

GLOBAL  
EDITION



# Starting Out With C++

## *From Control Structures Through Objects*

BRIEF VERSION

EIGHTH EDITION

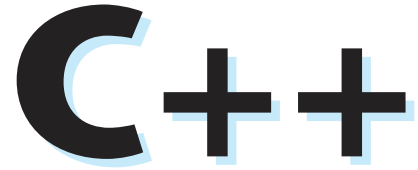
Tony Gaddis



ALWAYS LEARNING

PEARSON

STARTING OUT WITH



From Control Structures  
through Objects

**BRIEF VERSION  
EIGHTH EDITION  
GLOBAL EDITION**

# Starting Out with C++ from Control Structures through Objects, Brief Version, Global Edition

## Table of Contents

Cover

Title Page

Copyright Page

Supplements

Acknowledgments

About the Author

Contents

Preface

CHAPTER 1 Introduction to Computers and Programming

1.1 Why Program?

1.2 Computer Systems: Hardware and Software

1.3 Programs and Programming Languages

1.4 What Is a Program Made of?

1.5 Input, Processing, and Output

1.6 The Programming Process

1.7 Procedural and Object-Oriented Programming

CHAPTER 2 Introduction to C++

2.1 The Parts of a C++ Program

2.2 The cout Object

2.3 The #include Directive

2.4 Variables and Literals

2.5 Identifiers

2.6 Integer Data Types

# **Table of Contents**

- 2.7 The char Data Type
- 2.8 The C++ string Class
- 2.9 Floating-Point Data Types
- 2.10 The bool Data Type
- 2.11 Determining the Size of a Data Type
- 2.12 Variable Assignments and Initialization
- 2.13 Scope
- 2.14 Arithmetic Operators
- 2.15 Comments
- 2.16 Named Constants
- 2.17 Programming Style

## **CHAPTER 3 Expressions and Interactivity**

- 3.1 The cin Object
- 3.2 Mathematical Expressions
- 3.3 When You Mix Apples and Oranges: Type Conversion
- 3.4 Overflow and Underflow
- 3.5 Type Casting
- 3.6 Multiple Assignment and Combined Assignment
- 3.7 Formatting Output
- 3.8 Working with Characters and string Objects
- 3.9 More Mathematical Library Functions
- 3.10 Focus on Debugging: Hand Tracing a Program
- 3.11 Focus on Problem Solving: A Case Study

## **CHAPTER 4 Making Decisions**

- 4.1 Relational Operators
- 4.2 The if Statement
- 4.3 Expanding the if Statement
- 4.4 The if/else Statement
- 4.5 Nested if Statements
- 4.6 The if/else if Statement

# **Table of Contents**

- 4.7 Flags
- 4.8 Logical Operators
- 4.9 Checking Numeric Ranges with Logical Operators
- 4.10 Menus
- 4.11 Focus on Software Engineering: Validating User Input
- 4.12 Comparing Characters and Strings
- 4.13 The Conditional Operator
- 4.14 The switch Statement
- 4.15 More About Blocks and Variable Scope

## **CHAPTER 5 Loops and Files**

- 5.1 The Increment and Decrement Operators
- 5.2 Introduction to Loops: The while Loop
- 5.3 Using the while Loop for Input Validation
- 5.4 Counters
- 5.5 The do-while Loop
- 5.6 The for Loop
- 5.7 Keeping a Running Total
- 5.8 Sentinels
- 5.9 Focus on Software Engineering: Deciding Which Loop to Use
- 5.10 Nested Loops
- 5.11 Using Files for Data Storage
- 5.12 Optional Topics: Breaking and Continuing a Loop

## **CHAPTER 6 Functions**

- 6.1 Focus on Software Engineering: Modular Programming
- 6.2 Defining and Calling Functions
- 6.3 Function Prototypes
- 6.4 Sending Data into a Function
- 6.5 Passing Data by Value
- 6.6 Focus on Software Engineering: Using Functions in a Menu-Driven Program

# **Table of Contents**

- 6.7 The return Statement
- 6.8 Returning a Value from a Function
- 6.9 Returning a Boolean Value
- 6.10 Local and Global Variables
- 6.11 Static Local Variables
- 6.12 Default Arguments
- 6.13 Using Reference Variables as Parameters
- 6.14 Overloading Functions
- 6.15 The exit() Function
- 6.16 Stubs and Drivers

## **CHAPTER 7 Arrays**

- 7.1 Arrays Hold Multiple Values
- 7.2 Accessing Array Elements
- 7.3 No Bounds Checking in C++
- 7.4 Array Initialization
- 7.5 The Range-Based for Loop
- 7.6 Processing Array Contents
- 7.7 Focus on Software Engineering: Using Parallel Arrays
- 7.8 Arrays as Function Arguments
- 7.9 Two-Dimensional Arrays
- 7.10 Arrays with Three or More Dimensions
- 7.11 Focus on Problem Solving and Program Design: A Case Study
- 7.12 If You Plan to Continue in Computer Science: Introduction to the STL vector

## **CHAPTER 8 Searching and Sorting Arrays**

- 8.1 Focus on Software Engineering: Introduction to Search Algorithms
- 8.2 Focus on Problem Solving and Program Design: A Case Study
- 8.3 Focus on Software Engineering: Introduction to Sorting Algorithms
- 8.4 Focus on Problem Solving and Program Design: A Case Study
- 8.5 If You Plan to Continue in Computer Science: Sorting and Searching vectors

## **CHAPTER 9 Pointers**



# **Table of Contents**

9.1	Getting the Address of a Variable
9.2	Pointer Variables
9.3	The Relationship Between Arrays and Pointers
9.4	Pointer Arithmetic
9.5	Initializing Pointers
9.6	Comparing Pointers
9.7	Pointers as Function Parameters
9.8	Focus on Software Engineering: Dynamic Memory Allocation
9.9	Focus on Software Engineering: Returning Pointers from Functions
9.10	Using Smart Pointers to Avoid Memory Leaks
9.11	Focus on Problem Solving and Program Design: A Case Study
<b>CHAPTER 10 Characters, C-Strings, and More About the string Class</b>	
10.1	Character Testing
10.2	Character Case Conversion
10.3	C-Strings
10.4	Library Functions for Working with C-Strings
10.5	C-String/Numeric Conversion Functions
10.6	Focus on Software Engineering: Writing Your Own C-String-Handling Functions
10.7	More About the C++ string Class
10.8	Focus on Problem Solving and Program Design: A Case Study
<b>CHAPTER 11 Structured Data</b>	
11.1	Abstract Data Types
11.2	Focus on Software Engineering: Combining Data into Structures
11.3	Accessing Structure Members
11.4	Initializing a Structure
11.5	Arrays of Structures
11.6	Focus on Software Engineering: Nested Structures
11.7	Structures as Function Arguments

# **Table of Contents**

11.8 Returning a Structure from a Function

11.9 Pointers to Structures

11.10 Focus on Software Engineering: When to Use ., When to Use ->, and  
When to Use \*

11.11 Unions

11.12 Enumerated Data Types

## **CHAPTER 12 Advanced File Operations**

12.1 File Operations

12.2 File Output Formatting

12.3 Passing File Stream Objects to Functions

12.4 More Detailed Error Testing

12.5 Member Functions for Reading and Writing Files

12.6 Focus on Software Engineering: Working with Multiple Files

12.7 Binary Files

12.8 Creating Records with Structures

12.9 Random-Access Files

12.10 Opening a File for Both Input and Output

## **CHAPTER 13 Introduction to Classes**

13.1 Procedural and Object-Oriented Programming

13.2 Introduction to Classes

13.3 Defining an Instance of a Class

13.4 Why Have Private Members?

13.5 Focus on Software Engineering: Separating Class Specification from  
Implementation

13.6 Inline Member Functions

13.7 Constructors

13.8 Passing Arguments to Constructors

13.9 Destructors

13.10 Overloading Constructors

13.11 Private Member Functions



# **Table of Contents**

13.12 Arrays of Objects

13.13 Focus on Problem Solving and Program Design: An OOP Case Study

13.14 Focus on Object-Oriented Programming: Simulating Dice with Objects

13.15 Focus on Object-Oriented Programming: Creating an Abstract Array  
Data Type

13.16 Focus on Object-Oriented Design: The Unified Modeling Language  
(UML)

13.17 Focus on Object-Oriented Design: Finding the Classes and Their  
Responsibilities

## **CHAPTER 14 More About Classes**

14.1 Instance and Static Members

14.2 Friends of Classes

14.3 Memberwise Assignment

14.4 Copy Constructors

14.5 Operator Overloading

14.6 Object Conversion

14.7 Aggregation

14.8 Focus on Object-Oriented Design: Class Collaborations

14.9 Focus on Object-Oriented Programming: Simulating the Game of  
Cho-Han

## **CHAPTER 15 Inheritance, Polymorphism, and Virtual Functions**

15.1 What Is Inheritance?

15.2 Protected Members and Class Access

15.3 Constructors and Destructors in Base and Derived Classes

15.4 Redefining Base Class Functions

15.5 Class Hierarchies

15.6 Polymorphism and Virtual Member Functions

15.7 Abstract Base Classes and Pure Virtual Functions

15.8 Multiple Inheritance

## **APPENDIX A: Getting Started with Alice**

# **Table of Contents**

APPENDIX B: The ASCII Character Set

APPENDIX C: Operator Precedence and Associativity

Quick References

Index

Credit