

Modern C Programming Cookbook Ebook Now Just 5

A comprehensive guide with practical instructions for learning data structures, low-level programming, high-performance computing, networking and IoT to help you understand the latest standards in C programming such as C11 and C18 Key Features Tackle various challenges in C programming by making the most of its latest features Understand the workings of arrays, strings, functions, pointers, advanced data structures, and algorithms Become well-versed with process synchronization during multitasking and server-client process communication Book Description Used in everything from microcontrollers to operating systems, C is a popular programming language among developers because of its flexibility and versatility. This book helps you get hands-on with various tasks, covering the fundamental as well as complex C programming concepts that are essential for making real-life applications. You'll start with recipes for arrays, strings, user-defined functions, and pre-processing directives. Once you're familiar with the basic features, you'll gradually move on to learning pointers, file handling, concurrency, networking, and inter-process communication (IPC). The book then illustrates how to carry out searching and arrange data using different sorting techniques, before demonstrating the implementation of data structures such as stacks and queues. Later, you'll learn interesting programming features such as using graphics for drawing and animation, and the application of general-purpose utilities. Finally, the book will take you through advanced concepts such as low-level programming, embedded software, IoT, and security in coding, as well as techniques for improving code performance. By the end of this book, you'll have a clear understanding of C programming, and have the skills you need to develop robust apps. What you will learn Discover how to use arrays, functions, and strings to make large applications Perform preprocessing and conditional compilation for efficient programming Understand how to use pointers and memory optimally Use general-purpose utilities and improve code performance Implement multitasking using threads and process synchronization Use low-level programming and the inline assembly language Understand how to use graphics for animation Get to grips with applying security while developing C programs Who this book is for This intermediate-level book is for developers who want to become better C programmers by learning its modern features and programming practices. Familiarity with C programming is assumed to get the most out of this book.

Presents a collection of reusable design artifacts, called generic components, together with the techniques that make them possible. The author describes techniques for policy-based design, partial template specialization, typelists, and local classes, then goes on to implement generic components for smart pointers, object factories, functor objects, the Visitor design pattern, and multimethod engines. c. Book News Inc.

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

Take your C++ coding to the next level by leveraging the latest features and advanced techniques to building high performing, reliable applications. About This Book Get acquainted with the latest features in C++ 17 Take advantage of the myriad of features and possibilities that C++ offers to build real-world applications Write clear and expressive code in C++, and get insights into how to keep your code error-free Who This Book Is For This book is for experienced C++ developers. If you are a novice C++ developer, then it's highly recommended that you get a solid understanding of the C++ language before reading this book What You Will Learn Write modular C++ applications in terms of the existing and newly introduced features Identify code-smells, clean up, and refactor legacy C++ applications Leverage the possibilities provided by Cucumber and Google Test/Mock

to automate test cases Test frameworks with C++ Get acquainted with the new C++17 features Develop GUI applications in C++ Build portable cross-platform applications using standard C++ features In Detail C++ has come a long way and has now been adopted in several contexts. Its key strengths are its software infrastructure and resource-constrained applications. The C++ 17 release will change the way developers write code, and this book will help you master your developing skills with C++. With real-world, practical examples explaining each concept, the book will begin by introducing you to the latest features in C++ 17. It encourages clean code practices in C++ in general, and demonstrates the GUI app-development options in C++. You'll get tips on avoiding memory leaks using smart-pointers. Next, you'll see how multi-threaded programming can help you achieve concurrency in your applications. Moving on, you'll get an in-depth understanding of the C++ Standard Template Library. We show you the concepts of implementing TDD and BDD in your C++ programs, and explore template-based generic programming, giving you the expertise to build powerful applications. Finally, we'll round up with debugging techniques and best practices. By the end of the book, you'll have an in-depth understanding of the language and its various facets. Style and approach This straightforward guide will help you level up your skills in C++ programming, be it for enterprise software or for low-latency applications like games. Filled with real-world, practical examples, this book will take you gradually up the steep learning curve that is C++.

A recipe-based guide to refining your C++ programming skills with the help of coding best practices, advanced programming concepts, and the latest features of C++17 and C++20 Key Features Learn how to develop and design your own libraries Find solutions to your app development problems and implement them in a highly reusable manner, following library development best practices Explore advanced C++ features such as containers, coroutines, and modules Book Description If you think you've mastered C++ and know everything it takes to write robust applications, you'll be in for a surprise. With this book, you'll gain comprehensive insights into C++, covering exclusive tips and interesting techniques to enhance your app development process. You'll kick off with the basic principles of library design and development, which will help you understand how to write reusable and maintainable code. You'll then discover the importance of exception safety, and how you can avoid unexpected errors or bugs in your code. The book will take you through the modern elements of C++, such as move semantics, type deductions, and coroutines. As you advance, you'll delve into template programming - the standard tool for most library developers looking to achieve high code reusability. You'll explore the STL and learn how to avoid common pitfalls while implementing templates. Later, you'll learn about the problems of multithreaded programming such as data races, deadlocks, and thread starvation. You'll also learn high-performance programming by using benchmarking tools and libraries. Finally, you'll discover advanced techniques for debugging and testing to ensure code reliability. By the end of this book, you'll have become an expert at C++ programming and will have gained the skills to solve complex development problems with ease. What you will learn Solve common C++ development problems by implementing solutions in a more generic and reusable way Achieve different levels of exception safety guarantees by introducing precise declarations Write library-quality code that meets professional standards Practice writing reliable, performant code that exposes consistent behavior in programs Understand why you need to implement design patterns and how it's done Work with complex examples to understand various aspects of good library design Who this book is for This book is for intermediate and expert-level C++ developers who are looking to explore the lesser known functionalities of the language to improve the efficiency of their code and the way they develop applications. Basic knowledge of object-oriented programming concepts and the Standard Template Library (STL) is assumed.

C Programming Cookbook

Extreme C

A Friendly Introduction to C+ Language and C+11 to C+20 Standards

Hands-On System Programming with C++

Design modern systems using effective architecture concepts, design patterns, and techniques with C++20

Modern C++ Design

Build performant and concurrent Unix and Linux systems with C++17

Get started with writing simple programs in C while learning the skills that will help you work with practically any programming language

Key Features Learn essential C concepts such as variables, data structures, functions, loops, and pointers Get to grips with the core programming aspects that form the base of many modern programming languages Explore the expressiveness and versatility of the C language with the help of sample programs

Book Description C is a powerful general-purpose programming language that is excellent for beginners to learn. This book will introduce you to computer programming and software development using C. If you're an experienced developer, this book will help you to become familiar with the C programming language. This C programming book takes you through basic programming concepts and shows you how to implement them in C. Throughout the book, you'll create and run programs that make use of one or more C concepts, such as program structure with functions, data types, and conditional statements. You'll also see how to use looping and iteration, arrays, pointers, and strings. As you make progress, you'll cover code documentation, testing and validation methods, basic input/output, and how to write complete programs in C. By the end of the book, you'll have developed basic programming skills in C, that you can apply to other programming languages and will develop a solid foundation for you to advance as a programmer. What you will learn

Understand fundamental programming concepts and implement them in C

Write working programs with an emphasis on code indentation and readability

Break existing programs intentionally and learn how to debug code

Adopt good coding practices and develop a clean coding style

Explore general programming concepts that are applicable to more advanced projects

Discover how you can use building blocks to make more complex and interesting programs

Use C Standard Library functions and understand why doing this is desirable

Who this book is for This book is written for two very diverse audiences. If you're an absolute beginner who only has basic familiarity with operating a computer, this book will help you learn the most fundamental concepts and practices you need to know to become a successful C programmer. If you're an experienced programmer, you'll find the full range of C syntax as well as common C idioms. You can skim through the explanations and focus primarily on the source code provided.

Get started with C# and strengthen your knowledge of core programming concepts such as procedural, object-oriented, generic, functional, and asynchronous programming along with the latest features of C# 8

Key Features Learn the fundamentals of C# with the help of easy-to-follow examples and explanations Leverage the latest features of C# 8, including nullable reference types, pattern matching enhancements, and asynchronous streams

Explore object-oriented programming, functional programming, and multithreading concepts

Book Description The C# programming language is often developers' primary choice for creating a wide range of applications for desktop, cloud, and mobile. In nearly two decades of its existence, C# has evolved from a general-purpose, object-oriented language to a multi-paradigm language with impressive features. This book will take you through C# from the ground up in a step-by-step manner. You'll start with the building blocks of C#, which include basic data types, variables, strings, arrays, operators, control statements, and loops. Once comfortable with the basics, you'll then progress to learning object-oriented programming concepts such as classes and structures, objects, interfaces, and abstraction. Generics, functional programming, dynamic,

and asynchronous programming are covered in detail. This book also takes you through regular expressions, reflection, memory management, pattern matching, exceptions, and many other advanced topics. As you advance, you'll explore the .NET Core 3 framework and learn how to use the dotnet command-line interface (CLI), consume NuGet packages, develop for Linux, and migrate apps built with .NET Framework. Finally, you'll understand how to run unit tests with the Microsoft unit testing frameworks available in Visual Studio. By the end of this book, you'll be well-versed with the essentials of the C# language and be ready to start creating apps with it. What you will learn

- Get to grips with all the new features of C# 8
- Discover how to use attributes and reflection to build extendable applications
- Utilize LINQ to uniformly query various sources of data
- Use files and streams and serialize data to JSON and XML
- Write asynchronous code with the async-await pattern
- Employ .NET Core tools to create, compile, and publish your applications
- Create unit tests with Visual Studio and the Microsoft unit testing frameworks

Who this book is for If you have little experience in coding or C# and want to learn the essentials of C# programming to develop powerful programming techniques, this book is for you. It will also help aspiring programmers to write scripts or programs to accomplish specific tasks.

Push the limits of what C - and you - can do, with this high-intensity guide to the most advanced capabilities of C

- Key Features
- Make the most of C's low-level control, flexibility, and high performance
- A comprehensive guide to C's most powerful and challenging features
- A thought-provoking guide packed with hands-on exercises and examples

Book Description There's a lot more to C than knowing the language syntax. The industry looks for developers with a rigorous, scientific understanding of the principles and practices. Extreme C will teach you to use C's advanced low-level power to write effective, efficient systems. This intensive, practical guide will help you become an expert C programmer. Building on your existing C knowledge, you will master preprocessor directives, macros, conditional compilation, pointers, and much more. You will gain new insight into algorithm design, functions, and structures. You will discover how C helps you squeeze maximum performance out of critical, resource-constrained applications. C still plays a critical role in 21st-century programming, remaining the core language for precision engineering, aviations, space research, and more. This book shows how C works with Unix, how to implement OO principles in C, and fully covers multi-processing. In Extreme C, Amini encourages you to think, question, apply, and experiment for yourself. The book is essential for anybody who wants to take their C to the next level. What you will learn

- Build advanced C knowledge on strong foundations, rooted in first principles
- Understand memory structures and compilation pipeline and how they work, and how to make most out of them
- Apply object-oriented design principles to your procedural C code
- Write low-level code that's close to the hardware and squeezes maximum performance out of a computer system
- Master concurrency, multithreading, multi-processing, and integration with other languages
- Unit Testing and debugging, build systems, and inter-process communication for C programming

Who this book is for Extreme C is for C programmers who want to dig deep into the language and its capabilities. It will help you make the most of the low-level control C gives you.

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.

As scientific and engineering projects grow larger and more complex, it is increasingly likely that those projects will be written in C++. With embedded hardware growing more powerful, much of its software is moving to C++, too. Mastering C++ gives you strong skills for programming at nearly every level, from "close to the hardware" to the highest-level abstractions. In short, C++ is a language that scientific and technical practitioners need to know. Peter Gottschling's *Discovering Modern C++* is an intensive introduction that guides you smoothly to sophisticated approaches based on advanced features. Gottschling introduces key concepts using examples from many technical problem domains, drawing on his extensive experience training professionals and teaching C++ to students of physics, math, and engineering. This book is designed to help you get started rapidly and then master increasingly robust features, from lambdas to expression templates. You'll also learn how to take advantage of the powerful libraries available to C++ programmers: both the Standard Template Library (STL) and scientific libraries for arithmetic, linear algebra, differential equations, and graphs. Throughout, Gottschling demonstrates how to write clear and expressive software using object orientation, generics, metaprogramming, and procedural techniques. By the time you're finished, you'll have mastered all the abstractions you need to write C++ programs with exceptional quality and performance.

Learn socket programming in C and write secure and optimized network code

Discovering Modern C++

C++ System Programming Cookbook

Learn C Programming

Beginning C++ Programming

The The Modern C++ Challenge

Expert C Programming

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.

The author of Midnight's Children and The Satanic Verses describes his 1986 trip to Nicaragua and shares his impressions of the true Nicaragua--the people, politics, land, poetry, and problems behind the headlines. Reprint.

Software -- Programming Languages.

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 FeaturesDesign professional-grade, maintainable apps by learning advanced concepts such as functional programming, templates, and networkingApply design patterns and best practices to solve real-world problemsImprove the performance of your projects by designing concurrent data structures and algorithmsBook 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.

Modern C++ at your fingertips! About This Book This book gets you started with the exciting world of C++ programming. It will enable you to write C++ code that uses the standard library, has a level of object orientation, and uses memory in a safe and effective way. It forms the basis of programming and covers concepts such as data structures and the core programming language.

Who This Book Is For A computer, an internet connection, and the desire to learn how to code in C++ is all you need to get started with this book.

What You Will Learn Get familiar with the structure of C++ projects. Identify the main structures in the language: functions and classes. Feel confident about being able to identify the execution flow through the code. Be aware of the facilities of the standard library. Gain insights into the basic concepts of object orientation. Know how to debug your programs. Get acquainted with the standard C++ library.

In Detail C++ has come a long way and is now adopted in several contexts. Its key strengths are its software infrastructure and resource-constrained applications, including desktop applications, servers, and performance-critical applications, not to forget its importance in game programming. Despite its strengths in these areas, beginners usually tend to shy away from learning the language because of its steep learning curve. The main mission of this book is to make you familiar and comfortable with C++. You will finish the book not only being able to write your own code, but more importantly, you will be able to read other projects. It is only by being able to read others' code that you will progress from a beginner to an advanced programmer. This book is the first step in that progression. The first task is to familiarize you with the structure of C++ projects so you will know how to start reading a project. Next, you will be able to identify the main structures in the language, functions, and classes, and feel confident being able to identify the execution flow through the code. You will then become aware of the facilities of the standard library and be able to determine whether you need to write a routine yourself, or use an existing routine in the standard library. Throughout the book, there is a big emphasis on memory and pointers. You will understand memory usage, allocation, and access, and be able to write code that does not leak memory. Finally, you will learn about C++ classes and get an introduction to object orientation and polymorphism.

Style and approach This straightforward tutorial will help you build strong skills in C++ programming, be it for enterprise software or for

low-latency applications such as games or embedded programming. Filled with examples, this book will take you gradually up the steep learning curve of C++.

Modern C++ Programming Cookbook - Second Edition

Effective Modern C++

A Guide to System Programming

Mastering C++ Programming

C++ Cookbook

Modern C++

Professional C++

This book presents an introduction to the C programming language, featuring a structured approach and aimed at professionals and students with some experience of high-level languages. Features *includes embedded summary material in bulleted form *highlights common traps and pitfalls in C programming.

A comprehensive guide with curated recipes to help you gain a deeper understanding of modern C. Key Features Learn how to make your applications swift and robust by leveraging powerful features of C Understand the workings of arrays, strings, functions, and more down to how they operate in memory Master process synchronization during multi-tasking and server-client process communication Book Description C is a high-level language that's popular among developers. It enables you to write drivers for different devices, access machine-level hardware, apply dynamic memory allocation, and much more. With self-contained tutorials, known as recipes, this book will guide you in dealing with C and its idiosyncrasies and help you benefit from its latest features. Beginning with common tasks, each recipe addresses a specific problem followed by explaining the solution to get you acquainted with what goes on under the hood. You will explore core concepts of the programming language, including how to work with strings, pointers, and single and multi-dimensional arrays. You will also learn how to break a large application into small modules by creating functions, handling files, and using a database. Finally, the book will take you through advanced concepts such as concurrency and interprocess communication. By the end of this book, you'll have a clear understanding and deeper knowledge of C programming, which will help you become a better developer. What you will learn Manipulate single and multi-dimensional arrays Perform complex operations on strings Understand how to use pointers and memory optimally Discover how to use arrays, functions, and strings to make large applications Implement multitasking using threads and process synchronization Establish communication between two or more processes using different techniques Store simple text in files and store data in a database Who this book is for If you're a programmer with basic experience in C and want to leverage its features through modern programming practices, then this book is for you.

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.

Create apps in C++ and leverage its latest features using modern programming techniques. Key Features

- Develop strong C++ skills to build a variety of applications
- Explore features of C++17, such as containers, algorithms, and threads
- Grasp the standard support for threading and concurrency and use them in basic daily tasks

Book Description C++ is one of the most widely used programming languages. It is fast, flexible, and used to solve many programming problems. This Learning Path gives you an in-depth and hands-on experience of working with C++, using the latest recipes and understanding most recent developments. You will explore C++ programming constructs by learning about language structures, functions, and classes, which will help you identify the execution flow through code. You will also understand the importance of the C++ standard library as well as memory allocation for writing better and faster programs. Modern C++: Efficient and Scalable Application Development deals with the challenges faced with advanced C++ programming. You will work through advanced topics such as multithreading, networking, concurrency, lambda expressions, and many more recipes. By the end of this Learning Path, you will have all the skills to become a master C++ programmer. This Learning Path includes content from the following Packt products: Beginning C++ Programming by Richard Grimes Modern C++ Programming Cookbook by Marius Bancila The Modern C++ Challenge by Marius Bancila

What you will learn

- Become familiar with the structure of C++ projects
- Identify the main structures in the language: functions and classes
- Learn to debug your programs
- Leverage C++ features to obtain increased robustness and performance
- Explore functions and callable objects with a focus on modern features
- Serialize and deserialize JSON and XML data
- Create client-server applications that communicate over TCP/IP
- Use design patterns to solve real-world problems

Who this book is for This Learning Path is designed for developers who want to gain a solid foundation in C++. The desire to learn

how to code in C++ is all you need to get started with this Learning Path

Discover the newest major features of C++20, including modules, concepts, spaceship operators, and smart pointers. This book is a handy code cookbook reference guide that covers the C++ core language standard as well as some of the code templates available in standard template library (STL). In C++20 Recipes: A Problem-Solution Approach, you'll find numbers, strings, dates, times, classes, exceptions, streams, flows, pointers, and more. Also, you'll see various code samples, templates for C++ algorithms, parallel processing, multithreading, and numerical processes. It also includes 3D graphics programming code. A wealth of STL templates on function objects, adapters, allocators, and extensions are also available. This is a must-have, contemporary reference for your technical library to help with just about any project that involves the C++ programming language. What You Will Learn See what's new in C++20 Write modules Work with text, numbers, and classes Use the containers and algorithms available in the standard library Work with templates, memory, concurrency, networking, scripting, and more Code for 3D graphics Who This Book Is For Programmers with at least some prior experience with C++.

Hands-On Design Patterns with C++

Software Architecture with C++

The Jaguar Smile

Generic Programming and Design Patterns Applied

Solutions for modern C developers to create efficient and well-structured programs

MASTERING ALGORITHMS WITH C. Avec une disquette

Write high-level abstractions while retaining full control of the hardware, performances, and maintainability. Key Features Transform your ideas into modern C++ code, with both C++11 and C++17 Explore best practices for creating high-performance solutions Understand C++ basics and work with concrete real-world examples Book Description C++ Fundamentals begins by introducing you to the C++ syntax. You will study the semantics of variables along with their advantages and trade-offs, and see how they can be best used to write safe and efficient code. With the help of this book, you'll be able to compile fully working C++ programs and understand how variables, references, and pointers can be used to manipulate the state of the program. Then you'll explore functions and classes — the features that C++ offers to organize a program — and use them to solve more complex problems such as functions and classes. You'll also understand common pitfalls and modern best practices, especially the ones that diverge from the C++98 guideline. As you advance through the chapters, you'll study the advantages of generic programming and write your own templates to make generic algorithms that work with any type. This C++ book will guide you in fully exploiting standard containers and understanding how to pick the appropriate container for each problem. You will even work with a variety of memory management tools in C++. By the end of this book, you will not only be able to write efficient code, but also be equipped to improve the readability, performance, and maintainability of your programs using standard algorithms. What you will learn Work with the C++ compilation model and syntaxes Apply best practices for writing functions and classes Write safe, generic, and efficient code with templates Explore the containers that C++ standard offers Discover the new paradigms introduced with C++11, C++14, and C++17 Get to grips with the core language features of C++ Abstract complex problems using object-oriented programming in C++ Who this book is for If you're a developer looking to learn a new powerful language or are familiar with C++ but want to update your knowledge with modern paradigms of C++11, C++14, and C++17, this book is for you. To easily understand the concepts in the book, you must be familiar with the basics of programming. A comprehensive guide to programming with network sockets, implementing Internet protocols,

designing IoT devices, and much more with C Key FeaturesLeverage your C or C++ programming skills to build powerful network applicationsGet to grips with a variety of network protocols that allow you to load web pages, send emails, and do much moreWrite portable network code for operating systems such as Windows, Linux, and macOSBook 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 learnUncover cross-platform socket programming APIsImplement techniques for supporting IPv4 and IPv6Understand how TCP and UDP connections work over IPDiscover how hostname resolution and DNS workInterface with web APIs using HTTP and HTTPSAcquire 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. A hands-on guide to making system programming with C++ easy Key FeaturesWrite system-level code leveraging C++17Learn the internals of the Linux Application Binary Interface (ABI) and apply it to system programmingExplore C++ concurrency to take advantage of server-level constructsBook Description C++ is a general-purpose programming language with a bias toward system programming as it provides ready access to hardware-level resources, efficient compilation, and a versatile approach to higher-level abstractions. This book will help you understand the benefits of system programming with C++17. You will gain a firm understanding of various C, C++, and POSIX standards, as well as their respective system types for both C++ and POSIX. After a brief refresher on C++, Resource Acquisition Is Initialization (RAII), and the new C++ Guideline Support Library (GSL), you will learn to program Linux and Unix systems along with process management. As you progress through the chapters, you will become acquainted with C++'s support for IO. You will then study various memory management methods, including a chapter on allocators and how they benefit system programming. You will also explore how to program file input and output and learn about POSIX sockets. This book will help you get to grips with safely setting up a UDP and TCP server/client. Finally, you will be guided through Unix time interfaces, multithreading, and error handling with C++ exceptions. By the end of this book, you will be comfortable with using C++ to program high-quality systems. What you will learnUnderstand the benefits of using C++ for system programmingProgram Linux/Unix systems using C++Discover the advantages of Resource Acquisition Is Initialization (RAII)Program both console and file input and outputUncover the POSIX socket APIs and understand how to program themExplore advanced system programming topics, such as C++ allocatorsUse POSIX and C++ threads to program concurrent systemsGrasp how C++ can be used to create performant system applicationsWho this book is for If you are a fresh developer with intermediate knowledge of C++ but little or no knowledge of Unix and Linux system programming, this book will help you learn system programming with C++ in a practical way.

Apply business requirements to IT infrastructure and deliver a high-quality product by understanding architectures such as microservices, DevOps, and cloud-native using modern C++ standards and features

Key Features Design scalable large-scale applications with the C++ programming language Architect software solutions in a cloud-based environment with continuous integration and continuous delivery (CI/CD) Achieve architectural goals by leveraging design patterns, language features, and useful tools

Book Description Software architecture refers to the high-level design of complex applications. It is evolving just like the languages we use, but there are architectural concepts and patterns that you can learn to write high-performance apps in a high-level language without sacrificing readability and maintainability. If you're working with modern C++, this practical guide will help you put your knowledge to work and design distributed, large-scale apps. You'll start by getting up to speed with architectural concepts, including established patterns and rising trends, then move on to understanding what software architecture actually is and start exploring its components. Next, you'll discover the design concepts involved in application architecture and the patterns in software development, before going on to learn how to build, package, integrate, and deploy your components. In the concluding chapters, you'll explore different architectural qualities, such as maintainability, reusability, testability, performance, scalability, and security. Finally, you will get an overview of distributed systems, such as service-oriented architecture, microservices, and cloud-native, and understand how to apply them in application development. By the end of this book, you'll be able to build distributed services using modern C++ and associated tools to deliver solutions as per your clients' requirements. What you will learn

Understand how to apply the principles of software architecture

Apply design patterns and best practices to meet your architectural goals

Write elegant, safe, and performant code using the latest C++ features

Build applications that are easy to maintain and deploy

Explore the different architectural approaches and learn to apply them as per your requirement

Simplify development and operations using application containers

Discover various techniques to solve common problems in software design and development

Who this book is for This software architecture C++ programming book is for experienced C++ developers looking to become software architects or develop enterprise-grade applications.

A problem-solution-based guide to help you overcome hurdles effectively while working with kernel APIs, filesystems, networks, threads, and process communications

Key Features Learn to apply the latest C++ features (from C++11, 14, 17, and 20) to facilitate systems programming

Create robust and concurrent systems that make the most of the available hardware resources

Delve into C++ inbuilt libraries and frameworks to design robust systems as per your business needs

Book Description C++ is the preferred language for system programming due to its efficient low-level computation, data abstraction, and object-oriented features. System programming is about designing and writing computer programs that interact closely with the underlying operating system and allow computer hardware to interface with the programmer and the user. The C++ System Programming Cookbook will serve as a reference for developers who want to have ready-to-use solutions for the essential aspects of system programming using the latest C++ standards wherever possible. This C++ book starts out by giving you an overview of system programming and refreshing your C++ knowledge. Moving ahead, you will learn how to deal with threads and processes, before going on to discover recipes for how to manage memory. The concluding chapters will then help you understand how processes communicate and how to interact with the console (console I/O). Finally, you will learn how to deal with time interfaces, signals, and CPU scheduling. By the end of the book, you will become adept at developing robust systems applications using C++. What you will learn

Get up to speed with the fundamentals including makefile, man pages, compilation, and linking and debugging

Understand how to deal with time interfaces, signals, and CPU scheduling

Develop your knowledge of memory management

Use processes and threads for advanced synchronizations (mutexes and condition variables)

Understand interprocess communications (IPC): pipes, FIFOs, message queues, shared

memory, and TCP and UDPDiscover how to interact with the console (console I/O)Who this book is for This book is for C++ developers who want to gain practical knowledge of systems programming. Though no experience of Linux system programming is assumed, intermediate knowledge of C++ is necessary.

A Problem-Solution Approach

A Nicaraguan Journey

Master C++ core language and standard library features, with over 100 recipes, updated to C++20, 2nd Edition

Hands-On Network Programming with C

Using C on the UNIX System

C++20 Recipes

Become a proficient programmer by learning coding best practices with C++17 and C++20's latest features

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

Key FeaturesUpdated to C++20 with completely revised code and more content on error handling, benchmarking, memory allocators, and concurrent programmingExplore the latest C++20 features including concepts, ranges, and coroutinesUtilize 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 codeUse modern metaprogramming techniques to reduce runtime calculationsAchieve efficient memory management using custom memory allocatorsReduce boilerplate code using reflection techniquesReap the benefits of lock-free concurrent programmingGain insights into subtle optimizations used by standard library algorithmsCompose algorithms using ranges libraryDevelop the ability to apply metaprogramming aspects such as constexpr, constraints, and conceptsImplement 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.

Modern C++ Programming Cookbook - Second Edition

Learn the basics of the modern C++ programming language from scratch, including the

C++11 to C++20 standards, no experience necessary. You'll work with expressions and statements, variables, libraries, arguments, classes, functions, memory handling, and much more. Each section is filled with real-world examples and advice on how to avoid common mistakes. Modern C++ for Absolute Beginners will teach you more than just programming in C++20. It will provide you with a set of C++ skills, which will serve you if you ever decide to deepen your knowledge in C++, computer science, or learn more about advanced C++ techniques. The author will take you through the C++ programming language, the Standard Library, and the C++11 to C++20 standard basics. Each chapter is accompanied by the right amount of theory and plenty of source code examples. You will work with C++20 features and standards, yet you will also compare and take a look into previous versions of C++. You will do so with plenty of examples and real code writing to gain an even better level of understanding. What You Will Learn Use the basics of C++: types, operators, variables, constants, expressions, references, functions, classes, I/O, smart pointers, polymorphism, and more Set up the Visual Studio development environment where you can write your own code Declare and define functions, classes, and objects Discover object-oriented programming: classes and objects, encapsulation, inheritance, polymorphism, and more using the most advanced C++ features Employ best practices in organizing source code, controlling program workflow, C++ language dos and don'ts, and more Program using lambda, modules, inheritance, polymorphism, smart pointers, templates, contracts, STL, concepts, and exceptions Who This Book Is For Beginner or novice programmers who wish to learn C++ programming. No prior programming experience is required. 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 ...

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

A Book on C

Efficient and Scalable Application Development

Make full use of the standard library components in C++17

Mastering the C++17 STL

Hit the ground running with C++, the language that supports tech giants globally

Modern C++ for Absolute Beginners

Practical C++ Programming

Designed for the way many developers work, this practical problem-solving guide balances the need for rapid development with a trusted source of information.

Practical solutions to overcome challenges in creating console and web applications and working with systems-level and embedded code, network programming, deep neural networks, and much more. Key Features Work through recipes featuring advanced concepts such as concurrency, unsafe code, and macros to migrate your codebase to the Rust programming language Learn how to run machine learning models with Rust Explore error handling, macros, and modularization to write maintainable code Book Description Rust 2018, Rust's first major milestone since version 1.0, brings more advancement in the Rust language. The Rust Programming Cookbook is a practical guide to help you overcome challenges when writing Rust code. This Rust book covers recipes for configuring Rust for different environments and architectural designs, and provides solutions to practical problems. It will also take you through Rust's core concepts, enabling you to create efficient, high-performance applications that use features such as zero-cost

abstractions and improved memory management. As you progress, you'll delve into more advanced topics, including channels and actors, for building scalable, production-grade applications, and even get to grips with error handling, macros, and modularization to write maintainable code. You will then learn how to overcome common roadblocks when using Rust for systems programming, IoT, web development, and network programming. Finally, you'll discover what Rust 2018 has to offer for embedded programmers. By the end of the book, you'll have learned how to build fast and safe applications and services using Rust. What you will learn

Understand how Rust provides unique solutions to solve system programming language problems
Grasp the core concepts of Rust to develop fast and safe applications
Explore the possibility of integrating Rust units into existing applications for improved efficiency
Discover how to achieve better parallelism and security with Rust
Write Python extensions in Rust
Compile external assembly files and use the Foreign Function Interface (FFI)
Build web applications and services using Rust for high performance

Who this book is for The Rust cookbook is for software developers looking to enhance their knowledge of Rust and leverage its features using modern programming practices. Familiarity with Rust language is expected to get the most out of this book.

A comprehensive guide with extensive coverage on concepts such as OOP, functional programming, generic programming, and STL along with the latest features of C++

Key Features Delve into the core patterns and components of C++ in order to master application design
Learn tricks, techniques, and best practices to solve common design and architectural challenges
Understand the limitation imposed by C++ and how to solve them using design patterns

Book Description C++ is a general-purpose programming language designed with the goals of efficiency, performance, and flexibility in mind. Design patterns are commonly accepted solutions to well-recognized design problems. In essence, they are a library of reusable components, only for software architecture, and not for a concrete implementation. The focus of this book is on the design patterns that naturally lend themselves to the needs of a C++ programmer, and on the patterns that uniquely benefit from the features of C++, in particular, the generic programming. Armed with the knowledge of these patterns, you will spend less time searching for a solution to a common problem and be familiar with the solutions developed from experience, as well as their advantages and drawbacks. The other use of design patterns is as a concise and an efficient way to communicate. A pattern is a familiar and instantly recognizable solution to specific problem; through its use, sometimes with a single line of code, we can convey a considerable amount of information. The code conveys: "This is the problem we are facing, these are additional considerations that are most important in our case; hence, the following well-known solution was chosen." By the end of this book, you will have gained a comprehensive understanding of design patterns to create robust, reusable, and maintainable code. What you will learn

Recognize the most common design patterns used in C++
Understand how to use C++ generic programming to solve common design problems
Explore the most powerful C++ idioms, their strengths, and drawbacks
Rediscover how to use popular C++ idioms with generic programming
Understand the impact of design patterns on the program's performance

Who this book is for This book is for experienced C++ developers and programmers who wish to learn about software design patterns and principles and apply them to create robust, reusable, and easily maintainable apps.

This book is a collection of practical examples for understanding how embedded development is different from other desktop application development. You'll learn to build an embedded application and use specialized memory and custom allocators. By the end of the book, you'll be able to build robust and secure embedded applications with C++20.

The authors provide clear examples and thorough explanations of every feature in the C language. They teach C vis-a-vis the UNIX operating system. A reference and tutorial to the C programming language.

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

An Intensive Course for Scientists, Engineers, and Programmers

Practical Recipes to Help You Build Robust and Secure Embedded Applications on Linux

Learn C# Programming

Deep C Secrets

Practical recipes for Linux system-level programming using the latest C++ features

Explore the latest features of Rust 2018 for building fast and secure apps

Expert C++

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

Over 100 recipes to help you overcome your difficulties with C++ programming and gain a deeper understanding of the working of modern C++

About This Book 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. 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. 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 In Detail 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.

Style and approach This book follows a recipe-based approach, with examples that will empower you to implement the core programming language features and explore the newer aspects of C++.

Geared to experienced C++ developers who may not be familiar with the more advanced features of the language, and therefore are not using it to its full capabilities Teaches programmers how to think in C++-that is, how to design effective solutions that maximize the power of the language The authors drill down into this notoriously complex language, explaining poorly understood elements of the C++ feature set as well as common pitfalls to avoid Contains several in-depth case studies with working code that's been tested on Windows, Linux, and Solaris platforms

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

For intermediate to experienced C programmers who want to become UNIX system programmers. Explains system calls and special library routines available on the system. Annotation copyrighted by Book News, Inc., Portland, OR

C++17 STL Cookbook

42 Specific Ways to Improve Your Use of C++11 and C++14

Practical C Programming

Become an expert programmer by solving real-world problems

The Bulgarian C# Book

Modern C++ Programming Cookbook

Taking you to the limit in Concurrency, OOP, and the most advanced

capabilities of C

Over 90 recipes that leverage the powerful features of the Standard Library in C++17 About This Book Learn the latest features of C++ and how to write better code by using the Standard Library (STL). Reduce the development time for your applications. Understand the scope and power of STL features to deal with real-world problems. Compose your own algorithms without forfeiting the simplicity and elegance of the STL way. Who This Book Is For This book is for intermediate-to-advanced C++ programmers who want to get the most out of the Standard Template Library of the newest version of C++: C++ 17. What You Will Learn Learn about the new core language features and the problems they were intended to solve Understand the inner workings and requirements of iterators by implementing them Explore algorithms, functional programming style, and lambda expressions Leverage the rich, portable, fast, and well-tested set of well-designed algorithms provided in the STL Work with strings the STL way instead of handcrafting C-style code Understand standard support classes for concurrency and synchronization, and how to put them to work Use the filesystem library addition available with the C++17 STL In Detail C++ has come a long way and is in use in every area of the industry. Fast, efficient, and flexible, it is used to solve many problems. The upcoming version of C++ will see programmers change the way they code. If you want to grasp the practical usefulness of the C++17 STL in order to write smarter, fully portable code, then this book is for you. Beginning with new language features, this book will help you understand the language's mechanics and library features, and offers insight into how they work. Unlike other books, ours takes an implementation-specific, problem-solution approach that will help you quickly overcome hurdles. You will learn the core STL concepts, such as containers, algorithms, utility classes, lambda expressions, iterators, and more, while working on practical real-world recipes. These recipes will help you get the most from the STL and show you how to program in a better way. By the end of the book, you will be up to date with the latest C++17 features and save time and effort while solving tasks elegantly using the STL. Style and approach This recipe-based guide will show you how to make the best use of C++ together with the STL to squeeze more out of the standard language

C++ Fundamentals

A beginner's guide to learning C programming the easy and disciplined way

Become an expert C++ programmer by mastering concepts like templates, concurrency, and type deduction

Programming in C

Solve common C++ problems with modern design patterns and build robust applications

Advanced C++ Programming Cookbook
The C Book, Featuring the ANSI C Standard