

Safari*i>>*

Books
Online

Entire Site

Favorites & Folders

All Shared Lists

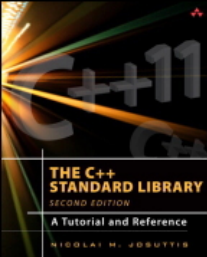
Bookmarks

Notes & Tags

Reviews

Help

Pralay Patoria



< Return to Search Results

RESUME READING >>

ADD TO FAVORITES

DOWNLOAD

BUY FROM PUBLISHER

Safari Books Online Subscribers
Get Up to 35% OFF

Mobi ePub

Subscriber Reviews

REVIEW THIS BOOK

ADD NOTE / TAGS

Table of Contents

Title Page
Copyright Page
Dedication Page
Contents
Preface to the Second Edition
Acknowledgments for the Second Edition
Preface to the First Edition
Acknowledgments for the First Edition
Chapter 1. About This Book
1.1 Why This Book
1.2 Before Reading This Book
1.3 Style and Structure of the Book
1.4 How to Read This Book
1.5 State of the Art
1.6 Example Code and Additional Information
1.7 Feedback
Chapter 2. Introduction to C++ and the Standard Library
2.1 History of the C++ Standards
2.2 Complexity and Big-O Notation
Chapter 3. New Language Features
3.1 New C++11 Language Features
3.2 Old "New" Language Features
Chapter 4. General Concepts
4.1 Namespace std
4.2 Header Files
4.3 Error and Exception Handling

4.4 Callable Objects

4.5 Concurrency and Multithreading

4.6 Allocators

Chapter 5. Utilities

5.1 Pairs and Tuples

5.2 Smart Pointers

5.3 Numeric Limits

5.4 Type Traits and Type Utilities

5.5 Auxiliary Functions

5.6 Compile-Time Fractional Arithmetic with Class `ratio`

5.7 Clocks and Timers

5.8 Header Files `<cstdlib>`, `<cstdliblib>`, and `<cstring>`

Chapter 6. The Standard Template Library

6.1 STL Components

6.2 Containers

6.3 Iterators

6.4 Algorithms

6.5 Iterator Adapters

6.6 User-Defined Generic Functions

6.7 Manipulating Algorithms

6.8 Functions as Algorithm Arguments

6.9 Using Lambdas

6.10 Function Objects

6.11 Container Elements

6.12 Errors and Exceptions inside the STL

6.13 Extending the STL

Chapter 7. STL Containers

7.1 Common Container Abilities and Operations

7.2 Arrays

7.3 Vectors

7.4 Deques

7.5 Lists

7.6 Forward Lists

7.7 Sets and Multisets

7.8 Maps and Multimaps

7.9 Unordered Containers

7.10 Other STL Containers

7.11 Implementing Reference Semantics

7.12 When to Use Which Container

Chapter 8. STL Container Members in Detail

8.1 Type Definitions

8.2 Create, Copy, and Destroy Operations

8.3 Nonmodifying Operations

8.4 Assignments

8.5 Direct Element Access

8.6 Operations to Generate Iterators

8.7 Inserting and Removing Elements

8.8 Special Member Functions for Lists and Forward Lists

8.9 Container Policy Interfaces

8.10 Allocator Support

Chapter 9. STL Iterators

9.1 Header Files for Iterators

9.2 Iterator Categories

9.3 Auxiliary Iterator Functions

9.4 Iterator Adapters

9.5 Iterator Traits

9.6 Writing User-Defined Iterators

Chapter 10. STL Function Objects and Using Lambdas

10.1 The Concept of Function Objects

10.2 Predefined Function Objects and Binders

10.3 Using Lambdas

Chapter 11. STL Algorithms

11.1 Algorithm Header Files

11.2 Algorithm Overview

11.3 Auxiliary Functions

11.4 The `for_each()` Algorithm

11.5 Nonmodifying Algorithms

11.6 Modifying Algorithms

11.7 Removing Algorithms

11.8 Mutating Algorithms

11.9 Sorting Algorithms

11.10 Sorted-Range Algorithms

11.11 Numeric Algorithms

Chapter 12. Special Containers

12.1 Stacks

12.2 Queues

12.3 Priority Queues

12.4 Container Adapters in Detail

12.5 Bitsets

Chapter 13. Strings

13.1 Purpose of the String Classes

13.2 Description of the String Classes

13.3 String Class in Detail

Chapter 14. Regular Expressions

14.1 The Regex Match and Search Interface

14.2 Dealing with Subexpressions

14.3 Regex Iterators

14.4 Regex Token Iterators

14.5 Replacing Regular Expressions

14.6 Regex Flags

14.7 Regex Exceptions

14.8 The Regex ECMAScript Grammar

14.9 Other Grammars

14.10 Basic Regex Signatures in Detail

Chapter 15. Input/Output Using Stream Classes

- 15.1 Common Background of I/O Streams
- 15.2 Fundamental Stream Classes and Objects
- 15.3 Standard Stream Operators << and >>
- 15.4 State of Streams
- 15.5 Standard Input/Output Functions
- 15.6 Manipulators
- 15.7 Formatting
- 15.8 Internationalization
- 15.9 File Access
- 15.10 Stream Classes for Strings
- 15.11 Input/Output Operators for User-Defined Types
- 15.12 Connecting Input and Output Streams
- 15.13 The Stream Buffer Classes
- 15.14 Performance Issues

Chapter 16. Internationalization

- 16.1 Character Encodings and Character Sets
- 16.2 The Concept of Locales
- 16.3 Locales in Detail
- 16.4 Facets in Detail

Chapter 17. Numerics

- 17.1 Random Numbers and Distributions
- 17.2 Complex Numbers
- 17.3 Global Numeric Functions
- 17.4 Valarrays

Chapter 18. Concurrency

- 18.1 The High-Level Interface: `async()` and Futures
- 18.2 The Low-Level Interface: Threads and Promises
- 18.3 Starting a Thread in Detail
- 18.4 Synchronizing Threads, or the Problem of Concurrency
- 18.5 Mutexes and Locks
- 18.6 Condition Variables
- 18.7 Atomics

Chapter 19. Allocators

- 19.1 Using Allocators as an Application Programmer
- 19.2 A User-Defined Allocator
- 19.3 Using Allocators as a Library Programmer

Bibliography

- Newsgroups and Forums
- Books and Web Sites

Index

Supplementary Chapter