

高等教育国家级教学成果二等奖

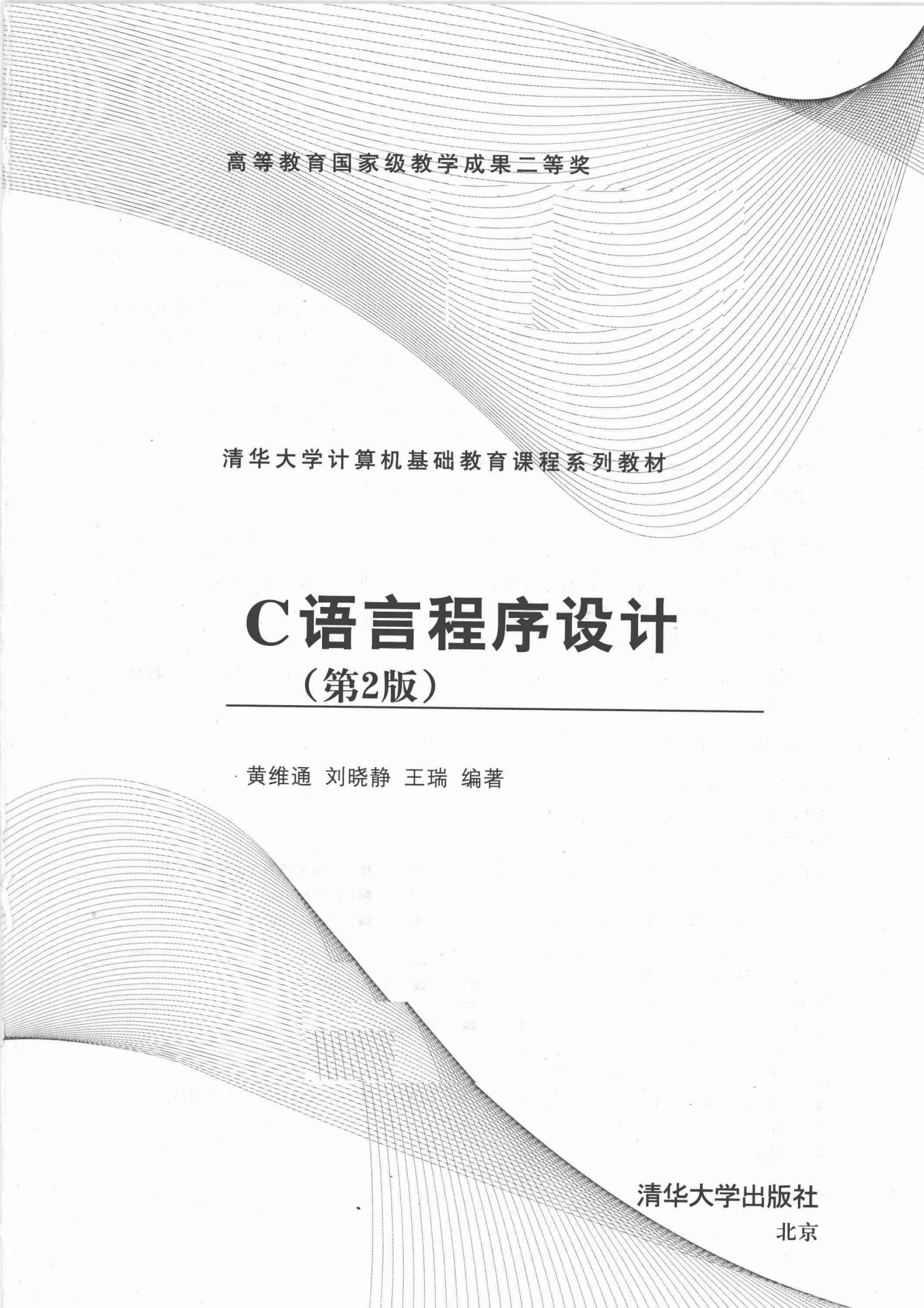
清华大学计算机基础教育课程系列教材

C语言程序设计

(第2版)

黄维通 刘晓静 王瑞 编著

清华大学出版社



高等教育国家级教学成果二等奖

清华大学计算机基础教育课程系列教材

C语言程序设计

(第2版)

黄维通 刘晓静 王瑞 编著

清华大学出版社
北京

内 容 简 介

为适应和推进新时期计算机基础教育的改革,推进精品课程建设以及与之配套的精品教材建设,从素质教育的理念出发,结合信息化社会对高素质、复合型人才的需求,特出版此教材,力求较为全面地介绍 C 语言的基本内容与程序设计思想。

本教材从 C 语言程序设计的基本原理及程序设计的基本思想出发,贯穿“基础-应用-专题”这一主线,紧扣基础,循序渐进,面向应用。为方便授课和学习,本教材配套了电子课件和习题解析。

基础部分的主要内容包括程序设计中的基本概念与应用,如变量、数组、控制结构及判断结构等,并在掌握了这些基本概念与应用的基础上适时引入函数的结构与应用、指针的概念及其应用、算法设计与实现、结构型数据的应用及文件的操作等面向应用的知识点介绍,然后在上述知识点的基础上进一步介绍数据结构专题,包括链表、栈、队列和二叉树的概念与应用。

本教材可供高等院校、水平考试、各类成人教育等教学使用,也可供计算机爱好者自学。

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

C 语言程序设计/黄维通,刘晓静,王瑞编著. —2 版. —北京:清华大学出版社,2011.1
(清华大学计算机基础教育课程系列教材)

ISBN 978-7-302-23750-1

I. ①C… II. ①黄… ②刘… ③王… III. ①C 语言—程序设计—高等学校—教材
IV. ①TP312

中国版本图书馆 CIP 数据核字(2010)第 168104 号

责任编辑:谢 琛 薛 阳

责任校对:李建庄

责任印制:何 芊

出版发行:清华大学出版社

地 址:北京清华大学学研大厦 A 座

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

邮 编:100084

社 总 机:010-62770175

邮 购:010-62786544

投稿与读者服务:010-62795954, jsjic@tup.tsinghua.edu.cn

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

印 刷 者:北京四季青印刷厂

装 订 者:三河市溧源装订厂

经 销:全国新华书店

开 本:186×260

印 张:20.5

字 数:510 千字

版 次:2011 年 1 月第 2 版

印 次:2011 年 1 月第 1 次印刷

印 数:1~5000

定 价:30.00 元

产品编号:027920-01

序

计算机科学技术的发展不仅极大地促进了整个科学技术的发展,而且明显地加快了经济信息化和社会信息化的进程。因此,计算机教育在各国备受重视,计算机知识与能力已成为 21 世纪人才素质的基本要素之一。

清华大学自 1990 年开始将计算机教学纳入基础课的范畴,作为校重点课程进行建设和管理,并按照“计算机文化基础”、“计算机技术基础”和“计算机应用基础”三个层次的课程体系组织教学:

第一层次“计算机文化基础”的教学目的是培养学生掌握在未来信息化社会里更好地学习、工作和生活所必须具备的计算机基础知识和基本操作技能,并进行计算机文化道德规范教育。

第二层次“计算机技术基础”是讲授计算机软硬件的基础知识、基本技术与方法,从而为学生进一步学习计算机的后续课程,并利用计算机解决本专业及相关领域中的问题打下必要的基础。

第三层次“计算机应用基础”则是讲解计算机应用中带有基础性、普遍性的知识,讲解计算机应用与开发中的基本技术、工具与环境。

以上述课程体系为依据,设计了计算机基础教育系列课程。随着计算机技术的飞速发展,计算机教学的内容与方法也在不断更新。近几年来,清华大学不断丰富和完善教学内容,在有关课程中先后引入了面向对象技术、多媒体技术、Internet 与互联网技术等。与此同时,在教材与 CAI 课件建设、网络化的教学环境建设等方面也正在大力开展工作,并积极探索适应 21 世纪人才培养的教学模式。

为进一步加强计算机基础教学工作,适应高校正在开展的课程体系与教学内容的改革,及时反映清华大学计算机基础教学的成果,加强与兄弟院校的交流,清华大学在原有工作的基础上,重新规划了“清华大学计算机基础教育课程系列教材”。

该系列教材有如下几个特色:

1. 自成体系:该系列教材覆盖了计算机基础教学三个层次的教学内容。其中既包括所有大学生都必须掌握的计算机文化基础,又包括适用于各专业的软、硬件基础知识;既包括基本概念、方法与规范,又包括计算机应用开发的工具与环境。

2. 内容先进:该系列教材注重将计算机技术的最新发展适当地引入教学中来,保持了教学内容的先进性。例如,系列教材中包括了面向对象与可视化编程、多媒体技术与应用、Internet 与互联网技术、大型数据库技术等。

3. 适应面广：该系列教材照顾了理、工、文等各种类型专业的教学要求。

4. 立体配套：为适应教学模式、教学方法和手段的改革，该系列教材中多数都配有习题集和实验指导、多媒体电子教案，有的还配有 CAI 课件以及相应的网络教学资源。

本系列教材源于清华大学计算机基础教育的教学实践，凝聚了工作在第一线的任课教师的教学经验与科研成果。我希望本系列教材不断完善，不断更新，为我国高校计算机基础教育做出新的贡献。



注：周远清，曾任教育部副部长，原清华大学副校长、计算机专业教授。

前 言

人类已经步入信息化的 21 世纪,信息时代的到来,使社会经济向知识经济发展。为了推进 21 世纪计算机基础教育改革,推进精品课程建设以及与之配套的精品教材建设,从素质教育的理念出发,结合信息化社会对高素质、复合型人才的需求,特出版此教材。此教材力求较为全面地介绍 C 语言的基本内容与程序设计思想。

C 语言是目前国内外广泛使用的程序设计语言之一,是国内外大学都在开设的重要的基础课之一。C 语言功能丰富、表达能力强、使用方便灵活、程序执行效率高并且可移植性好,既具有高级语言的特点,又具有汇编语言的特点,具有较强的系统处理能力。它支持自顶向下逐步求精的程序设计技术,其函数式结构为实现程序的模块化设计提供了强有力的保障。因此,它被广泛应用于系统软件和应用软件的开发。

本教材从 C 语言程序设计的基本原理及程序设计的基本思想出发,贯穿“基础-应用-专题”这一主线,紧扣基础,面向应用,循序渐进地引导读者学习程序设计的思想和方法。基础部分的主要内容包括程序设计中的基本概念与应用,如变量、数组、控制结构及判断结构等,并在掌握了这些基本概念与应用的基础上适时引入函数的结构与应用、指针的概念及其应用、算法设计与实现、结构型数据的应用及文件的操作等面向应用的知识点介绍,然后在上述知识点的基础上进一步介绍数据结构专题,包括链表、栈、队列和二叉树的概念与应用。为方便授课和学习,本教材还配套有电子教案和习题解析。

本书的特点之一是通俗易懂,突出“三基”(基本概念、基本原理与基本应用)的介绍与应用。

本书的特点之二是在介绍“算法设计与实现”这部分内容时,重点介绍经典的排序与查找的算法及其实现,同时通过几种不同算法的比较,讨论算法的效率及代码实现的效率。计算机程序设计,很大一部分工作是为了实现某一特定的模型或算法。本书通过介绍算法设计并通过计算机程序加以实现,可以使读者更好地学习程序设计的思想、体系结构和方法,尤其是优化的程序设计方法。

本书的特点之三是介绍了数据结构的基本内容,如链表、栈、队列和二叉树的“三基”,将这些内容作为综合应用的专题,使读者从实际应用中对 C 语言程序设计的基本知识得以融会贯通和进一步提高。

本书的特点之四是教材中有代码的详细注释。为了用最简洁的语言讲解代码结构及功能,本教材在例题代码中给出了详细的代码注释,以利于读者更好地理解代码,同时,教材中的所有代码均在 TC 3.0 下面完全通过调试。

本书的特点之五是重点突出、难点分散。这样从学习者对知识的掌握角度来说是有利的,而且这样也有利于自学。本教材标有“*”的内容为选学内容,各学校可以根据自己的实际情况决定教材的学习内容。如果这些内容不作为教学内容学习的话,也不影响其他内容在知识点上的承接关系。

本教材中将“编译预处理”内容放到附录中去,由于此部分的内容跨越了很多知识点,如果单独作为一章内容,那么只能将这部分内容放到较靠后的位置。但基础知识的学习中也牵涉编译预处理的内容,因此,将这部分内容放到附录中,可以根据需要及时查询。读者可以根据实际情况安排学习编译预处理内容的进度。

本书可供高等院校、计算机水平考试培训、各类成人教育学校作为开设程序设计课程的教材,也可供计算机爱好者自学。

参与本书编写的有黄维通、刘晓静(第 10、11、12、13 章内容)、王瑞(第 1、8、9 章内容),马力妮、王晓青、金鑫、解辉、贾续涵核对了全书的例题代码,王晓英对本教材进行了通读审核,感谢清华大学出版社为编写本书给予的大力支持。

由于作者水平有限,加上时间仓促,书中缺点和错误在所难免,恳请读者批评指正并通过作者的电子信箱告知。不胜感激,谢谢!

作者联系信箱:huangwt@tsinghua.edu.cn

黄维通

2010 年 12 月于清华园

目 录

第 1 章 C 语言的基本概念	1
1.1 程序设计语言的历史和发展	1
1.2 程序设计过程中的几个基本概念	2
1.2.1 程序	2
1.2.2 程序设计	2
1.2.3 算法	3
1.2.4 数据结构	3
1.3 软件工程的观念	4
1.4 C 语言的发展与特点	4
1.4.1 C 语言的发展和 ANSI C 标准	4
1.4.2 C 语言的特点	5
1.5 C 语言程序的基本标识符	6
1.6 C 程序的几个简单实例	7
1.7 C 程序的结构特点	10
1.8 C 语言程序的编译和执行	11
1.9 习题	12
第 2 章 C 语言程序的基本数据类型及其运算	13
2.1 C 语言的数据类型	13
2.1.1 数据类型的一般概念	13
2.1.2 常量	14
2.2 数据类型及变量	17
2.2.1 基本数据类型	17
2.2.2 变量及变量的定义	17
2.2.3 变量的初始化	18
2.2.4 数据类型转换	19
2.3 运算符和表达式	21
2.3.1 运算符和表达式概述	21
2.3.2 赋值运算符和赋值表达式	22
2.3.3 算术运算符和算术表达式	23
2.3.4 关系运算符和关系表达式	24
2.3.5 逻辑运算符和逻辑表达式	25

2.3.6	条件运算符	26
2.3.7	其他运算符	27
2.4	位运算符	28
2.4.1	按位取反运算符	28
2.4.2	移位运算符	28
2.4.3	按位“与”、按位“或”、按位“异或”	29
2.5	C 语言基本输入输出函数	30
2.5.1	字符输入输出函数	31
2.5.2	格式化输入输出函数	32
2.6	习题	37
第 3 章	基本控制结构及其应用	39
3.1	算法及结构化程序设计	39
3.1.1	算法及其特征	39
3.1.2	算法的类型与结构	41
3.2	顺序结构程序设计	42
3.3	分支结构程序设计	43
3.3.1	if 分支	43
3.3.2	if-else 分支	43
3.3.3	多分支 if-else if-else 形式	44
3.3.4	条件分支的嵌套	45
3.3.5	开关分支	46
3.4	循环结构程序设计	50
3.4.1	while 语句	50
3.4.2	do-while 语句	50
3.4.3	for 语句	51
3.4.4	三种循环的比较	52
3.4.5	多重循环	52
3.4.6	循环和开关分支的中途退出	53
3.4.7	goto 语句	54
3.4.8	结构化程序设计的注意事项	55
3.5	结构化程序设计及应用举例	57
3.6	习题	58
第 4 章	数组及其应用	62
4.1	一维数组	62
4.1.1	一维数组的定义	62
4.1.2	一维数组的初始化	63

4.1.3	一维数组的引用	64
4.1.4	一维数组的应用举例	65
4.2	多维数组	67
4.2.1	多维数组的定义	67
4.2.2	多维数组的存储形式	67
4.2.3	多维数组的引用	68
4.2.4	多维数组的初始化	68
4.2.5	数组应用举例	70
4.3	字符型数组与字符串	71
4.3.1	字符型数组的概念	71
4.3.2	字符型数组的初始化	72
4.3.3	字符型数组的输入输出	73
4.3.4	字符型数组的应用举例	74
4.4	综合应用举例	75
4.5	习题	77
第 5 章	函数及其应用	79
5.1	函数的定义与调用	79
5.1.1	C 源程序的结构	79
5.1.2	函数的定义	80
5.1.3	函数的调用	82
5.2	函数间的信息传递方式	85
5.2.1	实参-形参之间的信息传递	86
5.2.2	函数调用结果的返回	89
5.3	函数与数组	92
5.3.1	数组元素做实参	92
5.3.2	一维数组名做实参	92
5.3.3	多维数组名做参数	94
5.3.4	字符数组做函数的参数	95
5.4	递归函数与递归调用	96
5.4.1	递归的概念	96
5.4.2	递归程序的设计	97
5.5	变量的存储类型及作用域	98
5.5.1	auto 型变量	99
5.5.2	extern 型变量	99
5.5.3	静态变量	102
5.6	习题	104

第 6 章 指针	107
6.1 指针的基本概念及定义方式	107
6.1.1 指针的基本概念.....	107
6.1.2 指针的定义.....	107
6.1.3 指针的初始化.....	108
6.1.4 指针运算符.....	110
6.2 指针的运算	110
6.2.1 指针的算术运算.....	110
6.2.2 关系运算.....	112
6.2.3 指针的赋值运算.....	112
6.3 指针与数组	112
6.3.1 指向一维数组的指针.....	113
6.3.2 指向多维数组的指针.....	115
6.4 字符指针和字符串	116
6.5 指针数组	118
6.5.1 指针数组的概念.....	118
6.5.2 指针数组的应用.....	119
6.5.3 指针数组在带形参的 main 函数中的应用	120
6.6 指针在函数参数传递中的应用	122
6.7 指针型函数	122
6.7.1 指针型函数的定义和引用.....	123
6.7.2 指针型函数的应用举例.....	123
6.8 多级指针	124
6.8.1 多级指针的概念及定义.....	124
6.8.2 多级指针应用举例.....	125
6.9 指向函数的指针	126
6.9.1 指向函数的指针的概念.....	126
6.9.2 指向函数的指针的应用.....	127
6.10 动态指针.....	129
6.10.1 动态内存分配的概念	129
* 6.10.2 动态内存分配的应用	130
6.11 习题.....	132
第 7 章 排序及查找算法及其实现	134
7.1 排序概述	134
7.1.1 排序的概念.....	134
7.1.2 排序的定义.....	134
7.1.3 排序的方法.....	135

7.1.4	排序效率	135
7.2	冒泡排序法的设计及其实现	137
7.2.1	冒泡算法设计思想	137
7.2.2	冒泡算法的实现	138
7.3	选择排序法的设计及其实现	139
7.3.1	选择排序法设计思想	139
7.3.2	选择排序法的实现	140
7.4	插入排序法的设计及其实现	141
7.4.1	插入排序法设计思想	141
7.4.2	插入排序法的实现	142
7.5	SHELL 排序法的设计及其实现	143
7.5.1	SHELL 排序法设计思想	143
7.5.2	SHELL 排序法的实现	144
7.6	字符串数组的排序设计及其实现	146
7.6.1	字符串数组的排序算法设计思想	146
7.6.2	字符串数组排序算法的实现	147
7.7	查找概述	148
7.8	顺序查找及其应用	148
7.8.1	顺序查找算法的设计思想	148
7.8.2	顺序查找算法的实现	149
7.9	折半查找及其应用	150
7.9.1	折半查找算法的设计思想	150
7.9.2	折半查找算法的实现	151
7.10	习题	152
第 8 章	结构体、联合体和枚举	153
8.1	结构体的说明和定义	153
8.1.1	结构体的概念	153
8.1.2	结构体的说明及结构体变量的定义	154
8.2	结构体成员的引用与结构体变量的初始化	157
8.2.1	结构体成员的引用	157
8.2.2	结构体变量的初始化	158
8.3	结构体数组	159
8.3.1	结构体数组的定义及初始化	159
8.3.2	结构体数组的应用举例	159
8.4	结构体指针	161
8.4.1	结构体指针及其定义	161
8.4.2	通过指针引用结构体成员	162

8.4.3	结构体指针的应用举例	163
8.5	结构体在函数间的传递	166
8.5.1	结构体变量的传递	166
8.5.2	结构体数组在函数间的传递	169
8.6	结构体指针型和结构体型函数	170
8.6.1	结构体指针型函数	170
8.6.2	结构体型函数	172
8.7	结构体嵌套	174
8.7.1	结构体嵌套的概念	174
8.7.2	嵌套结构体类型变量的引用	175
8.7.3	结构体嵌套应用举例	176
8.8	联合体	177
8.8.1	联合体的说明及联合体变量的定义	177
8.8.2	使用联合体变量应注意的问题	182
8.9	枚举类型	184
8.9.1	枚举类型数据的概念及其定义	184
8.9.2	枚举型变量的使用	184
8.10	自定义类型	186
8.10.1	自定义类型及其表示形式	186
8.10.2	自定义类型的优点	187
8.11	位字段结构体	188
8.12	习题	189
第 9 章	文件操作	193
9.1	文件概述	193
9.2	文件的基本操作	197
9.2.1	文件的打开	197
9.2.2	文件的关闭	198
9.2.3	文件的删除	198
9.2.4	文件的重命名	199
9.2.5	临时文件的创建	199
9.3	文件的读写操作	200
9.3.1	文件的非格式化读写	200
9.3.2	文件的格式化写操作	206
9.3.3	文件的格式化读操作	208
9.4	文件的定位	209
9.4.1	fgetpos 函数	209
9.4.2	fsetpos 函数	209

9.4.3	ftell 函数	209
9.4.4	fseek 函数	210
9.4.5	feof 函数	211
9.5	错误处理	211
9.5.1	ferror 函数	211
9.5.2	perror 函数	211
9.6	习题	212
* 第 10 章	线性表及其应用	218
10.1	线性表的基本概念	218
10.2	结构简介	219
10.2.1	单链表	220
10.2.2	循环链表	228
10.2.3	双向链表	229
10.3	线性表的应用	229
10.3.1	单链表的应用: 单链表归并问题	229
10.3.2	循环链表的应用: 求解约瑟夫问题	231
10.4	习题	232
* 第 11 章	栈及其应用	233
11.1	栈的定义及其基本操作	233
11.2	栈的实现	234
11.2.1	栈的数组实现	234
11.2.2	栈的链式存储实现	236
11.3	栈的应用	238
11.3.1	栈在进制转换中的应用	238
11.3.2	栈在递归中的应用	241
11.4	习题	243
* 第 12 章	队列及其应用	244
12.1	队列的定义及基本操作	244
12.2	队列的实现	245
12.2.1	用循环数组实现队列	245
12.2.2	用链表实现队列	248
12.3	队列的应用	251
12.3.1	打印杨辉三角形	251
12.3.2	舞伴问题	254
12.4	习题	258

第13章 二叉树及其应用	259
13.1 树的定义和基本术语	259
13.1.1 树的定义	259
13.1.2 树的基本术语	260
13.2 二叉树的特点及其数学性质	261
13.2.1 二叉树的特点	261
13.2.2 二叉树的数学性质	261
13.3 二叉树的基本操作及其实现	263
13.3.1 二叉树的基本操作	263
13.3.2 二叉树基本操作的实现	263
13.4 二叉树的应用	268
13.4.1 二叉树的遍历	268
13.4.2 利用二叉树的遍历建立、打印二叉树	270
13.4.3 利用二叉树的遍历按树状打印二叉树	273
13.5 习题	275
附录 A 预处理命令的应用	276
A.1 预处理概述	276
A.2 宏定义	276
A.2.1 无参宏定义	276
A.2.2 带参宏定义	280
A.3 文件包含	284
A.4 条件编译	284
A.4.1 第一种条件编译形式	284
A.4.2 第二种条件编译形式	285
A.4.3 第三种条件编译形式	286
附录 B Turbo C 3.0 的上机过程	288
B.1 文件的建立与保存	288
B.2 文件的编辑	289
B.3 编译连接	290
B.4 程序的运行	291
附录 C Turbo C 3.0 程序的调试	292
C.1 程序的调试	292
C.1.1 有关调试的设置	292
C.1.2 设置断点及动态调试的基本操作	293

C.2 程序调试实例	295
C.2.1 添加断点	295
C.2.2 观察变量	296
C.2.3 单步执行	296
C.2.4 观察调用关系	297
C.2.5 终止调试	297
C.2.6 条件断点	297
C.2.7 动态修改变量值	298
 附录 D Turbo C 3.0 部分库函数表	 299
 附录 E ASCII 码表	 311
 参考文献	 312

第 1 章

C 语言的基本概念

1.1 程序设计语言的历史和发展

自 1946 年第一台计算机问世以来,计算机学科的发展逐渐引起人们的重视,计算机学科的应用也越来越广泛。目前,更是随着计算机网络的普及应用,计算机的应用已经渗入到我们的日常生活中。

计算机系统由计算机软件系统和硬件系统两部分构成。对计算机硬件系统来说,若没有软件系统支持,那么计算机硬件只能是一堆“废铁”,俗称“裸机”。那现在各种极为丰富的信息是如何被计算机硬件所识别并执行的呢?计算机硬件通俗意义上来说就像是电灯开关,它只能识别“打开”或者“关闭”这两种状态,因此对于计算机硬件来说,它需通过 0 和 1 这两种状态来识别。

1. 机器语言

上文中提到的计算机硬件只有 0 和 1 两种状态,所以说计算机硬件只能识别由 0 和 1 构成的信息。

机器语言是由 0 和 1 组成的二进制数字序列,其特点是它能直接被计算机硬件识别,所以它的执行速度快,执行效率高;但是由于所有的代码都由 0 和 1 组成,因此读写机器语言程序不直观,比较困难。

2. 汇编语言

鉴于对于机器语言程序的读写和纠错过程太困难,人们便开发了带有简洁自然语言和符号的汇编语言。

例如用 ADD 表示加法, SUB 表示减法。

但是计算机硬件不能够直接识别这些符号,所以需要一个专门的翻译程序将这些符号翻译成为计算机硬件能直接识别的机器语言。

汇编语言的产生大大提高了程序编写纠错的效率。但是随着程序规模的逐渐增加,兼容性问题也随之出现,程序员发现在某机器上用汇编语言编写的程序,换到另一台机器的时候却无法正常运行。也就是说汇编语言依赖于具体的计算机硬件,移植性较差。但是由于汇编语言具有较高的执行效率,针对具体硬件编写的汇编程序能够较好地发挥硬件作用,故在底层中开发汇编语言仍是一种较好的开发模式。

鉴于汇编语言的不足,人们越来越期望出现一种程序设计语言能够类似于我们的自然语言和常见的数学符号,既易学易懂,便于维护,又有较好的可移植性,在大部分计算机上可以通用。1954 年,人们期望中的这种语言终于问世了,那就是就是 FORTRAN 语