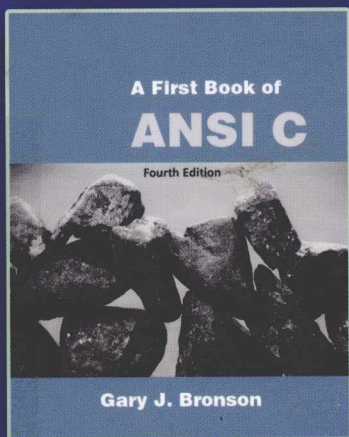


国外计算机科学教材系列

 CENGAGE
Learning™

标准C语言基础教程 (第四版)

A First Book of ANSI C
Fourth Edition



英文版

[美] Gary J. Bronson 著

 电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

<http://www.phei.com.cn>

国外计算机科学教材系列

标准 C 语言基础教程

(第四版)

(英文版)

A First Book of ANSI C
Fourth Edition

[美] Gary J. Bronson 著

電子工業出版社
Publishing House of Electronics Industry
北京 · BEIJING

内 容 简 介

这是一本讲授如何用C语言编程的经典教材。通过大量的实例和练习,本书系统介绍了C语言编程中涉及的输入输出、条件选择、循环、函数、数组、字符串、数据类型、宏、指针、动态数据结构、库函数等基本内容,使读者在学习之后就能很快掌握C语言编程的精髓。本书还对面向对象的C++编程技术进行了介绍,书中对计算机软硬件发展历史的回顾同样是这一版本的特色之一。书中每一章都附有大量习题,附录还列出了主要习题的答案。

本书结构合理,内容深入浅出,适合用做高等学校本科和专科教材,也适合初学编程的自学者。

A First Book of ANSI C, Fourth Edition by Gary J. Bronson. Copyright © 2006 Cengage Learning Asia Pte Ltd.

Original edition published by Cengage Learning. All Rights reserved.

本书原版由圣智学习出版公司出版。版权所有,盗印必究。

Publishing House of Electronics Industry is authorized by Cengage Learning to publish and distribute exclusively this reprint edition. This edition is authorized for sale in China Mainland. Unauthorized export of this edition is a violation of the Copyright Act. No part of this publication may be reproduced or distributed by any means, or stored in a database or retrieval system, without the prior written permission of the publisher.

本书英文影印版由圣智学习出版公司授权电子工业出版社独家出版发行。此版本经授权仅限在中国大陆销售。未经授权的本书出口将被视为违反版权法的行为。未经出版者预先书面许可,不得以任何方式复制或发行本书的任何部分。

978-1-4188-3556-0

本书封面贴有 Cengage Learning 防伪标签,无标签者不得销售。

版权贸易合同登记号 图字:01-2006-1680

图书在版编目(CIP)数据

标准C语言基础教程:第4版 = A First Book of ANSI C, Fourth Edition: 英文 / (美)布朗森(Bronson, G. J.)著.

北京:电子工业出版社,2012.8

国外计算机科学教材系列

ISBN 7-121-17900-6

I. ①标... II. ①布... III. ①C语言-程序设计-高等学校-教材-英文 IV. ①TP312

中国版本图书馆CIP数据核字(2012)第188049号

责任编辑:史鹏举

印 刷: 三河市鑫金马印装有限公司

装 订:

出版发行:电子工业出版社

北京市海淀区万寿路173信箱 邮编:100036

开 本:787×980 1/16 印张:38.5 字数:862千字

印 次:2012年8月第1次印刷

定 价:59.00元

凡所购买电子工业出版社的图书有缺损问题,请向购买书店调换;若书店售缺,请与本社发行部联系。联系及购电话:(010)88254888。

质量投诉请发邮件至 zltz@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线:(010)88258888。

出版说明

21世纪初的5至10年是我国国民经济和社会发展的关键时期,也是信息产业快速发展的关键时期。在我国加入WTO后的今天,培养一支适应国际化竞争的一流IT人才队伍是我国高等教育的重要任务之一。信息科学和技术方面人才的优劣与多寡,是我国面对国际竞争时成败的关键因素。

当前,正值我国高等教育特别是信息科学领域的教育调整、变革的重大时期,为使我国教育体制与国际化接轨,有条件的高等院校正在为某些信息学科和技术课程使用国外优秀教材和优秀原版教材,以使我国在计算机教学上尽快赶上国际先进水平。

电子工业出版社秉承多年来引进国外优秀图书的经验,翻译出版了“国外计算机科学教材系列”丛书,这套教材覆盖学科范围广、领域宽、层次多,既有本科专业课程教材,也有研究生课程教材,以适应不同院系、不同专业、不同层次的师生对教材的需求,广大师生可自由选择 and 自由组合使用。这些教材涉及的学科方向包括网络与通信、操作系统、计算机组织与结构、算法与数据结构、数据库与信息处理、编程语言、图形图像与多媒体、软件工程等。同时,我们也适当引进了一些优秀英文原版教材,本着翻译版本和英文原版并重的原则,对重点图书既提供英文原版又提供相应的翻译版本。

在图书选题上,我们大都选择国外著名出版公司出版的高校教材,如Pearson Education培生教育出版集团、麦格劳-希尔教育出版集团、麻省理工学院出版社、剑桥大学出版社等。撰写教材的许多作者都是蜚声世界的教授、学者,如道格拉斯·科默(Douglas E. Comer)、威廉·斯托林斯(William Stallings)、哈维·戴特尔(Harvey M. Deitel)、尤利斯·布莱克(Uyless Black)等。

为确保教材的选题质量和翻译质量,我们约请了清华大学、北京大学、北京航空航天大学、复旦大学、上海交通大学、南京大学、浙江大学、哈尔滨工业大学、华中科技大学、西安交通大学、国防科学技术大学、解放军理工大学等著名高校的教授和骨干教师参与了本系列教材的选题、翻译和审校工作。他们中既有讲授同类教材的骨干教师、博士,也有积累了几十年教学经验的老教授和博士生导师。

在该系列教材的选题、翻译和编辑加工过程中,为提高教材质量,我们做了大量细致的工作,包括对所选教材进行全面论证;选择编辑时力求达到专业对口;对排版、印制质量进行严格把关。对于英文教材中出现的错误,我们通过与作者联络和网上下载勘误表等方式,逐一进行了修订。

此外,我们还将与国外著名出版公司合作,提供一些教材的教学支持资料,希望能为授课老师提供帮助。今后,我们将继续加强与各高校教师的密切联系,为广大师生引进更多的国外优秀教材和参考书,为我国计算机科学教学体系与国际教学体系的接轨做出努力。

电子工业出版社

教材出版委员会

- | | | |
|----|-----|---|
| 主任 | 杨芙清 | 北京大学教授
中国科学院院士
北京大学信息与工程学部主任
北京大学软件工程研究所所长 |
| 委员 | 王 珊 | 中国人民大学信息学院院长、教授 |
| | 胡道元 | 清华大学计算机科学与技术系教授
国际信息处理联合会通信系统中国代表 |
| | 钟玉琢 | 清华大学计算机科学与技术系教授
中国计算机学会多媒体专业委员会主任 |
| | 谢希仁 | 中国人民解放军理工大学教授
全军网络技术研究中心主任、博士生导师 |
| | 尤晋元 | 上海交通大学计算机科学与工程系教授
上海分布计算技术中心主任 |
| | 施伯乐 | 上海国际数据库研究中心主任、复旦大学教授
中国计算机学会常务理事、上海市计算机学会理事长 |
| | 邹 鹏 | 国防科学技术大学计算机学院教授、博士生导师
教育部计算机基础课程教学指导委员会副主任委员 |
| | 张昆藏 | 青岛大学信息工程学院教授 |

Preface

As with its predecessors, the primary purpose of this edition of *A First Book of ANSI C* is to make C accessible as an applications programming language. The success of past editions, and the many comments I have received from both students and faculty stating that the book really did help them learn and teach C, have been extremely gratifying. Thus, the goal of this fourth edition remains the same as the first three editions: to present all topics clearly, unambiguously, and accessibly to beginning students. This text can be used as an introduction to programming in general, as an introduction to the C language in particular, and as a basis for further study of the C++ language.

This fourth edition includes a number of enhancements, including:

- New case studies illustrating real-world program applications
- A thorough explanation of input data validation techniques
- Expanded exercises that now include both short-answer questions and programming exercises
- A table of Common Compiler Errors (at the end of most chapters), for both Unix- and Windows-based compilers, in addition to the list of Common Programming Errors also provided at the end of most chapters
- Historical Notes relating to computer science topics
- Introductory chapter on Computer Hardware and Software Engineering
- Updates to all programs and descriptions that reflect the latest C99 ANSI standard

Distinctive Features of This Book

Emphasis: This text presents topics in a manner that helps students in the real-world of programming. I think this emphasis is best described by a reviewer, who wrote. “The depth [of this text] focuses on the kinds of problems that beginning students stumble across. This is in contrast to [many texts] that somehow manage the feat of providing massive volume without including useful hints and short cuts.”

Writing Style: I firmly believe that introductory texts do not teach students—professors teach students. An introductory textbook, if it is to be useful, must be the supporting actor to the professor’s leading role. Once the professor sets the stage, however, the textbook must encourage the student in mastering the material presented in class. To do this, the text must make sense to the student. My primary concern, and one of the distinctive features of this book, is that it has been written for the student. As one reviewer said of an earlier edition, “This book addresses the student and not the professional.”

Software Engineering: This text introduces students to the fundamentals of software engineering right from the start. In Section 1.3, students learn about algorithms and the various ways that an algorithm can be described.

The emphasis on software engineering continues in Section 1.4, which introduces the Software Development Process, and in all subsequent case studies, which demonstrate practical applications of the Software Development Process.

Introduction to Pointers: One of the unique features of the first edition was the introduction to pointers, in which the `printf()` function was used to initially display the addresses of variables; only then were variables used to actually store these addresses. This approach always seemed a more logical and intuitive method of explaining pointer variables than the indirection description in vogue at the time the first edition was released. Since the first edition, I have been pleased to see that using the `printf()` function to display addresses has become the standard way to introduce pointers. Although this approach, therefore, is no longer a unique feature of my book, I am very proud of its presentation and continue to use it in this new edition.

Program Testing: Every C program in this text has been successfully compiled and run using both Microsoft’s Visual C++ .NET and UNIX compilers. All programs have been written following the C99 ANSI standard. Source code files for

all program examples used in the text are available online. This permits students to experiment with the programs and more easily modify them as required by a number of end-of-section exercises.

Pedagogical Features

To make C accessible for a first-level course, the text includes the following pedagogical features:

End-of-Section Exercises: Almost every section in the book contains numerous short answer questions, in addition to the programming exercises provided in earlier editions. Additionally, solutions to all short answer questions are provided in Appendix G and the solutions to all programming exercises are available online to instructors.

Pseudocode and Flowchart Description: Pseudocode is stressed throughout the text. Students also learn about flow-chart symbols and how to use flowcharts to visually present flow-of-control constructs.

Common Programming and Compiler Errors and Chapter Summary: Each chapter ends with a section on common programming errors. New to this edition is an easy-to-read table of compiler errors and associated error messages generated by both UNIX- and Windows-based compilers. Each chapter also contains a summary of the main topics covered in the chapter.

Programming and Historical Notes: Scattered through the chapters are shaded boxes labeled “Programming Notes” that highlight important concepts, useful technical points, and programming techniques used by professional programmers. Similarly, the “Historical Notes” highlight significant historical events and people related to the history of computer hardware and software development.

Appendices and Solutions: An expanded set of appendices is provided in this fourth edition. These include appendices on operator precedence, ASCII codes, the standard C library, I/O and standard error redirection, floating-point number storage, and creating a personal C library. A final appendix offers solutions to all short answer questions. Solutions for all programming exercises are available at www.cengage.com/us/.

Supplemental Materials The following supplemental materials are available when this book is used in a classroom setting.

Electronic Instructor's Manual. The Instructor's Manual that accompanies this textbook includes:

- Additional instructional material to assist in class preparation, including suggestions for lecture topics
- Solutions to all the end-of-chapter materials, including the Programming Exercises

ExamView®. This textbook is accompanied by ExamView, a powerful testing software package that allows instructors to create and administer printed, computer (LAN-based), and Internet exams. ExamView includes hundreds of questions that correspond to the topics covered in this text, enabling students to generate detailed study guides that include page references for further review. These computer-based and Internet testing components allow students to take exams at their computers, and save the instructor time because each exam is graded automatically.

PowerPoint Presentations. This book comes with Microsoft PowerPoint slides for each chapter. These are included as a teaching aid for classroom presentations, either to make available to students on the network for chapter review, or to be printed for classroom distribution. Instructors can add their own slides for additional topics that they introduce to the class.

Distance Learning. Course Technology is proud to present online courses in WebCT and Blackboard to provide the most complete and dynamic learning experience possible. When you add online content to one of your courses, you're adding a lot: Topic Reviews, Practice Tests, Review Questions, Assignments, PowerPoint presentations, and, most of all, a gateway to the 21st century's most important information resource. For more information on how to bring distance learning to your course, contact your local Course Technology sales representative.

Source Code. The source code for this text is available at www.cengage.com/us/.

Solution Files. The solution files for all programming exercises are available at www.cengage.com/us/.

Acknowledgments This fourth edition is a direct result of the success of the past editions. In this regard, my most heartfelt acknowledgment and appreciation go to the instructors and students who found these editions helpful in their quests to teach and learn C.

Special thanks also to my editor, Alyssa Pratt, at Course Technology. Alyssa's vision, continuous faith, and attention to schedule and detail were instrumental to the successful completion of this edition. Next, Ann Shaffer, the development editor, provided one of the most extensive and professional edits of the original manuscript that I have ever been fortunate to receive.

Additionally, I would like to express my gratitude to the following reviewers:

John Avitabile, The College of Saint Rose
Pamela Carter, University of Colorado at Colorado Springs
Andrew Hurd, Hudson Valley Community College
Thami Rachidi, University of Colorado at Denver
Eric Thompson, University of Colorado at Denver
John H. Town, Blinn College

Each reviewer supplied detailed and constructive reviews of the text. Their suggestions, attention to detail, and comments were extraordinarily helpful to me as the manuscript evolved and matured throughout the editorial process.

Once the review process was completed, the task of turning the final manuscript into a textbook depended on many people other than myself. I especially want to thank Jennifer Roehrig, Production Editor, Serge Palladino, Quality Assurance Tester, Chris Scriver, Quality Assurance Manager, Nicole Ashton, who created the solutions, and once again, Ann Shaffer, the overall coordinator. The dedication of this second team of people was incredible and very important to me. I am very grateful to each of these individuals for the work they did on this text.

Special acknowledgment goes to three of my colleagues who provided material for this text. First, in addition to numerous editing and technical contributions made by Assistant Professor Andrew J. Hurd, of Hudson Valley Community College, I am very grateful for his provision of the compiler errors. I am also extremely grateful to R. Kenneth Walter, now retired from Weber State University, who graciously provided the material used in the Historical Notes. Special thanks must also go to my first mathematics teacher, Marie Scully-Bell, who taught me that any subject, no matter how difficult, can be mastered, both in academics, and in all of life. She is one of those special people that we are blessed to have in our lives. As always, any errors in the text (as in most of my life) rest solely on my shoulders.

I gratefully acknowledge the direct encouragement and support of Fairleigh Dickinson University. This includes the constant encouragement, support, and positive academic climate provided by the campus provost, Dr. Kenneth Greene, and my Chairperson, Dr. Paul Yoon. Without their support, this text could not have been written.

Finally, I deeply appreciate the patience, understanding, and love provided by my friend, wife, and partner, Rochelle.

To: Rochelle, Matthew, Jeremy, David Bronson

Gary Bronson

2006

Photo Credits

Figures 1.1, 1.2, 1.3, 1.7 and 1.8

Figure 1.5 Courtesy of Intel

Courtesy of IBM Archives
Corporation

教学支持说明

圣智学习出版公司 (Cengage Learning) 作为为终身教育提供全方位信息服务的全球知名教育出版集团,为秉承其在全球对教材产品的一贯教学支持服务,将为采用其教材图书的每位老师提供教学辅助资料。任何一位通过 Cengage Learning 北京代表处注册的老师都可直接下载所有在线提供的、最为丰富的教学辅助资料,包括教师用书、PPT、习题库等等。

鉴于部分资源仅适用于老师教学使用,烦请索取的老师配合填写如下情况说明表,并邮寄或传真给电子工业出版社。

证 明

兹证明 _____ 大学 _____ 系 / 院 _____ 专业 _____ 学年 (学期) 开设的课程, 共 _____ 学时, 现采用电子工业出版社出版的英文原版 / 简体中文版 (书名 / 作者) 作为主要教材。任课教师为 _____, 学生 _____ 个班共 _____ 人。

任课教师需要与本书配套的教师指导手册和习题解答。

电 话: _____

传 真: _____

E-mail: _____

联系地址: _____

邮 编: _____

建议和要求:

系 / 院主任: _____ (签字)

(系 / 院办公室章)

_____ 年 _____ 月 _____ 日



Publishing House of Electronics Industry

电子工业出版社: www.phei.com.cn

www.hxedu.com.cn

北京市万寿路 173 信箱外版教材事业部

邮编: 100036

电话: 010-88254555

传真: 010-88254560

E-mail: Te_service@phei.com.cn



Cengage Learning Beijing Office

圣智学习出版公司北京代表处

地址: 100080 北京市海淀区科学院南路 2 号

融科资讯中心 C 座南楼 12 层 1201 室

电话: 010-82862095/96/97

传真: 010-82862089

E-Mail: meng.li@Cengage.com

<http://www.Cengage.com>

Contents

Part 1 Fundamentals	1
Chapter 1 Introduction to Computer Programming.....	1
1.1 History and Hardware	1
Computer Hardware	3
Bits and Bytes	3
Components	4
Exercises 1.1	8
1.2 Programming Languages	8
Machine Language	9
Assembly Language	9
Low- and High-Level Languages	9
Procedural and Object-Oriented Languages	10
Application and System Software	11
The Development of C	12
Exercises 1.2	12
1.3 Algorithms	13
Exercises 1.3	17
1.4 The Software Development Process	17
Phase I: Specify the Program's Requirements	18
Phase II: Design and Development	19
Phase III: Documentation	22
Phase IV: Maintenance	22
Backup	23
Exercises 1.4	24
1.5 Case Study: Design and Development	25
Step 1: Analyze the Problem	26
Step 2: Select an Overall Solution Algorithm	26
Step 3: Write the Program	27
Step 4: Test and Correct the Program	27
Exercises 1.5	28
1.6 Common Programming Errors	29
1.7 Chapter Summary	29
1.8 Chapter Appendix: Numerical Storage Codes	30

Chapter 2 Getting Started in C Programming	32
2.1 Introduction to C Programming	32
Identifiers	34
The <code>main()</code> Function	35
The <code>printf()</code> Function	37
Exercises 2.1	40
2.2 Programming Style	42
Indentation	42
Comments	43
Exercises 2.2	44
2.3 Data Types	45
Integer Data Types	46
Floating-Point Data Types	48
Exponential Notation	50
Exercises 2.3	51
2.4 Arithmetic Operations	51
Displaying Numerical Values	52
Expression Types	54
Integer Division	55
Negation	55
Operator Precedence and Associativity	56
Exercises 2.4	57
Programming Exercises	59
2.5 Variables and Declarations	60
Declaration Statements	62
Initialization	66
Exercises 2.5	67
2.6 Case Study: Temperature Conversion	69
Step 1: Analyze the Problem	69
Step 2: Select a Solution Algorithm	69
Step 3: Write the Program	70
Step 4: Test and Debug the Program	71
Exercises 2.6	71
2.7 Programming and Compiler Errors	73
Programming Errors	73
Compiler Errors	74
2.8 Chapter Summary	75
2.9 Chapter Supplement: Memory Allocation	76
Integer Data Types	76
The <code>sizeof()</code> Operator	78
Addresses	79
Exercises 2.9	81

Chapter 3 Processing and Interactive Input	82
3.1 Assignment	82
Implicit Type Conversions	85
Explicit Type Conversions (Casts)	85
Assignment Variations	85
Accumulating	86
Counting	87
Exercises 3.1	89
3.2 Mathematical Library Functions	91
Exercises 3.2	94
3.3 Interactive Input	96
Caution: The Phantom Newline Character	99
A First Look at User-Input Validation	101
Exercises 3.3	102
3.4 Formatted Output	106
Format Modifiers	107
Other Number Bases (Optional)	108
Exercises 3.4	111
3.5 Symbolic Constants	113
Exercises 3.5	114
3.6 Case Study: Interactive Input	115
Requirements Specification	116
Exercises 3.6	118
3.7 Common Programming and Compiler Errors	120
Programming Errors	121
Compiler Errors	122
3.8 Chapter Summary	123
3.9 Chapter Supplement: Introduction to Abstraction	123
Built-in and Abstract Data Types	124
Procedural Abstraction	124
Part 2 Flow of Control.....	126
Chapter 4 Selection	126
4.1 Relational Expressions	126
Logical Operators	129
Exercises 4.1	131
4.2 The if and if-else Statements	132
Compound Statements	133
The if-else statement	134
Exercises 4.2	139
4.3 The if-else Chain	141
Exercises 4.3	145

4.4	The switch Statement	148
	Exercises 4.4	152
4.5	Case Study: Data Validation	154
	Requirements Specification	154
	Analyze the Problem	154
	Select an Algorithm	154
	Write the Program	155
	Test and Debug the Program	156
	Exercises 4.5	156
4.6	Common Programming and Compiler Errors	158
	Programming Errors	158
	Compiler Errors	159
4.7	Chapter Summary	160
4.8	Chapter Supplement: Errors, Testing, and Debugging	161
	Testing and Debugging	164
Chapter 5	Repetition	166
5.1	Basic Loop Structures	166
	Pretest and Posttest Loops	167
	Counter-Controlled and Condition-Controlled Loops	167
	Exercises 5.1	168
5.2	The while Statement	169
	Exercises 5.2	174
5.3	Computing Sums and Averages Using a while Loop	175
	Sentinels	181
	The break and continue Statements	183
	The Null Statement	184
	Exercises 5.3	184
5.4	The for Statement	186
	Computing Sums and Averages Using a for Loop	191
	Exercises 5.4	192
5.5	Case Studies: Loop Programming Techniques	194
	Technique 1: Selection within a Loop	194
	Technique 2: Input Data Validation	195
	Technique 3: Interactive Loop Control	197
	Technique 4: Evaluating Equations	198
	Exercises 5.5	200
5.6	Nested Loops	201
	Exercises 5.6	203
5.7	The do-while Statement	204
	Exercises 5.7	207
5.8	Common Programming and Compiler Errors	208
	Programming Errors	208

Compiler Errors	209
5.9 Chapter Summary	209
Chapter 6 Modularity Using Functions: Part I	211
6.1 Function and Parameter Declarations	211
Function Prototypes	213
Calling a Function	214
Function Header Line	215
Placement of Statements	221
Exercises 6.1	222
6.2 Returning a Value	223
Function Stubs	229
Functions with Empty Parameter Lists	230
Exercises 6.2	230
6.3 Case Study: Calculating Age Norms	232
Requirements Specification	233
Step 1: Analyze the Problem	233
Step 2: Select an Overall Solution Algorithm	234
Step 3: Write the Program	235
Step 4: Test and Correct the Program	236
Exercises 6.3	239
6.4 Standard Library Functions	241
Mathematical Library Functions	242
The rand() and srand() Functions	242
Scaling	245
Coin Toss Simulation	245
Input/Output Library Functions	248
Character Processing Functions	248
Conversion Functions	250
Exercises 6.4	251
6.5 Common Programming and Compiler Errors	253
Compiler Errors	254
6.6 Chapter Summary	255
Chapter 7 Modularity Using Functions: Part II	256
7.1 Variable Scope	257
When to Use Global Declarations	260
Misuse of Global Variables	260
Exercises 7.1	261
7.2 Variable Storage Class	263
Local Variable Storage Classes	263
Global Variable Storage Classes	266
Exercises 7.2	268

7.3	Pass by Reference	269
	Passing Addresses to a Function	270
	Storing Addresses	270
	Using Addresses	271
	Declaring and Using Pointers	272
	Passing Addresses to a Function	274
	Exercises 7.3	278
7.4	Case Study: Swapping Values	280
	Requirements Specification	280
	Analyze the Problem	280
	Code the Function	281
	Test and Debug the Program	284
	Exercises 7.4	285
7.5	Recursion 4	286
	Mathematical Recursion	286
	Recursion versus Iteration	290
	Exercises 7.5	290
7.6	Common Programming and Compiler Errors	291
	Compiler Errors	292
7.7	Chapter Summary	292

Part 3 Completing the Basics 294

Chapter 8 Arrays 294

8.1	One-Dimensional Arrays	295
	Input and Output of Array Values	298
	Exercises 8.1	301
8.2	Array Initialization	303
	Exercises 8.2	306
8.3	Arrays as Function Arguments	307
	Exercises 8.3	310
8.4	Case Study: Computing Averages and Standard Deviations	311
	Requirements Specification	311
	Analyze the Problem	312
	Select an Overall Solution Algorithm	313
	Write the Functions	313
	Test and Debug the Functions	314
	Exercises 8.4	315
8.5	Two-Dimensional Arrays	317
	Internal Array Element Location Algorithm (Optional)	322
	Larger Dimensional Arrays	323
	Exercises 8.5	324
8.6	Common Programming and Compiler Errors	325

Programming Errors	325
Compiler Errors	325
8.7 Chapter Summary	326
8.8 Chapter Supplement: Searching and Sorting Methods	327
Search Algorithms	327
Big O Notation	334
Sort Algorithms	334
Exercises 8.8	346
Chapter 9 Character Strings	348
9.1 String Fundamentals	348
String Input and Output	348
String Processing	350
Exercises 9.1	356
9.2 Library Functions	358
Character Routines	361
Conversion Routines	363
Exercises 9.2	364
9.3 Input Data Validation	365
Creating a Personal Library	370
Exercises 9.3	371
9.4 Formatting Strings (Optional)	371
In-Memory String Conversions	372
Format Strings	373
Exercises 9.4	373
9.5 Case Study: Character and Word Counting	374
Program Requirement: Character Counting	374
Analyze the Problem	374
Code the Function	375
Test and Debug the Function	375
Requirement Specification: Word Counting	376
Analyze the Problem	376
Code the Function	377
Test and Debug the Function	378
Exercises 9.5	379
9.6 Common Programming and Compiler Errors	380
Programming Errors	380
Compiler Errors	380
9.7 Chapter Summary	380
Chapter 10 Data Files	382
10.1 Declaring, Opening, and Closing File StreamsFile Streams	382
Files	382

File Streams	384
Declaring a File Stream	385
Opening a File Stream	385
Embedded and Interactive Filenames	390
Closing a File Stream	391
Exercises 10.1	392
10.2 Reading from and Writing to Text Files	393
Reading from a Text File	394
Standard Device Files	397
Other Devices	398
Exercises 10.2	399
10.3 Random File Access	401
Exercises 10.3	403
10.4 Passing and Returning Filenames	403
Exercises 10.4	406
10.5 Case Study: Creating and Using a Table of Constants	406
Requirements Specification	406
Analysis for the First Function	407
Code the Function	409
Test and Debug the Function	410
Analysis for the Second Function	411
Code the Function	412
Test and Debug the Function	412
Exercises 10.5	414
10.6 Writing and Reading Binary Files (Optional)	415
Exercises 10.6	419
10.7 Common Programming and Compiler Errors	419
Programming Errors	419
Compiler Errors	420
10.8 Chapter Summary	421
10.9 Chapter Supplement: Control Codes	422
Part 4 Additional Topics	425
Chapter 11 Arrays, Addresses, and Pointers	425
11.1 Array Names as Pointers	425
Exercises 11.1	430
11.2 Manipulating Pointers	431
Pointer Arithmetic	431
Pointer Initialization	434
Exercises 11.2	434
Programming Exercises	435
11.3 Passing and Using Array Addresses	436
Advanced Pointer Notation	439