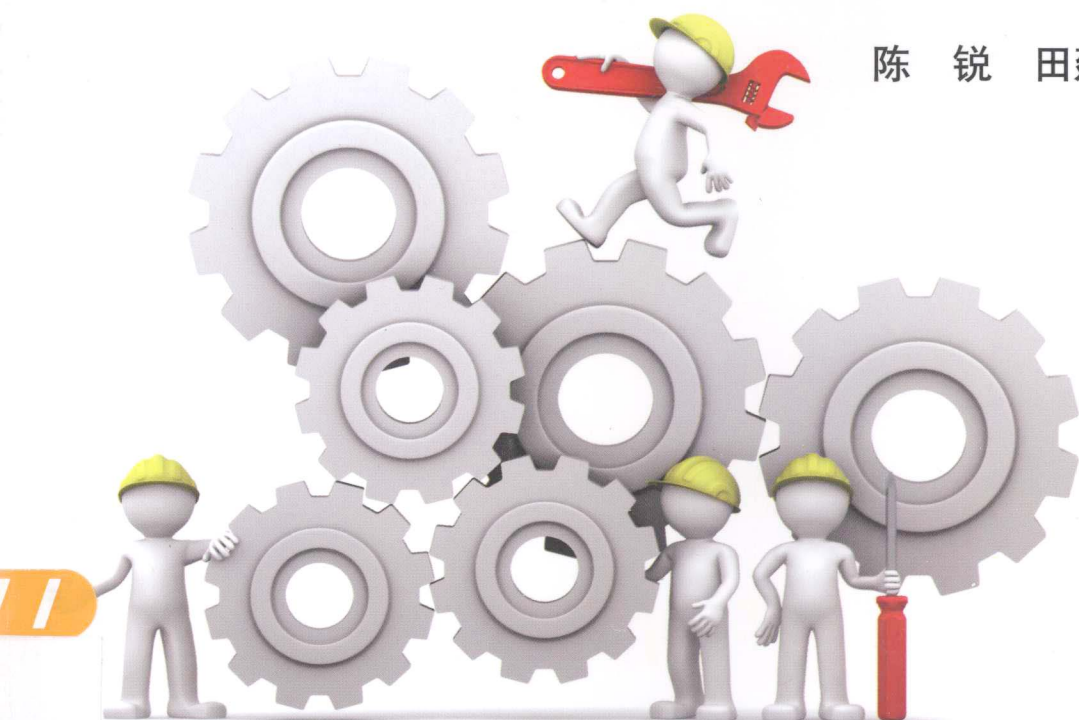


跟我学 C语言

陈 锐 田建新 编著



内容全面，由浅入深：本书涵盖了C语言课程中的所有知识点，从最基础语法开始讲解，覆盖图形、网络、算法等方面的应用。

- **结合实例，强化理解：**提供众多实例对每个知识点进行讲解，以强化读者的理解。
- **案例丰富，分析全面：**全书提供C语言多方面的应用案例，详细讲解案例的知识背景及实现。
- **图文并茂，重点突出：**丰富的图、表更利于读者理解与掌握，在知识的讲解上具有很强的连贯性。结构层次清晰，重点明确，所有细节处都给予重点标识。

清华大学出版社



013064692

TP312C
2202

内容简介

跟我学 C 语言

陈 锐 田建新 编著



清华大学出版社
北 京



北航

C1672336

TP312C
2202

内 容 简 介

C 语言是计算机专业的基础课和核心课程。本书内容全面, 结构清晰, 语言通俗, 重点难点突出, 所有程序都能够直接运行。本书内容包括 C 语言概述、常用的程序开发环境、基本数据类型、表达式、顺序结构程序设计、选择结构程序设计、循环结构程序设计、数组、函数、预处理命令、指针、结构体和共用体、位运算、文件、链表、键盘和鼠标、网络编程基础、常用算法、栈和队列、排序、学生成绩管理系统、常见错误和程序调试技术。

本书可作为学习 C 程序设计、从事计算机软件开发、参加等级考试和软考的人员的参考书, 也可以作为计算机及相关专业的教材。

本书封面贴有清华大学出版社防伪标签, 无标签者不得销售。

版权所有, 侵权必究。侵权举报电话: 010-62782989 13701121933

图书在版编目(CIP)数据

跟我学 C 语言/陈锐, 田建新编著. —北京: 清华大学出版社, 2013

ISBN 978-7-302-33008-0

I. ①跟… II. ①陈… ②田… III. ①C 语言—程序设计 IV. ①TP312

中国版本图书馆 CIP 数据核字(2013)第 145937 号

责任编辑: 张彦青 桑任松

封面设计: 杨玉兰

责任校对: 周剑云

责任印制: 宋 林

出版发行: 清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址: 北京清华大学学研大厦 A 座 邮 编: 100084

社 总 机: 010-62770175

邮 购: 010-62786544

投稿与读者服务: 010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈: 010-62772015, zhiliang@tup.tsinghua.edu.cn

课 件 下 载: <http://www.tup.com.cn>, 010-62791865

印 刷 者: 清华大学印刷厂

装 订 者: 三河市新茂装订有限公司

经 销: 全国新华书店

开 本: 185mm×260mm

印 张: 41.75

字 数: 1020 千字

版 次: 2013 年 9 月第 1 版

印 次: 2013 年 9 月第 1 次印刷

印 数: 1~3000

定 价: 78.00 元

产品编号: 049948-01

前 言

C 语言是一门使用最为广泛的程序设计语言,也是计算机等相关专业的必修课程,它是今后继续学习数据结构和算法的基础课程。C 语言功能丰富,表达能力强,使用方便灵活,程序执行效率高,移植性好,既具有高级语言的特点,又具有低级语言的特征。现在流行的 Visual C++ 和 C++ Builder 就是以 C 语言为基础的开发工具。

本书适合初、中级程序员阅读,是入门级用户的首选教材和参考用书。如果您还在犹豫应选择哪门程序设计语言作为开始,或者您刚下定决心要学好 C 语言,选择本书一定没有错。但如果您是一名高级程序员,想了解一些专门领域的内容,那么这本书不适合您,希望您将本书放回书架比较显眼的位置,在此表示感谢。

本书全面地介绍 C 语言的基本概念、基本语法及程序设计方法,以学习程序设计的基本路线进行讲解,通过实例,分析程序的结构和含义,每个实例都给出具体的代码,并配合流程图进行讲解。

本书内容全面,不仅详细介绍 C 语言的基础知识,还涉及 C 语言相关的高级技术和理论知识,是一部难得的技术参考书和自学教材,主要内容包括 C 语言概述、常用的程序开发环境、基本数据类型、表达式、顺序结构程序设计、选择结构程序设计、循环结构程序设计、数组、函数、预处理命令、指针、结构体和共用体、位运算、文件、链表、键盘和鼠标、网络编程基础、常用算法、栈和队列、排序、C 语言常见错误与调试技术。

学完本书之后,读者基本上就可以熟练地掌握程序设计的方法了,并能够熟练地开发 C 语言程序。

1. 本书特点

(1) 内容全面,讲解详细

为了方便读者学习,本书首先对 C 语言的特点及优势进行讲解,然后讲解各种 C 语言开发工具。本书内容全面,覆盖 C 语言的全部知识,对于每个知识点,都使用具体实例和生活中的实例进行讲解、对比,以便读者能够迅速地掌握 C 语言。

(2) 层次清晰,结构合理

本书将 C 语言按章、节和小节划分知识点,将知识点细化,层次清晰、结构合理,易于读者理解和学习。每一章、每一节、每一小节都划分为单独的知识点,每一小节还可分为更多的知识点。在知识点的讲解过程中,循序渐进、由浅入深,先用提问的方式引出概念,然后再进行讲解,最后通过例子来强化知识点,这样的讲解方式使读者更容易理解和消化。

(3) 结合图表,叙述简单

针对每个知识点,都结合图表和具体的实例去说明,将抽象的概念具体化,以利于读者领会。在语言的叙述上,普遍采用短句子、易于理解的语言,而避免使用复杂句子和晦涩难懂的语言,使读者可以更加容易地学会 C 语言。



(4) 例子典型，深入剖析

在讲解每一个 C 语言知识点时，结合具体例子进行剖析。所选取的例子都是一些最为常见的典型程序，且每个例子都准确说明了相关的知识点。在每一章的最后或比较大的知识点后面，都给出一个完整的程序；给出程序的同时，对程序通过图进行具体讲解，并深入分析，在程序的最后给出运行结果。

(5) 语言诙谐，结合生活

在知识的讲解方面，本书采用诙谐的语言，将每个概念与现实生活结合起来对比，从而使晦涩难懂的知识变得有趣，让学习 C 语言程序设计变得易如反掌，使读者能够快速地从外行变为内行。

2. 本书内容

本书共分 21 章，各章内容介绍如下。

第 1 章：如果读者还不知道应该学习 C、C++、C#、Java 中的哪一种编程语言，则通过阅读该章，可以坚定学习 C 语言的决心。该章介绍 C 语言的特点、为什么要学习 C 语言和基本的理论基础。

第 2 章：对几种常用的 C 语言开发工具进行详细的讲解。主要包括 Turbo C、Win-TC、LCC、Visual C++ 6.0，读者可以选择一个自己喜欢的工具。

第 3 章：主要介绍变量和常量。首先讲解为什么要有变量、为什么要有数据类型，然后讲解各种类型的变量定义及使用，最后讲解常量。

第 4 章：主要介绍运算符和表达式。首先介绍运算符的分类、什么是表达式，然后讲解常用的算术表达式、赋值表达式、关系表达式、逻辑表达式、逗号表达式。通过对该章内容的学习，读者将学会各种表达式的使用。

第 5 章：主要介绍顺序结构程序设计。首先介绍语句的概念，然后介绍顺序结构程序的特点，最后通过具体实例讲解如何编写顺序结构的程序。

第 6 章：主要介绍选择结构程序设计。首先讲解条件运算符和条件表达式，然后讲解单分支的 if 选择语句、双分支的 if 选择语句、多分支的 if 选择语句、switch 选择语句，最后给出一个具体的实例。

第 7 章：主要介绍循环结构程序设计。首先介绍 3 种常用的循环语句——while 循环语句、do-while 循环语句、for 循环语句，接着介绍 goto 语句、break 语句和 continue 语句、多重循环结构的程序设计，最后给出几个典型的实例。

第 8 章：主要介绍数组。首先通过具体问题引出数组的概念，然后介绍一维数组、二维数组、字符数组。

第 9 章：主要介绍函数。首先介绍函数的分类，然后介绍函数的定义形式、局部变量和全局变量、函数的参数和函数的值、函数的调用、函数的嵌套调用、函数的递归调用、数组作为函数的参数，最后给出几个实例。通过学习该章的内容，读者将掌握模块化程序设计知识。

第 10 章：讲解预处理命令。依次介绍宏定义、文件包含命令、条件编译，通过学习该章的内容，可以提高所编写程序的编译效率。

第 11 章：主要介绍 C 语言的灵魂——指针。首先介绍指针与地址的区别，然后讲解指

针变量、数组与指针、字符串与指针、指针数组与指向指针的指针、函数与指针、返回指针值的函数。指针是 C 语言中最难的一个知识点，该章通过对比学习指针中容易混淆的概念，使读者可以编写出高效的 C 语言程序。

第 12 章：主要介绍结构体和共用体。首先介绍结构体的定义、引用、初始化，然后讲解结构体数组、指针与结构体、共用体、枚举类型。通过对该章内容的学习，读者将能够定义及使用比较复杂的数据，设计出更加实用的程序。

第 13 章：主要介绍位运算。首先介绍位运算符及位运算的优先级与结合性，然后介绍各种位运算符，并通过一个典型实例讲解位运算符的使用方法。通过对该章内容的学习，读者将了解到计算机的底层操作。

第 14 章：主要介绍文件的知识。首先介绍文件的分类，然后讲解打开和关闭文件、读取文件和写入文件、文件的定位等操作，最后讲解常见的其他文件操作函数。通过对该章内容的学习，读者将掌握文件的相关操作。

第 15 章：主要介绍链表的知识。首先介绍链表的概念、动态存储分配，然后讲解创建链表、插入链表、删除链表等操作，最后给出具体的应用实例。通过学习该章内容，读者将掌握如何进行动态内存分配，构造出最简单的数据结构。

第 16 章：主要介绍键盘和鼠标操作。首先介绍键盘和鼠标操作的一些基础知识，然后通过实例分析键盘和鼠标操作的具体应用技术。通过对该章内容的学习，读者将掌握计算机内部的工作原理，理解控制键盘和鼠标的技术。

第 17 章：主要介绍网络编程基础。首先介绍网络的基本知识，然后介绍 Winsock 基础知识和相关函数，最后给出一个基于 TCP 和 UDP 的网络聊天程序。通过对该章内容的学习，读者将学会如何使用 C 语言编写简单的网络应用程序。

第 18 章：主要介绍常用算法。首先介绍算法设计的基础知识，然后分别介绍迭代算法、递推算法、穷举算法、递归算法、分治算法、贪心算法和矩阵算法。通过对该章内容的学习，读者将掌握软件开发中最常见的算法，进一步提高软件开发水平。

第 19 章：主要介绍栈和队列。首先介绍队列和栈的有关概念，然后介绍队列和栈的表示与实现，最后介绍队列和栈的应用。通过对该章内容的学习，读者将能够利用栈和队列这种最简单的数据结构对事物进行抽象。

第 20 章：主要介绍排序技术。首先介绍排序的有关概念，然后分别介绍插入排序、选择排序、交换排序、归并排序和分配排序，最后对各种排序算法进行比较。

第 21 章：主要介绍学生成绩管理系统。通过一个具体的软件项目来告诉读者如何进行软件开发，首先对需要开发的软件系统进行目标分析，然后进行总体设计、详细设计，最后是编码和测试。通过对该章内容的学习和实践，读者将能够掌握软件开发的流程，初步学会软件开发技术。

第 22 章：主要介绍 C 语言常见错误和程序调试技术。首先介绍 C 语言开发过程中的常见错误，然后举例说明如何利用 Visual C++ 6.0 进行程序调试。根据一些热心读者和学生在学习 C 语言过程中遇到的问题，进行有针对性的分析和讲解。通过对该章内容的学习，读者将学会识别最常见的 C 语言错误信息，并初步掌握 Visual C++ 6.0 程序调试技术。

本书的第 1~9 章及第 15~20 章、第 22 章由陈锐编写，第 10 章、第 11 章、第 21 章由杨宇晶编写，第 12~14 章由郭天娇编写。



由于作者水平有限，书中难免存在一些疏漏和不足之处，恳请读者批评指正。请通过 nwuchenrui@126.com 与作者联系。

3. 适合的读者

本书适合下列人员阅读和使用：

- 大中专院校的学生。
- 准备参加计算机等级考试的人员。
- 准备参加软考的人员。
- 软件开发人员。
- 计算机相关的科研工作者。

4. 致谢

感谢我的导师张蕾教授，她丰富的知识储备及敏锐的洞察力极大地影响了我的学习态度和认识能力，使我在职业生涯中受益，也为本书的编写奠定了良好的基础。

感谢我的家人，是因为有他们默默的付出和鼓励，我才能顺利地做好各项工作。

最后特别感谢温县教育局及所有支持我写作的朋友们！

陈 锐

2013 年 7 月

目 录

第 1 章 C 语言概述.....1

1.1 为什么要选择 C 语言.....2

1.1.1 选择 C 语言的好处.....2

1.1.2 C 语言的特点.....2

1.1.3 如何学好 C 语言.....3

1.2 进制转换.....4

1.2.1 十进制数的表示.....4

1.2.2 二进制数的表示.....5

1.2.3 二进制数、十六进制数

和八进制数转换为十进制数.....6

1.2.4 十进制数转换为二进制数.....6

1.2.5 十进制数转换为十六进制数.....9

1.2.6 十进制数转换为八进制数.....11

1.2.7 二进制数与十六进制数、

八进制数的转换.....12

1.3 计算机中数的表示.....13

1.3.1 计算机中的正数与负数表示.....14

1.3.2 原码和补码.....14

1.3.3 浮点数.....17

1.4 小结.....19

第 2 章 C 语言开发环境.....21

2.1 Turbo C 2.0 开发环境介绍.....22

2.1.1 运行 Turbo C 2.0.....22

2.1.2 Turbo C 2.0 环境设置.....23

2.1.3 Turbo C 2.0 的使用.....25

2.2 Win-TC 开发环境介绍.....27

2.2.1 Win-TC 开发环境概述.....27

2.2.2 Win-TC 的使用.....28

2.3 LCC 开发环境介绍.....30

2.3.1 使用 LCC 开发环境创建

项目.....30

2.3.2 LCC 开发环境的介绍.....32

2.3.3 LCC 的使用.....33

2.4 Visual C++ 6.0 开发环境介绍.....34

2.4.1 使用 Visual C++ 6.0 新建

项目.....34

2.4.2 使用 Visual C++ 6.0 新建源

程序文件.....36

2.4.3 使用 Visual C++ 6.0 开发

环境.....37

2.5 小结.....38

第 3 章 基本数据类型.....39

3.1 变量.....40

3.1.1 为什么要使用变量.....40

3.1.2 什么是变量——变量在计算机

中的表示.....40

3.1.3 如何定义变量.....41

3.1.4 为什么要有数据类型.....41

3.1.5 C 语言有哪些数据类型.....42

3.1.6 什么符号可以作为变量名——

用户定义的标识符.....43

3.1.7 关键字——已经被占用的

标识符.....44

3.1.8 初识变量赋值.....44

3.2 整型变量.....45

3.2.1 为什么要使用整型变量.....45

3.2.2 基本整型变量.....46

3.2.3 基本整型变量的输入——使用

scanf 函数.....47

3.2.4 基本整型变量的输出——使用

printf 函数.....49

3.2.5 长整型变量——解决大整数

问题.....50

3.2.6 短整型变量——节省内存

空间.....52



3.2.7 无符号整型变量——只包括非负的整数.....	53	4.3.2 类型转换.....	89
3.2.8 整型变量的应用举例.....	54	4.3.3 赋值表达式的应用举例.....	91
3.3 实型变量.....	57	4.4 关系运算符与关系表达式及其应用 ...	93
3.3.1 为什么使用实型变量.....	57	4.4.1 关系运算符和关系表达式.....	93
3.3.2 单精度浮点型变量.....	57	4.4.2 关系表达式应用举例.....	94
3.3.3 双精度浮点型变量.....	60	4.5 逻辑运算符与逻辑表达式及其应用 ...	95
3.3.4 长双精度浮点型变量.....	61	4.5.1 逻辑运算符和逻辑表达式.....	95
3.3.5 实型变量的应用举例.....	62	4.5.2 逻辑表达式应用举例.....	97
3.4 字符型数据类型.....	64	4.6 逗号表达式.....	98
3.4.1 为什么要使用字符型变量.....	64	4.6.1 逗号运算符与逗号表达式.....	99
3.4.2 字符型变量.....	64	4.6.2 逗号表达式中应注意的问题.....	99
3.4.3 字符型数据的输入与字符型变量值的输出——使用%c和%s.....	65	4.6.3 逗号表达式应用举例.....	99
3.4.4 字符型变量的应用举例.....	66	4.7 小结.....	100
3.5 常量.....	68	第5章 顺序结构程序设计.....	101
3.5.1 为什么使用常量.....	68	5.1 语句和程序.....	102
3.5.2 整型常量.....	68	5.1.1 什么是简单语句.....	102
3.5.3 浮点型常量.....	69	5.1.2 为什么使用复合语句.....	103
3.5.4 字符型常量与字符串型常量.....	69	5.1.3 什么是复合语句.....	103
3.5.5 常量与常数.....	70	5.1.4 使用复合语句应注意的问题.....	104
3.6 小结.....	70	5.1.5 为什么要有程序.....	105
习题.....	70	5.1.6 C语言程序的框架结构.....	106
第4章 运算符与表达式.....	73	5.2 顺序结构程序设计.....	107
4.1 表达式.....	74	5.2.1 顺序结构程序设计的特点——自上而下执行每一个语句.....	107
4.1.1 运算符有哪些.....	74	5.2.2 顺序结构程序设计——求圆的面积.....	107
4.1.2 什么是表达式.....	74	5.2.3 顺序结构程序——求一元二次方程 $ax^2+bx+c=0$ 的根.....	110
4.1.3 运算符的优先级与结合性.....	75	5.2.4 顺序结构程序设计应用举例——将小写字母转换为大写字母.....	111
4.1.4 自动类型转换.....	76	5.3 小结.....	113
4.2 算术运算符与算术表达式.....	77	第6章 选择结构程序设计.....	115
4.2.1 基本算术表达式.....	77	6.1 条件运算符.....	116
4.2.2 自增表达式——++a和a++.....	79	6.1.1 为什么要有条件运算符.....	116
4.2.3 自减表达式——--a和a--.....	82	6.1.2 条件运算符与条件表达式.....	116
4.2.4 算术运算符的混合运算——算术运算符的优先级与结合性.....	83	6.2 if选择语句.....	119
4.2.5 算术表达式应用举例.....	84		
4.3 赋值运算符与赋值表达式及其应用.....	86		
4.3.1 赋值运算符与赋值表达式.....	86		

6.2.1	为什么要有 if 选择语句——理解容易	119
6.2.2	if 选择语句——单分支选择结构	120
6.2.3	if-else 选择语句——双分支的选择结构	124
6.2.4	if...else if...else 选择语句——多分支选择结构	126
6.2.5	if 选择语句的嵌套	131
6.2.6	设置标志变量	135
6.2.7	if 选择语句与条件运算符——相互转换	136
6.3	switch 选择语句	137
6.3.1	switch 选择语句	137
6.3.2	switch 选择语句应用举例	138
6.4	选择结构程序设计应用举例	143
6.5	小结	151
第 7 章	循环结构程序设计	153
7.1	为什么要有循环结构	154
7.1.1	如何重复输出多个 hello world	154
7.1.2	如何求连续的 n 个自然数的和	154
7.2	while 循环语句	155
7.2.1	while 循环语句——当型循环	155
7.2.2	while 循环语句应用举例	158
7.3	do-while 循环语句	160
7.3.1	为什么要有 do-while	160
7.3.2	do-while 循环语句——直到型循环	160
7.3.3	比较 while 循环结构和 do-while 循环结构	162
7.4	for 循环语句	165
7.4.1	for 循环语句概述	165
7.4.2	for 循环语句的灵活性	169
7.4.3	几种循环语句的比较	174

7.5	goto 语句与 goto 语句构成的循环语句	175
7.5.1	goto 语句——无条件转移语句	175
7.5.2	goto 语句构成的循环语句——向前跳转	176
7.5.3	goto 语句与 while 语句、for 语句的比较	177
7.6	break 语句	178
7.6.1	循环中的“提前开溜”	178
7.6.2	使用 break 语句	179
7.7	continue 语句	181
7.7.1	为什么要有 continue 语句	181
7.7.2	使用 continue 语句	181
7.7.3	continue 语句与 break 语句的区别	182
7.7.4	continue 语句应用举例	183
7.8	多重循环程序设计	184
7.8.1	为什么要有循环的嵌套——一层循环是不能解决所有问题的	184
7.8.2	循环的嵌套——循环结构中还可以有循环结构	185
7.8.3	多重循环应用举例——输出乘法口诀表	187
7.9	循环结构程序应用举例	190
7.9.1	循环结构程序应用举例——求 π 的近似值	190
7.9.2	循环结构程序应用举例——求 101~200 之间的所有质数	193
7.9.3	循环结构程序应用举例——质数问题的改进算法	195
7.9.4	循环结构程序应用举例——百鸡问题	196
7.10	小结	198
第 8 章	数组	199
8.1	为什么要有数组	200



8.1.1 多变量的解决之道——数组的引入.....	200
8.1.2 数组就是这个样子——初识数组.....	201
8.1.3 数组的数组——维度的出现.....	202
8.2 一维数组.....	204
8.2.1 一维数组是这样定义的.....	204
8.2.2 一维数组的引用——其实很简单.....	205
8.2.3 一维数组的初始化——方法有许多.....	206
8.2.4 一维数组的应用举例——求数组各元素的平均值.....	208
8.2.5 一维数组的应用举例——数组元素的倒排.....	209
8.2.6 一维数组的应用举例——冒泡排序.....	211
8.3 二维数组.....	214
8.3.1 二维数组原来是这样定义的.....	214
8.3.2 二维数组的引用——其实并不难.....	216
8.3.3 二维数组的初始化——多样的初始化.....	217
8.3.4 二维数组应用举例——计算各科成绩的平均分.....	220
8.3.5 二维数组应用举例——矩阵的转置.....	222
8.3.6 二维数组应用举例——有趣的魔方阵.....	223
8.4 字符数组.....	227
8.4.1 字符数组是这样定义的.....	227
8.4.2 字符数组的初始化.....	228
8.4.3 字符数组的连续输出——格式符%s.....	230
8.4.4 字符数组的连续输入——格式符%s.....	231
8.4.5 常用的字符串处理函数.....	232
8.4.6 字符数组的应用举例.....	236
8.5 小结.....	240
第9章 函数.....	241
9.1 为什么要有函数及函数的分类.....	242
9.1.1 为什么要有函数.....	242
9.1.2 函数有哪些——库函数和用户函数.....	243
9.2 函数的定义形式.....	243
9.2.1 不带参数的函数定义形式.....	243
9.2.2 带参数的函数定义形式——有参数列表.....	244
9.2.3 带参数的函数定义形式——形参定义的传统方式.....	245
9.3 局部变量与全局变量.....	246
9.3.1 局部变量.....	246
9.3.2 全局变量.....	248
9.4 函数的参数与函数的值.....	250
9.4.1 形式参数与实际参数.....	250
9.4.2 函数的返回值.....	252
9.5 函数的调用.....	254
9.5.1 函数调用的一般形式.....	254
9.5.2 函数调用的方式.....	256
9.5.3 对被调用函数的声明.....	257
9.6 函数的嵌套调用.....	260
9.6.1 函数的嵌套调用过程.....	260
9.6.2 函数的嵌套应用举例.....	261
9.7 函数的递归调用.....	262
9.7.1 什么是递归——自己调用自己.....	262
9.7.2 递归函数应用举例——求 n!.....	263
9.7.3 递归函数调用应用举例——Fibonacci 数列.....	265
9.8 数组作为函数的参数.....	266
9.8.1 数组元素作为函数的参数——只能传递一个数组元素.....	266
9.8.2 数组名作为函数的参数——可以传递整个数组.....	268

9.8.3	数组名作为参数传递——实际上传递的是数组的首地址.....	271
9.8.4	多维数组名作为函数的参数——传递整个数组.....	275
9.9	变量的存储.....	277
9.9.1	什么是动态存储与静态存储——生存期不同.....	278
9.9.2	auto 变量——自动类型变量.....	278
9.9.3	static 变量——静态变量.....	278
9.9.4	register 变量——使运行速度更快的变量.....	281
9.9.5	extern 变量——声明外部变量.....	282
9.10	内部函数与外部函数.....	284
9.10.1	内部函数——使用 static.....	284
9.10.2	外部函数——使用 extern.....	285
9.11	函数应用举例.....	286
9.11.1	函数应用举例——递归求解 n 个数的最大值.....	286
9.11.2	函数应用举例——递归求解 a 与 b 的最大公约数.....	288
9.11.3	函数应用举例——字符数组元素逆序存放.....	289
9.12	小结.....	290
第 10 章	预处理命令.....	291
10.1	宏定义.....	292
10.1.1	为什么要有宏定义.....	292
10.1.2	不带参数的宏定义.....	292
10.1.3	宏定义与常量定义的区别.....	294
10.1.4	带参数的宏定义.....	294
10.1.5	宏定义中的参数与函数的参数的区别.....	295
10.2	文件包含命令.....	297
10.2.1	为什么要有文件包含命令.....	297
10.2.2	文件包含命令——#include.....	297
10.3	条件编译.....	299

10.3.1	第一种条件编译命令——#ifdef.....	300
10.3.2	第二种条件编译命令——#ifdef...#elif...#endif.....	301
10.3.3	第三种条件编译命令——#ifndef.....	302
10.3.4	第四种条件编译命令——#if.....	303
10.4	小结.....	304
第 11 章	指针.....	305
11.1	指针的相关概念.....	306
11.1.1	为什么要有指针和指针变量.....	306
11.1.2	什么是地址.....	306
11.1.3	什么是指针变量——存放地址的变量.....	307
11.2	指针变量.....	308
11.2.1	定义指针变量.....	308
11.2.2	使用指针变量.....	310
11.2.3	指针变量作为函数参数.....	313
11.3	数组与指针.....	317
11.3.1	指向数组元素的指针.....	317
11.3.2	通过指针引用数组元素.....	318
11.3.3	指针变量的自增运算和自减运算.....	322
11.3.4	数组指针作为函数的参数.....	323
11.3.5	指向多维数组的指针变量.....	329
11.4	字符串与指针.....	336
11.4.1	指向字符串的指针变量.....	336
11.4.2	字符串指针作为函数的参数.....	342
11.4.3	字符数组与字符指针变量的比较.....	346
11.5	指针数组与指向指针的指针.....	347
11.5.1	指针数组.....	347
11.5.2	指向指针的指针.....	351
11.5.3	指针数组作为 main 函数的参数.....	353



11.6 函数与指针.....	354	12.5.5 共用体应用举例.....	390
11.6.1 函数指针调用函数.....	354	12.6 枚举类型.....	393
11.6.2 指向函数的指针作为函数 参数.....	355	12.6.1 为什么要有枚举类型——变量 的值只有有限的几种.....	393
11.7 返回指针值的函数.....	358	12.6.2 定义枚举类型及变量.....	394
11.7.1 为什么函数要返回指针值—— 可以返回多个值.....	358	12.6.3 使用枚举类型时应注意的 问题.....	394
11.7.2 返回指针类型的函数.....	358	12.6.4 枚举类型应用举例.....	395
11.8 小结.....	360	12.7 小结.....	396
第 12 章 结构体和共用体.....	361	第 13 章 位运算.....	397
12.1 结构体.....	362	13.1 为什么要有位运算与位运算符.....	398
12.1.1 为什么要有结构体.....	362	13.1.1 为什么要有位运算——提高 效率.....	398
12.1.2 结构体类型的定义.....	363	13.1.2 位运算符.....	398
12.1.3 定义结构体变量.....	363	13.1.3 位运算符的优先级 与结合性.....	399
12.1.4 引用结构体变量.....	365	13.2 位运算符和位运算.....	399
12.1.5 结构体变量的初始化.....	367	13.2.1 按位与运算符和按位 与运算.....	399
12.2 结构体数组.....	369	13.2.2 按位或运算符与按位 或运算.....	401
12.2.1 为什么要有结构体数组.....	369	13.2.3 按位异或运算符与按位 异或运算.....	401
12.2.2 定义结构体数组.....	369	13.2.4 按位取反运算符与按位 取反运算.....	403
12.2.3 结构体数组的初始化.....	370	13.2.5 左移运算符与左移运算.....	404
12.2.4 结构体数组应用举例.....	371	13.2.6 右移运算符与右移运算.....	405
12.3 指针与结构体.....	375	13.2.7 与位运算符相结合的赋值 运算符.....	405
12.3.1 指向结构体变量的指针.....	375	13.3 位运算应用举例.....	406
12.3.2 指向结构体数组的指针.....	376	13.3.1 以二进制形式输出一个数—— 按位与运算并输出 1 或 0.....	406
12.3.3 结构体变量和指向结构体的 指针作为函数的参数.....	378	13.3.2 取一个整数的中间几位.....	407
12.4 typedef 类型定义.....	382	13.4 位段.....	409
12.4.1 typedef——为数据类型重新 起个名字.....	382	13.4.1 定义位段.....	409
12.4.2 使用 typedef 时应注意的 问题.....	384	13.4.2 引用位段成员.....	410
12.4.3 typedef 的应用举例.....	384	13.4.3 使用位段需要说明的问题.....	411
12.5 共用体.....	387	13.5 小结.....	412
12.5.1 为什么要有共用体——节省 内存, 多选一.....	387		
12.5.2 定义共用体.....	388		
12.5.3 引用共用体变量.....	389		
12.5.4 使用共用体应该注意的 问题.....	390		

第 14 章 文件	413
14.1 文件的相关概念	414
14.1.1 为什么要有文件	414
14.1.2 文件的分类——二进制文件 和文本文件	414
14.1.3 缓冲文件系统	414
14.2 打开与关闭文件	415
14.2.1 文件类型指针——FILE*	415
14.2.2 打开文件——使用 fopen 函数	416
14.2.3 关闭文件——使用 fclose 函数	417
14.3 读取文件与写入文件	418
14.3.1 使用 fputc 函数写文件和 使用 fgetc 函数读取文件	418
14.3.2 使用 fputs 函数写文件和使用 fgets 函数读取文件	422
14.3.3 使用 fwrite 函数写文件和 使用 fread 函数读取文件	424
14.3.4 使用 fprintf 函数写文件和 使用 fscanf 函数读取文件	426
14.4 文件的定位	428
14.4.1 rewind 函数——移动位置 指针到文件的开始位置	428
14.4.2 fseek 函数——移动位置指针 到任意位置	429
14.4.3 ftell 函数——得到位置指针 的当前位置	431
14.5 文件状态检测	432
14.5.1 feof 函数——检测位置指针 是否到了文件末尾	432
14.5.2 ferror 函数——检测是否操作 文件错误	432
14.5.3 clearerr 函数——清除错误 标志	433
14.6 文件的输入与输出操作函数 总结	433
14.7 小结	434

第 15 章 链表	435
15.1 链表的相关概念	436
15.1.1 为什么要有链表——节省 内存单元, 不用事先定义 空间大小	436
15.1.2 什么是链表	436
15.1.3 简单链表——静态链表	438
15.1.4 动态存储分配	442
15.2 链表的操作	443
15.2.1 创建链表	443
15.2.2 链表的输出操作	447
15.2.3 链表的插入操作	449
15.2.4 链表的删除操作	453
15.2.5 链表的综合操作	456
15.3 链表操作应用举例	457
15.3.1 链表操作应用举例——逆置 链表	457
15.3.2 链表操作应用举例—— 约瑟夫问题	463
15.4 小结	467
第 16 章 键盘和鼠标	469
16.1 键盘操作	470
16.1.1 键盘编码	470
16.1.2 键盘操作函数	470
16.2 鼠标操作	472
16.2.1 鼠标的工作原理	473
16.2.2 鼠标综合应用举例	477
16.3 小结	484
第 17 章 网络编程基础	485
17.1 网络基础知识	486
17.1.1 什么是计算机网络	486
17.1.2 网络协议	486
17.1.3 协议分层	487
17.1.4 网络参考模型	488
17.1.5 端口	491
17.2 WinSocket 基础	491
17.2.1 套接字 Socket	491



17.2.2 基于 TCP 的 Socket 编程	491	18.2.3 角谷猜想	512
17.2.3 基于 UDP 的 Socket 编程	493	18.2.4 牛顿迭代法	513
17.3 WinSocket 相关函数	493	18.3 递推算法	515
17.3.1 WSAStartup 函数——启动套接字库	494	18.3.1 认识递推	515
17.3.2 Socket 函数——建立套接字	494	18.3.2 斐波那契数列	516
17.3.3 bind 函数——绑定本地 IP 地址和端口	495	18.3.3 分西瓜	517
17.3.4 listen 函数——侦听客户端请求	496	18.3.4 该存多少钱	518
17.3.5 accept 函数——等待客户端的请求	496	18.4 穷举算法	518
17.3.6 send 函数——发送数据	496	18.4.1 算法思想	519
17.3.7 recv 函数——接收数据	497	18.4.2 完全数	519
17.3.8 connect 函数——建立连接	497	18.4.3 背包问题	520
17.3.9 recvfrom 函数——接收数据	497	18.5 递归算法	521
17.3.10 sendto 函数——发送数据	497	18.5.1 算法思想	522
17.4 基于 TCP 的简单网络程序	498	18.5.2 数制转换	522
17.4.1 服务器端的程序实现	498	18.5.3 组合问题	523
17.4.2 客户端程序的实现	501	18.6 分治算法	525
17.5 基于 UDP 的简单网络聊天程序	503	18.6.1 算法思想	525
17.5.1 服务器端程序的实现	503	18.6.2 求 n 个数的最大值和最小值	525
17.5.2 客户端程序的实现	505	18.6.3 赛程安排问题	527
17.6 小结	506	18.7 贪心算法	530
第 18 章 常用算法	507	18.7.1 算法思想	530
18.1 算法基础	508	18.7.2 加油站问题	531
18.1.1 什么是算法及算法的描述语言	508	18.7.3 找零钱问题	532
18.1.2 算法的特性	509	18.8 矩阵算法	534
18.1.3 算法设计的目标	509	18.8.1 打印魔方阵	534
18.1.4 算法的时间复杂度和空间复杂度	510	18.8.2 打印拉丁方阵	536
18.2 迭代算法	511	18.8.3 将矩阵旋转 90 度	537
18.2.1 算法思想	511	18.9 小结	539
18.2.2 求一个数的平方根	511	第 19 章 简单数据结构——栈和队列	541
		19.1 队列	542
		19.1.1 队列的定义	542
		19.1.2 队列的表示与实现	543
		19.1.3 顺序循环队列	544
		19.1.4 顺序循环队列的实现	545
		19.1.5 链式队列的表示与实现	547
		19.1.6 队列的应用——商品货架模拟	549

19.2 栈.....	553	20.8 小结.....	599
19.2.1 栈的定义.....	553	第 21 章 学生成绩管理系统.....	601
19.2.2 顺序栈的存储结构与实现.....	553	21.1 系统总体设计.....	602
19.2.3 链式栈的存储结构与实现.....	555	21.1.1 项目开发目标.....	602
19.2.4 栈的应用举例——算术 表达式求值.....	557	21.1.2 系统功能描述.....	602
19.3 小结.....	563	21.2 系统详细设计.....	603
第 20 章 常用技术——排序.....	565	21.2.1 主函数的运行流程.....	603
20.1 排序的基础知识.....	566	21.2.2 功能模块设计.....	604
20.1.1 排序的相关概念.....	566	21.2.3 数据结构设计.....	606
20.1.2 排序算法的分类.....	567	21.2.4 函数功能描述.....	606
20.2 插入类排序.....	567	21.3 系统实现与系统测试.....	608
20.2.1 直接插入排序.....	568	21.3.1 编码.....	608
20.2.2 折半插入排序.....	569	21.3.2 系统测试.....	621
20.2.3 希尔排序.....	571	21.4 小结.....	625
20.3 选择类排序.....	573	第 22 章 C 语言常见错误与程序调试 技术.....	627
20.3.1 简单选择排序.....	573	22.1 常见错误.....	628
20.3.2 堆排序.....	575	22.1.1 错误分类.....	628
20.4 交换类排序.....	582	22.1.2 常见错误举例.....	628
20.4.1 冒泡排序.....	582	22.2 程序调试.....	632
20.4.2 快速排序.....	584	22.2.1 Visual C++ 6.0 开发环境的 程序调试.....	632
20.5 归并类排序.....	587	22.2.2 程序调试应用举例.....	638
20.5.1 二路归并排序算法思想.....	587	22.3 小结.....	643
20.5.2 二路归并排序算法实现.....	588	后记.....	645
20.6 分配类排序.....	591	参考文献.....	651
20.6.1 基数排序算法思想.....	591		
20.6.2 基数排序算法实现.....	592		
20.7 各种排序方法的比较.....	598		



第 1 章 C 语言概述

C 语言是一门十分优秀的经典开发语言。1972 年，美国贝尔实验室的 Dennis M. Ritchie 在 B 语言的基础上设计发明了 C 语言，并首次用 C 语言改写了 Unix 操作系统。随后 C 语言被使用在多种计算机上，并广泛应用在各个领域，风靡全球。后来，美国国家标准学会 (ANSI) 为 C 语言制定了一个统一的标准，成为现行的 C 语言标准。

通过阅读本章，您可以：

- 了解 C 语言的优势。
- 掌握二进制数与十进制数的转换方法。
- 掌握二进制数与十六进制数的转换方法。
- 掌握二进制数与八进制数的转换方法。
- 掌握计算机中定点数与浮点数的表示方法。