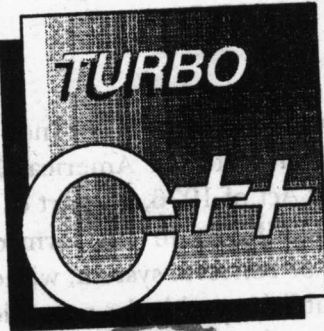


USING TURBO C++

F  
TP312.10  
2

# USING TURBO C++

Herbert Schildt



Osborne McGraw-Hill

Berkeley New York St. Louis San Francisco  
Auckland Bogotá Hamburg London Madrid  
Mexico City Milan Montreal New Delhi Panama City  
Paris São Paulo Singapore Sydney  
Tokyo Toronto



X005871

Handwritten signature and number: 913319

**Osborne McGraw-Hill**  
2600 Tenth Street  
Berkeley, California 94710  
U.S.A.

For information on translations and book distributors outside of the U.S.A., please write to Osborne McGraw-Hill at the above address.

TAB BOOKS is a McGraw-Hill Company. TAB BOOKS offers software for sale. For information and a catalog, please contact TAB Software Department, Blue Ridge Summit, PA 17294-0850.

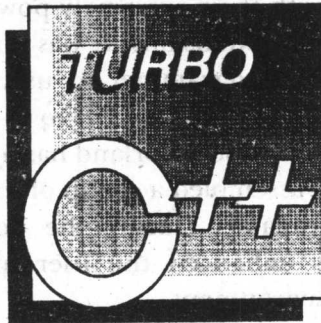
### **Using Turbo C++**

Copyright © 1990 by McGraw-Hill, Inc. All rights reserved. Printed in the United States of America. Except as permitted under the Copyright Act of 1976, no part of this publication may be reproduced or distributed in any form or by any means, or stored in a database or retrieval system, without the prior written permission of the publisher, with the exception that the program listings may be entered, stored, and executed in a computer system, but they may not be reproduced for publication.

234567890 DOC 99876543210

ISBN 0-07-881610-6

Information has been obtained by Osborne McGraw-Hill from sources believed to be reliable. However, because of the possibility of human or mechanical error by our sources, Osborne McGraw-Hill, or others, Osborne McGraw-Hill does not guarantee the accuracy, adequacy, or completeness of any information and is not responsible for any errors or omissions or the results obtained from use of such information.



## ***Introduction***

As the 1980s drew to a close, a new way to approach the job of programming was beginning to take hold. This new method is called object oriented programming. Object oriented programming (OOP for short) embodies all features of structured programming (its predecessor), but provides the programmer with an exciting new way to analyze and solve programming tasks. In the course of this book, you will learn why OOP is important and how OOP is implemented in Turbo C++.

As you probably know, C++ is the object oriented version of C. C was the preeminent language of the 1980s and is still in wide use at the time of this writing. (In fact, it will most likely be a very long time before C is deemed obsolete.) C is known for its efficiency, power, and elegance. C++ preserves these important qualities but adds support for OOP. It is this combination that makes C++ the single most important object oriented programming language available today.

In 1988, Borland International, Inc. began secret work on Turbo C++. They started with their extremely powerful Turbo C and added to it the C++ OOP extensions. It was not an easy task. C++, although easy for the programmer to use, is a rigorous exercise in compiler construction. In fact, Turbo C++ is the largest and most ambitious language project Borland has undertaken. To create Turbo C++, Borland assembled a group of some of the best compiler programmers available. During the development of Turbo C++, Borland also enhanced the user interface to its interactive development environment.

As you will learn while working through this book, Turbo C++ provides a powerful yet flexible programming environment. It produces tight, efficient code and supports hundreds of library functions and classes. Although Turbo C++ was an ambitious project, it is fair to say that it lives up to its ancestor's reputation and deserves the title *Turbo!*

## **About This Book**

---

This book teaches the Turbo C++ programming language, beginning with its base language, C. Because C++ is a superset of the C language, you need to know C before you can learn C++. Therefore, if you don't already know C, you can still use this book because it contains all the background information you need. The book also covers the Turbo C++ integrated environment, the editor, and many library functions. By the time you finish this book, you will definitely be able to call yourself a Turbo C++ programmer.

This book is based on the philosophy that learning is best accomplished by doing. Toward this end, the book contains many sample programs that you can compile and run. In fact, virtually every major feature of Turbo C++ has a sample program to demonstrate it.

## **How This Book Is Organized**

---

This book is organized in such a way that it can be effectively used by two different types of programmers: those who already know C and just want to learn about the C++ extensions, and those who are new to both C and C++.

Part I presents an overview of Turbo C++, introduces the Turbo C++ integrated environment, and discusses the editor. In Part II, the C-like features of C++ are presented. If you don't already know C, be sure to carefully work through Part II. However, if you are proficient at C, you can skip to Part III, in which the C++ extensions to C are fully described.

Organizing the book in this fashion prevents the experienced C programmer from having to wade through reams of material he or she already knows, yet still provides the beginner with sufficient background material.

## **Conventions Used in This Book**

---

In this book, keywords, operators, function names, and variable names are shown in bold when referenced in text. General forms are shown in italics. Also, when referencing a function name in text, the name is followed by parentheses. In this way, you can easily distinguish a variable name from a function name.

## **Additional Help from Osborne/McGraw-Hill**

---

Osborne/McGraw-Hill provides top-quality books for computer users at every level of computing experience. To help you build

your skills, we suggest that you look for the books in the following Osborne/McGraw-Hill series that best address your needs.

The "Teach Yourself" series is perfect for beginners who have never used a computer before or who want to gain confidence in using program basics. These books provide a simple, slow-paced introduction to the fundamental usage of popular software packages and programming languages. The "Mastery Learning" format ensures that concepts are learned thoroughly before progressing to new material. Plenty of exercises and examples (with answers at the back of the book) are used throughout the text.

The "Made Easy" series is also for beginners or users who may need a refresher on the new features of an upgraded product. These in-depth introductions guide users step-by-step from the program basics to intermediate-level usage. Plenty of "hands on" exercises and examples are used in every chapter.

The "Using" series presents fast-paced guides that quickly cover beginning concepts and move on to intermediate-level techniques, and even some advanced topics. These books are written for users already familiar with computers and software, and who want to get up to speed fast with a certain product.

The "Advanced" series assumes that the reader is already an experienced user who has reached at least an intermediate skill level, and is ready to learn more sophisticated techniques and refinements.

The "Complete Reference" books are a series of handy desktop references for popular software and programming languages that list every command, feature, and function of the product, along with brief, detailed descriptions of how they are used. Books are fully indexed and often include tear-out command cards. The "Complete Reference" series is ideal for all users, beginners and pros.

The "Pocket Reference" is a pocket-sized, shorter version of the "Complete Reference" series, and provides only the essential commands, features, and functions of software and programming

languages for users who need a quick reminder of the most important commands. This series is also written for all users and every level of computing ability.

The "Secrets, Solutions, Shortcuts" series is written for beginning users who are already somewhat familiar with the software, and for experienced users at intermediate and advanced levels. This series provides clever tips and points out shortcuts for using the software to greater advantage. Traps to avoid are also mentioned.

Osborne/McGraw-Hill also publishes many fine books that are not included in the series described above. If you have questions about which Osborne book is right for you, ask the salesperson at your local book or computer store, or call us toll-free at 1-800-262-4729.

## ***Diskette Offer***

---

There are many useful and interesting functions and programs contained in this book. If you're like me, you probably would like to use them, but hate typing them into the computer. When I key in routines from a book it always seems that I type something wrong and spend hours trying to get the program to work. For this reason, I am offering the source code on diskette for all the functions and programs contained in this book, for \$24.95. Just fill in the order blank on the next page and mail it, along with your payment, to the address shown. Or, if you're in a hurry, just call 217-586-4021 (the number of my consulting office) and place your order by telephone. (Visa and Mastercard accepted.)



Please send me \_\_\_\_\_ copies, at \$24.95 each, of the programs in *Using Turbo C++* on an IBM-compatible diskette.

Foreign orders only: Checks must be drawn on a U.S. bank. Please add \$5 shipping and handling.

\_\_\_\_\_  
Name

\_\_\_\_\_  
Address

\_\_\_\_\_  
City

\_\_\_\_\_  
State

\_\_\_\_\_  
Zip

\_\_\_\_\_  
Telephone

Diskette size (check one): 5 1/4" \_\_\_\_\_ 3 1/2" \_\_\_\_\_

Method of payment: check \_\_\_\_\_ Visa \_\_\_\_\_ MC \_\_\_\_\_

Credit card number: \_\_\_\_\_

Expiration date: \_\_\_\_\_

Signature: \_\_\_\_\_

Send to:

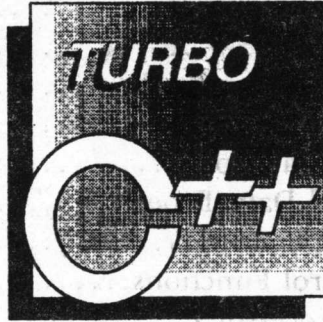
Herbert Schildt

RR 1, Box 130

Mahomet, IL 61853

or phone: 217-586-4021

This offer subject to change or cancellation at any time.



# Contents at a Glance

Why This Book Is for You.....	1
<b>PART I</b>	<b>I</b>
<b>Getting Started</b>	<b>3</b>
1: Turbo C++ in Perspective .....	5
2: The Turbo C++ Integrated Development Environment .....	13
3: Using the Turbo C++ Editor .....	39
<b>PART II</b>	<b>II</b>
<b>The C Language</b>	<b>61</b>
4: Some C Essentials.....	63
5: Variables, Constants, Operators, and Expressions ...	93

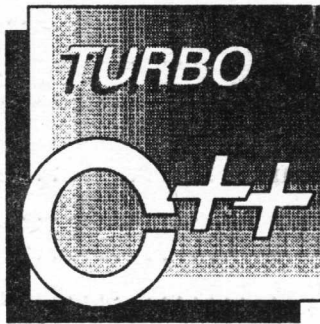
6: Program-Control Statements.....	121
7: Arrays and Strings.....	157
8: Pointers.....	185
9: Functions: A Closer Look.....	215
10: Input, Output, and Disk Files.....	263
11: User-Defined Data Types.....	317
12: User-Defined Data Types.....	343
13: Advanced Operators.....	387
14: Screen Control Functions.....	405
15: The C Preprocessor.....	437

## **PART**

## **III**

### ***Using Turbo C++'s Object Oriented Features* 455**

16: An Overview of C++.....	457
17: A Closer Look at Classes.....	483
18: Function and Operator Overloading.....	523
19: Inheritance, Virtual Functions, and Polymorphism..	561
20: Using C++'s I/O Class Library.....	585
21: Miscellaneous C++ Topics.....	619
<b>A:</b> Some Common C Library Functions.....	647
<b>B:</b> Using Turbo C++'s Debugger.....	699
<b>C:</b> Turbo C++'s Memory Models.....	711
<b>D:</b> Using the VROOMM Overlay Technology.....	723
<b>E:</b> Using the Command-Line Compiler.....	727
<b>F:</b> Compiling Multiple File Programs.....	733



# Contents

	<b>Introduction .....</b>	<b>xxi</b>
<b>1</b>	<b>Why This Book Is for You .....</b>	<b>1</b>
	<i>Part I</i> <b>Getting Started</b>	
<b>1</b>	<b>Turbo C++ in Perspective .....</b>	<b>5</b>
	The Origins of C++ .....	6
	What Is Object Oriented Programming? .....	7
	C++: True to the Spirit of C .....	8
	What Can C++ Be Used For? .....	9
	Compilers Versus Interpreters .....	10
	A Word to C Programmers .....	11
<b>2</b>	<b>The Turbo C++ Integrated Development Environment .....</b>	<b>13</b>

TURBO C++

21

Executing Turbo C++ .....	14
Using the Mouse .....	15
The Main Menu .....	16
Dialog Boxes .....	18
Turning On Full Menus .....	20
Exploring the Main Menu .....	21
The System Menu .....	22
File .....	22
Edit .....	23
Search .....	24
Run .....	24
Compile .....	25
Debug .....	26
Project .....	26
Options .....	26
Window .....	27
Help .....	30
The Hot Keys .....	30
Using Turbo C++'s Context-Sensitive Help .....	31
Understanding Windows .....	31
Sizing and Moving Windows .....	35
The Editor Window .....	35
The Message Window .....	36
The Status Line .....	36
<b>3 Using the Turbo C++ Editor .....</b>	<b>39</b>
Editor Commands .....	40
Invoking the Editor and Entering Text .....	40
Deleting Characters, Words, and Lines .....	42
Moving, Copying, and Deleting Blocks of Text .....	44
Using the Clipboard .....	46
More on Cursor Movement .....	47

	Characters and Strings .....	81
	Strings.....	82
	printf( ): A Quick Review .....	84
	scanf( ): A Quick Review .....	86
	Semicolons, Braces, and Comments.....	87
	Indentation Practices.....	88
	The C Library.....	89
	The C Keywords.....	90
	Review of Terms.....	91
<b>5</b>	<b>Variables, Constants, Operators, and Expressions ...</b>	<b>93</b>
	Identifier Names .....	94
	Data Types.....	95
	Type Modifiers .....	96
	Declaring Variables .....	99
	Where Variables Are Declared .....	100
	Constants .....	102
	Hexadecimal and Octal Constants .....	103
	String Constants .....	104
	Backslash Character Constants .....	104
	Variable Initializations .....	105
	Operators.....	107
	Arithmetic Operators.....	107
	Increment and Decrement.....	108
	Relational and Logical Operators.....	110
	The Assignment Operator .....	115
	Expressions .....	116
	Type Conversion in Expressions .....	116
	Casts .....	118
	Spacing and Parentheses .....	119
<b>6</b>	<b>Program-Control Statements .....</b>	<b>121</b>
	The if Statement.....	122

Find and Replace .....	47
Setting and Finding Place Markers .....	51
Saving and Loading Your File .....	51
Understanding Auto-Indentation .....	53
Moving Blocks of Text to and from Disk Files .....	53
Pair Matching .....	54
Miscellaneous Commands .....	55
Command Summary .....	56
Changing the Editor Defaults .....	56
Invoking Turbo C++ with a Filename .....	60

***Part II***  
***The C Language***

<b>4</b>	<b>Some C Essentials .....</b>	<b>63</b>
	Preparing the IDE .....	64
	C Is Case Sensitive .....	65
	A Simple C Program .....	65
	A Closer Look .....	66
	Handling Errors .....	69
	Errors Versus Warnings .....	70
	A Second Program .....	71
	A Variation .....	73
	A Quick Review .....	73
	What Is a C Function? .....	74
	Functions with Arguments .....	75
	Functions Returning Values .....	77
	The General Form of a Function .....	78
	Two Simple Commands .....	78
	The if .....	79
	The for Loop .....	79
	Blocks of Code .....	80

Using the else Statement .....	124
The if-else-if Ladder .....	125
The Conditional Expression .....	127
Nested ifs .....	128
The switch Statement .....	129
The default Statement.....	131
A Closer Look at the break Statement .....	132
Nested switch Statements .....	134
Loops .....	136
The for Loop .....	137
for Loop Basics .....	137
Variations of the for Loop .....	139
The Infinite Loop .....	142
Breaking Out of a for Loop.....	143
Using for Loops with No Bodies .....	143
The while Loop .....	144
The do-while Loop .....	146
Nested Loops .....	148
Loop Breaking .....	151
The continue Statement.....	153
Labels and goto .....	155
<b>7   Arrays and Strings .....</b>	<b>157</b>
One-Dimensional Arrays .....	158
No Bounds Checking .....	159
One-Dimensional Arrays Are Lists.....	160
Strings.....	161
Reading a String From the Keyboard.....	162
Some String Library Functions.....	163
Using the Null Terminator .....	168
A printf( ) Variation.....	169
Two-Dimensional Arrays .....	170
Arrays of Strings.....	172



	Multidimensional Arrays.....	174
	Array Initialization .....	175
	Unsize Array Initializations .....	176
	A Sub Hunt Game .....	178
<b>8</b>	<b>Pointers .....</b>	<b>185</b>
	Pointers Are Addresses .....	186
	Pointer Variables .....	187
	The Pointer Operators .....	187
	The Base Type Is Important .....	189
	Pointer Expressions .....	190
	Pointer Assignments .....	190
	Pointer Arithmetic .....	191
	Pointer Comparisons.....	193
	Pointers and Arrays.....	193
	Indexing a Pointer.....	196
	Pointers and Strings .....	196
	Getting the Address of an Array Element .....	198
	Arrays of Pointers.....	199
	An Interesting Example Using Arrays and Pointers.....	202
	Pointers to Pointers.....	207
	Initializing Pointers .....	209
	Problems with Pointers.....	210
<b>9</b>	<b>Functions: A Closer Look .....</b>	<b>215</b>
	The General Form of a Function .....	216
	The return Statement.....	217
	Returning from a Function .....	217
	Returning a Value .....	218
	Functions Returning Noninteger Values .....	222
	Using Function Prototypes.....	223
	Returning Pointers.....	225