

---

# Efficient C Performance Programming Techniques

---

The Art of Writing Efficient Programs  
Accuracy and Reliability in Scientific Computing  
The The C++ Workshop  
Learning R Programming  
Sams Teach Yourself C++ in One Hour a Day,  
Seventh Edition  
Writing Efficient C Code  
WebAssembly Essentials  
Effective C#  
Efficient C++  
C++  
Software Architecture with C++  
Expert C++  
Practical FPGA Programming in C  
Pro C++ Performance Optimization  
Effektives modernes C++  
Computer Literature Bibliography: 1946-1963  
PC Mag  
Practical C++ Financial Programming  
Professional C++  
Computer Architecture Techniques for Power-  
efficiency

Ray Tracing from the Ground Up  
Real-Time C++  
Optimized C++  
Eine Tour durch C++  
Writing Efficient C Code  
Monte Carlo Frameworks  
Java Performance and Scalability: Server-side  
programming techniques  
Efficient C/C++ Programming  
Modern C++  
Hands-On High Performance Programming with  
Qt 5  
Programming Pearls  
Sams Teach Yourself C++ in One Hour a Day  
The C++ Report  
Code Optimization: Effective Memory Usage  
Modern C++ Programming Cookbook  
Professional C++  
C++ High Performance  
C++ High Performance  
Effective TCP/IP Programming

Efficient C  
Performance  
Programming  
Techniques Downloaded from  
ccobankspyservices.ecobank.com  
by guest

---

**CAROLYN  
BAILEE**

---

**The Art of  
Writing  
Efficient  
Programs**

Addison-  
Wesley

Professional  
Build efficient  
and fast Qt  
applications,  
target  
performance  
problems, and  
discover  
solutions to  
refine your

code Key  
FeaturesBuild  
efficient and  
concurrent  
applications in  
Qt to create  
cross-platform  
applicationsId  
entify  
performance

bottlenecks and apply the correct algorithm to improve application performance. Delve into parallel programming and memory management to optimize your code. Book Description: Achieving efficient code through performance tuning is one of the key challenges faced by many programmers. This book looks at Qt programming from a performance perspective. You'll explore

the performance problems encountered when using the Qt framework and means and ways to resolve them and optimize performance. The book highlights performance improvements and new features released in Qt 5.9, Qt 5.11, and 5.12 (LTE). You'll master general computer performance best practices and tools, which can help you identify the reasons

behind low performance, and the most common performance pitfalls experienced when using the Qt framework. In the following chapters, you'll explore multithreading and asynchronous programming with C++ and Qt and learn the importance and efficient use of data structures. You'll also get the opportunity to work through techniques such as memory management

and design guidelines, which are essential to improve application performance. Comprehensive sections that cover all these concepts will prepare you for gaining hands-on experience of some of Qt's most exciting application fields - the mobile and embedded development domains. By the end of this book, you'll be ready to build Qt applications that are more efficient, concurrent, and

performance-oriented in nature. What you will learn: Understand classic performance best practices. Get to grips with modern hardware architecture and its performance impact. Implement tools and procedures used in performance optimization. Grasp Qt-specific work techniques for graphical user interface (GUI) and platform programming. Make Transmission Control Protocol (TCP)

and Hypertext Transfer Protocol (HTTP) performant and use the relevant Qt classes. Discover the improvements Qt 5.9 (and the upcoming versions) holds in store. Explore Qt's graphic engine architecture, strengths, and weaknesses. Who this book is for: This book is designed for Qt developers who wish to build highly performance applications for desktop and embedded devices.

<p>Programming Experience with C++ is required. <i>Accuracy and Reliability in Scientific Computing</i> Pearson Education</p> <p>Um richtig in C++11 und C++14 einzusteigen, reicht es nicht aus, sich mit den neuen Features vertraut zu machen. Die Herausforderung liegt darin, sie effektiv einzusetzen, so dass Ihre Software korrekt, effizient, wartbar und portabel ist. Hier kommt dieses</p>	<p>praxisnahe Buch ins Spiel: Es beschreibt, wie Sie wirklich gute Software mit C++11 und C++14 erstellen - also modernes C++ einsetzen. Scott Meyers' <i>Effective C++</i>-Bestseller gelten seit mehr als 20 Jahren als herausragende C++-Ratgeber. Seine klaren, verbindlichen Erläuterungen komplexer technischer Materie haben ihm eine weltweite Anhängerschaft beschert. In diesem Buch</p>	<p>nutzt Scott Meyers wieder das bewährte beispielorientierte Konzept seiner früheren Bücher, um Ihnen den optimalen Einsatz von C++11 und C++14 zu veranschaulichen. Das Buch ist Pflichtlektüre für jeden modernen C++-Softwareentwickler. <u><a href="#">The C++ Workshop</a></u> Packt Publishing Ltd "In the last few years, power dissipation has become an important</p>
--	--	---

design constraint, on par with performance, in the design of new computer systems. Whereas in the past, the primary job of the computer architect was to translate improvements in operating frequency and transistor count into performance, now power efficiency must be taken into account at every step of the design process." "This book aims to document some of the most

important architectural techniques that were invented, proposed, and applied to reduce both dynamic power and static power dissipation in processors and memory hierarchies. A significant number of techniques have been proposed for a wide range of situations and this book synthesizes those techniques by focusing on their common characteristics."  
 ."--BOOK JACKET.  
*Learning R*

*Programming*  
 Pearson Education  
 Build high-performance applications with C++ by using the best optimization techniques available. This book is an authoritative guide to teaching you the tools to do the most effective C++ performance analysis and tuning for maximum optimization. After you've read it you'll be able to start tuning your application right away. The book includes the

latest C++20 standard. What You Will Learn Optimize and tune your C++20 code Write a better algorithm for a particular problem Use the full power of the CPU in your system by creating multiple threads in your program Avoid locks that occur during a multithreading program Optimize input and output actions in your program Read large inputs from files easily Apply template programming

for better-optimized applications and reduce development time Who This Book Is For C++ programmers with an interest in software performance optimization analysis and tuning. *Sams Teach Yourself C++ in One Hour a Day, Seventh Edition* GitforGits With the increase in computing speed and due to the high quality of the optical effects it achieves, ray tracing is becoming a

popular choice for interactive and animated rendering. This book takes readers through the whole process of building a modern ray tracer from scratch in C++. All concepts and processes are explained in detail with the aid of **Writing Efficient C Code** Packt Publishing Ltd 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.



<p>This Learning Path includes content from the following Packt products:</p> <p>Beginning C++ Programming by Richard Grimes</p> <p>Modern C++ Programming Cookbook by Marius Bancila</p> <p>The Modern C++ Challenge by Marius Bancila</p> <p>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</p>	<p>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</p>	<p>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</p> <p><a href="#"><u>WebAssembly Essentials</u></a></p> <p>Packt Publishing Ltd</p> <p>Unleash the Speed: Master WebAssembly and Take Your Web Apps to the Next Level Through practical examples, you'll learn how WebAssembly can transform C++, Rust, and other</p>
---	--	---

languages into lightning-fast web apps. Step-by-step, you'll master techniques to integrate WebAssembly modules into JavaScript projects for seamless communication. But we go deeper. You'll discover strategies to optimize code, leverage multi-threading, and apply cutting-edge methods like lazy loading. We also tackle memory management and dive into profiling, benchmarking, and browser

tools to push performance to the limits. This isn't just essentials of webassembly - it's a roadmap to learn intricacies, best practices, and solutions to common pitfalls. The goal is to equip you with the expertise to develop insanely fast applications. The future is native-level performance on the web. Join the revolution and skill up with this guide to mastering WebAssembly. Your web apps will never be

the same! Key Learnings Grasp WASMs core concepts, bridging web and native apps, boosting performance seamlessly. Dive into tools like Emscripten and Binaryen, facilitating efficient cross language compilations to Wasm. Learn techniques to convert C++, Python, and Java apps, unlocking web application potential. Seamlessly integrate JavaScript and WebAssembly, ensuring smooth inter-

<p>module communications. Harness strategies for fine-tuning code, guaranteeing peak application performance. Explore multi- threading, leveraging simultaneous operations, accelerating execution times. Implement lazy loading and code splitting, optimizing web app responsiveness and load times. Dive deep into efficient memory usage, ensuring</p>	<p>resourceful application execution. Proficiently use browser developer tools, dissecting and enhancing application performance. Enhance user experience with effective caching, ensuring rapid web application accessibility Prerequisites This book is for every programmer aspiring to put their all applications onto web browser with the same speed, efficiency and product</p>	<p>performance. Knowing fundamentals of any programming language and basics of web app development is sufficient to tear out the book wholly. <u>Effective C#</u> Addison- Wesley Professional Write code that scales across CPU registers, multi-core, and machine clusters Key Features Explore concurrent programming in C++ Identify memory management problems Use</p>
--	--	---

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

<p>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</p>	<p>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</p>	<p>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</p>
--	--	---

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.

**Efficient C++**  
Packt Publishing Ltd  
Efficient C/C++ Programming describes a practical, real-world approach to efficient

C/C++ programming. Topics covered range from how to save storage using a restricted character set and how to speed up access to records by employing hash coding and caching. A selective mailing list system is used to illustrate rapid access to and rearrangement of information selected by criteria specified at runtime. Comprised of eight chapters, this

book begins by discussing factors to consider when deciding whether a program needs optimization. In the next chapter, a supermarket price lookup system is used to illustrate how to save storage by using a restricted character set and how to speed up access to records with the aid of hash coding and caching. Attention is paid to rapid retrieval of prices. A

selective mailing list system is then used to illustrate rapid access to and rearrangement of information selected by criteria specified at runtime. The book also considers the Huffman coding and arithmetic coding methods of data compression; a token-threaded interpreter whose code can run faster than equivalent compiled C code, due to its greater

code density; a customer database program with variable-length records; and index and key access to variable-length records. The final chapter summarizes the characteristics of the algorithms encountered in previous chapters, as well as the future of the art of optimization. This monograph will be a useful resource for practicing computer

programmers and those who intend to be working programmers. **C++** Packt Publishing Ltd Learn to create high-performance, error-free programs by understanding the core principles and techniques behind programming in C++ Key Features Gain a solid understanding of the syntax and anatomy of C++ Implement best practices when building high-performance C++

programs. Preparation for real-world development tasks by tackling engaging activities. Book Description: C++ is the backbone of many games, GUI-based applications, and operating systems. Learning C++ effectively is more than a matter of simply reading through theory, as the real challenge is understanding the fundamentals in depth and being able to use them in the real world.

If you're looking to learn C++ programming efficiently, this Workshop is a comprehensive guide that covers all the core features of C++ and how to apply them. It will help you take the next big step toward writing efficient, reliable C++ programs. The C++ Workshop begins by explaining the basic structure of a C++ application, showing you how to write and run your first program

to understand data types, operators, variables and the flow of control structures. You'll also see how to make smarter decisions when it comes to using storage space by declaring dynamic variables during program runtime. Moving ahead, you'll use object-oriented programming (OOP) techniques such as inheritance, polymorphism, and class hierarchies to



make your code structure organized and efficient. Finally, you'll use the C++ standard library's built-in functions and templates to speed up different programming tasks. By the end of this C++ book, you will have the knowledge and skills to confidently tackle your own ambitious projects and advance your career as a C++ developer. What you will learn

Understand how a C++ program is written,

executed, and compiled

Efficiently work with the essential C++ data types and variables

Build your own C++ applications by writing clear and error-free code

Grasp the core principles behind object-oriented programming

Simplify your code by using templates and the standard library

Debug logical errors and handle exceptions in your program

Who this book is for

This Workshop is for anyone who is new to

C++ who wants to build a strong foundation for C++ game programming or application development.

Basic prior knowledge of data structures and OOP concepts, as well as experience in any other programming language, will help you grasp the concepts covered in this book more easily.

Software Architecture with C++

"O'Reilly Media, Inc."

This book covers methods that

can be used to increase the speed of programs via optimisation. Typical mistakes, made by programmers, that lessen the performance of the system are discussed along with easily implemented solutions. Detailed descriptions of the devices and mechanisms of interaction between the computer components, effective ways of programming and a technique for

optimising programs are provided. Programmers also will learn how to effectively implement, in a high-level language, programming methods that are usually done in the assembler. Particular attention is given to the RAM and cache subsystems. The working principles of RAM and the way in which it is coupled with the processor, as well as descriptions of programming methods that

allow programmers to over clock the memory to reach maximum performance, are included. It is oriented toward application programmers with some experience in C/C++ programming, as well as toward system programmers with knowledge of the assembly language. However, the optimisation techniques described here are not bound to any high-level programming language.

Thus, knowledge of C is required only for understanding the source code provided in this book. This book also will be useful to hardware specialists and technicians especially those who build and fine-tune computers. It provides detailed descriptions of the hardware operating principles and discusses the bottlenecks of the most common hardware components. Expert C++ Packt

Publishing Ltd 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 Practical FPGA Programming in C John Wiley & Sons Become a better programmer with performance improvement techniques such as concurrency, lock-free

programming, atomic operations, parallelism, and memory management

**Key Features** Learn proven techniques from a heavyweight and recognized expert in C++ and high-performance computing. Understand the limitations of modern CPUs and their performance impact. Find out how you can avoid writing inefficient code and get the best optimizations from the

compiler. Learn the tradeoffs and costs of writing high-performance programs.

**Book Description** The great free lunch of "performance taking care of itself" is over. Until recently, programs got faster by themselves as CPUs were upgraded, but that doesn't happen anymore. The clock frequency of new processors has almost peaked, and while new architectures provide small improvements to existing

programs, this only helps slightly. To write efficient software, you now have to know how to program by making good use of the available computing resources, and this book will teach you how to do that. The Art of Efficient Programming covers all the major aspects of writing efficient programs, such as using CPU resources and memory efficiently, avoiding unnecessary computations, measuring performance,

and how to put concurrency and multithreading to good use. You'll also learn about compiler optimizations and how to use the programming language (C++) more efficiently. Finally, you'll understand how design decisions impact performance. By the end of this book, you'll not only have enough knowledge of processors and compilers to write efficient programs, but

you'll also be able to understand which techniques to use and what to measure while improving performance. At its core, this book is about learning how to learn. What you will learnDiscover how to use the hardware computing resources in your programs effectivelyUnderstand the relationship between memory order and memory barriersFamiliarize yourself with the performance implications of

different data structures and organizationsAssess the performance impact of concurrent memory accessed and how to minimize itDiscover when to use and when not to use lock-free programming techniquesExplore different ways to improve the effectiveness of compiler optimizationsDesign APIs for concurrent data structures and high-performance data structures to

avoid inefficiencies. Who this book is for This book is for experienced developers and programmers who work on performance-critical projects and want to learn new techniques to improve the performance of their code. Programmers in algorithmic trading, gaming, bioinformatics, computational genomics, or computational fluid dynamics communities will get the most out of

the examples in this book, but the techniques are fairly universal. Although this book uses the C++ language, the concepts demonstrated in the book can be easily transferred or applied to other compiled languages such as C, Java, Rust, Go, and more. [Pro C++ Performance Optimization](#), Apress Writing Efficient C Code: A Thorough Introduction for Java

Programmers was written for two groups of readers: Java programmers who want to learn C from the beginning, and practicing C programmers who want to sharpen their skills. Our goal with the book is to give the reader a deep understanding of both the ISO C programming language and a method based on performance measurements to write efficient C code. We present essentially all

of C99 and the new revision of the ISO C standard, called C11. In addition to C, we introduce elementary computer architecture and essential C development tools including the gcc compiler, the gdb debugger, profilers, and the Valgrind suite of tools for performance analysis and automatic detection of software defects. Using performance measurements and a deep knowledge about which

code transformation s optimizing compilers can perform automatically, as well as their limitations, as the basis for the method for writing efficient C code, the readers of this book will hopefully become more productive and more competent in writing correct, maintainable and fast C code. In order to achieve this goal, and to help C programmers visualize the machine code

and the clock cycle counts of their code, the book contains one chapter on the internals of modern optimizing compilers, and the necessary background on how C is translated to machine code for a RISC processor. The book has a web site [www.writing-efficient-c-code.com](http://www.writing-efficient-c-code.com) where the authors answer questions related to the book."

**Effektives  
modernes  
C++** Carl  
Hanser Verlag

<p>GmbH Co KG PLEASE PROVIDE DESCRIPTION <b>Computer Literature Bibliography : 1946-1963</b> Addison- Wesley Professional This is one of the first books that describe all the steps that are needed in order to analyze, design and implement Monte Carlo applications. It discusses the financial theory as well as the mathematical and numerical background that is needed to write</p>	<p>flexible and efficient C++ code using state-of-the- art design and system patterns, object- oriented and generic programming models in combination with standard libraries and tools. Includes a CD containing the source code for all examples. It is strongly advised that you experiment with the code by compiling it and extending it to suit your needs. Support is offered via a</p>	<p>user forum on <a href="http://www.datasimfinancial.com">www.datasimfinancial.com</a> where you can post queries and communicate with other purchasers of the book. This book is for those professionals who design and develop models in computational finance. This book assumes that you have a working knowledge of C ++. <i>PC Mag</i> Addison- Wesley Professional Programming in TCP/IP can seem deceptively simple.</p>
---	--	---



Nonetheless, many network programmers recognize that their applications could be much more robust. Effective TCP/IP Programming is designed to boost programmers to a higher level of competence by focusing on the protocol suite's more subtle features and techniques. It gives you the know-how you need to produce highly effective TCP/IP programs. In

forty-four concise, self-contained lessons, this book offers experience-based tips, practices, and rules of thumb for learning high-performance TCP/IP programming techniques. Moreover, it shows you how to avoid many of TCP/IP's most common trouble spots. Effective TCP/IP Programming offers valuable advice on such topics as: Exploring IP addressing, subnets, and CIDR Preferring the

sockets interface over XTI/TLI Using two TCP connections Making your applications event-driven Using one large write instead of multiple small writes Avoiding data copying Understanding what TCP reliability really means Recognizing the effects of buffer sizes Using tcpdump, traceroute, netstat, and ping effectively Numerous examples demonstrate essential

ideas and concepts. Skeleton code and a library of common functions allow you to write applications without having to worry about routine chores. Through individual tips and explanations, you will acquire an overall understanding of TCP/IP's inner workings and the practical knowledge needed to put it to work. Using Effective TCP/IP

Programming, you'll speed through the learning process and quickly achieve the programming capabilities of a seasoned pro.

**Practical C++ Financial Programming** Apress  
 Learn C++ in Just One Hour a Day  
 Completely updated for the C++11 standard, Sams Teach Yourself C++ in One Hour a Day presents the language from a practical point of view, helping you

learn how to use C++11 to create faster, simpler, and more efficient C++ applications. Master the fundamentals of C++ and object-oriented programming. Understand how C++11 features help you write compact and efficient code using concepts such as lambda expressions, move constructors, and assignment operators. Learn the Standard Template Library,

including containers and algorithms used in most real-world C++ applications. Test your knowledge and expertise using exercises at the end of every lesson. Learn on your own time, at your own pace: No previous programming experience required. Learn C++11, object-oriented programming, and analysis. Write fast and powerful C++ programs, compile the

source code with a gcc compiler, and create executable files. Use the Standard Template Library's (STL) algorithms and containers to write feature-rich yet stable C++ applications. Develop sophisticated programming techniques using lambda expressions, smart pointers, and move constructors. Learn to expand your program's power with inheritance and

polymorphism. Master the features of C++ by learning from programming experts. Learn C++11 features that allow you to program compact and high-performance C++ applications. TABLE OF CONTENTS PART I: THE BASICS LESSON 1: Getting Started with C++11 LESSON 2: The Anatomy of a C++ Program LESSON 3: Using Variables, Declaring

Constants	Inheritance	Classes
LESSON 4: Managing Arrays and Strings	LESSON 11: Polymorphism	LESSON 18: STL list and forward_list
LESSON 5: Working with Expressions, Statements, and Operators	LESSON 12: Operator Types and Operator Overloading	LESSON 19: STL Set Classes
LESSON 6: Controlling Program Flow	LESSON 13: Casting Operators	LESSON 20: STL Map Classes PART IV: MORE STL
LESSON 7: Organizing Code with Functions	LESSON 14: An Introduction to Macros and Templates	LESSON 21: Understanding Function Objects
LESSON 8: Pointers and References Explained PART II: FUNDAMENTA LS OF OBJECT- ORIENTED C++ PROGRAMMIN G	LESSON 15: An Introduction to the Standard Template Library	LESSON 22: C++11 Lambda Expressions
LESSON 9: Classes and Objects	LESSON 16: The STL String Class	LESSON 23: STL Algorithms
LESSON 10: Implementing	LESSON 17: STL Dynamic Array	LESSON 24: Adaptive Containers: Stack and ... <u>Professional</u> C++ John Wiley & Sons PCMag.com is a leading authority on

technology, delivering Labs-based, independent reviews of the latest products and services. Our expert industry analysis and practical solutions help you make better buying decisions and get more from technology.

**Computer Architecture Techniques for Power-efficiency**

Packt Publishing Ltd  
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	reflection	constexpr,
you will learn	techniques	constraints,
Write	Reap the	and concepts
specialized	benefits of	Implement
data	lock-free	lazy
structures for	concurrent	generators
performance-	programming	and
critical code	Gain insights	asynchronous
Use modern	into subtle	tasks using
metaprogram	optimizations	C++20
ming	used by	coroutines
techniques to	standard	Who this book
reduce	library	is for If you're
runtime	algorithms	a C++
calculations	Compose	developer
Achieve	algorithms	looking to
efficient	using ranges	improve the
memory	library	efficiency of
management	Develop the	your code or
using custom	ability to	just keen to
memory	apply	upgrade your
allocators	metaprogram	skills to the
Reduce	ming aspects	next level, this
boilerplate	such as	book is for
code using		you.

Related with Efficient C Performance  
Programming Techniques:

[© Efficient C Performance Programming  
Techniques Rdw Process In Math](#)

[© Efficient C Performance Programming  
Techniques Reading A Triple Beam Balance](#)

Worksheet

© Efficient C Performance Programming  
Techniques Raytheon Technologies Scholar  
Program