

Mastering C Pointers Tools For Programming Power Robert J Traister

Getting the books **Mastering C Pointers Tools For Programming Power Robert J Traister** now is not type of challenging means. You could not deserted going taking into account books growth or library or borrowing from your friends to approach them. This is an agreed easy means to specifically get guide by on-line. This online proclamation Mastering C Pointers Tools For Programming Power Robert J Traister can be one of the options to accompany you like having other time.

It will not waste your time. tolerate me, the e-book will certainly tone you further matter to read. Just invest tiny period to door this on-line statement **Mastering C Pointers Tools For Programming Power Robert J Traister** as capably as review them wherever you are now.

Dr. Dobb's Journal - 1993

The Pragmatic Programmer - Andrew Hunt 1999-10-20

What others in the trenches say about The Pragmatic Programmer...

"The cool thing about this book is that it's great for keeping the programming process fresh. The book helps you to continue to grow and clearly comes from people who have been there." —Kent Beck, author of Extreme Programming Explained: Embrace Change "I found this book to be a great mix of solid advice and wonderful analogies!" —Martin Fowler, author of Refactoring and UML Distilled "I would buy a copy, read it twice, then tell all my colleagues to run out and grab a copy. This is a book I would never loan because I would worry about it being lost." —Kevin Ruland, Management Science, MSG-Logistics "The wisdom and practical experience of the authors is obvious. The topics presented are relevant and useful.... By far its greatest strength for me has been the outstanding analogies—tracer bullets, broken windows, and the fabulous helicopter-based explanation of the need for orthogonality, especially in a crisis situation. I have little doubt that this book will eventually become an excellent source of useful information for journeymen programmers and expert mentors alike." —John Lakos, author of Large-Scale C++ Software Design "This is the sort of book I will buy a dozen copies of when it comes out so I can give it to my clients." —Eric Vought, Software Engineer "Most modern books on software development fail to cover the basics of what makes a great software developer, instead spending their time on syntax or technology where in reality the greatest leverage possible for any software team is in having talented developers who really know their craft well. An excellent book." —Pete McBreen, Independent Consultant "Since reading this book, I have implemented many of the practical suggestions and tips it contains. Across the board, they have saved my company time and money while helping me get my job done quicker! This should be a desktop reference for everyone who works with code for a living." —Jared Richardson, Senior Software Developer, iRenaissance, Inc. "I would like to see this issued to every new employee at my company...." —Chris Cleeland, Senior Software Engineer, Object Computing, Inc. "If I'm putting together a project, it's the authors of this book that I want. . . . And failing that I'd settle for people who've read their book." —Ward Cunningham Straight from the programming trenches, The Pragmatic Programmer cuts through the increasing specialization and technicalities of modern software development to examine the core process—taking a requirement and producing working, maintainable code that delights its users. It covers topics ranging from personal responsibility and career development to architectural techniques for keeping your code flexible and easy to adapt and reuse. Read this book, and you'll learn how to Fight software rot; Avoid the trap of duplicating knowledge; Write flexible, dynamic, and adaptable code; Avoid programming by coincidence; Bullet-proof your code with contracts, assertions, and exceptions; Capture real requirements; Test ruthlessly and effectively; Delight your users; Build teams of pragmatic programmers; and Make your developments more precise with automation. Written as a series of self-contained sections and filled with entertaining anecdotes, thoughtful examples, and interesting analogies, The Pragmatic Programmer illustrates the best practices and major pitfalls of many different aspects of software development. Whether you're a new coder, an experienced programmer, or a manager responsible for software projects, use these lessons daily, and you'll quickly see improvements in personal productivity, accuracy, and job satisfaction. You'll learn skills and develop habits and attitudes that form the foundation for long-term success in your career. You'll become a Pragmatic Programmer.

Expert C Programming - Peter Van der Linden 1994

Software -- Programming Languages.

Programming Embedded Systems - Michael Barr 2006-10-11

Authored by two of the leading authorities in the field, this guide offers readers the knowledge and skills needed to achieve proficiency with embedded software.

A Book on C - Al Kelley 1990

The authors provide clear examples and thorough explanations of every feature in the C language. They teach C vis-a-vis the UNIX operating system. A reference and tutorial to the C programming language. Annotation copyrighted by Book News, Inc., Portland, OR

A Complete Guide to Programming in C++ - Ulla Kirch-Prinz 2002

This guide was written for readers interested in learning the C++ programming language from scratch, and for both novice and advanced C++ programmers wishing to enhance their knowledge of C++. The text is organized to guide the reader from elementary language concepts to professional software development, with in depth coverage of all the C++ language elements en route.

C - Paul J.. Deitel 2007

For introductory courses in C Programming. Also for courses in Programming for Engineers, Programming for Business, and Programming for Technology. The Deitels' groundbreaking How to Program series offers unparalleled breadth and depth of object-oriented programming concepts and intermediate-level topics for further study. Using the Deitels' signature Live-Code Approach, this complete, authoritative introduction to C programming offers strong treatment of structured algorithm and program development in ANSI/ISO C with 150 working C programs. Includes rich, 300-page treatment of object-oriented programming in C++ that helps students interpret the code more effectively.

Practical C++ Programming - Steve Oualline 2003

Practical C++ Programming thoroughly covers: C++ syntax · Coding standards and style · Creation and use of object classes · Templates · Debugging and optimization · Use of the C++ preprocessor · File input/output.

Journal of Object-oriented Programming - 2000

UNDERSTANDING POINTERS IN C - 1997

Computer Language - 1991

Books in Print - 1991

Mastering Algorithms with C - Kyle Loudon 1999

A comprehensive guide to understanding the language of C offers solutions for everyday programming tasks and provides all the necessary information to understand and use common programming techniques. Original. (Intermediate).

The Art of R Programming - Norman Matloff 2011-10-11

R is the world's most popular language for developing statistical software: Archaeologists use it to track the spread of ancient civilizations, drug companies use it to discover which medications are safe and effective, and actuaries use it to assess financial risks and keep economies running smoothly. The Art of R Programming takes you on a guided tour of software development with R, from basic types and data structures to advanced topics like closures, recursion, and anonymous functions. No statistical knowledge is required, and your programming skills can range from hobbyist to pro. Along the way, you'll learn about functional and object-oriented programming, running mathematical simulations, and rearranging complex data into simpler, more useful formats. You'll also learn to: -Create artful graphs to visualize complex

data sets and functions -Write more efficient code using parallel R and vectorization -Interface R with C/C++ and Python for increased speed or functionality -Find new R packages for text analysis, image manipulation, and more -Squash annoying bugs with advanced debugging techniques Whether you're designing aircraft, forecasting the weather, or you just need to tame your data, *The Art of R Programming* is your guide to harnessing the power of statistical computing.

The C Programming Language - Brian W. Kernighan 1988

Introduces the features of the C programming language, discusses data types, variables, operators, control flow, functions, pointers, arrays, and structures, and looks at the UNIX system interface

The Book of R - Tilman M. Davies 2016-07-16

The Book of R is a comprehensive, beginner-friendly guide to R, the world's most popular programming language for statistical analysis. Even if you have no programming experience and little more than a grounding in the basics of mathematics, you'll find everything you need to begin using R effectively for statistical analysis. You'll start with the basics, like how to handle data and write simple programs, before moving on to more advanced topics, like producing statistical summaries of your data and performing statistical tests and modeling. You'll even learn how to create impressive data visualizations with R's basic graphics tools and contributed packages, like ggplot2 and ggvis, as well as interactive 3D visualizations using the rgl package. Dozens of hands-on exercises (with downloadable solutions) take you from theory to practice, as you learn: -The fundamentals of programming in R, including how to write data frames, create functions, and use variables, statements, and loops -Statistical concepts like exploratory data analysis, probabilities, hypothesis tests, and regression modeling, and how to execute them in R -How to access R's thousands of functions, libraries, and data sets -How to draw valid and useful conclusions from your data -How to create publication-quality graphics of your results Combining detailed explanations with real-world examples and exercises, this book will provide you with a solid understanding of both statistics and the depth of R's functionality. Make *The Book of R* your doorway into the growing world of data analysis.

The Standard C Library - P. J. Plauger 1992

First comprehensive treatment of ANSI and ISO standards for the C Library. Includes practical advice on using all 15 headers of the Library and covers the concept design and utilization of libraries. Contains complete codes of C Library and is the companion volume to *C Programming Language*. An independent consultant, author Plauger is one of the world's leading experts on C and the C Library.

C Programming FAQs - Steve Summit 1996

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.

Think Like a Programmer - V. Anton Spraul 2012-08-12

The real challenge of programming isn't learning a language's syntax—it's learning to creatively solve problems so you can build something great. In this one-of-a-kind text, author V. Anton Spraul breaks down the ways that programmers solve problems and teaches you what other introductory books often ignore: how to Think Like a Programmer. Each chapter tackles a single programming concept, like classes, pointers, and recursion, and open-ended exercises throughout challenge you to apply your knowledge. You'll also learn how to: -Split problems into discrete components to make them easier to solve -Make the most of code reuse with functions, classes, and libraries -Pick the perfect data structure for a particular job -Master more advanced programming tools like recursion and dynamic memory -Organize your thoughts and develop strategies to tackle particular types of problems Although the book's examples are written in C++, the creative problem-solving concepts they illustrate go beyond any particular language; in fact, they often reach outside the realm of computer science. As the most skillful programmers know, writing great code is a creative art—and the first step in creating your masterpiece is learning to Think Like a Programmer.

Mastering the C++17 STL - Arthur O'Dwyer 2017-09-28

This book breaks down the C++ STL, teaching you how to extract its gems and apply them to your programming. About This Book Boost your productivity as a C++ developer with the latest features of C++17 Develop high-quality, fast, and portable applications with the varied features of the STL Migrate from older versions (C++11, C++14) to

C++17 Who This Book Is For This book is for developers who would like to master the C++ STL and make full use of its components. Prior C++ knowledge is assumed. What You Will Learn Make your own iterator types, allocators, and thread pools. Master every standard container and every standard algorithm. Improve your code by replacing new/delete with smart pointers. Understand the difference between monomorphic algorithms, polymorphic algorithms, and generic algorithms. Learn the meaning and applications of vocabulary type, product type and sum type. In Detail Modern C++ has come a long way since 2011. The latest update, C++17, has just been ratified and several implementations are on the way. This book is your guide to the C++ standard library, including the very latest C++17 features. The book starts by exploring the C++ Standard Template Library in depth. You will learn the key differences between classical polymorphism and generic programming, the foundation of the STL. You will also learn how to use the various algorithms and containers in the STL to suit your programming needs. The next module delves into the tools of modern C++. Here you will learn about algebraic types such as `std::optional`, vocabulary types such as `std::function`, smart pointers, and synchronization primitives such as `std::atomic` and `std::mutex`. In the final module, you will learn about C++'s support for regular expressions and file I/O. By the end of the book you will be proficient in using the C++17 standard library to implement real programs, and you'll have gained a solid understanding of the library's own internals. Style and approach This book takes a concise but comprehensive approach to explaining and applying the C++ STL, one feature at a time.

New Technical Books - New York Public Library 1994

Hacking- The art Of Exploitation - J. Erickson 2018-03-06

This text introduces the spirit and theory of hacking as well as the science behind it all; it also provides some core techniques and tricks of hacking so you can think like a hacker, write your own hacks or thwart potential system attacks.

Intermediate C Programming - Yung-Hsiang Lu 2015-06-17

Teach Your Students How to Program Well Intermediate C Programming provides a stepping-stone for intermediate-level students to go from writing short programs to writing real programs well. It shows students how to identify and eliminate bugs, write clean code, share code with others, and use standard Linux-based tools, such as `ddd` and `valgrind`. The text covers numerous concepts and tools that will help your students write better programs. It enhances their programming skills by explaining programming concepts and comparing common mistakes with correct programs. It also discusses how to use debuggers and the strategies for debugging as well as studies the connection between programming and discrete mathematics.

Voltage Regulator Circuit Manual - Gerard Meurant 2012-12-02

Voltage Regulator Circuit Manual highlights the techniques in DC regulator design. This book contains seven chapters that cover different circuit types, from the simple incorporation of silicon chips to the complex IC manufacturing. After providing an overview of the changes in power supply design, this book goes on discussing the various circuit configurations applicable to linear IC voltage regulators and switching regulator designs. The following chapters contain schematic diagrams of a general assortment of regulators. In these chapters, the circuits are based on three-terminal, linear regulator ICs that offer simplicity of design, low cost, minimal circuit complexity, and relatively fast construction times. A chapter focuses on a wide assortment of regulators that fall into the general category of "switchers", which is a very broad class of circuit that encompasses several highly different configurations. The discussion then shifts to the switching power-supply circuits that fall into the category of flyback regulators, also known as ringing choke regulators. The last chapters deal with DC regulators that perform true value voltage conversions and their distinct characteristics. These chapters also include circuits that did not exactly fit the other circuit categories, such as battery chargers and motor controllers. Technicians and electronic engineers and designers who are interested in electronic design will find this book beneficial.

Cumulative Book Index - 1995

A world list of books in the English language.

Pointers on C - Kenneth A. Reek 1998

Pointers On C brings the power of pointers to your C programs. Designed for professionals and advanced students, Pointers on C provides a comprehensive resource for those needing in-depth coverage of the C programming language. An extensive explanation of pointer basics and a thorough exploration of their advanced features allows programmers to

incorporate the power of pointers into their C programs. Complete coverage, detailed explanations of C programming idioms, and thorough discussion of advanced topics makes Pointers on C a valuable tutorial and reference for students and professionals alike. Highlights: Provides complete background information needed for a thorough understanding of C. Covers pointers thoroughly, including syntax, techniques for their effective use and common programming idioms in which they appear. Compares different methods for implementing common abstract data structures. Offers an easy, conversant writing style to clearly explain difficult topics, and contains numerous illustrations and diagrams to help visualize complex concepts. Includes Programming Tips, discussing efficiency, portability, and software engineering issues, and warns of common pitfalls using Caution! Sections. Describes every function on the standard C library. 0673999866B04062001

Modeling with Data - Ben Klemens 2008-10-06

Modeling with Data fully explains how to execute computationally intensive analyses on very large data sets, showing readers how to determine the best methods for solving a variety of different problems, how to create and debug statistical models, and how to run an analysis and evaluate the results. Ben Klemens introduces a set of open and unlimited tools, and uses them to demonstrate data management, analysis, and simulation techniques essential for dealing with large data sets and computationally intensive procedures. He then demonstrates how to easily apply these tools to the many threads of statistical technique, including classical, Bayesian, maximum likelihood, and Monte Carlo methods. Klemens's accessible survey describes these models in a unified and nontraditional manner, providing alternative ways of looking at statistical concepts that often befuddle students. The book includes nearly one hundred sample programs of all kinds. Links to these programs will be available on this page at a later date. *Modeling with Data* will interest anyone looking for a comprehensive guide to these powerful statistical tools, including researchers and graduate students in the social sciences, biology, engineering, economics, and applied mathematics.

From Fortran to C - James F. Kerrigan 1991

This guide shows Fortran programmers how to adapt their language's rich legacy of engineering and scientific routines to C without having to go back to the basics. James F. Kerrigan shows how to use existing knowledge to translate Fortran programming constructs into equivalent C elements. He provides an in-depth comparison of Fortran and ANSI C programming concepts, with extensive code examples to illustrate the features of each language. In addition to a parallel analysis of program structures, the book examines each Fortran command and defines its counterpart in C, describes input/output and error control mechanisms in both languages, and identifies specific C attributes that do not appear in Fortran.

Mastering C Pointers - Robert J. Traister 2014-05-10

Mastering C Pointers: Tools for Programming Power focuses on the pointer operations of the C programming language, explaining exactly what pointers are and how to master them through easy-to-understand phrasing and by presenting many simple program examples. The functions of pointers with respect to memory access and memory allocation are also discussed. Comprised of 10 chapters, this book begins with the author's personal reflection on his first encounters with the C programming language and its pointers. The next two chapters presents steps to learning pointers, with emphasis on the essential processes that occur (invisibly and internally) when declaring standard numeric variables in C language and how to deal with C language character arrays and C strings. The reader is then introduced to string pointers and declared pointers of numeric types; the use of C language pointers and the memory allocation functions; and C language functions. The book also explores some of the other "entities" that pointers are used to access, including structures and unions, before concluding with an examination of the source code format of C language. This monograph is intended for both beginning and experienced C language programmers.

Professional CUDA C Programming - John Cheng 2014-09-09

Break into the powerful world of parallel GPU programming with this down-to-earth, practical guide. Designed for professionals across multiple industrial sectors, *Professional CUDA C Programming* presents CUDA -- a parallel computing platform and programming model designed to ease the development of GPU programming -- fundamentals in an easy-to-follow format, and teaches readers how to think in parallel and implement parallel algorithms on GPUs. Each chapter covers a specific topic, and includes workable examples that demonstrate the development process, allowing readers to explore both the "hard" and

"soft" aspects of GPU programming. Computing architectures are experiencing a fundamental shift toward scalable parallel computing motivated by application requirements in industry and science. This book demonstrates the challenges of efficiently utilizing compute resources at peak performance, presents modern techniques for tackling these challenges, while increasing accessibility for professionals who are not necessarily parallel programming experts. The CUDA programming model and tools empower developers to write high-performance applications on a scalable, parallel computing platform: the GPU. However, CUDA itself can be difficult to learn without extensive programming experience. Recognized CUDA authorities John Cheng, Max Grossman, and Ty McKercher guide readers through essential GPU programming skills and best practices in *Professional CUDA C Programming*, including: CUDA Programming Model GPU Execution Model GPU Memory model Streams, Event and Concurrency Multi-GPU Programming CUDA Domain-Specific Libraries Profiling and Performance Tuning. The book makes complex CUDA concepts easy to understand for anyone with knowledge of basic software development with exercises designed to be both readable and high-performance. For the professional seeking entrance to parallel computing and the high-performance computing community, *Professional CUDA C Programming* is an invaluable resource, with the most current information available on the market.

Scientific Programming and Computer Architecture - Divakar Viswanath 2017-07-28

A variety of programming models relevant to scientists explained, with an emphasis on how programming constructs map to parts of the computer. What makes computer programs fast or slow? To answer this question, we have to get behind the abstractions of programming languages and look at how a computer really works. This book examines and explains a variety of scientific programming models (programming models relevant to scientists) with an emphasis on how programming constructs map to different parts of the computer's architecture. Two themes emerge: program speed and program modularity. Throughout this book, the premise is to "get under the hood," and the discussion is tied to specific programs. The book digs into linkers, compilers, operating systems, and computer architecture to understand how the different parts of the computer interact with programs. It begins with a review of C/C++ and explanations of how libraries, linkers, and Makefiles work. Programming models covered include Pthreads, OpenMP, MPI, TCP/IP, and CUDA. The emphasis on how computers work leads the reader into computer architecture and occasionally into the operating system kernel. The operating system studied is Linux, the preferred platform for scientific computing. Linux is also open source, which allows users to peer into its inner workings. A brief appendix provides a useful table of machines used to time programs. The book's website (<https://github.com/divakarvi/bk-spca>) has all the programs described in the book as well as a link to the html text.

Books in Print Supplement - 1994

Mastering C Pointers :tools for Programming Power - Robert J. Traister 1990

This book features complete coverage on using and controlling C language pointers to make C applications more powerful and expressive. This edition is updated and revised to reflect the changes that have been brought about with the full adoption of ANSI C.

Dr. Dobb's Journal of Software Tools for the Professional Programmer - 1990

Refactoring - Paul Becker 1999

Refactoring is gaining momentum amongst the object oriented programming community. It can transform the internal dynamics of applications and has the capacity to transform bad code into good code. This book offers an introduction to refactoring.

Head First C - David Griffiths 2012-04-03

Learn key topics such as language basics, pointers and pointer arithmetic, dynamic memory management, multithreading, and network programming. Learn how to use the compiler, the make tool, and the archiver.

Recent Indian Publications on Display at World Book Fair - Sudhir Chandra Mathur 1996

Catalog of books on display at the 12th New Delhi World Book Fair, held at New Delhi in February 1996.

Mastering Go - Mihalis Tsoukalos 2021-08-31

Master key features of Go, including advanced concepts like concurrency

and working with JSON, to create and optimize real-world services, network servers, and clients

Key FeaturesThis third edition of the bestselling guide to advanced Go programming has been overhauled and expanded to cover RESTful servers, the WebSocket protocol, and Go generics

Use real-world exercises to build high-performance network servers and powerful command line utilities

Packed with practical examples and utilities to apply to your own development work and administrative tasks

Get clear explanations about Go nuances and features to simplify Go development

Book Description Mastering Go is the essential guide to putting Go to work on real production systems. This freshly updated third edition includes topics like creating RESTful servers and clients, understanding Go generics, and developing gRPC servers and clients. Mastering Go was written for programmers who want to explore the capabilities of Go in practice. As you work your way through the chapters, you'll gain confidence and a deep understanding of advanced Go concepts, including concurrency and the operation of the Go Garbage Collector, using Go with Docker, writing powerful command-line utilities, working with JavaScript Object Notation (JSON) data, and interacting with databases. You'll also improve your understanding of Go internals to optimize Go code and use data types and data structures in new and unexpected ways. This essential Go programming book will also take you through the nuances and idioms of Go with exercises and resources to fully embed your newly acquired knowledge. With the help of Mastering Go, you'll become an expert Go programmer by building Go systems and implementing advanced Go techniques in your projects.

What you will learn

- Use Go in production
- Write reliable, high-performance concurrent code
- Manipulate data structures including slices, arrays, maps, and pointers
- Develop reusable packages with reflection and interfaces
- Become familiar with generics for effective Go

programming

Create concurrent RESTful servers, and build gRPC clients and servers

Define Go structures for working with JSON data

Who this book is for You'll need to know the basics of Go before you get started with this book, but beyond that, anyone can sink their teeth into it. It's written primarily for Go programmers who have a bit of experience with the language and want to become expert practitioners.

C Programming - Greg M. Perry 2013

Provides instructions for writing C code to create games and mobile applications using the new C11 standard.

Understanding and Using C Pointers - Richard M Reese 2013-05-01

Improve your programming through a solid understanding of C pointers and memory management. With this practical book, you'll learn how pointers provide the mechanism to dynamically manipulate memory, enhance support for data structures, and enable access to hardware. Author Richard Reese shows you how to use pointers with arrays, strings, structures, and functions, using memory models throughout the book. Difficult to master, pointers provide C with much flexibility and power—yet few resources are dedicated to this data type. This comprehensive book has the information you need, whether you're a beginner or an experienced C or C++ programmer or developer. Get an introduction to pointers, including the declaration of different pointer types

Learn about dynamic memory allocation, de-allocation, and alternative memory management techniques

Use techniques for passing or returning data to and from functions

Understand the fundamental aspects of arrays as they relate to pointers

Explore the basics of strings and how pointers are used to support them

Examine why pointers can be the source of security problems, such as buffer overflow

Learn several pointer techniques, such as the use of opaque pointers, bounded pointers and, the restrict keyword