

C Traps And Pitfalls

One of the most complete and comprehensive books written on "C" language, this text requires no prior knowledge of programming, no mathematical skills, and no prior training in problem-solving. A Step-by-Step Guide to C Programming has a workbook-like approach so readers can pace themselves. The following features make this book unique: the approach is informal, and each concept is explained by example; explanations are organized like lecture notes; most examples are code complete; readers learn how to put segments of codes together, not just how to use functions and syntax; advice on good programming practices is included; one- and two-dimensional arrays are presented; and the importance of how to write portable functions is stressed.

A fascinating journey through history and culture, examining how makeup affects self-empowerment, how people have used it to define (and defy) their roles in society, and why we all need to care. There is a history and a cultural significance that comes with wearing cat-eye-inspired liner or a bold red lip, one that many women feel to this day, even if we don't realize exactly why. Increasingly, people of all genders are wrestling with what it means to be a woman living in a patriarchy, and part of that is how looking like a woman—whatever that means—affects people's real lives. Through the stories of famous women like Cleopatra, Empress Wu, Madam C. J. Walker, Elizabeth Taylor, and Marsha P. Johnson, Rae Nudson unpacks makeup's cultural impact—including how it can be used to shape a personal or cultural narrative, how often beauty standards align with whiteness, how and when it can be used for safety, and its function in the workplace, to name a few examples. Every woman has had to make a very personal choice about her relationship with makeup, and consciously or unconsciously, every woman knows that the choice is never entirely hers to make. This book also holds space for complicating factors, especially the ways that beauty standards differ across race, class, and culture. Engaging and informative, All Made Up will expand the discussion around what it means to participate in creating your own self-image.

As networks, devices, and systems continue to evolve, software engineers face the unique challenge of creating reliable distributed applications within frequently changing environments. C++ Network Programming, Volume 1, provides practical solutions for developing and optimizing complex distributed systems using the ADAPTIVE Communication Environment (ACE), a revolutionary open-source framework that runs on dozens of hardware platforms and operating systems. This book guides software professionals through the traps and pitfalls of developing efficient, portable, and flexible networked applications. It explores the inherent design complexities of concurrent networked applications and the tradeoffs that must be considered when working to master them. C++ Network Programming begins with an overview of the issues and tools involved in writing distributed concurrent applications. The book then provides the essential design dimensions, patterns, and principles needed to develop flexible and efficient concurrent networked applications. The book's expert author team shows you how to enhance design skills while applying C++ and patterns effectively to develop object-oriented networked applications. Readers will find coverage of: C++ network programming, including an overview and strategies for addressing common development challenges The ACE Toolkit Connection protocols, message exchange, and message-passing versus shared memory Implementation methods for reusable networked application services Concurrency in object-oriented network programming Design principles and patterns for ACE wrapper facades With this book, C++ developers have at their disposal the most complete toolkit available for developing successful, multiplatform, concurrent networked applications with ease and efficiency.

"Every programming language has its quirks. This lively book reveals oddities of the Java programming language through entertaining and thought-provoking programming puzzles." --Guy Steele, Sun Fellow and coauthor of The Java™ Language Specification "I laughed, I cried, I threw up (my hands in admiration)." --Tim Peierls, president, Prior Artisans LLC, and member of the JSR 166 Expert Group How well do you really know Java? Are you a code sleuth? Have you ever spent days chasing a bug caused by a trap or pitfall in Java or its libraries? Do you like brainteasers? Then this is the book for you! In the tradition of Effective Java™, Bloch and Gafter dive deep into the subtleties of the Java programming language and its core libraries. Illustrated with visually stunning optical illusions, Java™ Puzzlers features 95 diabolical puzzles that educate and entertain. Anyone with a working knowledge of Java will understand the puzzles, but even the most seasoned veteran will find them challenging. Most of the puzzles take the form of a short program whose behavior isn't what it seems. Can you figure out what it does? Puzzles are grouped loosely according to the features they use, and detailed solutions follow each puzzle. The solutions go well beyond a simple explanation of the program's behavior--they show you how to avoid the underlying traps and pitfalls for good. A handy catalog of traps and pitfalls at the back of the book provides a concise taxonomy for future reference. Solve these puzzles and you'll never again fall prey to the counterintuitive or obscure behaviors that can fool even the most experienced programmers.

C Traps and Pitfalls Pearson Education India

"As a rule, I have found that the greater brain a man has, and the better he is educated, the easier it has been to mystify him" (Harry Houdini to Arthur Conan Doyle). Smart people are not only just as prone to making mistakes as everyone else-- they may be even more susceptible to them. This is the "intelligence trap," the subject of David Robson's fascinating and provocative book. The Intelligence Trap explores cutting-edge ideas in our understanding of intelligence and expertise, including "strategic ignorance," "meta-forgetfulness," and "functional stupidity." Robson reveals the surprising ways that even the brightest minds and most talented organizations can go wrong--from some of Thomas Edison's worst ideas to failures at NASA, Nokia, and the FBI. And he offers practical advice to avoid mistakes based on the timeless lessons of Benjamin Franklin, Richard

Feynman, and Daniel Kahneman.

More Exceptional C++ continues where Herb Sutter's best-selling Exceptional C++ left off, delivering 40 puzzles that illuminate the most challenging -- and most powerful -- aspects of C++. More Exceptional C++ offers many new puzzles focused on generic programming and the C++ Standard Template Library, including important techniques such as traits and predicates, as well as key considerations in using standard containers and algorithms -- many of them never covered elsewhere. More Exceptional C++ contains a detailed new section (and two appendices) on optimization in single- and multithreaded environments. It also provides important new insights on crucial topics first introduced in Exceptional C++, including exception safety, generic programming, and memory management. For all C++ programmers.

Literate programming is a programming methodology that combines a programming language with a documentation language, making programs more easily maintained than programs written only in a high-level language. A literate programmer is an essayist who writes programs for humans to understand. When programs are written in the recommended style they can be transformed into documents by a document compiler and into efficient code by an algebraic compiler. This anthology of essays includes Knuth's early papers on related topics such as structured programming as well as the Computer Journal article that launched literate programming. Many examples are given, including excerpts from the programs for TeX and METAFONT. The final essay is an example of CWEB, a system for literate programming in C and related languages. Index included.

This book presents an introduction to the C programming language, featuring a structured approach and aimed at professionals and students with some experience of high-level languages. Features

*includes embedded summary material in bulleted form *highlights common traps and pitfalls in C programming.

With the award-winning book Agile Software Development: Principles, Patterns, and Practices, Robert C. Martin helped bring Agile principles to tens of thousands of Java and C++ programmers. Now .NET programmers have a definitive guide to agile methods with this completely updated volume from Robert C. Martin and Micah Martin, Agile Principles, Patterns, and Practices in C#. This book presents a series of case studies illustrating the fundamentals of Agile development and Agile design, and moves quickly from UML models to real C# code. The introductory chapters lay out the basics of the agile movement, while the later chapters show proven techniques in action. The book includes many source code examples that are also available for download from the authors' Web site. Readers will come away from this book understanding Agile principles, and the fourteen practices of Extreme Programming Spiking, splitting, velocity, and planning iterations and releases Test-driven development, test-first design, and acceptance testing Refactoring with unit testing Pair programming Agile design and design smells The five types of UML diagrams and how to use them effectively Object-oriented package design and design patterns How to put all of it together for a real-world project Whether you are a C# programmer or a Visual Basic or Java programmer learning C#, a software development manager, or a business analyst, Agile Principles, Patterns, and Practices in C# is the first book you should read to understand agile software and how it applies to programming in the .NET Framework.

Insect Sampling in Forest Ecosystems highlights the problems faced by entomologists working in forest ecosystems. Insects play a major part in all aspects of ecology Brings together the methodology needed to investigate insects through the various strata of the forest canopy Covers techniques associated with various specialised groups of forest insects Each chapter is backed up by a sound approach to experimental design and data analysis Essential reading for advanced students and researchers as well as teachers

Avoid data blunders and create truly useful visualizations Avoiding Data Pitfalls is a reputation-saving handbook for those who work with data, designed to help you avoid the all-too-common blunders that occur in data analysis, visualization, and presentation. Plenty of data tools exist, along with plenty of books that tell you how to use them—but unless you truly understand how to work with data, each of these tools can ultimately mislead and cause costly mistakes. This book walks you step by step through the full data visualization process, from calculation and analysis through accurate, useful presentation.

Common blunders are explored in depth to show you how they arise, how they have become so common, and how you can avoid them from the outset. Then and only then can you take advantage of the wealth of tools that are out there—in the hands of someone who knows what they're doing, the right tools can cut down on the time, labor, and myriad decisions that go into each and every data presentation.

Workers in almost every industry are now commonly expected to effectively analyze and present data, even with little or no formal training. There are many pitfalls—some might say chasms—in the process, and no one wants to be the source of a data error that costs money or even lives. This book provides a full walk-through of the process to help you ensure a truly useful result. Delve into the "data-reality gap" that grows with our dependence on data Learn how the right tools can streamline the visualization process Avoid common mistakes in data analysis, visualization, and presentation Create and present clear, accurate, effective data visualizations To err is human, but in today's data-driven world, the stakes can be high and the mistakes costly. Don't rely on "catching" mistakes, avoid them from the outset with the expert instruction in Avoiding Data Pitfalls.

A detailed introduction to the C programming language for experienced programmers. The world runs on code written in the C programming language, yet most schools begin the curriculum with Python or Java. Effective C bridges this gap and brings C into the modern era--covering the modern C17 Standard as well as potential C2x features. With the aid of this instant classic, you'll soon be writing professional, portable, and secure C programs to power robust systems and solve real-world problems. Robert C. Seacord introduces C and the C Standard Library while addressing best practices, common errors, and open debates in the C community. Developed together with other C Standards committee experts, Effective C will teach you how to debug, test, and analyze C programs. You'll benefit from Seacord's concise explanations of C language constructs and behaviors, and from his 40 years of coding experience. You'll learn:

- How to identify and handle undefined behavior in a C program
- The range and representations of integers and floating-point values
- How dynamic memory allocation works and how to use nonstandard functions
- How to use character encodings and types
- How to perform I/O with terminals and filesystems using C Standard streams and POSIX file descriptors
- How to understand the C compiler's translation phases and the role of the preprocessor
- How to test, debug, and analyze C programs

Effective C will teach you how to write professional, secure, and portable C code that will stand the test of time and help strengthen the foundation of the computing world.

This guide provides the reader with clear, ready-to-use techniques that will make them a better trial lawyer and sharpen their understanding of the basics. serves as a resource for questioning and selecting a jury.

Bjarne Stroustrup's own C++ In-Depth Series is now available all together in one attractive gift box, at a special reduced price! All books in this series have been hand-picked by Bjarne Stroustrup, the creator of the C++ programming language, as being worthy additions to the C++ literature. They give programmers concise, focused guides to specific topics. The series' practical approach is designed to lift professionals to the next level in their programming skills. They are all written by acknowledged experts. The books included are: Modern C++ Design, by Andrei Alexandrescu Accelerated C++, by Andrew Koenig and Barbara Moo Essential C++, by Stan Lippman Exceptional C++, by Herb Sutter More Exceptional C++, by Herb Sutter These are five great books of use to all C++ programmers. They are

gathered into one handsome and sturdy gift box, and they are specially priced at over \$30 off the cost of buying them individually. The C++ In-Depth Box Set will be a welcome gift for any C++ programmer. 0201775816B12112002

Written by the originator of the USENET C FAQ, this book addresses the real-world problems on C programming that are asked, again and again, on the "comp.lang.c" newsgroup. The book is aimed at C programmers who need quick, concise answers to the stubborn questions which invariably arise when programming in C. It provides accurate answers, insightful explanations, and extensive code examples. The authors begin by explaining why C++ is worth learning and then move on to the most important elements of C++. This book emphasizes understanding and practical use of the language. It explores the basics, covers inheritance and object-oriented programming, discusses templates and the powerful kind of abstraction they provide, and shows how to design and use libraries.

This guide fills in the knowledge gaps for intermediate-advanced programmers who have been trained without regard to software performance from a program level. Information on advanced tips and cautions can prepare programmers for cross-platform or multilanguage projects.

Consistent, high-quality coding standards improve software quality, reduce time-to-market, promote teamwork, eliminate time wasted on inconsequential matters, and simplify maintenance. Now, two of the world's most respected C++ experts distill the rich collective experience of the global C++ community into a set of coding standards that every developer and development team can understand and use as a basis for their own coding standards. The authors cover virtually every facet of C++ programming: design and coding style, functions, operators, class design, inheritance, construction/destruction, copying, assignment, namespaces, modules, templates, genericity, exceptions, STL containers and algorithms, and more. Each standard is described concisely, with practical examples. From type definition to error handling, this book presents C++ best practices, including some that have only recently been identified and standardized--techniques you may not know even if you've used C++ for years. Along the way, you'll find answers to questions like What's worth standardizing--and what isn't? What are the best ways to code for scalability? What are the elements of a rational error handling policy? How (and why) do you avoid unnecessary initialization, cyclic, and definitional dependencies? When (and how) should you use static and dynamic polymorphism together? How do you practice "safe" overriding? When should you provide a no-fail swap? Why and how should you prevent exceptions from propagating across module boundaries? Why shouldn't you write namespace declarations or directives in a header file? Why should you use STL vector and string instead of arrays? How do you choose the right STL search or sort algorithm? What rules should you follow to ensure type-safe code? Whether you're working alone or with others, C++ Coding Standards will help you write cleaner code--and write it faster, with fewer hassles and less frustration.

The author uses practical, concise code examples to illuminate a useful programming stratagem or warn against a dangerous practice. Readers will come away with a better understanding of how C++ is used in the real world.

Designed for a compulsory fundamental course, C: From Theory to Practice uses a hands-on approach to teach the C programming language, using numerous examples and a clear, concise presentation. Easy to use and classroom tested, this textbook includes more than 500 exercises and examples of progressive difficulty to help students in understanding all the aspects and peculiarities of C. The exercises test students on various levels of programming and the examples enhance their concrete understanding of programming know-how. Divided into three parts, this book: Introduces the basic concepts of C, like getting input from a user, C's operators, selection statements, and loops. Emphasizes major features of C such as arrays, pointers, functions and strings. Covers advanced topics such as like searching and sorting arrays' algorithms, structures and unions, memory management, the preprocessor and files. The book tests the skills of beginners and advanced developers by providing an easy-to-read compilation of the C theory enriched with tips and advice as well as difficulty-scaled solved programming exercises. It decodes the secrets of the C language, providing inside information and programming knowledge through practical examples and meaningful advice. The examples are designed to be short, concrete, and substantial, quickly giving students the know-how they need.

Perl is a powerful programming language that has grown in popularity since it first appeared in 1988. The first edition of this book, Programming Perl, hit the shelves in 1990, and was quickly adopted as the undisputed bible of the language. Since then, Perl has grown with the times, and so has this book. Programming Perl is not just a book about Perl. It is also a unique introduction to the language and its culture, as one might expect only from its authors. Larry Wall is the inventor of Perl, and provides a unique perspective on the evolution of Perl and its future direction. Tom Christiansen was one of the first champions of the language, and lives and breathes the complexities of Perl internals as few other mortals do. Jon Orwant is the editor of The Perl Journal, which has brought together the Perl community as a common forum for new developments in Perl. Any Perl book can show the syntax of Perl's functions, but only this one is a comprehensive guide to all the nooks and crannies of the language. Any Perl book can explain typeglobs, pseudohashes, and closures, but only this one shows how they really work. Any Perl book can say that my is faster than local, but only this one explains why. Any Perl book can have a title, but only this book is affectionately known by all Perl programmers as "The Camel." This third edition of Programming Perl has been expanded to cover version 5.6 of this maturing language. New topics include threading, the compiler, Unicode, and other new features that have been added since the previous edition.

C Unleashed is a very comprehensive book on the ANSI C programming language. The book promotes solid, portable programming using ANSI C, thus benefitting programmers on any platform, including mainframes. Covers the New Standard for C, known as C9X, and includes: Embedded systems, Simulation Processing, Threading and

Multiprocessing, Digital Signal Processing, and Natural Language Processing.

C++ is a powerful, highly flexible, and adaptable programming language that allows software engineers to organize and process information quickly and effectively. But this high-level language is relatively difficult to master, even if you already know the C programming language. The new second edition of "Practical C++ Programming is a complete introduction to the C++ language for programmers who are learning C++. Reflecting the latest changes to the C++ standard, this new edition takes a useful down-to-earth approach, placing a strong emphasis on how to design clean, elegant code. In short, to-the-point chapters, all aspects of programming are covered including style, software engineering, programming design, object-oriented design, and debugging. It also covers common mistakes and how to find (and avoid) them. End of chapter exercises help you ensure you've mastered the material. Steve Oualline's clear, easy-going writing style and hands-on approach to learning make "Practical C++ Programming a nearly painless way to master this complex but powerful programming language.

C++ was written to help professional C# developers learn modern C++ programming. The aim of this book is to leverage your existing C# knowledge in order to expand your skills. Whether you need to use C++ in an upcoming project, or simply want to learn a new language (or reacquaint yourself with it), this book will help you learn all of the fundamental pieces of C++ so you can begin writing your own C++ programs. This updated and expanded second edition of Book provides a user-friendly introduction to the subject, Taking a clear structural framework, it guides the reader through the subject's core elements. A flowing writing style combines with the use of illustrations and diagrams throughout the text to ensure the reader understands even the most complex of concepts. This succinct and enlightening overview is a required reading for all those interested in the subject. We hope you find this book useful in shaping your future career & Business.

The puzzles and problems in Exceptional C++ not only entertain, they will help you hone your skills to become the sharpest C++ programmer you can be. Many of these problems are culled from the famous Guru of the Week feature of the Internet newsgroup comp.lang.c++, moderated, expanded and updated to conform to the official ISO/ANSI C++ Standard. Try your skills against the C++ masters and come away with the insight and experience to create more efficient, effective, robust, and portable C++ code.

A lifesaver for any Java programmer-proven workarounds and time-saving solutions Although using the Java language provides a substantial boost to a programmer's productivity, it still has its share of subtleties and weaknesses. This book is designed to save you time and frustration by carefully guiding you through this potential minefield. A team of Java experts, led by programming guru Michael Daconta, offers a collection of proven solutions to 50 difficult, real-world problems chosen from their own extensive experiences. You'll find workarounds for problems caused by shortcomings in both the Java language itself and in its APIs and utilities, including java.util, java.io, java.awt, and javax.swing. The authors also share techniques for improving the performance of your Java applications. For easy reference, the book is organized into categories so that similar solutions are grouped together. Examples of topics covered include: * Language syntax, for example, using the String equals() method instead of the == operator (Item 2) * Language support, for example, method dispatching with reflection, interfaces, and anonymous classes (Item 16) * Utilities and collections, like choosing between a PropertyFile and ResourceBundle (Item 20) * Input/output, including subtleties in sending serialized objects over a network (Item 25) * GUI presentation, for example, tackling the common pitfall of using repaint() instead of validate() for relaying out components (Item 29) * Performance, including tips like lazy loading your way to better performance (Item 43)

Far too many programmers and software designers consider efficient C++ to be an oxymoron. They regard C++ as inherently slow and inappropriate for performance-critical applications. Consequently, C++ has had little success penetrating domains such as networking, operating system kernels, device drivers, and others. Efficient C++ explodes that myth. Written by two authors with first-hand experience wringing the last ounce of performance from commercial C++ applications, this book demonstrates the potential of C++ to produce highly efficient programs. The book reveals practical, everyday object-oriented design principles and C++ coding techniques that can yield large performance improvements. It points out common pitfalls in both design and code that generate hidden operating costs. This book focuses on combining C++'s power and flexibility with high performance and scalability, resulting in the best of both worlds. Specific topics include temporary objects, memory management, templates, inheritance, virtual functions, inlining, reference-counting, STL, and much more. With this book, you will have a valuable compendium of the best performance techniques at your fingertips. 0201379503B04062001

Software -- Programming Languages.

Building on the success of Java Pitfalls (0-471-36174-7), this book provides more specific programming solutions to fifty difficult Java programming problems Shows experienced programmers how to identify and avoid weaknesses in Java and related J2EE technologies that can cause programs to go haywire Explores advanced topics including networking, XML and Java programming, and the Java Virtual Machine

This book helps to prevent such problems by showing how C programmers get themselves into trouble. Each of the book's many examples has trapped a professional programmer. Distilled from the author's experience over a decade of programming in C, this book is an ideal resource for anyone, novice or expert, who has ever written a C program.

Even C experts encounter problems that require days of debugging. This book shows how to prevent such problems. Also includes advice for mastering often-misunderstood parts of C.

Annotation copyrighted by Book News, Inc., Portland, OR

Shows how to avoid and capitalize on the playing errors and psychological problems experienced by most chess enthusiasts

"Emily Kane shows clearly that most parents understand children's personality to be some combination of nature and nurture, and many wish they could help nurture their children to escape gender traps. Yet these parents are themselves trapped by the gender structure itself, especially the accountability they feel to other people's expectations, and the fear that if their boys are free to explore activities usually associated with girls they will be punished by the world around them. The author shows clearly that to help parents navigate childrearing, we have to change

the world around them. A good read, perfect for the undergraduate classroom, and clear enough even to give to those new parents in your family or the neighborhood."--Cover.

Want to write iOS apps or desktop Mac applications? This introduction to programming and the Objective-C language is your first step on the journey from someone who uses apps to someone who writes them. Based on Big Nerd Ranch's popular Objective-C Bootcamp, Objective-C Programming: The Big Nerd Ranch Guide covers C, Objective-C, and the common programming idioms that enable developers to make the most of Apple technologies. Compatible with Xcode 5, iOS 7, and OS X Mavericks (10.9), this guide features short chapters and an engaging style to keep you motivated and moving forward. At the same time, it encourages you to think critically as a programmer. Here are some of the topics covered: Using Xcode, Apple's documentation, and other tools Programming basics: variables, loops, functions, etc. Objects, classes, methods, and messages Pointers, addresses, and memory management with ARC Properties and Key-Value Coding (KVC) Class extensions Categories Classes from the Foundation framework Blocks Delegation, target-action, and notification design patterns Key-Value Observing (KVO) Runtime basics

Solving complex problems and selling their solutions is critical for personal and organizational success. For most of us, however, it doesn't come naturally and we haven't been taught how to do it well. Research shows a host of pitfalls trips us up when we try: We're quick to believe we understand a situation and jump to a flawed solution. We seek to confirm our hypotheses and ignore conflicting evidence. We view challenges incompletely through the frameworks we know instead of with a fresh pair of eyes. And when we communicate our recommendations, we forget our reasoning isn't obvious to our audience. How can we do it better? In *Cracked It!*, seasoned strategy professors and consultants Bernard Garrette, Corey Phelps and Olivier Sibony present a rigorous and practical four-step approach to overcome these pitfalls. Building on tried-and-tested (but rarely revealed) methods of top strategy consultants, research in cognitive psychology, and the latest advances in design thinking, they provide a step-by-step process and toolkit that will help readers tackle any challenging business problem. Using compelling stories and detailed case examples, the authors guide readers through each step in the process: from how to state, structure and then solve problems to how to sell the solutions. Written in an engaging style by a trio of experts with decades of experience researching, teaching and consulting on complex business problems, this book will be an indispensable manual for anyone interested in creating value by helping their organizations crack the problems that matter most.

[Copyright: ac7df4b4f40055d3e7c5a11bfe1e5fd1](#)