saddle River
Amsterdam Cape Town Dubai London Madrid Milan Munich Paris Montreal Toronto
Delhi Mexico City Sao Paulo Sydney Hong Kong Seoul Singapore Taipei Tokyo

内 容 简 介

本书是一本 C++ 编程方面的优秀教材,全面介绍了面向对象编程的原理和方法,详细分析了与 C++ 编程有关的技术,具体包括类与对象、控制语句、函数与递归、数组、指针、运算符重载、继承、多态、模板、流输入/输出、异常处理、文件处理、搜索与排序、数据结构、标准模板库等内容,本书的同步学习网站上还包含了更多的扩展内容。全书以“活代码”方式详细分析了每个知识要点,是初学者和中高级程序员学习 C++ 编程的理想用书。

本书可作为高等院校相关专业的 C++ 程序设计课程的双语教材,也是软件设计人员学习 C++ 编程的理想读物。

Original edition, entitled C++ How to Program, Eighth Edition, ISBN 9780132662369, by Paul Deitel and Harvey Deitel, published by Pearson Education, Inc., publishing as Prentice Hall, Copyright © 2012 Pearson Education, Inc.

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from Pearson Education, Inc.

China edition published by PEARSON EDUCATION ASIA LTD., and PUBLISHING HOUSE OF ELECTRONICS INDUSTRY copyright © 2015.

This edition is manufactured in the People's Republic of China, and is authorized for sale and distribution only in the People's Republic of China exclusively(exclude Taiwan, Hong Kong SAR and Macau SAR).

本书英文影印版专有出版权由 Pearson Education(培生教育出版集团)授予电子工业出版社。未经出版者预先书面许可,不得以任何方式复制或抄袭本书的任何部分。

本书在中国大陆地区出版,仅限在中国大陆发行。

本书贴有 Pearson Education(培生教育出版集团)激光防伪标签,无标签者不得销售。

版权贸易合同登记号 图字:01-2015-1187

图书在版编目(CIP)数据

C++ 大学教程:第8版:英文/(美)戴特尔(Deitel, P.), (美)戴特尔(Deitel, H.)著.

北京:电子工业出版社,2015.8

书名原文:C++ How to Program, Eighth Edition

国外计算机科学教材系列

ISBN 978-7-121-26647-8

I. ①C… II. ①戴…②戴… III. ①C 语言-程序设计-高等学校-教材 IV. ①TP312

中国版本图书馆 CIP 数据核字(2015)第 161549 号

策划编辑:冯小贝

责任编辑:冯小贝

印 刷:三河市鑫金马印装有限公司

装 订:三河市鑫金马印装有限公司

出版发行:电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本:787×1092 1/16 印张:52 字数:1990 千字

版 次:2005 年 9 月第 1 版(原书第 4 版)

2015 年 8 月第 4 版(原书第 8 版)

印 次:2015 年 8 月第 1 次印刷

定 价:135.00 元

凡所购买电子工业出版社图书有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,联系及邮购电话:(010)88254888。

质量投诉请发邮件至 zlt@phei.com.cn,盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线:(010)88258888。

Vice President and Editorial Director: *Marcia J. Horton*
Editor-in-Chief: *Michael Hirsch*
Associate Editor: *Carole Snyder*
Vice President, Marketing: *Patrice Jones*
Marketing Manager: *Yezan Alayan*
Senior Marketing Coordinator: *Kathryn Ferranti*
Vice President, Production: *Vince O'Brien*
Managing Editor: *Jeff Holcomb*
Associate Managing Editor: *Robert Engelhardt*
Operations Specialist: *Lisa McDowell*
Art Director: *Linda Knowle*
Cover Design: *Abbey S. Deitel, Harvey M. Deitel, Marta Samsel*
Cover Photo Credit: © *James Hardy/PhotoAlto/Getty Images*
Media Editor: *Daniel Sandin*
Media Project Manager: *Wanda Rockwell*

Credits and acknowledgments borrowed from other sources and reproduced, with permission, in this textbook appear on page vi.

The authors and publisher of this book have used their best efforts in preparing this book. These efforts include the development, research, and testing of the theories and programs to determine their effectiveness. The authors and publisher make no warranty of any kind, expressed or implied, with regard to these programs or to the documentation contained in this book. The authors and publisher shall not be liable in any event for incidental or consequential damages in connection with, or arising out of, the furnishing, performance, or use of these programs.

Copyright © 2012, 2008, 2005, 2003, 2001 Pearson Education, Inc., publishing as Prentice Hall. All rights reserved. Manufactured in the United States of America. This publication is protected by Copyright, and permission should be obtained from the publisher prior to any prohibited reproduction, storage in a retrieval system, or transmission in any form or by any means, electronic, mechanical, photocopying, recording, or likewise. To obtain permission(s) to use material from this work, please submit a written request to Pearson Education, Inc., Permissions Department, 501 Boylston Street, Suite 900, Boston, Massachusetts 02116.

Many of the designations by manufacturers and sellers to distinguish their products are claimed as trademarks. Where those designations appear in this book, and the publisher was aware of a trademark claim, the designations have been printed in initial caps or all caps.

Library of Congress Cataloging-in-Publication Data

Deitel, Paul J.

C++ : how to program / P.J. Deitel, H.M. Deitel. -- 8th ed.
p. cm.

Includes index.

ISBN 978-0-13-266236-9

1. C++ (Computer program language) I. Deitel, Harvey M. II. Title.

QA76.73.C153D45 2012

005.13'3--dc22

2011000245

10 9 8 7 6 5 4 3 2 1

ISBN-10: 0-13-266236-1

ISBN-13: 978-0-13-266236-9

Prentice Hall
is an imprint of

PEARSON

C++

HOW TO PROGRAM

EIGHTH EDITION

Deitel® How to Program Series Cover Theme

The cover theme for the DEITEL® HOW TO PROGRAM SERIES emphasizes social consciousness issues such as going green, clean energy, recycling, sustainability and more. Within the text, in addition to conventional programming exercises, we've included our Making a Difference exercise set to raise awareness of issues such as global warming, population growth, affordable healthcare, accessibility, privacy of electronic records and more. In this book, you'll use C++ to program applications that relate to these issues. We hope that what you learn in *C++ How to Program*, 8/e will help you to make a difference.

Rainforests



The world's rainforests are often referred to as the "Earth's lungs," the "jewels of the Earth" and the "world's largest pharmacy." Approximately 50% of the world's tropical rainforests are in Central and South America, over 33% are in Asia and Oceania (which consists of Australia, New Zealand and various South Pacific Islands), and 15% are in Africa. Rainforests absorb from the atmosphere vast amounts of carbon dioxide—a gas that many scientists blame for global warming—and they provide approximately 40% of the world's oxygen. They regulate water flow to surrounding areas preventing mudslides and crop loss. Rainforests also support the livelihoods of 1.6 billion people, providing food, fresh water, medicines and more. Approximately 25% of Western medicines used to treat infections, viruses, cancer and more are derived from plants

found in rainforests. The U.S. National Cancer Institute has found about 2100 rainforest plant species that are effective against cancer cells. Fewer than one percent of rainforest plant species have been tested for medical use.

Rainforests are being deforested at an alarming rate. According to a March 2010 report by the United Nations Food and Agriculture Organization, deforestation has slowed over the last 10 years, but more than 30 million acres of forests are still lost annually, and they're not easily renewed. The United Nations Environment Programme Plant for the Planet: Billion Tree Campaign is one of many reforestation initiatives. To learn more about how you can make a difference, visit www.unep.org/billiontreecampaign/index.asp. For further information visit:

www.rain-tree.com/facts.htm

www.savetherainforest.org/savetherainforest_007.htm

en.wikipedia.org/wiki/Rainforest

www.rainforestfoundation.org/

About Deitel & Associates, Inc.

Deitel & Associates, Inc., is an internationally recognized authoring and corporate training organization. The company offers instructor-led courses delivered at client sites worldwide on programming languages and other software topics such as C++, Visual C++®, C, Java, C#®, Visual Basic®, Objective-C®, XML®, Python®, JavaScript, object technology, Internet and web programming, and Android and iPhone app development. The company's clients include many of the world's largest companies, as well as government agencies, branches of the military and academic institutions. To learn more about Deitel Pearson Higher Education publications and Dive Into® Series corporate training, e-mail deitel@deitel.com or visit www.deitel.com/training/. Follow Deitel on Facebook® at www.deitel.com/deitelfan/ and on Twitter® @deitel.

Continued from Back Cover

- "I really like the Making a Difference programming exercises. The game programming [in the Functions chapter] gets students excited."*—Virginia Bailey, Jackson State University
- "It's great that the text introduces object-oriented programming early. The car analogy was well-thought out. An extremely meticulous treatment of control structures. The virtual function figure and corresponding explanation in the Polymorphism chapter is thorough and truly commendable."*—Gregory Dai, eBay, Inc.
- "An excellent 'objects first' coverage of C++ that remains accessible to beginners. The example-driven presentation is enriched by the optional OO design case study that contextualizes the material in a software engineering project."*
—Gavin Osborne, Saskatchewan Institute of Applied Science and Technology
- "It is excellent that [the authors] use the STL and standard libraries early."*—John Dibling, SpryWare
- "Provides a complete basis of fundamental instruction in all core aspects of C++. Examples provide a solid grounding in the construction of C++ programs. A solid overview of C++ Stream I/O."*
—Peter DePasquale, The College of New Jersey
- "Great discussion about the mistakes resulted from using = for == and vice versa!"*
—Wing-Ning Li, University of Arkansas
- "Thorough and detailed coverage of exceptions from an object-oriented point of view."*
—Dean Mathias, Utah State University
- "Chapter 20 (Data Structures) is very good. The examples are accessible to CS, IT, software engineering and business students."*—Thomas J. Borrelli, Rochester Institute of Technology
- "The Simpletron exercises are brilliant. The Polymorphism chapter explains one of the hardest topics to understand in OOP in a clear manner. Great job! The writing is excellent, the examples are well developed and the exercises are interesting."*—José Antonio González Seco, Parliament of Andalusia, Spain
- "Introducing the UML early is a great idea."*—Raymond Stephenson, Microsoft
- "Good use of diagrams, especially of the activation call stack and recursive functions."*
—Amar Raheja, California State Polytechnic University, Pomona
- "Terrific discussion of pointers—the best I have seen."*—Anne B. Horton, Lockheed Martin
- "Great coverage of polymorphism and how the compiler implements polymorphism 'under the hood.' I wish I had such a clear presentation of data structures when I was a student."*—Ed James-Beckham, Borland
- "A nice introduction to searching and sorting, and Big-O."*—Robert Myers, Florida State University
- "Will get you up and running quickly with the memory management and regular expression libraries."*
—Ed Brey, Kohler Co.
- "Excellent introduction to the Standard Template Library (STL). The best book on C++ programming for the serious student!"*—Richard Albright, Goldey-Beacom College
- "Each code example is completely reviewed. This is a critical step for students to learn good programming practices."*
—Jack R. Hagemester, Washington State University
- "The most thorough C++ treatment I've seen. Replete with real-world case studies covering the full software development lifecycle. Code examples are extraordinary!"*—Terrell Hull, Logicalis Integration Solutions

PEARSON

北京培生信息中心
北京市东城区北三环东路 36 号
北京环球贸易中心 D 座 1208 室
邮政编码: 100013
电话: (8610) 57355171/57355169/57355176
传真: (8610) 58257961

Beijing Pearson Education
Information Centre
Suit 1208, Tower D, Beijing Global Trade Centre,
36 North Third Ring Road East,
Dongcheng District, Beijing, China 100013
TEL: (8610) 57355171/57355169/57355176
FAX: (8610) 58257961

尊敬的老师:

您好!

为了确保您及时有效地申请教辅资源, 请您务必完整填写如下教辅申请表, 加盖学院公章后将扫描件用电子邮件的形式发送给我们, 我们将会 在 2-3 个工作日内为您开通属于您个人的唯一账号以供您下载与教材配套的教师资源。

请填写所需教辅的开课信息:

采用教材	<input type="checkbox"/> 中文版 <input type="checkbox"/> 英文版 <input type="checkbox"/> 双语版		
作 者		出版社	
版 次		ISBN	
课程时间	始于 年 月 日	学生人数	
	止于 年 月 日	学生年级	<input type="checkbox"/> 专科 <input type="checkbox"/> 本科 1/2 年级 <input type="checkbox"/> 研究生 <input type="checkbox"/> 本科 3/4 年级

请填写您的个人信息:

学 校			
院系/专业			
姓 名		职 称	<input type="checkbox"/> 助教 <input type="checkbox"/> 讲师 <input type="checkbox"/> 副教授 <input type="checkbox"/> 教授
通信地址/邮编			
手 机		电 话	
传 真			
official email(必填) (eg:XXX@ruc.edu.cn)		email (eg:XXX@163.com)	
是否愿意接受我们定期的新书讯息通知: <input type="checkbox"/> 是 <input type="checkbox"/> 否			

Publishing House of Electronics Industry
电子工业出版社: www.phei.com.cn
www.hxedu.com.cn
北京市万寿路 173 信箱高等教育分社(100036)
联系电话: 010-88254555
E-mail: Te_service@phei.com.cn

系 / 院主任: _____ (签字)

(系 / 院办公室章)

____年____月____日

Deitel® Series Page

How To Program Series

C++ How to Program, 8/E
C How to Program, 6/E
Java™ How to Program, 9/E
Java™ How to Program, Late Objects Version, 8/E
Internet & World Wide Web How to Program, 4/E
Visual C++® 2008 How to Program, 2/E
Visual Basic® 2010 How to Program
Visual C#® 2010 How to Program, 3/E
Small Java™ How to Program, 6/E
Small C++ How to Program, 5/E

Simply Series

Simply C++: An App-Driven Tutorial Approach
Simply Java™ Programming: An App-Driven Tutorial Approach
Simply C#: An App-Driven Tutorial Approach
Simply Visual Basic® 2008, 3/E: An App-Driven Tutorial Approach

CourseSmart Web Books

www.deitel.com/books/CourseSmart/
C++ How to Program, 5/E, 6/E, 7/E & 8/E
Simply C++: An App-Driven Tutorial Approach
Java™ How to Program, 6/E, 7/E, 8/E & 9/E

(continued next column)

(continued)

Simply Visual Basic 2008: An App-Driven Tutorial Approach, 3/E
Visual Basic® 2010 How to Program
Visual Basic® 2008 How to Program
Visual C#® 2010 How to Program, 4/E
Visual C#® 2008 How to Program, 3/E

Deitel® Developer Series

C++ for Programmers
AJAX, Rich Internet Applications and Web Development for Programmers
Android for Programmers: An App-Driven Approach
C# 2010 for Programmers, 3/E
iPhone for Programmers: An App-Driven Approach
Java™ for Programmers
JavaScript for Programmers

LiveLessons Video Learning Products

www.deitel.com/books/LiveLessons/
C++ Fundamentals
Java™ Fundamentals
C# Fundamentals
iPhone® App Development Fundamentals
JavaScript Fundamentals
Visual Basic Fundamentals

To receive updates on Deitel publications, Resource Centers, training courses, partner offers and more, please register for the free *Deitel® Buzz Online* e-mail newsletter at:

www.deitel.com/newsletter/subscribe.html

follow us on Twitter®

@deitel

and become a Deitel & Associates fan on Facebook®

www.deitel.com/deitelfan/

To communicate with the authors, send e-mail to:

deitel@deitel.com

For information on government and corporate *Dive-Into® Series* on-site seminars offered by Deitel & Associates, Inc. worldwide, visit:

www.deitel.com/training/

or write to

deitel@deitel.com

For continuing updates on Prentice Hall/Deitel publications visit:

www.deitel.com
www.pearsonhighered.com/deitel/

Check out our Resource Centers for valuable web resources that will help you master C++, other important programming languages, software, and Internet- and web-related topics:

www.deitel.com/ResourceCenters.html

*In memory of Ken Olsen,
Founder of Digital Equipment Corporation (DEC):*

*We are deeply grateful for the opportunities
DEC extended to us, enabling us to form and grow
Deitel & Associates, Inc.*

Paul and Harvey Deitel

Trademarks

DEITEL, the double-thumbs-up bug and DIVE INTO are registered trademarks of Deitel and Associates, Inc.

Microsoft and the Windows logo are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

UNIX is a registered trademark of The Open Group.

Throughout this book, trademarks are used. Rather than put a trademark symbol in every occurrence of a trademarked name, we state that we are using the names in an editorial fashion only and to the benefit of the trademark owner, with no intention of infringement of the trademark.

Preface

“The chief merit of language is clearness ...”

—Galen

For the Student

Welcome to the C++ computer programming language and *C++ How to Program, Eighth Edition*! This book presents leading-edge computing technologies, and is particularly appropriate for introductory course sequences based on the curriculum recommendations of two key professional organizations—the ACM and the IEEE.

The new Chapter 1 presents intriguing facts and figures. Our goal is to get you excited about studying computers and programming. The chapter includes a table of some of the research made possible by computers; current technology trends and hardware discussions; the data hierarchy; social networking; a table of business and technology publications and websites that will help you stay up-to-date with the latest technology news, trends and career opportunities; additional Making a Difference exercises and more.

We focus on software engineering best practices. At the heart of the book is our signature “live-code approach”—programming concepts are presented in the context of complete working programs, rather than in code snippets. Each C++ code example is accompanied by live sample executions, so you can see exactly what each program does when it’s run on a computer. All the source code is available at www.deitel.com/books/cpphttp8/ and www.pearsonhighered.com/deitel/.

Much of this Preface is addressed to instructors. Please be sure to read the sections entitled Pedagogic Features; Teaching Approach; Software Used in *C++ How to Program, 8/e*; C++.

We believe that this book and its support materials will give you an informative, interesting, challenging and entertaining C++ educational experience. As you read the book, if you have questions, send an e-mail to deitel@deitel.com—we’ll respond promptly. For updates on this book, visit www.deitel.com/books/cpphttp8/, follow us on Facebook (www.deitel.com/deitelfan) and Twitter (@deitel), and subscribe to the *Deitel Buzz Online* newsletter (www.deitel.com/newsletter/subscribe.html). Good luck!

New and Updated Features

Here are the updates we’ve made for *C++ How to Program, 8/e*:

Impending New C++ Standard

- **Optional sections.** We cover various features of the new standard (sometimes called C++0x and due late in 2011 or early in 2012) in *optional modular sections* and in Chapter 23. These are *easy to include or omit*. Popular compilers such as Microsoft Visual C++ 2010 and GNU C++ 4.5 already implement many of these features. To enable the new standard features in GNU C++, use the `-std=C++0x` flag when you compile the corresponding programs.
- **Boost C++ Libraries, Technical Report 1 (TR1) and C++0x.** In Chapter 23, we introduce the Boost C++ Libraries, Technical Report 1 (TR1) and C++0x. The free Boost open source libraries are created by members of the C++ community. Technical Report 1 describes the proposed changes to the C++ Standard Library, many of which are based on current Boost libraries. The C++ Standards Committee is revising the C++ Standard—the main goals are to make C++ easier to learn, improve library building capabilities, and increase compatibility with the C programming language. The new standard will include many of the libraries in TR1 and changes to the core language. We overview the Boost libraries and provide code examples for the “regular expression” and “smart pointer” libraries. Regular expressions are used to match specific character patterns in text. They can be used, for example, to validate data to ensure that it’s in a particular format, to replace parts of one string with another, or to split a string. Many common bugs in C and C++ code are related to pointers, a powerful programming capability you’ll study in Chapter 8. Smart pointers help you avoid errors by providing additional functionality to standard pointers.

- ***unique_ptr* vs. *auto_ptr*.** We replaced our `auto_ptr` example with the impending standard's class `unique_ptr`, which fixes various problems that were associated with class `auto_ptr`. Use of `auto_ptr` is deprecated and `unique_ptr` is already implemented in many popular compilers, including Visual C++ 2010 and GNU C++ 4.5.
- ***Initializer lists for user-defined types.*** These enable objects of your own types to be initialized using the same syntax as built-in arrays.
- ***Range-based for statement.*** A version of the `for` statement that iterates over all the elements of an array or container (such as an object of the `vector` class).
- ***Lambda expressions.*** These enable you to create anonymous functions that can be passed to other functions as arguments.
- ***auto storage class specifier.*** The keyword `auto` can no longer be used as a storage class specifier.
- ***auto.*** This keyword now deduces the type of a variable from its initializer.
- ***nullptr.*** This keyword is a replacement for assigning zero to a null pointer.
- ***static_assert.*** This capability allows you to test certain aspects of the program at compile time.
- ***New long long and unsigned long long types.*** These new types were introduced for use with 64-bit machines.

Pedagogic Features

- ***Enhanced Making a Difference exercises set.*** We encourage you to use computers and the Internet to research and solve significant social problems. These exercises are meant to increase awareness and discussion of important issues the world is facing. We hope you'll approach them with your own values, politics and beliefs. Check out our new Making a Difference Resource Center at www.deitel.com/MakingADifference for additional ideas you may want to investigate further.
- ***Page numbers for key terms in chapter summaries.*** For key terms that appear in the chapter summaries, we include the page number of each term's defining occurrence in the chapter.
- ***Modular presentation.*** We've grouped the chapters into teaching modules. The Chapter Dependency Chart (later in this Preface) reflects the modularization.

Object Technology

- ***Object-oriented programming and design.*** We introduce the basic concepts and terminology of object technology in Chapter 1. Students develop their first customized classes and objects in Chapter 3. Presenting objects and classes early gets students "thinking about objects" immediately and mastering these concepts more thoroughly. [For courses that require a late-objects approach, consider *C++ How to Program, Late Objects Version, Seventh Edition*, which begins with six chapters on programming fundamentals (including two on control statements) and continues with seven chapters that gradually introduce object-oriented programming concepts.]
- ***Integrated case studies.*** We provide several case studies that span multiple sections and chapters. These include development of the `GradeBook` class in Chapters 3-7, the `Time` class in Chapters 9-10, the `Employee` class in Chapters 12-13, and the optional OOD/UML ATM case study in Chapters 25-26.
- ***Integrated GradeBook case study.*** The `GradeBook` case study uses classes and objects in Chapters 3-7 to incrementally build a `GradeBook` class that represents an instructor's grade book and performs various calculations based on a set of student grades, such as calculating the average grade, finding the maximum and minimum, and printing a bar chart.
- ***Exception handling.*** We integrate basic exception handling early in the book. Instructors can easily pull more detailed material forward from Chapter 16, *Exception Handling: A Deeper Look*.
- ***Prefer vectors to C arrays.*** C++ offers two types of arrays—`vector` class objects (which we start using in Chapter 7) and C-style, pointer-based arrays. As appropriate, we use class template `vector` instead of C arrays throughout the book. However, we begin by discussing C arrays in Chapter 7 to prepare you for working with legacy code and to use as a basis for building your own customized `Array` class in Chapter 11.
- ***Prefer string objects to C strings.*** Similarly, C++ offers two types of strings—`string` class objects (which we use starting in Chapter 3) and C-style, pointer-based strings. We continue to include some early

discussions of C strings to give you practice with pointer manipulations, to illustrate dynamic memory allocation with `new` and `delete` and to prepare you for working with C strings in the legacy code that you'll encounter in industry. In new development, you should favor `string` class objects. We've replaced most occurrences of C strings with instances of C++ class `string` to make programs more robust and eliminate many of the security problems that can be caused by using C strings.

- **Optional case study: Using the UML to develop an object-oriented design and C++ implementation of an ATM.** The UML (Unified Modeling Language) is the industry-standard graphical language for modeling object-oriented systems. Chapters 25-26 include an *optional* online case study on object-oriented design using the UML. We design and implement the software for a simple automated teller machine (ATM). We analyze a typical requirements document that specifies the system to be built. We determine the classes needed to implement that system, the attributes the classes need to have, the behaviors the classes need to exhibit and specify how the classes must interact with one another to meet the system requirements. From the design we produce a complete C++ implementation. Students often report having a “light-bulb moment”—the case study helps them “tie it all together” and really understand object orientation.
- **Standard Template Library (STL).** This might be one of the most important topics in the book in terms of your appreciation of software reuse. The STL defines powerful, template-based, reusable components that implement many common data structures and algorithms used to process those data structures. Chapter 22 introduces the STL and discusses its three key components—containers, iterators and algorithms. The STL components provide tremendous expressive power, often reducing many lines of code to a single statement.

Other Features

- **Printed book contains core content; additional chapters are online.** Several online chapters are included for more advanced courses and for professionals. These are available in searchable PDF format.
- **Reorganized Chapter 11, Operator Overloading; Class `string`.** We reorganized this chapter to begin with standard library class `string` so readers can see an elegant use of operator overloading before they implement their own. We also moved the section on proxy classes to the end of Chapter 10, where it's a more natural fit.
- **Enhanced use of `const`.** We increased the use of `const` book-wide to encourage better software engineering.
- **Software engineering concepts.** Chapter 1 briefly introduces very current software engineering terminology, including agile software development, Web 2.0, Ajax, SaaS (Software as a Service), PaaS (Platform as a Service), cloud computing, web services, open source software, design patterns, refactoring, LAMP and more.
- **Compilation and linking process for multiple-source-file programs.** Chapter 3 includes a detailed diagram and discussion of the compilation and linking process that produces an executable program.
- **Function Call Stack Explanation.** In Chapter 6, we provide a detailed discussion with illustrations of the function call stack and activation records to explain how C++ is able to keep track of which function is currently executing, how automatic variables of functions are maintained in memory and how a function knows where to return after it completes execution.
- **Tuned Treatment of Inheritance and Polymorphism.** Chapters 12-13 have been carefully tuned using a concise `Employee` class hierarchy. We use this same treatment in our C++, Java, C# and Visual Basic books—one of our reviewers called it the best he had seen in 25 years as a trainer and consultant.
- **Discussion and illustration of how polymorphism works “under the hood.”** Chapter 13 contains a detailed diagram and explanation of how C++ can implement polymorphism, virtual functions and dynamic binding internally. This gives students a solid understanding of how these capabilities work.
- **ISO/IEC C++ standard compliance.** We've audited our presentation against the ISO/IEC C++ standard document.
- **Debugger appendices.** We provide two Using the Debugger appendices on the book's Companion Website—Appendix H, Using the Visual Studio Debugger, and Appendix I, Using the GNU C++ Debugger.
- **Code tested on multiple platforms.** We tested the code examples on various popular C++ platforms including GNU C++ on Linux and Microsoft Windows, and Visual C++ on Windows. For the most part, the book's examples port to popular standard-compliant compilers.

- **Game Programming.** Because of limited interest, we've removed from the book Chapter 27, Game Programming with Ogre (which covers only Linux). For instructors who would like to continue using this material with *C++ How to Program, 8/e*, we've included the version from *C++ How to Program, 7/e* on the Website.

Our Text + Digital Approach to Content

We surveyed hundreds of instructors teaching C++ courses and learned that most want a book with content focused on their introductory courses. With that in mind, we moved various advanced chapters to the web. Having this content in digital format makes it easily searchable, and gives us the ability to fix errata and add new content as appropriate. The book's Companion Website, which is accessible at

www.pearsonhighered.com/deitel/

contains the following chapters in *searchable PDF format*^①:

- Chapter 25, ATM Case Study, Part 1: Object-Oriented Design with the UML
- Chapter 26, ATM Case Study, Part 2: Implementing an Object-Oriented Design
- Game Programming with Ogre (from *C++ How to Program, 7/e*)
- Appendix F, C Legacy Code Topics
- Appendix G, UML 2: Additional Diagram Types
- Appendix H, Using the Visual Studio Debugger
- Appendix I, Using the GNU C++ Debugger

The following materials are posted at the Companion Website and at www.deitel.com/books/cpphttp8/:

- An array of function pointers example and additional function pointer exercises (from Chapter 8).
- String Class Operator Overloading Case Study (from Chapter 11).
- Building Your Own Compiler exercise descriptions (from Chapter 20).

Dependency Chart

The chart on the next page shows the dependencies among the chapters to help instructors plan their syllabi. *C++ How to Program, 8/e* is appropriate for CS1 and CS2 courses.

Teaching Approach

C++ How to Program, 8/e, contains a rich collection of examples. We stress program clarity and concentrate on building well-engineered software.

Live-code approach. The book is loaded with “live-code” examples—most new concepts are presented in the context of *complete working C++ applications*, followed by one or more executions showing program inputs and outputs. In the few cases where we use a code snippet, we tested it in a complete working program, then copied and pasted it into the book.

Code highlighting. We place light blue shaded rectangles around each program's key code segments.

Using fonts for emphasis. We emphasize on-screen components in the **bold Helvetica** font (e. g., the File menu) and C++ program text in the *Lucida* font (for example, `intx = 5;`).

Objectives. The opening quotes are followed by a list of chapter objectives.

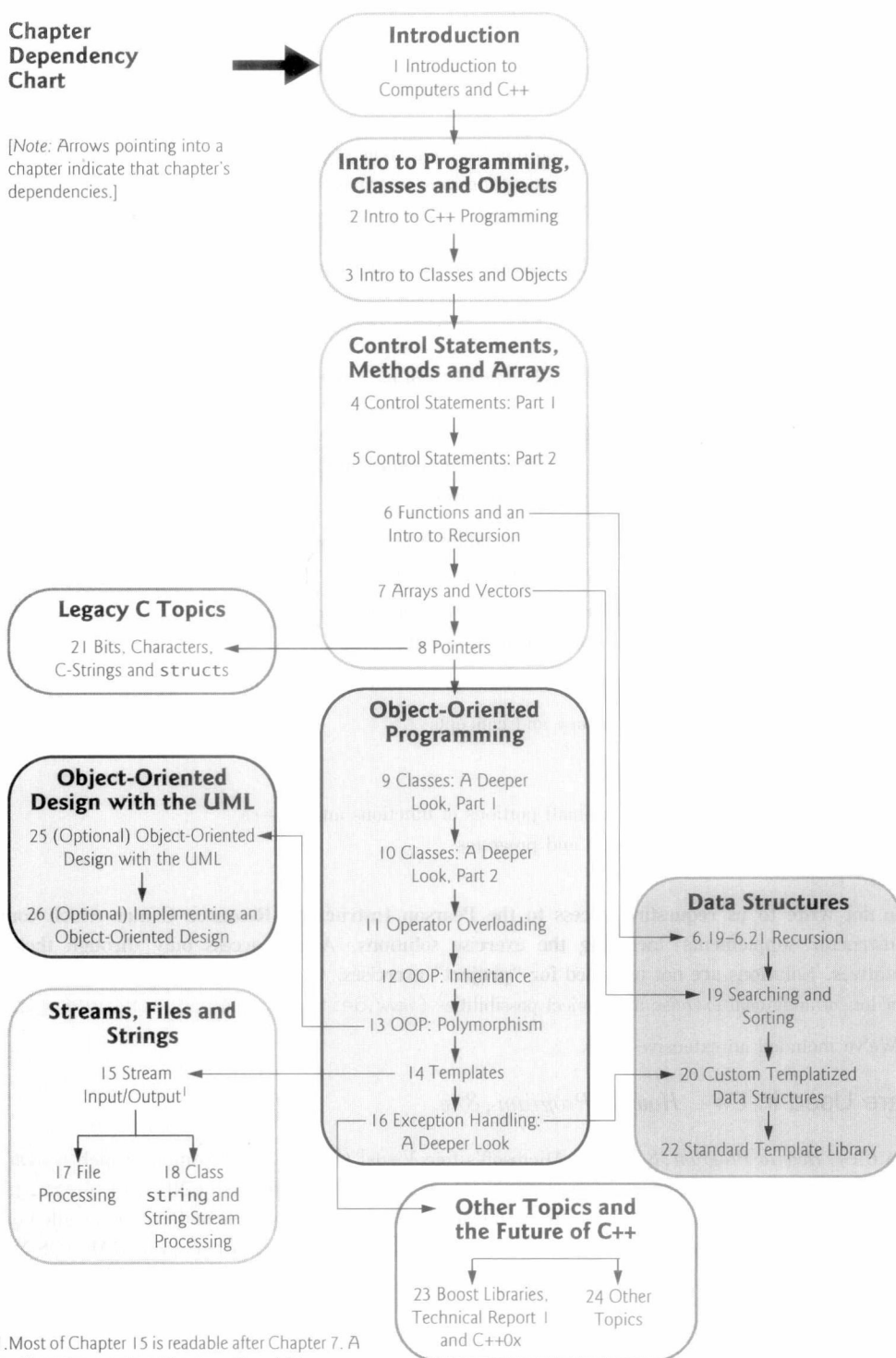
Illustrations/ figures. Abundant tables, line drawings, UML diagrams, programs and program outputs are included.

Programming tips. We include programming tips to help you focus on important aspects of program development. These tips and practices represent the best we've gleaned from a combined seven decades of programming and teaching experience.

① 在线章节及附录可登录华信教育资源网 (www.hxedu.com.cn) 免费注册下载。

Chapter Dependency Chart

[Note: Arrows pointing into a chapter indicate that chapter's dependencies.]



Good Programming Practices

The Good Programming Practices call attention to techniques that will help you produce programs that are clearer, more understandable and more maintainable.



Common Programming Errors

Pointing out these Common Programming Errors reduces the likelihood that you'll make them.



Error-Prevention Tips

These tips contain suggestions for exposing bugs and removing them from your programs; many describe aspects of C++ that prevent bugs from getting into programs in the first place.



Performance Tips

These tips highlight opportunities for making your programs run faster or minimizing the amount of memory that they occupy.



Portability Tips

The Portability Tips help you write code that will run on a variety of platforms.



Software Engineering Observations

The Software Engineering Observations highlight architectural and design issues that affect the construction of software systems, especially large-scale systems.

Summary bullets. We present a section-by-section bullet-list summary of the chapter with the page references to the defining occurrence for many of the key terms in each section.

Self-review exercises and answers. Extensive self-review exercises *and* answers are included for self study. All of the exercises in the optional ATM case study are fully solved.

Exercises. Each chapter concludes with a substantial set of exercises including:

- simple recall of important terminology and concepts
- What's wrong with this code?
- What does this code do?
- writing individual statements and small portions of functions and classes
- writing complete functions, classes and programs
- major projects.

Please do not write to us requesting access to the Pearson Instructor's Resource Center which contains the book's instructor supplements, including the exercise solutions. Actain access only through their Pearson representatives. Solutions are not provided for "project" exercises. Check out our Programming Projects Resource Center for lots of additional exercise and project possibilities (www.deitel.com/ProgrammingProjects/).

Index. We've included an extensive index.

Software Used in C++ *How to Program*, 8/e

We wrote C++ *How to Program*, 8/e using Microsoft's free Visual C++ Express Edition (which is available free for download at www.microsoft.com/express/downloads/) and the free GNU C++ (gcc.gnu.org/install/binaries.html), which is already installed on most Linux systems and can be installed on Mac OS X and Windows systems. Apple includes GNU C++ in their Xcode development tools, which Mac OS X users can download from developer.apple.com/technologies/tools/xcode.html.

C++ IDE Resource Kit

Your instructor may have ordered through your college bookstore a Value Pack edition of C++ *How to Program*, 8/e that comes bundled with the C++ IDE Resource Kit. This kit contains CD or DVD versions of:

- Microsoft® Visual Studio 2010 Express Edition (www.microsoft.com/express/)
- Dev C++ (www.bloodshed.net/download.html)
- NetBeans (netbeans.org/downloads/index.html)