C: The Complete Reference

73.8?

C: The Complete Reference
Second Edition

Herbert Schildt

Osborne McGraw-Hill

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

DS71/22

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., write to Osborne **McGraw-Hill** at the above address.

A complete list of trademarks appears on page 803.

C: The Complete Reference, Second Edition

34567890 DOC 90

ISBN 0-07-881538-X

PREFACE

Most of the time, writing a book is challenging, tedious, exciting, frustrating, and a lot of work all rolled into one. But the creation of *C: The Complete Reference, Second Edition,* had an extra, almost intangible component. I had been given the mission of creating a complete reference guide to the C programming language. Not just *any* book, but the *complete* book. Whenever I write a book, I try to be thorough, but in this book I had to be even more thorough. It was the word "complete" that kept haunting me. I knew that programmers from around the world would be buying my book because it was the *complete* reference. Finally, I realized, as I'm sure you must, that no one book on C can hope to cover all of its nuances, tricks, and uses. The C language is just too rich and powerful. But I do say this: The writing of *C: The Complete Reference, Second Edition,* was a labor of love. I am confident that you will be pleased with the result.

Is This Book for You?

This C reference is designed for all C programmers, regardless of their experience level. It does assume, however, a reader able to create at least a simple C program. If you are just learning C, this book will make an excellent companion to any C tutorial and serve as a source of answers to your specific questions.

What's New in the Second Edition

For the most part, I have left the basic structure of the book unchanged from the first edition. A couple of chapters have been moved, but the presentation of material within each chapter reflects the original book. Also, much new material has been added, including expanded coverage of graphics.

The major differences between the first and the second editions are caused by conformance to the new ANSI C standard. When the first edition was written, the ANSI C standard, still in its infancy, was being continually changed. Also, at that time no compilers conformed closely to the evolving standard. Therefore, most of the original edition reflected the more common de facto C standard as defined by Kernighan and Ritchie. However, now that the standard is stable, and given the fact that all major C compilers comply with the standard, it seemed time to switch to the ANSI C standard. (Important points and differences between the old de facto standard and the new ANSI standard are still covered by this book, however, so you don't have to be using an ANSI C standard compiler to utilize the book.)

The major changes caused by conformance to the ANSI C standard are function prototyping, which is used throughout the book, and inclusion of all header files. In older versions of C, most header files were not needed for programs to run correctly. However, in order to take advantage of function prototyping, all header files must be included because they contain the prototypes of the standard library functions.

Another change caused by the adoption of the ANSI C standard is that the modern form of function declaration is used. (The older—sometimes called "classic"—form is still explained in detail, but the code examples use the modern form.)

What's Inside

This book covers in detail all aspects of the C language and its libraries. It includes both the old de facto C standard and the ANSI C standard. However, the main emphasis is on the ANSI C standard. The book is divided into five parts, covering

- The C language
- The C libraries
- Common algorithms and applications
- The C programming environment
- C++: C's newest direction

Part One provides a thorough discussion of the keywords, preprocessor directives, and features of the C language.

Part Two discusses the standard C library. Both the K&R and ANSI standard library functions are covered. Also included are some functions common to the DOS programming environment, including DOS calls and graphics. For DOS-specific functions, Microsoft C was chosen for the examples.

Part Three covers some of the more common and important algorithms and applications that all C programmers should have in their toolbox. You will find Chapter 22, which deals with AI problem-solving techniques, particularly interesting.[1]

Part Four covers the C programming environment including such things as interfacing to assembly code, efficiency, porting, and debugging.

Part Five examines C's newest direction, C++. C++ is a superset of C designed to handle the rigorous needs of large projects and to aid in program verification.

There are many, many working functions and programs contained in this book. If you are like me, you like having a lot of routines to draw upon but hate entering them into the computer. It always seems that I type something wrong and spend hours trying to get the program to work. For this reason, I am offering on disk the source code to 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

Herbert Schildt
RR 1, Box 130
Mahomet IL 61853

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).

—HS

[1]For a wider assortment of C applications, refer to my book *Advanced C, Second Edition* (Berkeley, Calif.: Osborne/McGraw-Hill, 1988). For those interested in AI, see my book *Artificial Intelligence Using C* (Berkeley, Calif.: Osborne/McGraw-Hill, 1987).

Order Form

Please send me ____________ copies, at $24.95 each, of the programs in *C: The Complete Reference, 2nd Edition.* Foreign orders, please add $5 shipping and handling.

____
Name

____
Address

______________________ ____________ ____________
City State ZIP

____
Telephone

Disk size: 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 is solely the offering of the author. Osborne/McGraw-Hill takes no responsibility for the fulfillment of this offer. Please allow four to six weeks for delivery.

CONTENTS

PART ONE

The C Language

The first part of this reference guide presents a thorough discussion of the C programming language. Chapter 1 provides a quick overview of the C language—the more knowledgeable programmer may wish to skip directly to Chapter 2. Chapter 2 examines C's built-in data types, variables, operators, and expressions. Next, Chapter 3 presents program control statements. Chapter 4 discusses arrays and strings. Chapter 5 looks at pointers. Chapter 6 deals with functions, and Chapter 7 discusses structures, unions, and user-defined types. Chapter 8 examines console I/O. Chapter 9 covers file I/O, and Chapter 10 discusses the C preprocessor and comments.

The material in this section (and most of the material in the book) reflects the ANSI standard for C. However, the original C standard as derived from the UNIX version 5 C is also covered, and important differences are noted. The book covers both ANSI and the original C to ensure that you find information relevant to your C programming environment.

An Overview of C

ONE

The purpose of this chapter is to present an overview of the C programming language, its origins, its uses, and its underlying philosophy. This chapter is mainly for newcomers to C.

The Origins of C

C was invented and first implemented by Dennis Ritchie on a DEC PDP-11 that used the UNIX operating system. C is the result of a development process that started with an older language called BCPL, which is still in use primarily in Europe. BCPL was developed by Martin Richards, and it influenced a language called B, which was invented by Ken Thompson. B led to the development of C in the 1970s.

For many years, the de facto standard for C was the version supplied with the UNIX version 5 operating system. It is described in *The C Programming Language* by Brian Kernighan and Dennis Ritchie (Englewood Cliffs, N.J.: Prentice-Hall, 1978). With the popularity of microcomputers, a large number of C implementations were created. In a near miracle, the source codes of most of these implementations were