Ane's Student Edition

# Trouble Free C++


Hari Mohan Pandey

Ane Books
Pvt. Ltd.

# Trouble Free C++

**H.M. Pandey**
*Mukesh Patel School of Technology Management and Engineering, Shirpur Campus,*
*SVKM's NMIMS University, Mumbai.*

**Ane Books Pvt. Ltd.**

New Delhi ♦ Chennai ♦ Mumbai
Bengaluru ♦ Kolkata ♦ Thiruvananthapuram ♦ Lucknow

**Trouble Free C++**

H.M. Pandey



*Published by*

**Ane Books Pvt. Ltd.**
4821, Parwana Bhawan, 1st Floor, 24 Ansari Road,
Darya Ganj, **New Delhi** - 110 002, India
Tel.: +91(011) 23276843-44, Fax: +91(011) 23276863
e-mail: kapoor@anebooks.com, Website: www.anebooks.com

*Branches*

- Avantika Niwas, 1st Floor, 19 Doraiswamy Road, T. Nagar,
  **Chennai** - 600 017, Tel.: +91(044) 28141554, 28141209
  e-mail: anebooks_tn@airtelmail.in
- Plot No. 59, Sector-1, Shirwane, Nerul, **Navi Mumbai** - 400 706,
  Tel.: +91(022) 27720842, 27720851
  e-mail: anebooksmum@mtnl.net.in
- 38/1, 1st Floor, Model House, First Street, Opp. Shamanna Park,
  Basavannagudi, **Bengaluru** - 560 004, Tel.: +91(080) 41681432, 26620045
  e-mail: anebang@airtelmail.in
- Flat No. 16A, 220 Vivekananda Road, Maniktalla,
  **Kolkata** - 700 006, Tel.: +91(033) 23547119, 23523639
  e-mail: anebooks_vsn@bsnl.in
- # 6, TC 25/2710, Kohinoor Flats, Lukes Lane, Ambujavilasam Road,
  **Thiruvananthapuram** - 01, Kerala, Tel.: +91(0471) 4068777, 4068333
  e-mail: anebookstvm@airtelmail.in

*Representative Office*

- C-26, Sector-A, Mahanagar, **Lucknow** - 226 006
  Mobile - +91 93352 29971

ISBN (13): 978-93-8015-662-0



Printed at : Gopalji Enterprises, Delhi.

# Trouble Free C++

# Preface

This book is a comprehensive, hands-on guide to C++ programming- but one that does not assume you have programmed before. (People familiar with earlier programming or another structured programming language will, of course, have an easier time and can move through the early chapters quickly.)

Soon, you will write sophisticated programs that take full advantages of C++ exciting and powerful object-oriented nature. You will start at the beginning and when you have finished this book, you will have moved far along the road to C++ mastery. I have tried hard to cover at the least the fundamentals of every technique that a professional C++ will need to master.

I have also tried hard to stress the new ways of thinking needed to master C++ programming, so even experts in more traditional programming languages can benefit from this book. I have taken this approach because trying to force C++ into the framework of older programming languages is ultimately self-defeating you can not take advantage of its power if you continue to think within an older paradigm.

To make this book even more useful, there are extensive discussions of important topics left out of most other introductory books. There are whole chapter on objects, including non-trivial examples of building your own objects with C++. When I teach you process of inheritance at that time I have introduced you to develop C++ code on behalf of inherited diagrams. There is a chapter for input-output manipulators which shows you number of functions used in C++ for input/output control. In the same chapter I teach you about manipulator and ways to develop your own manipulator. There is a whole chapter for miscellaneous new features of C++. There is even a whole chapter on exception handling. There are methods of keeping file information through the chapter file handling. There are a chapter for Standard Template Library and for the String Class. There is a whole chapter for showing the hidden secrets of C++. The book also includes lots of examples with step by step explanation. The book also includes an extensive discussion of sorting and searching techniques and lots of tips and tricks. In sum, unlike many of the introductory books out there, I not only want to introduce you to a topic, but I go into it in enough depth that you can actually use the techniques for writing practical programs.

Now a confession: My original goal was to make this book a "**one-stop resource**", but, realistically, C++ has gotten far too big and far too powerful for any one book to do this. Nonetheless, if you finish this book, I truly believe that you will be in a position to begin writing

commercial-quality C++ programs! True mastery will take longer: I have tried to give suggestions that can take you to next level.

## ➔ HOW THIS BOOK IS ORGANISED?

The subject-matter of this book is divided into **18 chapters**. Each chapter has been written and developed with immense simplified programs **(except chapater-1, which is foundation chapter for various programming methodology)** which will clear the core concepts of the C++ language.

The book **"TROUBLE FREE C++ "**has been written specially for those students who are tyro in the field of programming. Inside the book you will find numerous programs instead of just code snippet to illustrate even the basic concept.

The book assumes no previous exposure to the C++ programming language. It also contains some good programming examples which might be useful for experienced programmers. All the programming examples given in the book have been tested on **VC++ compiler**, **Turbo C++ 3.0** and **Turbo C++ 4.5 compilers** under **windows** and **DOS**.

Each chapter contains a number of **examples** to explain the theoretical as well as practical concepts. **Every chapter is followed by questions to test the student performance and receptivity**.

**Here are short descriptions of the chapters:**

**Chapter 1** covers the topics of basic introduction of programming methodology and introduction of OOP like **structured programming** by which one can understand the elementary elements (**sequence structure, Loop or iteration and Decision structure**) of any programming language. The chapter also explains the basic approaches (**Bottom-up** and **Top-down**) which compiler applied for the compilation of programs. This chapter explains the basic concepts of object-oriented programming too. This chapter gives idea to programmer to categorise any programming language into **object-oriented or object based language.**

**Chapter 2** In this chapter, I have explained the historical development of C++ language. The chapter also gives introductory idea of tokens, variables, data types and basic structure of C++ program. In the same chapter, I have explained the method of **compiling** and **executing** the C++ program on **Turbo C++3.0, Turbo C++ 4.5 and VC++.**

**Chapter 3** This chapter introduces programmers about the behavior of operators used in C++. In this chapter, I have explained the most of the common features applied in C and C++ both, because as we say C++ is super set of C then **operator and expression** used in C must be implemented with C++ too.

**Chapter 4** covers the **operators used only with C++ not with C**. In this chapter, I have covered the operators like **scope resolution operator, reference variables, bool data type**. This chapter gives idea of **dynamic memory allocation** and operators **new and delete** for dynamic memory allocation in C++.

**Chapter 5** gives the idea of **declaring function (prototyping),** function of **main ()** function, introduction of **recursion**. It also gives the meaning of **call by reference** and **call by address** and difference between **call by reference** and **call by address**. Here I have explained the functionality of **inline function** and **function overloading** too.

**Chapter 6** gives the introduction of **class and objects** used in C++. Here I have put the comparison of **structure and class**, way of **accessing private data** and gives idea to **passing and returning objects**. In this chapter I have given some very crucial elements of C++ like **array of objects, friend function, Static class members and constant member function**. All these concepts play very important role in software development.

**Chapter 7** covers the behaviors of **constructor.** Here I show the role of different types of constructor like **default constructor, constructor with parameters, copy constructor.** This chapter also gives the ideas of dynamic constructor and destructor.

**Chapter 8** gives ideas to programmer to **overload different types of operators** used in C++ like, **binary operators, assignment operators, unary operators**. Overloading with the help of **friend function** and rules of overloading any operator and way for **type conversion** too.

**Chapter 9** In this chapter we will deal the concept of **inheritance** different types of inheritance *i.e.* **single level, multilevel, multiple, hierarchical and hybrid**. Here I have also defined the different visibility modifier with respect to inheritance. Application of **constructor and destructor** in inheritance and concept of **containership** is also defined in the same chapter.

**Chapter 10** gives the way of implementing the concepts like **pointers to objects, this pointer**, and way of **binding**, what is **virtual function** and how to work with **virtual function**, rules for **virtual function**? This chapter also gives the comparison of **virtual function, pure virtual** function the fundamental concepts of **object slicing** and **virtual destructor.**

**Chapter 11** explains the concepts of C++ **stream classes** and **formatted** and **unformatted** input and output operation applied in C++. In the same chapter I have also given the concept of **manipulator.**

**Chapter 12** gives the idea of how to handle **file** in C++ programming language? This chapter introduces programmer about the fundamental concepts of **file streams, way of opening and closing file, different modes of opening** a text file in C++. This chapter also gives the approaches to **check end of any file, Random access in file**. In the same chapter I have put the introductory ideas of **command line argument** and way of working with **binary mode** and **error handling mechanism** with file handling.

**Chapter 13** In this chapter, I have put miscellaneous new features of C++ like **explicit keyword, mutable keyword, constant object, namespace, typied operator**, etc.

**Chapter 14** covers the introductory idea of **template programming** *i.e.* **function template and class template.**

**Chapter 15** This chapter deals with basic concepts of **exception handling mechanism**, **how to handle exception with the help of class**? How to **re-throw** an exception, how to **catch** all exceptions.

**Chapter 16** gives the introduction of **standard template library**. Gives the behavior of **container classes**, introduces approaches for working with **vector** and working with **list.**

**Chapter 17** covers the topics like declaring and using **strings.** Introduces the principle by which one can differentiate **C++ strings with C- style strings**. Introduces operators for **string objects, methods for string classes, arrays of string and iterating string** etc.

**Chapter 18** gives the fundamental ideas of very basic elements used in C++. This chapter also clarifies some of the critical questions generated in students/faculty/professionals mind.

**Appendix 1** gives the explanations of some critical questions and answer.

**Appendix 2** presents some language-technical elements.

**Appendix 3** discusses the technical questions which are generally asked in technical interviews.

All efforts have been made to keep errors to barest minimum. However, there is possibility that some errors might have crept in inadvertently. I would, therefore, be grateful if such oversights are pointed out by the readers. Suggestions for improvement of the text from readers are welcome.

**–Author**

# Contents