

C Interfaces And Implementations Techniques For Creating Reusable Software

21st Century C
 Effective Modern C++
 42 Specific Ways to Improve Your Use of C++11 and C++14
 Real World Instrumentation with Python
 Programming Embedded Systems
 The Bulgarian C# Book
 Effective C++
 Head First Design Patterns
 C Traps and Pitfalls
 101 Rules, Guidelines, and Best Practices
 Design Patterns
 Maximizing Reuse with Object Technology
 Proven Techniques for Heightened Performance
 Tackling Complexity in the Heart of Software
 Advanced R
 Nonsequential and Distributed Programming with Go
 Expert C Programming
 Synchronization of Concurrent Processes: Communication - Cooperation - Competition
 Designing Interfaces
 Learning C# 3.0
 Blue Book
 Learn C the Hard Way
 Deep C Secrets
 Implementation Patterns
 Domain-driven Design
 Programming C# 8.0
 Web Database Applications with PHP and MySQL
 A Retargetable C Compiler
 Design and Implementation
 Build Cloud, Web, and Desktop Applications
 Embedded C Programming
 Interface-oriented Design
 A Tour of C++
 Elements of Reusable Object-Oriented Software
 API Design for C++
 DirectX 9 User Interfaces
 Patterns for Effective Interaction Design
 Talking Directly to the Kernel and C Library
 With C and GNU Development Tools
 TinyOS Programming

C Interfaces And Implementations Techniques For Creating Reusable Software

Downloaded from ecobankpayservices.ecobank.com by guest

EMILIANO KIERA

21st Century C CRC Press

Learn how to develop your own applications to monitor or control instrumentation hardware. Whether you need to acquire data from a device or automate its functions, this practical book shows you how to use Python's rapid development capabilities to build interfaces that include everything from software to wiring. You get step-by-step instructions, clear examples, and hands-on tips for interfacing a PC to a variety of devices. Use the book's hardware survey to identify the interface type for your particular device, and then follow detailed examples to develop an interface with Python and C. Organized by interface type, data processing activities, and user interface implementations, this book is for anyone who works with instrumentation, robotics, data acquisition, or process control. Understand how to define the scope of an application and determine the algorithms necessary, and why it's important. Learn how to use industry-standard interfaces such as RS-232, RS-485, and GPIB. Create low-level extension modules in C to interface Python with a variety of hardware and test instruments. Explore the console, curses, TkInter, and wxPython for graphical and text-based user interfaces. Use open source software tools and libraries to reduce costs and avoid implementing functionality from scratch.

Effective Modern C++ Pearson Deutschland GmbH

User Interfaces for All is the first book dedicated to the issues of Universal Design and Universal Access in the field of Human-Computer Interaction (HCI). Universal Design (or Design for All) is an inclusive and proactive approach seeking to accommodate diversity in the users and usage contexts of interactive products, applications, and services, starting from the design phase of the development life cycle. The ongoing paradigm shift toward a knowledge-intensive information society is already bringing about radical changes in the way people work and interact with each other and with information. The requirement for Universal Design stems from the growing impact of the fusion of the emerging technologies, and from the different dimensions of diversity, which are intrinsic to the information society. This book unfolds the various aspects of this ongoing evolution from a variety of viewpoints. It's a collection of 30 chapters written by leading international authorities, affiliated with academic, research, and industrial organizations, and non-market institutions. The book provides a comprehensive overview of the state of the art in the field, and includes contributions from a variety of theoretical and applied disciplines and research themes. This book can also be used for teaching purposes in HCI courses at the undergraduate as well as graduate level. Students will be introduced to the human-, organizational-, and technology-oriented dimensions that call for a departure from traditional approaches to user interface development. Students will also get an overview of novel methods, techniques, tools, and frameworks for the design, implementation, and evaluation of user interfaces that are universally accessible and usable by the broadest possible end-user population. This comprehensive book is targeted to a broad readership, including HCI researchers, user interface designers, computer scientists, software engineers, ergonomists and usability engineers, Human Factors researchers and practitioners, organizational psychologists, system/product designers, sociologists, policy- and decision makers, scientists in government, industry and education, as well as assistive technology and rehabilitation experts.

42 Specific Ways to Improve Your Use of C++11 and C++14 Addison-Wesley Professional
 API Design for C++ provides a comprehensive discussion of Application Programming Interface (API) development, from initial design through implementation, testing, documentation, release, versioning, maintenance, and deprecation. It is the only book that teaches the strategies of C++ API

development, including interface design, versioning, scripting, and plug-in extensibility. Drawing from the author's experience on large scale, collaborative software projects, the text offers practical techniques of API design that produce robust code for the long term. It presents patterns and practices that provide real value to individual developers as well as organizations. API Design for C++ explores often overlooked issues, both technical and non-technical, contributing to successful design decisions that produce high quality, robust, and long-lived APIs. It focuses on various API styles and patterns that will allow you to produce elegant and durable libraries. A discussion on testing strategies concentrates on automated API testing techniques rather than attempting to include end-user application testing techniques such as GUI testing, system testing, or manual testing. Each concept is illustrated with extensive C++ code examples, and fully functional examples and working source code for experimentation are available online. This book will be helpful to new programmers who understand the fundamentals of C++ and who want to advance their design skills, as well as to senior engineers and software architects seeking to gain new expertise to complement their existing talents. Three specific groups of readers are targeted: practicing software engineers and architects, technical managers, and students and educators. The only book that teaches the strategies of C++ API development, including design, versioning, documentation, testing, scripting, and extensibility. Extensive code examples illustrate each concept, with fully functional examples and working source code for experimentation available online. Covers various API styles and patterns with a focus on practical and efficient designs for large-scale long-term projects.

Real World Instrumentation with Python Cambridge University Press

Do you spend a lot of time during the design process wondering what users really need? Do you hate those endless meetings where you argue how the interface should work? Have you ever developed something that later had to be completely redesigned? Paper Prototyping can help. Written by a usability engineer with a long and successful paper prototyping history, this book is a practical, how-to guide that will prepare you to create and test paper prototypes of all kinds of user interfaces. You'll see how to simulate various kinds of interface elements and interactions. You'll learn about the practical aspects of paper prototyping, such as deciding when the technique is appropriate, scheduling the activities, and handling the skepticism of others in your organization. Numerous case studies and images throughout the book show you real world examples of paper prototyping at work. Learn how to use this powerful technique to develop products that are more useful, intuitive, efficient, and pleasing: * Save time and money - solve key problems before implementation begins * Get user feedback early - use it to focus the development process * Communicate better - involve development team members from a variety of disciplines * Be more creative - experiment with many ideas before committing to one * Enables designers to solve design problems before implementation begins * Five case studies provide real world examples of paper prototyping at work * Delves into the specifics of what types of projects paper prototyping is and isn't good for.

Programming Embedded Systems Pearson Education

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.

The Bulgarian C# Book Addison-Wesley

Throw out your old ideas of C, and relearn a programming language that's substantially outgrown its origins. With 21st Century C, you'll discover up-to-date techniques that are absent from every other C text available. C isn't just the foundation of modern programming languages, it is a modern language, ideal for writing efficient, state-of-the-art applications. Learn to dump old habits that made sense on mainframes, and pick up the tools you need to use this evolved and aggressively simple language. No matter what programming language you currently champion, you'll agree that

C rocks. Set up a C programming environment with shell facilities, makefiles, text editors, debuggers, and memory checkers Use Autotools, C's de facto cross-platform package manager Learn which older C concepts should be downplayed or deprecated Explore problematic C concepts that are too useful to throw out Solve C's string-building problems with C-standard and POSIX-standard functions Use modern syntactic features for functions that take structured inputs Build high-level object-based libraries and programs Apply existing C libraries for doing advanced math, talking to Internet servers, and running databases

Effective C++ "O'Reilly Media, Inc."

Combines language tutorials with application design advice to cover the PHP server-side scripting language and the MySQL database engine.

Head First Design Patterns "O'Reilly Media, Inc."

The C++11 standard allows programmers to express ideas more clearly, simply, and directly, and to write faster, more efficient code. Bjarne Stroustrup, the designer and original implementer of C++, thoroughly covers the details of this language and its use in his definitive reference, *The C++ Programming Language, Fourth Edition*. In *A Tour of C++*, Stroustrup excerpts the overview chapters from that complete reference, expanding and enhancing them to give an experienced programmer—in just a few hours—a clear idea of what constitutes modern C++. In this concise, self-contained guide, Stroustrup covers most major language features and the major standard-library components—not, of course, in great depth, but to a level that gives programmers a meaningful overview of the language, some key examples, and practical help in getting started. Stroustrup presents the C++ features in the context of the programming styles they support, such as object-oriented and generic programming. His tour is remarkably comprehensive. Coverage begins with the basics, then ranges widely through more advanced topics, including many that are new in C++11, such as move semantics, uniform initialization, lambda expressions, improved containers, random numbers, and concurrency. The tour ends with a discussion of the design and evolution of C++ and the extensions added for C++11. This guide does not aim to teach you how to program (see Stroustrup's *Programming: Principles and Practice Using C++* for that); nor will it be the only resource you'll need for C++ mastery (see Stroustrup's *The C++ Programming Language, Fourth Edition*, for that). If, however, you are a C or C++ programmer wanting greater familiarity with the current C++ language, or a programmer versed in another language wishing to gain an accurate picture of the nature and benefits of modern C++, you can't find a shorter or simpler introduction than this tour provides.

C Traps and Pitfalls Wordware Publishing, Inc.

Describes ways to incorporate domain modeling into software development.

101 Rules, Guidelines, and Best Practices "O'Reilly Media, Inc."

A catalog of solutions to commonly occurring design problems, presenting 23 patterns that allow designers to create flexible and reusable designs for object-oriented software. Describes the circumstances in which each pattern is applicable, and discusses the consequences and trade-offs of using the pattern within a larger design. Patterns are compiled from real systems, and include code for implementation in object-oriented programming languages like C++ and Smalltalk. Includes a bibliography. Annotation copyright by Book News, Inc., Portland, OR

Design Patterns Xlibris Corporation

Summary Functional Programming in C++ teaches developers the practical side of functional programming and the tools that C++ provides to develop software in the functional style. This in-depth guide is full of useful diagrams that help you understand FP concepts and begin to think functionally. Purchase of the print book includes a free eBook in PDF, Kindle, and ePub formats from Manning Publications. About the Technology Well-written code is easier to test and reuse, simpler to parallelize, and less error prone. Mastering the functional style of programming can help you tackle the demands of modern apps and will lead to simpler expression of complex program logic, graceful error handling, and elegant concurrency. C++ supports FP with templates, lambdas, and other core language features, along with many parts of the STL. About the Book Functional Programming in C++ helps you unleash the functional side of your brain, as you gain a powerful new perspective on C++ coding. You'll discover dozens of examples, diagrams, and illustrations that break down the functional concepts you can apply in C++, including lazy evaluation, function objects and invocables, algebraic data types, and more. As you read, you'll match FP techniques with practical scenarios where they offer the most benefit. What's inside Writing safer code with no performance penalties Explicitly handling errors through the type system Extending C++ with new control structures Composing tasks with DSLs About the Reader Written for developers with two or more years of experience coding in C++. About the Author Ivan Čukić is a core developer at KDE and has been coding in C++ since 1998. He teaches modern C++ and functional programming at the Faculty of Mathematics at the University of Belgrade. Table of Contents Introduction to functional programming Getting started with functional programming Function objects Creating new functions from the old ones Purity: Avoiding mutable state Lazy evaluation Ranges Functional data structures Algebraic data types and pattern matching Monads Template metaprogramming Functional design for concurrent systems Testing and debugging

Maximizing Reuse with Object Technology "O'Reilly Media, Inc."

This book provides a hands-on introductory course on concepts of C programming using a PIC® microcontroller and CCS C compiler. Through a project-based approach, this book provides an easy to understand method of learning the correct and efficient practices to program a PIC® microcontroller in C language. Principles of C programming are introduced gradually, building on skill sets and knowledge. Early chapters emphasize the understanding of C language through experience and exercises, while the latter half of the book covers the PIC® microcontroller, its peripherals, and how to use those peripherals from within C in great detail. This book demonstrates the programming methodology and tools used by most professionals in embedded design, and will enable you to apply your knowledge and programming skills for any real-life application. Providing a step-by-step guide to the subject matter, this book will encourage you to alter, expand, and customize code for use in your own projects. A complete introduction to C programming using PIC microcontrollers, with a focus on real-world applications, programming methodology and tools Each chapter includes C code project examples, tables, graphs, charts, references, photographs, schematic diagrams, flow charts and compiler compatibility notes to channel your knowledge into real-world examples Online materials include presentation slides, extended tests, exercises, quizzes and answers, real-world case studies, videos and weblinks

Proven Techniques for Heightened Performance CRC Press

The free book "Fundamentals of Computer Programming with C#" is a comprehensive computer programming tutorial that teaches programming, logical thinking, data structures and algorithms, problem solving and high quality code with lots of examples in C#. It starts with the first steps in programming and software development like variables, data types, conditional statements, loops and arrays and continues with other basic topics like methods, numeral systems, strings and string processing, exceptions, classes and objects. After the basics this fundamental programming book enters into more advanced programming topics like recursion, data structures (lists, trees, hash-tables and graphs), high-quality code, unit testing and refactoring, object-oriented principles (inheritance, abstraction, encapsulation and polymorphism) and their implementation the C#

language. It also covers fundamental topics that each good developer should know like algorithm design, complexity of algorithms and problem solving. The book uses C# language and Visual Studio to illustrate the programming concepts and explains some C# / .NET specific technologies like lambda expressions, extension methods and LINQ. The book is written by a team of developers lead by Svetlin Nakov who has 20+ years practical software development experience. It teaches the major programming concepts and way of thinking needed to become a good software engineer and the C# language in the meantime. It is a great start for anyone who wants to become a skillful software engineer. The books does not teach technologies like databases, mobile and web development, but shows the true way to master the basics of programming regardless of the languages, technologies and tools. It is good for beginners and intermediate developers who want to put a solid base for a successful career in the software engineering industry. The book is accompanied by free video lessons, presentation slides and mind maps, as well as hundreds of exercises and live examples. Download the free C# programming book, videos, presentations and other resources from <http://introprogramming.info>. Title: Fundamentals of Computer Programming with C# (The Bulgarian C# Programming Book) ISBN: 9789544007737 ISBN-13: 978-954-400-773-7 (9789544007737) ISBN-10: 954-400-773-3 (9544007733) Author: Svetlin Nakov & Co. Pages: 1132 Language: English Published: Sofia, 2013 Publisher: Faber Publishing, Bulgaria Web site: <http://www.introprogramming.info> License: CC-Attribution-Share-Alike Tags: free, programming, book, computer programming, programming fundamentals, ebook, book programming, C#, CSharp, C# book, tutorial, C# tutorial; programming concepts, programming fundamentals, compiler, Visual Studio, .NET, .NET Framework, data types, variables, expressions, statements, console, conditional statements, control-flow logic, loops, arrays, numeral systems, methods, strings, text processing, StringBuilder, exceptions, exception handling, stack trace, streams, files, text files, linear data structures, list, linked list, stack, queue, tree, balanced tree, graph, depth-first search, DFS, breadth-first search, BFS, dictionaries, hash tables, associative arrays, sets, algorithms, sorting algorithm, searching algorithms, recursion, combinatorial algorithms, algorithm complexity, OOP, object-oriented programming, classes, objects, constructors, fields, properties, static members, abstraction, interfaces, encapsulation, inheritance, virtual methods, polymorphism, cohesion, coupling, enumerations, generics, namespaces, UML, design patterns, extension methods, anonymous types, lambda expressions, LINQ, code quality, high-quality code, high-quality classes, high-quality methods, code formatting, self-documenting code, code refactoring, problem solving, problem solving methodology, 9789544007737, 9544007733

Tackling Complexity in the Heart of Software Addison-Wesley Professional

Describes all phases of a modern compiler, including techniques in code generation and register allocation for imperative, functional and object-oriented languages.

Advanced R Addison-Wesley

Software -- Programming Languages.

Nonsequential and Distributed Programming with Go Addison-Wesley Professional

Software Development with C++: Maximizing Reuse with Object Technology is about software development and object-oriented technology (OT), with applications implemented in C++. The basis for any software development project of complex systems is the process, rather than an individual method, which simply supports the overall process. This book is not intended as a general, all-encompassing treatise on OT. The intent is to provide practical information that is directly applicable to a development project. Explicit guidelines are offered for the infusion of OT into the various development phases. The book is divided into five major parts. Part I describes why we need a development process, the phases and steps of the software process, and how we use individual methods to support this process. Part II lays the foundation for the concepts included in OT. Part III describes how OT is used in the various phases of the software development process, including the domain analysis, system requirements analysis, system design, software requirements analysis, software design, and implementation. Part IV deals exclusively with design issues for an anticipated C++ implementation. Part V is devoted to object-oriented programming with C++. This book is intended for practicing software developers, software managers, and computer science and software engineering students. Sufficient guidelines are included to aid project leaders in establishing an overall development process for small, medium, and large system applications.

Expert C Programming Addison-Wesley Professional

C Interfaces and Implementations Techniques for Creating Reusable Software Addison-Wesley Professional

Synchronization of Concurrent Processes: Communication - Cooperation - Competition Prentice Hall Professional

In today's fast and competitive world, a program's performance is just as important to customers as the features it provides. This practical guide teaches developers performance-tuning principles that enable optimization in C++. You'll learn how to make code that already embodies best practices of C++ design run faster and consume fewer resources on any computer—whether it's a watch, phone, workstation, supercomputer, or globe-spanning network of servers. Author Kurt Guntheroth provides several running examples that demonstrate how to apply these principles incrementally to improve existing code so it meets customer requirements for responsiveness and throughput. The advice in this book will prove itself the first time you hear a colleague exclaim, "Wow, that was fast. Who fixed something?" Locate performance hot spots using the profiler and software timers Learn to perform repeatable experiments to measure performance of code changes Optimize use of dynamically allocated variables Improve performance of hot loops and functions Speed up string handling functions Recognize efficient algorithms and optimization patterns Learn the strengths—and weaknesses—of C++ container classes View searching and sorting through an optimizer's eye Make efficient use of C++ streaming I/O functions Use C++ thread-based concurrency features effectively

Designing Interfaces "O'Reilly Media, Inc."

Coming to grips with C++11 and C++14 is more than a matter of familiarizing yourself with the features they introduce (e.g., auto type declarations, move semantics, lambda expressions, and concurrency support). The challenge is learning to use those features effectively—so that your software is correct, efficient, maintainable, and portable. That's where this practical book comes in. It describes how to write truly great software using C++11 and C++14—i.e. using modern C++. Topics include: The pros and cons of braced initialization, noexcept specifications, perfect forwarding, and smart pointer make functions The relationships among std::move, std::forward, rvalue references, and universal references Techniques for writing clear, correct, effective lambda expressions How std::atomic differs from volatile, how each should be used, and how they relate to C++'s concurrency API How best practices in "old" C++ programming (i.e., C++98) require revision for software development in modern C++ Effective Modern C++ follows the proven guideline-based, example-driven format of Scott Meyers' earlier books, but covers entirely new material. "After I learned the C++ basics, I then learned how to use C++ in production code from Meyer's series of Effective C++ books. Effective Modern C++ is the most important how-to book for advice on key guidelines, styles, and idioms to use modern C++ effectively and well. Don't own it yet? Buy this one. Now". -- Herb Sutter, Chair of ISO C++ Standards Committee and C++ Software Architect at Microsoft

Learning C# 3.0 Faber Publishing

If you're new to C#, this popular book is the ideal way to get started. Completely revised for the latest version of the language, Learning C# 3.0 starts with the fundamentals and takes you through intermediate and advanced C# features -- including generics, interfaces, delegates, lambda expressions, and LINQ. You'll also learn how to build Windows applications and handle data with C#. No previous programming experience is required -- in fact, if you've never written a line of code in your life, bestselling authors Jesse Liberty and Brian MacDonald will show you how it's done. Each chapter offers a self-contained lesson to help you master key concepts, with plenty of annotated

examples, illustrations, and a concise summary. With this book, you will: Learn how to program as you learn C# Grasp the principles of object-oriented programming through C# Discover how to use the latest features in C# 3.0 and the .NET 3.5 Framework--including LINQ and the Windows Presentation Foundation (WPF) Create Windows applications and data-driven applications You'll also find a unique Test Your Knowledge section in each chapter, with practical exercises and review quizzes, so you can practice new skills and test your understanding. If you're ready to dive into C# and .NET programming, this book is a great way to quickly get up to speed.

Related with C Interfaces And Implementations Techniques For Creating Reusable Software:

[© C Interfaces And Implementations Techniques For Creating Reusable Software Learn Algebra 1 In A Week](#)

[© C Interfaces And Implementations Techniques For Creating Reusable Software Learners Permit Practice Test Georgia](#)

[© C Interfaces And Implementations Techniques For Creating Reusable Software Learn2serve Food Handler Training Course Answers](#)