Welcome to the C programming language. This book presents leading-edge computing technologies for software development professionals.

At the heart of the book is the Deitel signature “live-code approach.” We present concepts in the context of complete working programs, rather than in code snippets. Each code example is followed by one or more sample executions. Read the online Before You Begin section (www.deitel.com/books/cfp/cfp_BYB.pdf) to learn how to set up your computer to run the 130 code examples and your own C programs. All the source code is available at www.deitel.com/books/cfp and www.pearsonhighered.com/deitel. Use the source code we provide to run every program as you study it.

This book will give you an informative, challenging and entertaining introduction to C. If you have questions, send an e-mail to deitel@deitel.com—we’ll respond promptly. For book updates, visit www.deitel.com/books/cfp, join our communities on Facebook (www.deitel.com/deitelfan), Twitter (@deitel), Google+ (gplus.to/deitel) and LinkedIn (bit.ly/deitelLinkedIn), and subscribe to the Deitel® Buzz Online newsletter (www.deitel.com/newsletter/subscribe.html).

Features

Here are some key features of *C for Programmers with an Introduction to C11*:

- **Coverage of the New C standard.** The book is written to the new C standard, often referred to as C11 or simply “the C standard” since its approval in 2011. Support for the new standard varies by compiler. Most of our readers use either the GNU gcc compiler—which supports many of the key features in the new standard—or the Microsoft Visual C++ compiler. Microsoft supports only a limited subset of the features that were added to C in C99 and C11—primarily the features that are also required by the C++ standard. To accommodate all of our readers, we placed the discussion of the new standard’s features in optional, easy-to-use-or-omit sections and in Appendix E, Additional Features of the C Standard. We’ve also replaced various deprecated capabilities with newer preferred versions as a result of the new C standard.

- **Chapter 1.** We’ve included test-drives that show how to run a command-line C program on Microsoft Windows, Linux and Mac OS X.

- **Secure C Programming Sections.** We’ve added notes about secure C programming to many of the C programming chapters. We’ve also posted a Secure C Programming Resource Center at www.deitel.com/SecureC/. For more details, see the section *A Note About Secure C Programming* in this Preface.
Focus on Performance Issues. C is often favored by designers of performance-intensive applications such as operating systems, real-time systems, embedded systems and communications systems, so we focus intensively on performance issues.

All Code Tested on Windows and Linux. We’ve tested every example program using Visual C++® and GNU gcc in Windows and Linux, respectively.

Sorting: A Deeper Look. Sorting is an interesting problem because different sorting techniques achieve the same final result but they can vary hugely in their consumption of memory, CPU time and other system resources—algorithm performance is crucial. We begin our presentation of sorting in Chapter 6 and, in Appendix D, we present a deeper look. We consider several algorithms and compare them with regard to their memory consumption and processor demands. For this purpose, we introduce Big O notation, which indicates how hard an algorithm may have to work to solve a problem. Appendix D discusses the selection sort, insertion sort and recursive merge sort.

Debugger Appendices. We include Visual Studio® and GNU gdb debugging appendices.

Order of Evaluation. We discuss subtle order of evaluation issues to help you avoid errors.

C++-Style // Comments. We use the newer, more concise C++-style // comments in preference to C’s older style /* ... */ comments.

C Standard Library. Section 1.3 references en.cppreference.com/w/c where you can find thorough searchable documentation for the C Standard Library functions.

A Note About Secure C Programming

Experience has shown that it’s difficult to build industrial-strength systems that stand up to attacks from viruses, worms, etc. Today, via the Internet, such attacks can be instantaneous and global in scope. Software vulnerabilities often come from easy-to-avoid programming issues. Building security into software from the start of the development cycle can greatly reduce costs and vulnerabilities.

The CERT® Coordination Center (www.cert.org) was created to analyze and respond promptly to attacks. CERT—the Computer Emergency Response Team—publishes and promotes secure coding standards to help C programmers and others implement industrial-strength systems that avoid the programming practices that open systems to attack. The CERT standards evolve as new security issues arise.

Our code conforms to various CERT recommendations as appropriate for a book at this level. If you’ll be building C systems in industry, consider reading two books by Robert Seacord—The CERT C Secure Coding Standard (Addison-Wesley Professional, 2009) and Secure Coding in C and C++ (Addison-Wesley Professional, 2013). The CERT guidelines are available free online at www.securecoding.cert.org. Seacord, a technical reviewer for this book, also provided specific recommendations on each of our new Secure C Programming sections. Mr. Seacord is the Secure Coding Manager at CERT at Carnegie Mellon University’s Software Engineering Institute (SEI) and an adjunct professor in the Carnegie Mellon University School of Computer Science.
The Secure C Programming sections at the ends of Chapters 2–13 discuss many important topics, including testing for arithmetic overflows, using unsigned integer types, new more secure functions in the C standard’s Annex K, the importance of checking the status information returned by standard-library functions, range checking, secure random-number generation, array bounds checking, techniques for preventing buffer overflows, input validation, avoiding undefined behaviors, choosing functions that return status information vs. similar functions that do not, ensuring that pointers are always NULL or contain valid addresses, preferring C functions to preprocessor macros, and more.

Teaching Approach

C for Programmers with an Introduction to C11 contains a rich collection of examples. We focus on good software engineering, stressing program clarity.

Syntax Shading. For readability, we syntax shade the code, similar to the way most IDEs and code editors syntax color code. Our syntax-shading conventions are:

- Comments appear like this
- Keywords appear like this
- Constants and literal values appear like this
- All other code appears in black

Code Highlighting. We place gray rectangles around the key code segments in each source-code program.

Using Fonts for Emphasis. We place the key terms and the index’s page reference for each defining occurrence in bold text for easy reference. 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, int x = 5;).

Objectives. Each chapter includes a list of chapter objectives.

Illustrations/Figures. Abundant charts, tables, line drawings, flowcharts, 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 eight decades of programming and corporate training experience.

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 and removing bugs from your programs; many describe aspects of C that prevent bugs from getting into programs in the first place.
Preface

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.

Index. We’ve included an extensive index, which is especially useful when you use the book as a reference. Defining occurrences of key terms are highlighted with a bold page number.

Software Used in C for Programmers with an Introduction to C11
We wrote this book using the free GNU C compiler (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 and Microsoft’s free Visual Studio Express 2012 for Windows Desktop (www.microsoft.com/express). The Visual C++ compiler in Visual Studio can compile both C and C++ programs. Apple includes the LLVM compiler in its Xcode development tools, which Mac OS X users can download for free from the Mac App Store. Many other free C compilers are available online.

C Fundamentals: Parts I and II LiveLessons Video Training Product
Our C Fundamentals: Parts I and II LiveLessons video training product (available Fall 2013) shows you what you need to know to start building robust, powerful software with C. It includes 10+ hours of expert training synchronized with C for Programmers with an Introduction to C11. For additional information about Deitel LiveLessons video products, visit www.deitel.com/livelessons or contact us at deitel@deitel.com. You can also access our LiveLessons videos if you have a subscription to Safari Books Online (www.safaribooksonline.com).

Acknowledgments
We’d like to thank Abbey Deitel and Barbara Deitel for long hours devoted to this project. We’re fortunate to have worked with the dedicated team of publishing professionals at Prentice Hall/Pearson. We appreciate the extraordinary efforts and mentorship of our friend and professional colleague of 17 years, Mark L. Taub, Editor-in-Chief of Pearson Technology Group. Carole Snyder did a marvelous job managing the review process. Chuti Prasertsith designed the cover with creativity and precision. John Fuller does a superb job managing the production of all our Deitel Developer Series books.
Reviewers
We wish to acknowledge the efforts of our reviewers, who under tight deadlines scrutinized the text and the programs and provided countless suggestions for improving the presentation: Dr. John F. Doyle (Indiana University Southeast), Hemanth H.M. (Software Engineer at SonicWALL), Vytautus Leonavičius (Microsoft), Robert Seacord (Secure Coding Manager at SEI/CERT, author of The CERT C Secure Coding Standard and technical expert for the international standardization working group for the programming language C) and José Antonio González Seco (Parliament of Andalusia).

Well, there you have it! C11 is a powerful programming language that will help you write high-performance programs quickly and effectively. C11 scales nicely into the realm of enterprise systems development to help organizations build their business-critical and mission-critical information systems. As you read the book, we would sincerely appreciate your comments, criticisms, corrections and suggestions for improving the text. Please address all correspondence to:

deitel@deitel.com

We’ll respond promptly and post corrections and clarifications on:

www.deitel.com/books/cfp

We hope you enjoy working with C for Programmers with an Introduction to C11 as much as we enjoyed writing it!

Paul and Harvey Deitel

About the Authors
Paul Deitel, CEO and Chief Technical Officer of Deitel & Associates, Inc., is a graduate of MIT, where he studied Information Technology. Through Deitel & Associates, Inc., he has delivered hundreds of programming courses to industry, government and military clients, including Cisco, IBM, Siemens, Sun Microsystems, Dell, Fidelity, NASA at the Kennedy Space Center, the National Severe Storm Laboratory, White Sands Missile Range, Rogue Wave Software, Boeing, SunGard Higher Education, Nortel Networks, Puma, iRobot, Invensys and many more. He and his co-author, Dr. Harvey M. Deitel, are the world’s best-selling programming-language textbook/professional book/video authors.

Dr. Harvey Deitel, Chairman and Chief Strategy Officer of Deitel & Associates, Inc., has more than 50 years of experience in the computer field. Dr. Deitel earned B.S. and M.S. degrees in Electrical Engineering (studying computing) from MIT and a Ph.D. in Mathematics (studying computer science) from Boston University. He has extensive industry and college teaching experience, including earning tenure and serving as the Chairman of the Computer Science Department at Boston College before founding Deitel & Associates, Inc., in 1991 with his son, Paul Deitel. Dr. Deitel has delivered hundreds of professional programming seminars to major corporations, academic institutions, government organizations and the military. The Deitels’ publications have earned international recognition, with translations published in traditional Chinese, simplified Chinese, Korean, Japanese, German, Russian, Spanish, French, Polish, Italian, Portuguese, Greek, Urdu and Turkish.
Corporate Training from Deitel & Associates, Inc.

Deitel & Associates, Inc., founded by Paul Deitel and Harvey Deitel, is an internationally recognized authoring, corporate training and software development organization specializing in computer programming languages, object technology, Android and iOS app development and Internet and web software technology. The company offers instructor-led training courses delivered at client sites worldwide on major programming languages and platforms, including C, C++, Visual C++®, Java™, Visual C#®, Visual Basic®, XML®, Python®, object technology, Internet and web programming, Android™ app development, Objective-C and iOS® app development and a growing list of additional programming and software development courses. The company’s clients include some of the world’s largest companies as well as government agencies, branches of the military, and academic institutions.

Through its 37-year publishing partnership with Prentice Hall/Pearson, Deitel & Associates, Inc., publishes leading-edge programming professional books, college textbooks and LiveLessons video courses. Deitel & Associates, Inc. and the authors can be reached at:

**deitel@deitel.com**

To learn more about Deitel’s Dive-Into® Series Corporate Training curriculum, visit:

**www.deitel.com/training**

To request a proposal for worldwide on-site, instructor-led training at your organization, send an e-mail to deitel@deitel.com.

This book is also available as an e-book to Safari Books Online subscribers at

**www.safaribooksonline.com**

The last printed page of the book tells you how to get a free 45-day trial subscription to access the e-book.

Individuals wishing to purchase Deitel books and LiveLessons video training can do so through **www.deitel.com**. Bulk orders by corporations, the government, the military and academic institutions should be placed directly with Pearson. For more information, visit

**www.informit.com/store/sales.aspx**