

📖 The C++ Programming Language - Complete Guide

Book Overview

The C++ Programming Language is the definitive guide to C++ written by **Bjarne Stroustrup**, the creator and original implementer of the C++ programming language .

Edition Information

Edition	Year	Key Features
1st Edition	1985	First introduction of C++ to the world
2nd Edition	1991	Major revision, added exception handling, templates
3rd Edition	1997	Complete rewrite (80% new material)
Special Edition	2000	Interim update
4th Edition	2013	Comprehensive C++11 coverage

Note: The 4th Edition is the most current comprehensive reference. Bjarne Stroustrup has also published "A Tour of C++" (3rd Edition, 2022) as a shorter overview .

About the Author - Bjarne Stroustrup

Background

- **Born:** 1950 in Aarhus, Denmark
- **Education:** Cand.Scient. (Mathematics & Computer Science), University of Aarhus; Ph.D. in Computer Science, Cambridge University (1979)
- **Current Position:** Professor of Computer Science at Columbia University, New York City

Major Honors & Awards

Year	Award	Significance
2018	Charles Stark Draper Prize	NAE's highest honor
2018	Computer Pioneer Award	IEEE Computer Society
2017	Faraday Medal	IET's highest honor
2015	Computer History Museum Fellow	For invention of C++
2015	Dahl-Nygaard Senior Prize	European Organization for OO Technologies
1993	ACM Grace Murray Hopper Award	For early work on C++

Dr. Stroustrup is a member of the US National Academy of Engineering and an ACM, IEEE, and CHM fellow .

Book Structure (4th Edition)

Part I: Introductory Material (Chapters 1-5)

Chapter	Title	Content
1	Notes to the Reader	How to use the book
2	A Tour of C++: The Basics	Basic syntax, types, control flow
3	A Tour: Abstraction Mechanisms	Classes, hierarchies, templates
4	A Tour: Containers & Algorithms	STL overview, vectors, maps
5	A Tour: Concurrency & Utilities	Threads, tasks, resource management

Part II: Basic Facilities (Chapters 6-15)

- **6:** Types and Declarations
- **7:** Pointers, Arrays, and References
- **8:** Structures, Unions, and Enumerations
- **9:** Statements
- **10:** Expressions
- **11:** Select Operations
- **12:** Functions
- **13:** Exception Handling
- **14:** Namespaces
- **15:** Source Files and Programs

Part III: Abstraction Mechanisms (Chapters 16-29)

- **16-18:** Classes, Construction, Overloading
- **19:** Special Operators
- **20-21:** Derived Classes & Hierarchies
- **22:** Run-Time Type Information
- **23-29:** Templates, Generic Programming, Metaprogramming

Part IV: The Standard Library (Chapters 30-44)

- **30-32:** STL Containers, Algorithms, Iterators
- **33:** Memory and Resources
- **34:** Utilities
- **35:** Strings
- **36:** Regular Expressions
- **37:** I/O Streams
- **38:** Locales
- **39:** Numerics
- **40-42:** Concurrency, Threads, Tasks
- **43:** The C Standard Library
- **44:** Compatibility

Key Topics Covered

1. Core Language Features

- **Types & Declarations** - Fundamental types, type deduction (auto), decltype
- **Object-Oriented Programming** - Classes, inheritance, polymorphism, virtual functions

- **Templates** - Function templates, class templates, variadic templates, template aliases
- **Exception Handling** - try/catch blocks, RAII, exception safety guarantees

2. C++11 New Features (4th Edition)

Feature	Description
Lambdas	Anonymous function objects
Move Semantics	Move constructors, move assignment
Uniform Initialization	<code>{}</code> syntax for all types
Concurrency	Threads, mutexes, condition variables
Smart Pointers	<code>unique_ptr</code> , <code>shared_ptr</code> , <code>weak_ptr</code>
Regular Expressions	Pattern matching library
Random Numbers	<code><random></code> library

3. Standard Template Library (STL)

- **Containers:** `vector`, `list`, `map`, `unordered_map`, `set`, `array`
- **Algorithms:** `sort`, `find`, `transform`, `accumulate`, `for_each`
- **Iterators:** Input, output, forward, bidirectional, random-access

4. Advanced Topics

- **Metaprogramming** - Compile-time computation using templates
- **Concurrency** - Multithreading, async tasks, futures
- **Memory Model** - Sequential consistency, atomic operations

What Makes This Book Unique

1. Authoritative Source

- Written by the **creator of C++** himself
- The book is the **definitional reference** for the language

2. Complete Coverage

- Every language feature a professional programmer needs
- Extensive cross-references to the ISO standard

3. Practical Examples

- "Pure C++11" examples throughout
- Focus on **real-world programming techniques**
- Emphasis on **resource management and performance**

4. Teaching Approach

- "Just in Time" learning - concepts introduced when needed
- Emphasis on **abstraction and modularity**
- Integration of language features with programming techniques

Book Statistics (3rd/4th Edition)

Metric	Count
Exercises	362
Advice Lists	28 (400+ suggestions)
Cross References	1,500+
Index Terms	4,000+
Chapters	25
Appendices	5
Words	350,000+

"All problems in computer science can be solved by another level of indirection, except for the problem of too many layers of indirection."

— David J. Wheeler (quoted in the book)

Target Audience

This book is intended for:

1. **C++ programmers** wanting to learn C++11 and modern techniques
2. **C programmers** curious about C++'s additional features
3. **Developers from other languages** (Java, C#, Python) seeking performance and low-level control

Prerequisite: Readers should already know basic programming concepts. Beginners should start with "**Programming: Principles and Practice Using C++**" also by Stroustrup .

Related Books by Bjarne Stroustrup

Book	Edition	Purpose
The C++ Programming Language	4th (2013)	Complete reference
A Tour of C++	3rd (2022)	Concise overview for experienced programmers
Programming: Principles and Practice Using C++	3rd (2024)	Introductory textbook
The Design and Evolution of C++	1st (1994)	History and design rationale

ISBN Information (4th Edition)

Format	ISBN-13	ISBN-10
Paperback	978-0-321-56384-2	0-321-56384-0

Format	ISBN-13	ISBN-10
eBook	978-0-13-352285-3	0-13-352285-7

Publisher: Addison-Wesley Professional (Pearson)

How to Use This Book

For Beginners

1. Read **Chapters 2-5** (Tour of C++)
2. Work through **Part II** (Basic Facilities)
3. Practice with examples before moving to advanced topics

For Intermediate Programmers

1. Review **Part II** for language fundamentals
2. Focus on **Part III** (Abstraction Mechanisms)
3. Reference **Part IV** for STL usage

For Advanced Programmers

1. Use as a **reference** for specific features
2. Study **metaprogramming** (Chapter 28)
3. Explore **concurrency** (Chapters 40-42)
4. Consult **compatibility** guidelines (Chapter 44)

Key Quotes from the Preface

"C++ feels like a new language. That is, I can express my ideas more clearly, more simply, and more directly in C++11 than I could in C++98. Furthermore, the resulting programs are better checked by the compiler and run faster."

"C++ is a language for someone who takes the task of programming seriously. Our civilization depends critically on software; it had better be quality software."

"The emphasis on stability implies that standards-conforming code you write today will still work a couple of decades from now."

Recommendations

1. **Buy the Book** - This is a reference you'll use throughout your career
 2. **Read Actively** - Type the examples, experiment with modifications
 3. **Use with Compiler** - Practice with a modern C++ compiler (GCC, Clang, MSVC)
 4. **Reference Often** - Keep it on your desk, not your shelf
 5. **Combine with Practice** - Work on real projects to internalize concepts
-

Summary

The C++ Programming Language (4th Edition) is the **definitive reference** written by the language's creator. It provides complete coverage of C++11 with expert insights into design techniques, performance considerations, and best practices. Whether you're learning C++ or have years of experience, this book belongs in your library.

"If the answer isn't there, you'd better avoid the feature."

— Bjarne Stroustrup on the book's completeness.