

## PREFACE

Object-Oriented Programming (OOP) has become the preferred programming approach by the software industries, as it offers a powerful way to cope with the complexity of real-world problems. Among the OOP languages today, C++ is by far the most widely used language.

Since its creation by Bjarne Stroustrup in early 1980s, C++ has undergone many changes and improvements. The language was standardized in 1998 by the American National Standards Institute (ANSI) and the International Standards Organization (ISO) by incorporating not only the new features but also the changes suggested by the user groups. The present edition completely follows the specifications of ANSI/ISO Standards.

Object Oriented Programming with C++ is for the programmers who wish to know all about C++ language and object-oriented programming. It explains in a simple and easy-to-understand style the what, why and how of object-oriented programming with C++. The book assumes that the readers are already familiar with C language, although she or he need not be an expert programmer.

The book provides numerous examples, illustrations and complete programs. The sample programs are meant to be both simple and educational. Wherever necessary, pictorial descriptions of concepts have been included to improve clarity and facilitate better understanding. The book also presents the concept of object-oriented approach and discusses important elements of object-oriented analysis and design of systems.

## THE SIXTH EDITION

With each edition, I have worked hard to provide a book that is useful, interesting, and engaging to students while introducing them to the core concepts. Keeping this in mind, the existing topics have been strengthened in the present edition with the enhancement of the pedagogical topics, based on the readers' suggestions. Corrections to typographical errors have also been incorporated.

## NEW TO THE EDITION

- Colored Edition
- 20 New Programs and Review Questions (As per University Pattern)
- Refreshed Minor and Major Projects
- Revived and enriched pedagogy includes
  - ▲ 115 Programs
  - ▲ 225 Review Questions
  - ▲ 67 Debugging Exercises
  - ▲ 90 Programming Exercises

## SALIENT FEATURES OF THE BOOK

- Simple and easy-to-understand language
- Extended coverage to topics like *Overloading, Abstract classes, Typecasting, Dynamic Memory Allocation, Structures and Unions, and RTTI*
- Projects to enhance programming skills
- Refreshed C++ proficiency test *covering latest technical interview questions along with answers*
- Refreshed and enhanced pedagogy which includes over *500 pedagogical aids like programs, review questions, debugging exercise with answers, programming exercises, and special boxes with notes wherever relevant*

## KEY FEATURES OF THE BOOK

FEATURES	BENEFITS
<p>Key Concepts and Introduction</p> <p><b>Key Concepts</b></p> <p>Tokens   Keywords   Identifiers   Data types   User-defined types   Derived types   Symbolic constants            Declaration of variables   Initialization   Reference variables   Type compatibility   Scope resolution            Dereferencing   Memory management   Formatting the output   Type casting   Constructing expressions            Special assignment expressions   Implicit conversion   Operator overloading   Control structures</p>	<p>Key concepts provide a quick insight of the concepts that are discussed in the chapters. These are followed by an Introduction which introduces the topics to be covered in that chapter and also relates them to those already learnt. <i>Refer to pages 1, 16, 29, 769, and 88</i></p>
<p>Program Codes</p> <p><b>Program 12.9 Template Function with Explicit Function</b></p> <pre>#include &lt;iostream.h&gt; #include &lt;string.h&gt;  using namespace std;  template &lt;class T&gt; void display(T x) //overloaded template function display {     cout&lt;&lt;endl&lt;&lt;overloaded Template Display 1: &lt;&lt;x&lt;&lt;endl&lt;&lt;endl; }</pre>	<p>Program codes with comments are given throughout the book to elaborate how the various lines of codes work; thus, enhancing the programming skills. <i>Refer to pages 21, 48, 49, 76, and 95</i></p>
<p>Key Terms</p> <p><b>Key Terms</b></p> <p>actual arguments   argument list   bubble sort   call by reference   call by value   called function            calling program   calling statement   cmath   const arguments   declaration statement   default arguments            default values   dummy variables   ellipses   empty argument list   exit value   formal arguments            friend functions   function call   function definition   function overloading   function polymorphism            function prototype   indirection   inline   inline functions   macros   main()   math library            math.h   overloading   pointers   polymorphism   prototyping   reference variable   return by reference            returnstatement   return type   return()   template   virtual functions</p>	<p>Key terms lists the important terms discussed in the chapters. <i>Refer to pages 15, 26, 64, 84, and 123</i></p>
<p>Notes</p> <p><b>Note</b> Notice the following statement in the program:</p> <pre>int item :: count; // definition of static data member</pre>	<p>Language tips and other special considerations are highlighted as notes wherever essential to add value to the concept explanation. <i>Refer to pages 24, 51, 82, 114, and 116</i></p>

(Continued)

FEATURES	BENEFITS
<p>Summary</p> <div style="border: 1px solid black; padding: 5px; margin: 10px 0;"> <p style="text-align: center; background-color: #e91e63; color: white; margin: 0;"><b>SUMMARY</b></p> <ul style="list-style-type: none"> <li><input type="checkbox"/> C++ supports a mechanism known as template to implement the concept of generic programming.</li> <li><input type="checkbox"/> Templates allows us to generate a family of classes or a family of functions to handle different data types.</li> <li><input type="checkbox"/> Template classes and functions eliminate code duplication for different types and thus make the program development easier and more manageable.</li> <li><input type="checkbox"/> We can use multiple parameters in both the class templates and function templates.</li> <li><input type="checkbox"/> A specific class created from a class template is called a template class and the process of creating a template class is known as instantiation. Similarly, a specific function created from a function template is called a template function.</li> </ul> </div>	<p>Summary gives the essence of each chapter in the form of bulleted points for a quick review during the examinations. <i>Refer to pages 13, 26, 63, 83, and 123</i></p>
<p>Exercise Problems</p> <div style="border: 1px solid black; padding: 5px; margin: 10px 0;"> <p style="background-color: #e91e63; color: white; padding: 2px;"><b>Review Questions</b></p> <hr/> <p>13.1 What is an exception? <span style="background-color: #e91e63; color: white; padding: 0 2px;">P</span><span style="background-color: #e91e63; color: white; padding: 0 2px;">A</span><span style="background-color: #e91e63; color: white; padding: 0 2px;">Q</span></p> <p>13.2 How is an exception handled in C++? <span style="background-color: #e91e63; color: white; padding: 0 2px;">P</span><span style="background-color: #e91e63; color: white; padding: 0 2px;">A</span><span style="background-color: #e91e63; color: white; padding: 0 2px;">Q</span></p> <p>13.3 What are the advantages of using exception handling mechanism in a program?</p> </div> <div style="border: 1px solid black; padding: 5px; margin: 10px 0;"> <p style="background-color: #e91e63; color: white; padding: 2px;"><b>Debugging Exercises</b></p> <hr/> <p>3.1 What will happen when you execute the following code?</p> <pre style="font-family: monospace; font-size: 0.9em;">#include &lt;iostream.h&gt; void main() {     int i=0;     i=400*400/400;     cout &lt;&lt; i; }</pre> </div> <div style="border: 1px solid black; padding: 5px; margin: 10px 0;"> <p style="background-color: #e91e63; color: white; padding: 2px;"><b>Programming Exercises</b></p> <hr/> <p>12.1 Write a function template for finding the minimum value contained in an array. <span style="background-color: #e91e63; color: white; padding: 0 2px;">W</span><span style="background-color: #e91e63; color: white; padding: 0 2px;">E</span><span style="background-color: #e91e63; color: white; padding: 0 2px;">B</span></p> <p>12.2 Write a class template to represent a generic vector. Include member functions to perform the following tasks:</p> <ul style="list-style-type: none"> <li>(a) To create the vector.</li> <li>(b) To modify the value of a given element.</li> <li>(c) To multiply by a scalar value.</li> <li>(d) To display the vector in the form (10, 20, 30,.....). <span style="background-color: #e91e63; color: white; padding: 0 2px;">W</span><span style="background-color: #e91e63; color: white; padding: 0 2px;">E</span><span style="background-color: #e91e63; color: white; padding: 0 2px;">B</span></li> </ul> <p>12.3 Write a function template to perform linear search in an array. <span style="background-color: #e91e63; color: white; padding: 0 2px;">W</span><span style="background-color: #e91e63; color: white; padding: 0 2px;">E</span><span style="background-color: #e91e63; color: white; padding: 0 2px;">B</span></p> </div>	<p>375 chapter-end exercise problems including review questions, debugging exercises and programming exercises are given for the readers to work out, practice and implement the concepts. <i>Refer to pages 14, 27, 65, 84, and 124</i></p>
<p>Programming Projects</p> <p><b>Typing Tutor</b></p> <p>Typing Tutor is a command-driven application that helps users to perfect their typing skills. It allows users to take typing tests and determine their typing speed and accuracy. Table A.1 lists the various coding elements of the application:</p>	<p>The two programming projects (PR2PO and Credit CalC) in Appendix A give an insight on how to integrate and apply the various features of C++ in real-life problems. <i>Refer to pages 447, and 455</i></p>
<p>Proficiency Test</p> <div style="border: 1px solid black; padding: 10px; margin: 10px 0;"> <p style="font-size: 1.2em; font-weight: bold; margin: 0;">C++ Proficiency Test with Answers</p> <div style="float: right; background-color: #e91e63; color: white; padding: 5px; text-align: center; width: 80px;"> <p style="margin: 0;">APPENDIX</p> <p style="margin: 0; font-size: 2em; font-weight: bold;">I</p> </div> </div>	<p>An appendix on proficiency test offers readers to assess their level of mastery on the language and prepare for technical interviews. <i>Refer to pages 527, and 528</i></p>

## CHAPTER ORGANIZATION

The book is organized into 17 chapters and 9 appendices. **Chapters 1** and **2** provide a strong foundation and a perspective on Object-Oriented Programming and C++. **Chapter 3** deals with the

various data types and control structures. This chapter now includes differences between Structure and Union, explained with appropriate programs and introduction of Storage Classes and its various types. **Chapter 4** elucidates the various functions of C++. A separate section discussing the Recursion function has been added in this chapter.

**Chapter 5** covers Classes and Objects, while **Chapter 6** gives a detailed study on Constructors and Destructors, the various kinds of Constructors, including topics like Parameterized Constructors, Constructor Overloading and the various exceptional cases to be noted when working with Destructors. The chapter also deals with the Dynamic Initialization of Objects. **Chapter 7** covers in detail the concept of Operator overloading and Type conversions, apart from discussing Overloading rules and String manipulations. The topic on Exceptions in Operator Overloading Operations has been added and explained with sufficient programs to support the theory. The various types of Inheritance like single, multiple, and hybrid inheritance and its other topics like Derived and Abstract classes can be studied in **Chapter 8**.

**Chapter 9** begins by explaining the vital feature of OOP, i.e., Polymorphism, and elucidating on Pointers and Virtual Functions, while the Input and Output Operations have been dealt with in **Chapter 10**. An in-depth study of the various File Stream Operations and File Pointers has been elucidated in **chapter 11**. Templates and Exception handling, like Exceptions in Constructors and Destructors and Exceptions in Operator Overloading Functions can be studied in **Chapters 12** and **13**, respectively. **Chapter 14** introduces the students to the standard template, its components and functions. **Chapter 15** covers the study of Strings and its related fields while **Chapter 16** deals with the new features of ANSI C++ standards. The concluding chapter, **Chapter 17**, helps students in understanding the object-oriented systems.

Each chapter ends with a set of summary, key terms, review questions (the frequently asked universities' questions earmarked with **F A Q** icon), debugging exercises and programming problems. For further assistance, students can also refer to the website for solutions to questions containing the **W E B** icon. In addition, two new projects – PR2PO (Major) and Credit CalC (Minor) are included in the present edition. Executing Turbo C++, Executing C++ under Windows, Glossary of ANSI C++ Keywords, C++ Operator Precedence, Points to Remember, Glossary of Important C++ and OOP Terms, and C++ Proficiency Test are provided in the appendices; helping students understand and implement the concepts in a better way and prove to be valuable source and tool in their study.

## COMPANION WEBSITE

The book is accompanied with an enhanced and exhaustive web supplement which includes the following additional information for both instructors and students at

**<http://www.mhhe.com/balagurusamy/oop6>**

*For instructors and students*

- Complete code with step-by-step description and user manual for the Payroll management and Hospital management Systems projects, Projects from the 5th edition, and newly added projects in executable format
- Solutions for selected programming exercises from the book
- Write-up differences between C and C++
- Implementation of data structures concepts using C++
- Codes for few selected programs from 5e, which are not present in 6e.
- Practice Tests— Questions given under C++ proficiency tests in the fifth edition
- Student self-assessment test

## ACKNOWLEDGEMENTS

Since the release of the first edition of this book a decade ago, lakhs of teachers and students and professional programmers have been using this book. Their overwhelming support encouraged me to bring out the Fifth Edition in 2011 and now the Sixth Edition.

Many thanks to the following faculty and student reviewers for their valuable suggestions and comments:

<b>Preetvanti Singh</b>	<i>Dayalbagh Educational Institute, Agra, Uttar Pradesh</i>
<b>Pradeep Kumar</b>	<i>Jaypee University of Information Technology, Solan, Himachal Pradesh</i>
<b>Pragya Jain</b>	<i>Indian Institute of Technology Delhi, New Delhi</i>
<b>N K Kamila</b>	<i>C. V. Raman College of Engineering, Bhubaneswar, Odisha</i>
<b>Kishore Kumar Senapati</b>	<i>Birla Institute of Technology Mesra, Ranchi, Jharkhand</i>
<b>Prasanta Jena</b>	<i>Indian School of Mines, Dhanbad, Jharkhand</i>
<b>Manisha J Somavanshi</b>	<i>Kommon Sense Consultancy and Training, Pune, Maharashtra</i>
<b>S Kannimuthu</b>	<i>Sri Krishna College of Engineering and Technology, Coimbatore, Tamil Nadu</i>
<b>N Shanthi</b>	<i>K S Rengasamy College of Technology, Tiruchengode, Tamil Nadu</i>
<b>Balamurugan Balusamy</b>	<i>Vellore Institute of Technology, Vellore, Tamil Nadu</i>
<b>T V Gopal</b>	<i>Anna University, Chennai, Tamil Nadu</i>
<b>S Suresh Babu</b>	<i>Thiruvalluvar Government Arts College, Rasipuram, Tamil Nadu</i>
<b>M Prasanna</b>	<i>PSG College of Technology, Coimbatore, Tamil Nadu</i>
<b>A Sharada</b>	<i>G. Narayanamma Institute of Technology and Science, Shaikpet, Hyderabad, Andhra Pradesh</i>
<b>G Shobha</b>	<i>R V College of Engineering, Bangalore, Karnataka</i>
<b>CH V K N S N Moorthy</b>	<i>DRK Institute of Science and Technology, Hyderabad, Andhra Pradesh</i>
<b>Annapa</b>	<i>National Institute of Technology, Surathkal, Karnataka</i>

I would also like to thank the following reviewers from the previous edition:

<b>Amit Jain</b>	<i>Bharat Institute of Technology, Meerut, Uttar Pradesh</i>
<b>Nitin Gupta</b>	<i>National Institute of Technology, Hamirpur, Himachal Pradesh</i>
<b>Prabhat Verma</b>	<i>Harcourt Butler Technological Institute, Kanpur, Uttar Pradesh</i>
<b>Manish Varshney</b>	<i>Sri Siddhi Vinayak Institute of Technology, Bareilly, Uttar Pradesh</i>
<b>Harmeet Malhotra</b>	<i>Institute of Information Technology and Management, New Delhi</i>

<b>S R Biradar</b>	<i>Mody Institute of Technology and Science, Rajasthan</i>
<b>Barnali Goswami</b>	<i>Narula Institute of Technology, Kolkata, West Bengal</i>
<b>Samaresh Mishra</b>	<i>Kalinga Institute of Industrial Technology, Bhubaneshwar, Orissa</i>
<b>Poornachandra Sarang</b>	<i>ABCOM Information Systems Pvt Ltd, Mumbai, Maharashtra</i>
<b>Manisha J Somavanshi</b>	<i>Indira Institute of Management, Pune, Maharashtra</i>
<b>V B Gadekar</b>	<i>Vishwakarma Institute of Technology, Pune, Maharashtra</i>
<b>Kavita Mahesh Kelkar</b>	<i>K J Somaiya College of Engineering, Mumbai, Maharashtra</i>
<b>Leena Thomas</b>	<i>Mar Athanasius College of Engineering, Kothamangalam, Kerala</i>
<b>A Sakthivel</b>	<i>Aditya Institute of Technology, Coimbatore, Tamil Nadu</i>
<b>A C Kaladevi</b>	<i>Sona College of Technology, Salem, Tamil Nadu</i>
<b>S Murali</b>	<i>PES College of Engineering, Mandya, Karnataka</i>
<b>K Chandra Sekaran</b>	<i>National Institute of Technology, Surathkal, Karnataka</i>
<b>Annapurna P Patil</b>	<i>MSR Institute of Technology, Bengaluru, Karnataka</i>

My sincere thanks are due to the editorial and publishing professionals of McGraw-Hill Education (India) for their keen interest and support in bringing out this edition in its present form.

**E Balagurusamy**

## **FEEDBACK**

**Remember to write to us.** We look forward to receiving your feedback, comments and ideas to enhance the quality of this book. You can reach us at [tmh.csefeedback@gmail.com](mailto:tmh.csefeedback@gmail.com). Please mention the title and the author's name as the subject.

**In case you spot piracy of this book, in any form, please do let us know.**