Peter A. Darnell Philip E. Margolis

Software Engineering in C

C 语言及软件工程 [英]

Springer-Verlag
World Publishing Corp

Peter A. Darnell Philip E. Margolis

Software Engineering in C

Springer-Verlag
World Publishing Corp

Peter A. Darnell
Senior Software Engineer
Stellar Computer Inc.
95 Wells Avenue
Newton, MA 02159, USA

Philip E. Margolis
Senior Writer
Apollo Computer Inc.
330 Billerica Road
Chelmsford, MA 01824, USA

Library of Congress Cataloging-in-Publication-Data
Margolis, Philip E.
Software engineering in C.
(Springer books on professional computing)
Bibliography: p.
Includes index.
1. Computer software–Development. 2. C (Computer program language) I. Darnell, Peter. II. Title.
III. Series.
QA76.76.D47M364 1988 005.13'3 87–27515

Text prepared by the authors using Interleaf/Apollo DN 3000 workstation.

Reprinted by World Publishing Corporation, Beijing, 1990
for distribution and sale in The People's Republic of China only
ISBN 7-5062-0784-2

ISBN 0-387-96574-2 Springer-Verlag New York Berlin Heidelberg
ISBN 3-540-96574-2 Springer-Verlag Berlin Heidelberg New York

Preface

This book describes the C programming language and software engineering principles of program construction. The book is intended primarily as a textbook for beginning and intermediate C programmers. It does not assume previous knowledge of C, nor of any high-level language, though it does assume that the reader has some familiarity with computers. While not essential, knowledge of another programming language will certainly help in mastering C.

Although the subject matter of this book is the C language, the emphasis is on software engineering—making programs readable, maintainable, portable, and efficient. One of our main goals is to impress upon readers that there is a huge difference between programs that merely work, and programs that are well-engineered, just as there is a huge difference between a log thrown over a river and a well-engineered bridge.

The book is organized linearly so that each chapter builds on information provided in the previous chapters. Consequently, the book will be most effective if chapters are read sequentially. Readers with some experience in C, however, may find it more useful to consult the table of contents and index to find sections of particular interest.

Each chapter is autonomous inasmuch as it covers a single, well-defined area of the C language, such as scalar data types or control flow. Moreover, the chapters themselves are organized linearly, so that each section uses information provided in earlier sections. Again, experienced C programmers may want to skim introductory sections.

Although this book covers all C features, it makes no claim to being a reference manual. The organization and pace is designed for those learning the language rather than those who already know the language. If you plan to do extensive programming in C, we recommend that you supplement our book with *C: A Reference Manual*, by Harbison and Steele.

This book describes all features of the C language defined by Kernighan and Ritchie (known as the K&R standard), as well as all features defined in the C Standard proposed by the American National Standards Institute (ANSI). Where the two versions differ, we highlight the difference either by explicitly describing each version in the text or by describing the ANSI feature in a shaded box. A list of differences between the two standards appears in Appendix E. For more information about the ANSI Standard, you should read the official specification, which you can obtain by writing to:

American National Standards Institute
1430 Broadway
New York, NY 10018

In addition to using shaded boxes to describe ANSI extensions, we also use boxes to highlight common errors made by C programmers. These "Bug Alerts" are intended as buoys to mark places where we and others have run aground.

The examples in this book have all been tested on three machines: A PC-compatible Zenith Z-151 running the Microsoft Version 3.0 C compiler, an Apollo DN3000 running the DOMAIN C compiler (Version 4.78), and a Sun Microsystems 3/50 computer running Version 3.1 of the Sun compiler. Whenever possible, we have tried to use realistic examples gleaned from our own experiences. Occasionally we refer to "our machine", which means any of these three computers. The most significant aspect of "our machine" is that it allocates four bytes for **int**s.

Appendix A describes all of the runtime library functions defined in the ANSI standard. Many of these functions are derived from UNIX functions and are present in current C runtime libraries. Be careful, though, because some ANSI functions behave differently from identically-named functions in older libraries.

Appendix B shows the syntax of the ANSI C language in the form of "railroad diagrams." Each rectangular box in a diagram represents another diagram defined elsewhere. Items that appear in ovals are C keywords and predefined names that must appear exactly as they are written. Circles are used to represent punctuation tokens. Unless stated otherwise, it is always legal to insert spaces and newlines between one item and another.

Appendix C lists all names reserved by the ANSI standard. This includes keywords, library function names, and type definitions used by the li-

brary. You should avoid declaring variables that conflict with these names.

Appendix D lists certain ranges that an ANSI-conforming compiler must support. This includes, for example, the range of values that must be representable in a floating-point number.

Appendix E lists the major differences between the ANSI Standard and the K&R standard. Each entry in this list contains a reference to the section in the book where the difference is described. Note that this list is not exhaustive.

Appendix F contains the source listings for a C interpreter. In Chapter 12, we refer to this program as an example of using good engineering techniques to produce a large software product.

Acknowledgements

First and foremost, we wish to acknowledge our debt to the authors of the two most influential books about C: Samuel Harbison, Brian Kernighan, Dennis Ritchie, and Guy Steele.

In addition to the books by these authors, we also leaned heavily on the Draft Proposed ANSI Standard, and we thank all of the members of the ANSI X3J11 Subcommittee for their efforts in creating this document.

Many people reviewed various parts of this book at various stages. We are indebted to all of them, particularly David Boyce, Gary Bray, Clem Cole, Karen Darnell, Norman Garfinkle, John Humphrys, Ben Kingsbury, Diane Margolis, Doug McGlathery, Beth O'Connell, John Peyton, Bill Plauger, Barry Rosenberg, Jim Van Sciver, Kincade Webb, Bob Weir, and John Weiss. We are also indebted to the software development team at Dynatech Data Systems, especially Elizabeth Stark and Jonathan Edney. Special thanks go to Chuck Connell, Sam Harbison, and Tom Pennello, who read the manuscript in its entirety and offered numerous invaluable suggestions. We would also like to thank Kathy Ford for her assistance in preparing the artwork, and Andrea Morris for her expert editorial advice. Naturally, we accept responsibility for any flaws that remain.

Finally we would like to thank Apollo Computer Inc. and Stellar Computer Inc. for providing the working environments in which to produce this book. The entire book was formatted using the Interleaf Version 3.0 electronic publishing system running on an Apollo DN3000 workstation.

Suggested Reading

We have found the following books extremely helpful in mastering C and in absorbing general software engineering principles.

Aho, Alfred V., and Jeffrey P. Ullman. *Principles of Compiler Design*. Addison-Wesley, 1972.

Brooks, Frederick P., Jr. *The Mythical Man Month: Essays on Software Engineering*. Addison-Wesley, 1974.

Date, C. J. *An Introduction to Database Systems*. 4th ed. Addsion-Wesley, 1986.

Foley, J. D., and A. Van Dam. *Fundamentals of Computer Graphics*. Addison-Wesley, 1980.

Harbison, Samuel P., and Guy L. Steele Jr. *C: A Reference Manual*. 2d ed. Prentice Hall, 1984.

Kernighan, Brian W., and P. J. Plauger. *Software Tools*. Addison-Wesley, 1976.

Kernighan, Brian W., and P. J. Plauger. *Elements of Programming Style*. McGraw-Hill, 1978.

Kernighan, Brian W., and Dennis M. Ritchie. *The C Programming Language*. Prentice-Hall, 1978.

Knuth, Donald E. *The Art of Computer Programming*. Addison-Wesley, 1973.

Shore, John. *The Sacher Torte Algorithm*. Penguin Books, 1986.

Table of Contents

Chapter 4

Chapter 5

Chapter 6

List of Figures

List of Tables

List of Boxes