

BoostAsio C Network Programming

When somebody should go to the book stores, search commencement by shop, shelf by shelf, it is in point of fact problematic. This is why we give the book compilations in this website. It will no question ease you to see guide **BoostAsio C Network Programming** as you such as.

By searching the title, publisher, or authors of guide you really want, you can discover them rapidly. In the house, workplace, or perhaps in your method can be every best area within net connections. If you set sights on to download and install the BoostAsio C Network Programming, it is certainly easy then, in the past currently we extend the associate to purchase and make bargains to download and install BoostAsio C Network Programming thus simple!

Modern C++ Design - Debbie Debbie Lafferty 2001

This title documents a convergence of programming techniques - generic programming, template metaprogramming, object-oriented programming and design patterns. It describes the C++ techniques used in generic programming and implements a number of industrial strength components.

C++ Network Programming, Volume 2 - Douglas Schmidt 2002-10-29

Do you need to develop flexible software that can be customized quickly? Do you need to add the power and efficiency of frameworks to your software? The ADAPTIVE Communication Environment (ACE) is an open-source toolkit for building high-performance networked applications and next-generation middleware. ACE's power and flexibility arise from object-oriented frameworks, used to achieve the systematic reuse of networked application software. ACE frameworks handle common network programming tasks and can be customized using C++ language features to produce complete distributed applications. C++ Network Programming, Volume 2, focuses on ACE frameworks, providing thorough coverage of the concepts, patterns, and usage rules that form their

structure. This book is a practical guide to designing object-oriented frameworks and shows developers how to apply frameworks to concurrent networked applications. C++ Networking, Volume 1, introduced ACE and the wrapper facades, which are basic network computing ingredients. Volume 2 explains how frameworks build on wrapper facades to provide higher-level communication services. Written by two experts in the ACE community, this book contains: An overview of ACE frameworks Design dimensions for networked services Descriptions of the key capabilities of the most important ACE frameworks Numerous C++ code examples that demonstrate how to use ACE frameworks C++ Network Programming, Volume 2, teaches how to use frameworks to write networked applications quickly, reducing development effort and overhead. It will be an invaluable asset to any C++ developer working on networked applications.

Foundations of Python Network Programming - John Goerzen 2004-08-16

* Covers low-level networking in Python –essential for writing a new networked application protocol. * Many working examples demonstrate concepts in action -- and can be used as starting points for new projects.

* Networked application security is demystified. * Exhibits and explains multitasking network servers using several models, including forking, threading, and non-blocking sockets. * Features extensive coverage of Web and E-mail. Describes Python's database APIs.

Beginning C++20 - Ivor Horton
2021-02-12

Begin your programming journey with C++ including the C++20 standard. You'll start with the basics and progress through step-by-step examples to become a working C++ programmer. This book will include new features like parallelism, coroutines, modules, networking, ranges, and reflections. All you need are Beginning C++20 and any recent C++ compiler and you'll soon be writing real C++ programs. There is no assumption of prior programming knowledge. All language concepts that are explained in the book are illustrated with working program examples, and all chapters include exercises for you to test and practice your knowledge. Free source code downloads are provided for all examples from the text and solutions to the exercises. This latest edition has been fully updated to the latest version of the language, C++20, and to all conventions and best practices of modern C++. Beginning C++20 also introduces the elements of the C++ Standard Library that provide essential support for the C++20 language. What You Will Learn Begin programming with C++20 standard Carry out modular programming in C++ Work with arrays and loops, pointers and references, strings, and more Write your own functions, types, and operators Discover the essentials of object-oriented programming Use overloading, inheritance, virtual functions, and polymorphism Write generic function templates and class templates Use coroutines,

parallelism, ranges, auto type declarations, move semantics, lambda expressions, and much more Who This Book Is For Programmers new to C++ and those who may be looking for a refresh primer on C++ in general.

Mastering C++ Multithreading - Maya Posch 2017-07-28

Master multithreading and concurrent processing with C++ About This Book Delve into the fundamentals of multithreading and concurrency and find out how to implement them Explore atomic operations to optimize code performance Apply concurrency to both distributed computing and GPGPU processing Who This Book Is For This book is for intermediate C++ developers who wish to extend their knowledge of multithreading and concurrent processing. You should have basic experience with multithreading and be comfortable using C++ development toolchains on the command line. What You Will Learn Deep dive into the details of the how various operating systems currently implement multithreading Choose the best multithreading APIs when designing a new application Explore the use of mutexes, spin-locks, and other synchronization concepts and see how to safely pass data between threads Understand the level of API support provided by various C++ toolchains Resolve common issues in multithreaded code and recognize common pitfalls using tools such as Memcheck, CacheGrind, DRD, Helgrind, and more Discover the nature of atomic operations and understand how they can be useful in optimizing code Implement a multithreaded application in a distributed computing environment Design a C++-based GPGPU application that employs multithreading In Detail Multithreaded applications execute multiple threads in a single processor environment, allowing developers achieve concurrency. This

book will teach you the finer points of multithreading and concurrency concepts and how to apply them efficiently in C++. Divided into three modules, we start with a brief introduction to the fundamentals of multithreading and concurrency concepts. We then take an in-depth look at how these concepts work at the hardware-level as well as how both operating systems and frameworks use these low-level functions. In the next module, you will learn about the native multithreading and concurrency support available in C++ since the 2011 revision, synchronization and communication between threads, debugging concurrent C++ applications, and the best programming practices in C++. In the final module, you will learn about atomic operations before moving on to apply concurrency to distributed and GPGPU-based processing. The comprehensive coverage of essential multithreading concepts means you will be able to efficiently apply multithreading concepts while coding in C++. Style and approach This book is filled with examples that will help you become a master at writing robust concurrent and parallel applications in C++.

Boost.Asio C++ Network Programming - Wisnu Anggoro 2015-09-16

Learn effective C++ network programming with Boost.Asio and become a proficient C++ network programmer About This Book Learn efficient C++ network programming with minimum coding using Boost.Asio Your one-stop destination to everything related to the Boost.Asio library Explore the fundamentals of networking to choose designs with more examples, and learn the basics of Boost.Asio Who This Book Is For This book is for C++ Network programmers with basic knowledge of network programming, but no knowledge of how to use Boost.Asio for network

programming. What You Will Learn Prepare the tools to simplify network programming in C++ using Boost.Asio Explore the networking concepts of IP addressing, TCP/IP ports and protocols, and LAN topologies Get acquainted with the usage of the Boost libraries Get to know more about the content of Boost.Asio network programming and Asynchronous programming Establish communication between client and server by creating client-server application Understand the various functions inside Boost.Asio C++ libraries to delve into network programming Discover how to debug and run the code successfully In Detail Boost.Asio is a C++ library used for network programming operations. Organizations use Boost because of its productivity. Use of these high-quality libraries speed up initial development, result in fewer bugs, reduce reinvention-of-the-wheel, and cut long-term maintenance costs. Using Boost libraries gives an organization a head start in adopting new technologies. This book will teach you C++ Network programming using synchronous and asynchronous operations in Boost.Asio with minimum code, along with the fundamentals of Boost, server-client applications, debugging, and more. You will begin by preparing and setting up the required tools to simplify your network programming in C++ with Boost.Asio. Then you will learn about the basic concepts in networking such as IP addressing, TCP/IP protocols, and LAN with its topologies. This will be followed by an overview of the Boost libraries and their usage. Next you will get to know more about Boost.Asio and its concepts related to network programming. We will then go on to create a client-server application, helping you to understand the networking concepts. Moving on, you will discover how to

use all the functions inside the Boost.Asio C++ libraries. Lastly, you will understand how to debug the code if there are errors found and will run the code successfully. Style and approach An example-oriented book to show you the basics of networking and help you create a network application simply using Boost.Asio, with more examples for you to get up and running with Boost.Asio quickly.

Data Parallel C++ - James Reinders
2020-11-19

Learn how to accelerate C++ programs using data parallelism. This open access book enables C++ programmers to be at the forefront of this exciting and important new development that is helping to push computing to new levels. It is full of practical advice, detailed explanations, and code examples to illustrate key topics. Data parallelism in C++ enables access to parallel resources in a modern heterogeneous system, freeing you from being locked into any particular computing device. Now a single C++ application can use any combination of devices—including GPUs, CPUs, FPGAs and AI ASICs—that are suitable to the problems at hand. This book begins by introducing data parallelism and foundational topics for effective use of the SYCL standard from the Khronos Group and Data Parallel C++ (DPC++), the open source compiler used in this book. Later chapters cover advanced topics including error handling, hardware-specific programming, communication and synchronization, and memory model considerations. Data Parallel C++ provides you with everything needed to use SYCL for programming heterogeneous systems. What You'll Learn Accelerate C++ programs using data-parallel programming Target multiple device types (e.g. CPU, GPU, FPGA) Use SYCL and SYCL compilers Connect with computing's

heterogeneous future via Intel's oneAPI initiative Who This Book Is For Those new data-parallel programming and computer programmers interested in data-parallel programming using C++.

Learning Boost C++ Libraries - Arindam Mukherjee 2015-07-31

Filled with dozens of working code examples that illustrate the use of over 40 popular Boost libraries, this book takes you on a tour of Boost, helping you to independently build the libraries from source and use them in your own code. The first half of the book focuses on basic programming interfaces including generic containers and algorithms, strings, resource management, exception safety, and a miscellany of programming utilities that make everyday programming chores easy. Following a short interlude that introduces template metaprogramming and functional programming, the later chapters are devoted to systems programming interfaces, focusing on directory handling, I/O, concurrency, and network programming

The Modern C++ Challenge - Marius Bancila 2018-05-23

C++ is one of the most widely-used programming languages and has applications in a variety of fields, such as gaming, GUI programming, and operating systems, to name a few. Through the years, C++ has evolved into (and remains) one of the top choices for software developers worldwide. This book will show you some notable C++ features and how to ...

Introduction to C++ for Financial Engineers - Daniel J. Duffy
2013-10-24

This book introduces the reader to the C++ programming language and how to use it to write applications in quantitative finance (QF) and related areas. No previous knowledge of C or C++ is required -- experience with

VBA, Matlab or other programming language is sufficient. The book adopts an incremental approach; starting from basic principles then moving on to advanced complex techniques and then to real-life applications in financial engineering. There are five major parts in the book: C++ fundamentals and object-oriented thinking in QF Advanced object-oriented features such as inheritance and polymorphism Template programming and the Standard Template Library (STL) An introduction to GOF design patterns and their applications in QF Applications The kinds of applications include binomial and trinomial methods, Monte Carlo simulation, advanced trees, partial differential equations and finite difference methods. This book includes a companion website with all source code and many useful C++ classes that you can use in your own applications. Examples, test cases and applications are directly relevant to QF. This book is the perfect companion to Daniel J. Duffy's book *Financial Instrument Pricing using C++* (Wiley 2004, 0470855096 / 9780470021620) *C++ by Example* - Steve Donovan 2002 The By Example Series builds a language tutorial, example by example, with necessary text explaining the examples. Additionally, the complete example at the end of the book allows you to connect all of the examples to create the big picture.

Modern C++ Programming Cookbook - Marius Bancila 2020-09-11

A pragmatic recipe book for acquiring a comprehensive understanding of the complexities and core fundamentals of C++ programming Key Features Explore the latest language and library features of C++20 such as modules, coroutines, concepts, and ranges Shed new light on the core concepts in C++

programming, including functions, algorithms, threading, and concurrency, through practical self-contained recipes Leverage C++ features like smart pointers, move semantics, `constexpr`, and more for increased robustness and performance Book Description C++ has come a long way to be one of the most widely used general-purpose languages that is fast, efficient, and high-performance at its core. The updated second edition of *Modern C++ Programming Cookbook* addresses the latest features of C++20, such as modules, concepts, coroutines, and the many additions to the standard library, including ranges and text formatting. The book is organized in the form of practical recipes covering a wide range of problems faced by modern developers. The book also delves into the details of all the core concepts in modern C++ programming, such as functions and classes, iterators and algorithms, streams and the file system, threading and concurrency, smart pointers and move semantics, and many others. It goes into the performance aspects of programming in depth, teaching developers how to write fast and lean code with the help of best practices. Furthermore, the book explores useful patterns and delves into the implementation of many idioms, including `pimpl`, named parameter, and attorney-client, teaching techniques such as avoiding repetition with the factory pattern. There is also a chapter dedicated to unit testing, where you are introduced to three of the most widely used libraries for C++: Boost.Test, Google Test, and Catch2. By the end of the book, you will be able to effectively leverage the features and techniques of C++11/14/17/20 programming to enhance the performance, scalability, and efficiency of your applications. What

you will learn Understand the new C++20 language and library features and the problems they solve Become skilled at using the standard support for threading and concurrency for daily tasks Leverage the standard library and work with containers, algorithms, and iterators Solve text searching and replacement problems using regular expressions Work with different types of strings and learn the various aspects of compilation Take advantage of the file system library to work with files and directories Implement various useful patterns and idioms Explore the widely used testing frameworks for C++ Who this book is for The book is designed for entry- or medium-level C++ programmers who have a basic knowledge of C++ and want to master the language and become prolific modern C++ developers. Experienced C++ programmers can leverage this book to strengthen their command of C++ and find a good reference to many language and library features of C++11/14/17/20.

The Definitive Guide to HTML5

WebSocket - Vanessa Wang 2013-03-21
The Definitive Guide to HTML5
WebSocket is the ultimate insider's WebSocket resource. This revolutionary new web technology enables you to harness the power of true real-time connectivity and build responsive, modern web applications. This book contains everything web developers and architects need to know about WebSocket. It discusses how WebSocket-based architectures provide a dramatic reduction in unnecessary network overhead and latency compared to older HTTP (Ajax) architectures, how to layer widely used protocols such as XMPP and STOMP on top of WebSocket, and how to secure WebSocket connections and deploy WebSocket-based applications to the enterprise. Build real-time web applications with HTML5. This

book: Introduces you to the WebSocket API and protocol Describes and provides real-world examples of protocol communication over WebSocket Explains WebSocket security and enterprise deployment

The C++ Programming Language - Bjarne Stroustrup 2000

Hands-On Network Programming with C - Lewis Van Winkle 2019-05-13

A comprehensive guide to programming with network sockets, implementing Internet protocols, designing IoT devices, and much more with C Key Features Leverage your C or C++ programming skills to build powerful network applications Get to grips with a variety of network protocols that allow you to load web pages, send emails, and do much more Write portable network code for operating systems such as Windows, Linux, and macOS Book Description Network programming, a challenging topic in C, is made easy to understand with a careful exposition of socket programming APIs. This book gets you started with modern network programming in C and the right use of relevant operating system APIs. This book covers core concepts, such as hostname resolution with DNS, that are crucial to the functioning of the modern web. You'll delve into the fundamental network protocols, TCP and UDP. Essential techniques for networking paradigms such as client-server and peer-to-peer models are explained with the help of practical examples. You'll also study HTTP and HTTPS (the protocols responsible for web pages) from both the client and server perspective. To keep up with current trends, you'll apply the concepts covered in this book to gain insights into web programming for IoT. You'll even get to grips with network monitoring and implementing security best practices. By the end of this book, you'll have experience

of working with client-server applications, and be able to implement new network programs in C. The code in this book is compatible with the older C99 version as well as the latest C18 and C++17 standards. Special consideration is given to writing robust, reliable, and secure code that is portable across operating systems, including Winsock sockets for Windows and POSIX sockets for Linux and macOS. What you will learn

Uncover cross-platform socket programming APIs
Implement techniques for supporting IPv4 and IPv6
Understand how TCP and UDP connections work over IP
Discover how hostname resolution and DNS work
Interface with web APIs using HTTP and HTTPS
Acquire hands-on experience with Simple Mail Transfer Protocol (SMTP)
Apply network programming to the Internet of Things (IoT)

Who this book is for
If you're a developer or a system administrator who wants to enter the world of network programming, this book is for you. Basic knowledge of C programming is assumed.

Learning C++ Functional Programming - Wisnu Anggoro 2017-08-10

Apply Functional Programming techniques to C++ to build highly modular, testable, and reusable code

About This Book
Modularize your applications and make them highly reusable and testable
Get familiar with complex concepts such as metaprogramming, concurrency, and immutability
A highly practical guide to building functional code in C++ filled with lots of examples and real-world use cases

Who This Book Is For
This book is for C++ developers comfortable with OOP who are interested in learning how to apply the functional paradigm to create robust and testable apps. What You Will Learn
Get to know the difference between imperative and functional approaches
See the use of first-class

functions and pure functions in a functional style
Discover various techniques to apply immutable state to avoid side effects
Design a recursive algorithm effectively
Create faster programs using lazy evaluation
Structure code using design patterns to make the design process easier
Use concurrency techniques to develop responsive software
Learn how to use the C++ Standard Template Library and metaprogramming in a functional way to improve code optimization

In Detail
Functional programming allows developers to divide programs into smaller, reusable components that ease the creation, testing, and maintenance of software as a whole. Combined with the power of C++, you can develop robust and scalable applications that fulfill modern day software requirements. This book will help you discover all the C++ 17 features that can be applied to build software in a functional way. The book is divided into three modules—the first introduces the fundamentals of functional programming and how it is supported by modern C++. The second module explains how to efficiently implement C++ features such as pure functions and immutable states to build robust applications. The last module describes how to achieve concurrency and apply design patterns to enhance your application's performance. Here, you will also learn to optimize code using metaprogramming in a functional way. By the end of the book, you will be familiar with the functional approach of programming and will be able to use these techniques on a daily basis.

Style and approach
This book uses a module-based approach, where each module will cover important aspects of functional programming in C++ and will help you develop efficient and robust applications through gaining a

practical understanding.

C# Programming :: - Harry. H. Chaudhary. 2014-06-02

This book gives a good start and complete introduction for C# Programming for Beginner's. While reading this book it is fun and easy to read it. This book is best suitable for first time C# readers, Covers all fast track topics of C# for all Computer Science students and Professionals. This book is targeted toward those who have little or no programming experience or who might be picking up C# as a second language. The book has been structured and written with a purpose: to get you productive as quickly as possible. I've used my experiences in writing applications with C# and teaching C# to create a book that I hope cuts through the fluff and teaches you what you need to know. All too often, authors fall into the trap of focusing on the technology rather than on the practical application of the technology. I've worked hard to keep this book focused on teaching you practical skills that you can apply immediately toward a development project. This book is divided into ten Chapters, each of which focuses on a different aspect of developing applications with C#. These parts generally follow the flow of tasks you'll perform as you begin creating your own programs with C#. I recommend that you read them in the order in which they appear. Using C#, this book develops the concepts and theory of Building the Program Logic and Interfaces analysis, Exceptions, Delegates and Events and other important things in a gradual, step-by-step manner, proceeding from concrete examples to abstract principles. Standish covers a wide range of both traditional and contemporary software engineering topics. This is a handy guide of

sorts for any computer science engineering Students, Thinking In C# Programming is a solution bank for various complex problems related to C# and .NET. It can be used as a reference manual by Computer Science Engineering students. This Book also covers all aspects of B.TECH CS, IT, and BCA and MCA, BSC IT. Preview introduced programmers to a new era called functional programming. C# focused on bridging the gap between programming languages and databases. This book covers all the language features from the first version through C# . It also provides you with the essentials of using Visual Studio 2005 to let you enjoy its capabilities and save you time by using features such as IntelliSense. Learning a new programming language can be intimidating. If you've never programmed before, the act of typing seemingly cryptic text to produce sleek and powerful applications probably seems like a black art, and you might wonder how you'll ever learn everything you need to know. The answer is, of course, one step at a time. The first step to learning a language is the same as that of any other activity: building confidence. Programming is part art and part science. Although it might seem like magic, it's more akin to illusion: After you know how things work a lot of the mysticism goes away, freeing you to focus on the mechanics necessary to produce any given desired result. Chapter 1 (Introduction To C# AND .NET) Chapter 2 (Your First Go at C# Programming) Chapter 3 (C# Data Types)' Chapter 4 (Building the Program Logic) Chapter 5 (Using Classes) Chapter 6 (Function Members) Chapter 7 (Structs, Enums, and Attributes) Chapter 8 (Interfaces) Chapter 9 (Exceptions) Chapter 10 (Delegates and Events) **Beginning C++17** - Ivor Horton 2018-03-24

Learn how to program using the updated C++17 language. You'll start with the basics and progress through step-by-step examples to become a working C++ programmer. All you need are Beginning C++17 and any recent C++ compiler and you'll soon be writing real C++ programs. There is no assumption of prior programming knowledge. All language concepts that are explained in the book are illustrated with working program examples, and all chapters include exercises for you to test and practice your knowledge. Code downloads are provided for all examples from the text and solutions to the exercises. This latest edition has been fully updated to the latest version of the language, C++17, and to all conventions and best practices of so-called modern C++. Beginning C++17 also introduces the elements of the C++ Standard Library that provide essential support for the C++17 language. What You'll Learn Define variables and make decisions Work with arrays and loops, pointers and references, strings, and more Write your own functions, types, and operators Discover the essentials of object-oriented programming Use overloading, inheritance, virtual functions and polymorphism Write generic function templates and class templates Get up to date with modern C++ features: auto type declarations, move semantics, lambda expressions, and more Examine the new additions to C++17 Who This Book Is For Programmers new to C++ and those who may be looking for a refresh primer on the C++17 programming language in general.

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.

C++ Design Patterns and Derivatives

Pricing - Mark S. Joshi 2004-08-05

Design patterns are the cutting-edge
paradigm for programming in object-
oriented languages. Here they are
discussed, for the first time in a
book, in the context of implementing
financial models in C++. Assuming
only a basic knowledge of C++ and
mathematical finance, the reader is
taught how to produce well-designed,
structured, re-usable code via
concrete examples. Each example is
treated in depth, with the whys and
wherefores of the chosen method of
solution critically examined. Part of
the book is devoted to designing re-
usable components that are then put
together to build a Monte Carlo
pricer for path-dependent exotic
options. Advanced topics treated
include the factory pattern, the
singleton pattern and the decorator
pattern. Complete ANSI/ISO-compatible
C++ source code is included on a CD
for the reader to study and re-use
and so develop the skills needed to
implement financial models with
object-oriented programs and become a
working financial engineer. Please
note the CD supplied with this book
is platform-dependent and PC users
will not be able to use the files
without manual intervention in order
to remove extraneous characters.
Cambridge University Press apologises
for this error. Machine readable
files for all users can be obtained
from www.markjoshi.com/design.

C++ Template Metaprogramming - David
Abrahams 2005

This book explains what
metaprogramming is and how it is best
used. It provides the foundation
you'll need to use the template
metaprogramming effectively in your

own work. This book is aimed at any
programmer who is comfortable with
idioms of the Standard Template
Library (STL). C++ power-users will
gain a new insight into their
existing work and a new fluency in
the domain of metaprogramming.
Intermediate-level programmers who
have learned a few advanced template
techniques will see where these
tricks fit in the big picture and
will gain the conceptual foundation
to use them with discipline.
Programmers who have caught the scent
of metaprogramming, but for whom it
is still mysterious, will finally
gain a clear understanding of how,
when, and why it works. All readers
will leave with a new tool of
unprecedented power at their disposal
- the Boost Metaprogramming Library.
*Modern C++ Programming Cookbook -
Second Edition* - MARIUS. BANCILA
2020-09-11

Over 100 recipes to help you overcome
your difficulties with C++
programming and gain a deeper
understanding of the working of
modern C++ Key Features Explore the
most important language and library
features of C++17, including
containers, algorithms, regular
expressions, threads, and more, Get
going with unit testing frameworks
Boost.Test, Google Test and Catch,
Extend your C++ knowledge and take
your development skills to new
heights by making your applications
fast, robust, and scalable. Book
Description C++ is one of the most
widely used programming languages.
Fast, efficient, and flexible, it is
used to solve many problems. The
latest versions of C++ have seen
programmers change the way they code,
giving up on the old-fashioned C-
style programming and adopting modern
C++ instead. Beginning with the
modern language features, each recipe
addresses a specific problem, with a
discussion that explains the solution

and offers insight into how it works. You will learn major concepts about the core programming language as well as common tasks faced while building a wide variety of software. You will learn about concepts such as concurrency, performance, meta-programming, lambda expressions, regular expressions, testing, and many more in the form of recipes. These recipes will ensure you can make your applications robust and fast. By the end of the book, you will understand the newer aspects of C++11/14/17 and will be able to overcome tasks that are time-consuming or would break your stride while developing. What you will learn

Get to know about the new core language features and the problems they were intended to solve

Understand the standard support for threading and concurrency and know how to put them on work for daily basic tasks

Leverage C++'s features to get increased robustness and performance

Explore the widely-used testing frameworks for C++ and implement various useful patterns and idioms

Work with various types of strings and look at the various aspects of compilation

Explore functions and callable objects with a focus on modern features

Leverage the standard library and work with containers, algorithms, and iterators

Use regular expressions for find and replace string operations

Take advantage of the new filesystem library to work with files and directories

Use the new utility additions to the standard library to solve common problems developers encounter including `string_view`, `any`, `optional` and `variant` types

Who this book is for

If you want to overcome difficult phases of development with C++ and leverage its features using modern programming practices, then this book is for you. The book is designed for both experienced C++

programmers as well as people with strong knowledge of OOP concepts.

Effective Modern C++ - Scott Meyers
2014-11-11

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

Expert C++ - Vardan Grigoryan

2020-04-10

Design and architect real-world scalable C++ applications by exploring advanced techniques in low-level programming, object-oriented programming (OOP), the Standard Template Library (STL), metaprogramming, and concurrency. Key features: Design professional-grade, maintainable apps by learning advanced concepts such as functional programming, templates, and networking. Apply design patterns and best practices to solve real-world problems. Improve the performance of your projects by designing concurrent data structures and algorithms. Book Description: C++ has evolved over the years and the latest release – C++20 – is now available. Since C++11, C++ has been constantly enhancing the language feature set. With the new version, you'll explore an array of features such as concepts, modules, ranges, and coroutines. This book will be your guide to learning the intricacies of the language, techniques, C++ tools, and the new features introduced in C++20, while also helping you apply these when building modern and resilient software. You'll start by exploring the latest features of C++, and then move on to advanced techniques such as multithreading, concurrency, debugging, monitoring, and high-performance programming. The book will delve into object-oriented programming principles and the C++ Standard Template Library, and even show you how to create custom templates. After this, you'll learn about different approaches such as test-driven development (TDD), behavior-driven development (BDD), and domain-driven design (DDD), before taking a look at the coding best practices and design patterns essential for building professional-grade applications. Toward the end of the book, you will gain useful

insights into the recent C++ advancements in AI and machine learning. By the end of this C++ programming book, you'll have gained expertise in real-world application development, including the process of designing complex software. What you will learn: Understand memory management and low-level programming in C++ to write secure and stable applications. Discover the latest C++20 features such as modules, concepts, ranges, and coroutines. Understand debugging and testing techniques and reduce issues in your programs. Design and implement GUI applications using Qt5. Use multithreading and concurrency to make your programs run faster. Develop high-end games by using the object-oriented capabilities of C++. Explore AI and machine learning concepts with C++. Who this book is for: This C++ book is for experienced C++ developers who are looking to take their knowledge to the next level and perfect their skills in building professional-grade applications. C++ Coding Standards - Herb Sutter 2004-10-25

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.

Functional C# - Wisnu Anggoro
2016-12-30

Uncover the secrets of functional programming using C# and change the way you approach your applications forever About This Book This book focuses on the functional paradigm of C#, which will give you a whole new angle on coding with C# It illustrates the advantages that functional programming brings to the table and the associated coding benefits This practical guide covers all the aspects of functional programming and provides solutions that can be applied in business

scenarios Who This Book Is For This book is suitable for C# developers with basic prior knowledge of C# and with no functional programming experience at all. What You Will Learn Develop an application using the functional approach Implement unit testing to functionally program code Create efficient code using functional programming Work through a LINQ query so you can work with data Compose asynchronous programs to create a responsive application Use recursion in function programming in order to simplify code Optimize the program code using Laziness and Caching Techniques In Detail Functional programming makes your application faster, improves performance, and increases your productivity. C# code is written at a higher level of abstraction, so that code will be closer to business requirements, abstracting away many low-level implementation details. This book bridges the language gap for C# developers by showing you how to create and consume functional constructs in C#. We also bridge the domain gap by showing how functional constructs can be applied in business scenarios. We'll take you through lambda expressions and extension methods, and help you develop a deep understanding of the concepts and practices of LINQ and recursion in C#. By the end of the book, you will be able to write code using the best approach and will be able to perform unit testing in functional programming, changing how you write your applications and revolutionizing your projects. Style and approach This book takes a pragmatic approach and shows you techniques to write better functional constructs in C#. We'll also show you how these concepts can be applied in business scenarios.

Programming Rust - Jim Blandy
2021-06-11

Systems programming provides the foundation for the world's computation. Writing performance-sensitive code requires a programming language that puts programmers in control of how memory, processor time, and other system resources are used. The Rust systems programming language combines that control with a modern type system that catches broad classes of common mistakes, from memory management errors to data races between threads. With this practical guide, experienced systems programmers will learn how to successfully bridge the gap between performance and safety using Rust. Jim Blandy, Jason Orendorff, and Leonora Tindall demonstrate how Rust's features put programmers in control over memory consumption and processor use by combining predictable performance with memory safety and trustworthy concurrency. You'll learn: Rust's fundamental data types and the core concepts of ownership and borrowing How to write flexible, efficient code with traits and generics How to write fast, multithreaded code without data races Rust's key power tools: closures, iterators, and asynchronous programming Collections, strings and text, input and output, macros, unsafe code, and foreign function interfaces This revised, updated edition covers the Rust 2021 Edition.

UNIX System V Network Programming - Stephen A. Rago 1993

"Steve Rago offers valuable insights into the kernel-level features of SVR4 not covered elsewhere; I think readers will especially appreciate the coverage of STREAMS, TLI, and SLIP." - W. Richard Stevens, author of UNIX Network Programming, Advanced Programming in the UNIX Environment, TCP/IP Illustrated Volume 1, and TCP/IP Illustrated Volume 2 Finally, with UNIX(R) System V Network Programming, an authoritative

reference is available for programmers and system architects interested in building networked and distributed applications for UNIX System V. Even if you currently use a different version of the UNIX system, such as the latest release of 4.3BSD or SunOS, this book is valuable to you because it is centered around UNIX System V Release 4, the version of the UNIX system that unified many of the divergent UNIX implementations. For those professionals new to networking and UNIX system programming, two introductory chapters are provided. The author then presents the programming interfaces most important to building communication software in System V, including STREAMS, the Transport Layer Interface library, Sockets, and Remote Procedure Calls. So that your designs are not limited to user-level, the author also explains how to write kernel-level communication software, including STREAMS drivers, modules, and multiplexors. Many examples are provided, including an Ethernet driver and a transport-level multiplexing driver. In the final chapter, the author brings the material from previous chapters together, presenting the design of a SLIP communication package.

0201563185B04062001

Boost.Asio C++ Network Programming Cookbook - Dmytro Radchuk 2016-01-25

Over 25 hands-on recipes to create robust and highly-efficient cross-platform distributed applications with the Boost.Asio library About This Book Build highly efficient distributed applications with ease Enhance your cross-platform network programming skills with one of the most reputable C++ libraries Find solutions to real-world problems related to network programming with ready-to-use recipes using this detailed and practical handbook Who

This Book Is For If you want to enhance your C++ network programming skills using the Boost.Asio library and understand the theory behind development of distributed applications, this book is just what you need. The prerequisite for this book is experience with general C++11. To get the most from the book and comprehend advanced topics, you will need some background experience in multithreading. What You Will Learn Boost your working knowledge of one of the most reputable C++ networking libraries—Boost.Asio Familiarize yourself with the basics of TCP and UDP protocols Create scalable and highly-efficient client and server applications Understand the theory behind development of distributed applications Increase the security of your distributed applications by adding SSL support Implement a HTTP client easily Use iostreams, scatter-gather buffers, and timers In Detail Starting with recipes demonstrating the execution of basic Boost.Asio operations, the book goes on to provide ready-to-use implementations of client and server applications from simple synchronous ones to powerful multithreaded scalable solutions. Finally, you are presented with advanced topics such as implementing a chat application, implementing an HTTP client, and adding SSL support. All the samples presented in the book are ready to be used in real projects just out of the box. As well as excellent practical examples, the book also includes extended supportive theoretical material on distributed application design and construction. Style and approach This book is a set of recipes, each containing the statement and description of a particular practical problem followed by code sample providing the solution to the problem and detailed step-by-step explanation. Recipes are grouped

by topic into chapters and ordered by the level of complexity from basic to advanced.

Boost C++ Application Development Cookbook - Antony Polukhin 2017-08-30
Learn to build applications faster and better by leveraging the real power of Boost and C++ About This Book Learn to use the Boost libraries to simplify your application development Learn to develop high quality, fast and portable applications Learn the relations between Boost and C++11/C++4/C++17 Who This Book Is For This book is for developers looking to improve their knowledge of Boost and who would like to simplify their application development processes. Prior C++ knowledge and basic knowledge of the standard library is assumed. What You Will Learn Get familiar with new data types for everyday use Use smart pointers to manage resources Get to grips with compile-time computations and assertions Use Boost libraries for multithreading Learn about parallel execution of different task Perform common string-related tasks using Boost libraries Split all the processes, computations, and interactions to tasks and process them independently Learn the basics of working with graphs, stacktracing, testing and interprocess communications Explore different helper macros used to detect compiler, platform and Boost features In Detail If you want to take advantage of the real power of Boost and C++ and avoid the confusion about which library to use in which situation, then this book is for you. Beginning with the basics of Boost C++, you will move on to learn how the Boost libraries simplify application development. You will learn to convert data such as string to numbers, numbers to string, numbers to numbers and more. Managing resources will become a piece of

cake. You'll see what kind of work can be done at compile time and what Boost containers can do. You will learn everything for the development of high quality fast and portable applications. Write a program once and then you can use it on Linux, Windows, MacOS, Android operating systems. From manipulating images to graphs, directories, timers, files, networking – everyone will find an interesting topic. Be sure that knowledge from this book won't get outdated, as more and more Boost libraries become part of the C++ Standard.

The Boost C++ Libraries - Boris Schäling 2011

The second edition of *The Boost C++ Libraries* introduces 72 Boost libraries that provide a wide range of useful capabilities. They help you manage memory and process strings more easily. They provide containers and other data structures that go well beyond what the standard library offers. They make it easy to build platform-independent network applications. Simply put, these 72 libraries greatly expand your C++ toolbox. The second edition contains more than 430 examples. All examples are as short as possible, but they are complete, so you can compile and run them as is. They show you what the Boost libraries offer and give you a head start on using the libraries in your own applications. The goal of this book is to increase your efficiency as a C++ developer and to simplify software development with C++. The Boost libraries introduced in this book will help you write less code with fewer bugs and finish projects faster. Your code will be more concise and self-explanatory and more easily adapted when requirements change. The second edition is based on the Boost libraries 1.55.0 and 1.56.0 with the latter version having been released

in August 2014. The examples are based on C++11 and have been tested with Visual Studio 2013, GCC 4.8 and Clang 3.3 on various platforms. For Boost libraries which were incorporated into the C++11 standard library, differences between Boost and the standard library are highlighted. The Boost libraries are one of the most important and influential open source C++ libraries. Their source code is available under a permissive free software license. Several Boost libraries have been incorporated into the C++11 standard library. The Boost libraries are developed and supported by the Boost community - a worldwide developer community with a strong interest in pushing C++ boundaries further.

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.

Boost. Asio C++ Network Programming - Second Edition - Wisnu Anggoro 2015-09-16

Learn effective C++ network programming with Boost.Asio and become a proficient C++ network programmer
About This Book- Learn efficient C++ network programming with minimum coding using Boost.Asio- Your one-stop destination to everything related to the Boost.Asio library- Explore the fundamentals of networking to choose designs with more examples, and learn the basics of Boost.Asio
Who This Book Is For
This book is for C++ Network programmers with basic knowledge of

network programming, but no knowledge of how to use Boost.Asio for network programming. What You Will Learn- Prepare the tools to simplify network programming in C++ using Boost.Asio- Explore the networking concepts of IP addressing, TCP/IP ports and protocols, and LAN topologies- Get acquainted with the usage of the Boost libraries- Get to know more about the content of Boost.Asio network programming and Asynchronous programming- Establish communication between client and server by creating client-server application- Understand the various functions inside Boost.Asio C++ libraries to delve into network programming- Discover how to debug and run the code successfully In Detail Boost.Asio is a C++ library used for network programming operations. Organizations use Boost because of its productivity. Use of these high-quality libraries speed up initial development, result in fewer bugs, reduce reinvention-of-the-wheel, and cut long-term maintenance costs. Using Boost libraries gives an organization a head start in adopting new technologies. This book will teach you C++ Network programming using synchronous and asynchronous operations in Boost.Asio with minimum code, along with the fundamentals of Boost, server-client applications, debugging, and more. You will begin by preparing and setting up the required tools to simplify your network programming in C++ with Boost.Asio. Then you will learn about the basic concepts in networking such as IP addressing, TCP/IP protocols, and LAN with its topologies. This will be followed by an overview of the Boost libraries and their usage. Next you will get to know more about Boost.Asio and its concepts related to network programming. We will then go on to create a client-server application, helping you to

understand the networking concepts. Moving on, you will discover how to use all the functions inside the Boost.Asio C++ libraries. Lastly, you will understand how to debug the code if there are errors found and will run the code successfully. Style and approach An example-oriented book to show you the basics of networking and help you create a network application simply using Boost.Asio, with more examples for you to get up and running with Boost.Asio quickly.

Introduction to the Boost C++ Libraries - Robert Demming 2010-11-01 C++ is one of the most important and influential programming languages for application development. It supports the modular, object-oriented and generic programming models and its flexibility has been one of the main reasons why it has been so successful. With the emergence of the Boost Libraries (www.boost.org) we see that C++ is brought to a new level, namely a set of reusable and modular template libraries that C++ developers can use in their applications. This book is dedicated to a number of Boost libraries for higher-order functions, data types and data structures, libraries for text and string processing, multi-threading, random number generation and more. We also discuss how Boost and design patterns are used to promote the flexibility of code. Each library is described in a step-by-step manner. Numerous examples are given to show the functionality of each library. The full source code is freely available to purchasers of the book. Coverage Includes Understanding and using 30 major Boost libraries. Learn about higher-order functions, data structures, memory management, multi-threading and more. Using Boost in new and existing applications. Integrating Boost and the Gang-Of-Four design patterns. Ready-to-run projects for Visual Studio.

Appendices and exercises."

C++ High Performance - Björn Andrist
2018-01-31

Write code that scales across CPU registers, multi-core, and machine clusters

Key Features

- Explore concurrent programming in C++
- Identify memory management problems
- Use SIMD and STL containers for performance improvement

Book Description

C++ is a highly portable language and can be used to write both large-scale applications and performance-critical code. It has evolved over the last few years to become a modern and expressive language. This book will guide you through optimizing the performance of your C++ apps by allowing them to run faster and consume fewer resources on the device they're running on without compromising the readability of your code base. The book begins by helping you measure and identify bottlenecks in a C++ code base. It then moves on by teaching you how to use modern C++ constructs and techniques. You'll see how this affects the way you write code. Next, you'll see the importance of data structure optimization and memory management, and how it can be used efficiently with respect to CPU caches. After that, you'll see how STL algorithm and composable Range V3 should be used to both achieve faster execution and more readable code, followed by how to use STL containers and how to write your own specialized iterators. Moving on, you'll get hands-on experience in making use of modern C++ metaprogramming and reflection to reduce boilerplate code as well as in working with proxy objects to perform optimizations under the hood. After that, you'll learn concurrent programming and understand lock-free data structures. The book ends with an overview of parallel algorithms using STL execution policies, Boost Compute, and OpenCL to utilize both the CPU

and the GPU. What you will learn

- Benefits of modern C++ constructs and techniques
- Identify hardware bottlenecks, such as CPU cache misses, to boost performance
- Write specialized data structures for performance-critical code
- Use modern metaprogramming techniques to reduce runtime calculations
- Achieve efficient memory management using custom memory allocators
- Reduce boilerplate code using reflection techniques
- Reap the benefits of lock-free concurrent programming
- Perform under-the-hood optimizations with preserved readability using proxy objects
- Gain insights into subtle optimizations used by STL algorithms
- Utilize the Range V3 library for expressive C++ code
- Parallelize your code over CPU and GPU, without compromising readability

Who this book is for

If you're a C++ developer looking to improve the speed of your code or simply wanting to take your skills up to the next level, then this book is perfect for you.

C++ Network Programming, Volume I - Douglas Schmidt
2001-12-10

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.

C++ Crash Course - Josh Lospinoso
2019-09-24

A fast-paced, thorough introduction to modern C++ written for experienced programmers. After reading C++ Crash Course, you'll be proficient in the core language concepts, the C++ Standard Library, and the Boost Libraries. C++ is one of the most widely used languages for real-world software. In the hands of a knowledgeable programmer, C++ can produce small, efficient, and readable code that any programmer would be proud of. Designed for intermediate to advanced programmers, C++ Crash Course cuts through the weeds to get you straight to the core of C++17, the most modern revision of the ISO standard. Part 1 covers the

core of the C++ language, where you'll learn about everything from types and functions, to the object life cycle and expressions. Part 2 introduces you to the C++ Standard Library and Boost Libraries, where you'll learn about all of the high-quality, fully-featured facilities available to you. You'll cover special utility classes, data structures, and algorithms, and learn how to manipulate file systems and build high-performance programs that communicate over networks. You'll learn all the major features of modern C++, including:

- Fundamental types, reference types, and user-defined types
- The object lifecycle including storage duration, memory management, exceptions, call stacks, and the RAII paradigm
- Compile-time polymorphism with templates and runtime polymorphism with virtual classes
- Advanced expressions, statements, and functions
- Smart pointers, data structures, dates and times, numerics, and probability/statistics facilities
- Containers, iterators, strings, and algorithms
- Streams and files, concurrency, networking, and application development

With well over 500 code samples and nearly 100 exercises, C++ Crash Course is sure to help you build a strong C++ foundation.

C++ High Performance - Björn Andrist
2020-12-30

A comprehensive guide to help aspiring and professional C++ developers elevate the performance of their apps by allowing them to run faster and consume fewer resources. Purchase of the print or Kindle book includes a free eBook in PDF format. Key Features Updated to C++20 with completely revised code and more content on error handling, benchmarking, memory allocators, and concurrent programming Explore the latest C++20 features including

concepts, ranges, and coroutines Utilize C++ constructs and techniques to carry out effective data structure optimization and memory management Book Description C++ High Performance, Second Edition guides you through optimizing the performance of your C++ apps. This allows them to run faster and consume fewer resources on the device they're running on without compromising the readability of your codebase. The book begins by introducing the C++ language and some of its modern concepts in brief. Once you are familiar with the fundamentals, you will be ready to measure, identify, and eradicate bottlenecks in your C++ codebase. By following this process, you will gradually improve your style of writing code. The book then explores data structure optimization, memory management, and how it can be used efficiently concerning CPU caches. After laying the foundation, the book trains you to leverage algorithms, ranges, and containers from the standard library to achieve faster execution, write readable code, and use customized iterators. It provides hands-on examples of C++ metaprogramming, coroutines, reflection to reduce boilerplate code, proxy objects to perform optimizations under the hood, concurrent programming, and lock-free data structures. The book concludes with an overview of parallel algorithms. By the end of this book, you will have the ability to use every tool as needed to boost the efficiency of your C++ projects. What you will learn Write specialized data structures for performance-critical code Use modern metaprogramming techniques to reduce runtime calculations Achieve efficient memory management using custom memory allocators Reduce boilerplate code using reflection techniques Reap the benefits of lock-free concurrent

programming Gain insights into subtle optimizations used by standard library algorithms Compose algorithms using ranges library Develop the ability to apply metaprogramming aspects such as `constexpr`, constraints, and concepts Implement lazy generators and asynchronous tasks using C++20 coroutines Who this book is for If you're a C++ developer looking to improve the efficiency of your code or just keen to upgrade your skills to the next level, this book is for you.

C++ Data Structures and Algorithms - Wisnu Anggoro 2018-04-26

Learn how to build efficient, secure and robust code in C++ by using data structures and algorithms - the building blocks of C++ Key Features Use data structures such as arrays, stacks, trees, lists, and graphs with real-world examples Learn the functional and reactive implementations of the traditional data structures Explore illustrations to present data structures and algorithms, as well as their analysis, in a clear, visual manner Book Description C++ is a general-purpose programming language which has evolved over the years and is used to develop software for many different sectors. This book will be your companion as it takes you through implementing classic data structures and algorithms to help you get up and running as a confident C++ programmer. We begin with an introduction to C++ data structures and algorithms while also covering essential language constructs. Next, we will see how to store data using linked lists, arrays, stacks, and queues. Then, we will learn how to implement different sorting algorithms, such as quick sort and heap sort. Along with these, we will dive into searching algorithms such as linear search, binary search and more. Our next mission will be to

attain high performance by implementing algorithms to string datatypes and implementing hash structures in algorithm design. We'll also analyze Brute Force algorithms, Greedy algorithms, and more. By the end of the book, you'll know how to build components that are easy to understand, debug, and use in different applications. What you will learn Know how to use arrays and lists to get better results in complex scenarios Build enhanced applications by using hashtables, dictionaries, and sets Implement searching algorithms such as linear search, binary search, jump search, exponential search, and more Have a positive impact on the efficiency of applications with tree traversal Explore the design used in sorting algorithms like Heap sort, Quick sort, Merge sort and Radix sort Implement various common algorithms in string data types Find out how to design an algorithm for a specific task using the common algorithm paradigms Who this book is for This book is for developers who would like to learn the Data Structures and Algorithms in C++. Basic C++ programming knowledge is expected.

Handbook of Open Source Tools -
Sandeep Koranne 2010-10-17
Handbook of Open Source Tools

introduces a comprehensive collection of advanced open source tools useful in developing software applications. The book contains information on more than 200 open-source tools which include software construction utilities for compilers, virtual-machines, database, graphics, high-performance computing, OpenGL, geometry, algebra, graph theory , GUIs and more. Special highlights for software construction utilities and application libraries are included. Each tool is covered in the context of a real like application development setting. This unique handbook presents a comprehensive discussion of advanced tools, a valuable asset used by most application developers and programmers; includes a special focus on Mathematical Open Source Software not available in most Open Source Software books, and introduces several tools (eg ACL2, CLIPS, CUDA, and COIN) which are not known outside of select groups, but are very powerful. Handbook of Open Source Tools is designed for application developers and programmers working with Open Source Tools. Advanced-level students concentrating on Engineering, Mathematics and Computer Science will find this reference a valuable asset as well.