



高等院校规划教材

韩春成 张丹彤 主 编

C语言程序设计教程

注重学科体系的完整性，兼顾考研学生需要
强调理论与实践相结合，注重培养专业技能



中国水利水电出版社
www.waterpub.com.cn

21 世纪高等院校规划教材

C 语言程序设计教程

韩春成 张丹彤 主编

中国水利水电出版社

内 容 提 要

C 语言是一种通用的程序设计语言,具有丰富的数据类型。由于其具有简洁灵活、目标程序效率高、可移植性好以及兼有高级语言和低级语言的特点等,因此,C 语言成为计算机科学专业和各专业理想的程序设计语言之一。本书主要包括:程序设计基本概念、C 程序的基本组成、三种基本程序结构、数据、指针、构造数据类型、位运算、文件等。本书例题丰富,难点多用图示表达,叙述详细,有利于读者理解。

本书可作为高等学校各专业学生学习 C 语言程序设计的教材,也可作为各行业人员自学的教材。

本书所配电子教案可以从中国水利水电出版社网站上免费下载,网址为 <http://www.waterpub.com.cn/softdown/>。

图书在版编目(CIP)数据

C 语言程序设计教程 / 韩春成, 张丹彤主编. —北京: 中国水利水电出版社, 2006

(21 世纪高等院校规划教材)

ISBN 7-5084-3788-8

I. C… II. ①韩… ②张… III. C 语言—程序设计—高等学校—教材
IV.TP312

中国版本图书馆 CIP 数据核字 (2006) 第 085724 号

书 名	C 语言程序设计教程
作 者	韩春成 张丹彤 主编
出版 发行	中国水利水电出版社 (北京市三里河路 6 号 100044) 网址: www.waterpub.com.cn E-mail: mchannel@263.net (万水) sales@waterpub.com.cn 电话: (010) 63202266 (总机)、68331835 (营销中心)、82562819 (万水)
经 售	全国各地新华书店和相关出版物销售网点
排 版	北京万水电子信息有限公司
印 刷	北京蓝空印刷厂
规 格	787mm×1092mm 16 开本 17.25 印张 417 千字
版 次	2006 年 8 月第 1 版 2006 年 8 月第 1 次印刷
印 数	0001—5000 册
定 价	26.00 元

凡购买我社图书,如有缺页、倒页、脱页的,本社营销中心负责调换

版权所有·侵权必究

序

随着计算机科学与技术的飞速发展,计算机的应用已经渗透到国民经济与人们生活的各个角落,正在日益改变着传统的人类工作方式和生活方式。在我国高等教育逐步实现大众化后,越来越多的高等院校会面向国民经济发展的第一线,为行业、企业培养各级各类高级应用型专门人才。为了大力推广计算机应用技术,更好地适应当前我国高等教育的跨越式发展,满足我国高等院校从精英教育向大众化教育的转变,符合社会对高等院校应用型人才培养的各类要求,我们成立了“21世纪高等院校规划教材编委会”,在明确了高等院校应用型人才培养模式、培养目标、教学内容和课程体系的框架下,组织编写了本套“21世纪高等院校规划教材”。

众所周知,教材建设作为保证和提高教学质量的重要支柱及基础,作为体现教学内容和教学方法的知识载体,在当前培养应用型人才中的作用是显而易见的。探索和建设适应新世纪我国高等院校应用型人才培养体系需要的配套教材已经成为当前我国高等院校教学改革和教材建设工作面临的紧迫任务。因此,编委会经过大量的前期调研和策划,在广泛了解各高等院校的教学现状、市场需求,探讨课程设置、研究课程体系的基础上,组织一批具备较高的学术水平、丰富的教学经验、较强的工程实践能力的学术带头人、科研人员和主要从事该课程教学的骨干教师编写出一批有特色、适用性强的计算机类公共基础课、技术基础课、专业及应用技术课的教材以及相应的教学辅导书,以满足目前高等院校应用型人才培养的需要。本套教材消化和吸收了多年来已有的应用型人才培养的探索与实践成果,紧密结合经济全球化时代高等院校应用型人才培养工作的实际需要,努力实践,大胆创新。教材编写采用整体规划、分步实施、滚动立项的方式,分期分批地启动编写计划,编写大纲的确定以及教材风格的定位均经过编委会多次认真讨论,以确保该套教材的高质量和实用性。

教材编委会分析研究了应用型人才与研究型人才在培养目标、课程体系和内容编排上的区别,分别提出了3个层面上的要求:在专业基础类课程层面上,既要保持学科体系的完整性,使学生打下较为扎实的专业基础,为后续课程的学习做好铺垫,更要突出应用特色,理论联系实际,并与工程实践相结合,适当压缩过多过深的公式推导与原理性分析,兼顾考研学生的需要,以原理和公式结论的应用为突破口,注重它们的应用环境和方法;在程序设计类课程层面上,把握程序设计方法和思路,注重程序设计实践训练,引入典型的程序设计案例,将程序设计类课程的学习融入案例的研究和解决过程中,以学生实际编程解决问题的能力为突破口,注重程序设算法的实现;在专业技术应用层面上,积极引入工程案例,以培养学生解决工程实际问题的能力为突破口,加大实践教学内容的比重,增加新技术、新知识、新工艺的内容。

本套规划教材的编写原则是:

在编写中重视基础,循序渐进,内容精炼,重点突出,融入学科方法论内容和科学理念,反映计算机技术发展要求,倡导理论联系实际和科学的思想方法,体现一级学科知识组织的层次结构。主要表现在:以计算机学科的科学体系为依托,明确目标定位,分类组织实施,兼容互补;理论与实践并重,强调理论与实践相结合,突出学科发展特点,体现

学科发展的内在规律；教材内容循序渐进，保证学术深度，减少知识重复，前后相互呼应，内容编排合理，整体结构完整；采取自顶向下设计方法，内涵发展优先，突出学科方法论，强调知识体系可扩展的原则。

本套规划教材的主要特点是：

(1) 面向应用型高等院校，在保证学科体系完整的基础上不过度强调理论的深度和难度，注重应用型人才的专业技能和工程实用技术的培养。在课程体系方面打破传统的研究型人才培养体系，根据社会经济发展对行业、企业的工程技术需要，建立新的课程体系，并在教材中反映出来。

(2) 教材的理论知识包括了高等院校学生必须具备的科学、工程、技术等方面的要求，知识点不要求大而全，但一定要讲透，使学生真正掌握。同时注重理论知识与实践相结合，使学生通过实践深化对理论的理解，学会并掌握理论方法的实际运用。

(3) 在教材中加大能力训练部分的比重，使学生比较熟练地应用计算机知识和技术解决实际问题，既注重培养学生分析问题的能力，也注重培养学生思考问题、解决问题的能力。

(4) 教材采用“任务驱动”的编写方式，以实际问题引出相关原理和概念，在讲述实例的过程中将本章的知识点融入，通过分析归纳，介绍解决工程实际问题的思想和方法，然后进行概括总结，使教材内容层次清晰，脉络分明，可读性、可操作性强。同时，引入案例教学和启发式教学方法，便于激发学习兴趣。

(5) 教材在内容编排上，力求由浅入深，循序渐进，举一反三，突出重点，通俗易懂。采用模块化结构，兼顾不同层次的需求，在具体授课时可根据各校的教学计划在内容上适当加以取舍。此外还注重了配套教材的编写，如课程学习辅导、实验指导、综合实训、课程设计指导等，注重多媒体的教学方式以及配套课件的制作。

(6) 大部分教材配有电子教案，以使教材向多元化、多媒体化发展，满足广大教师进行多媒体教学的需要。电子教案用 PowerPoint 制作，教师可根据授课情况任意修改。相关教案的具体情况请到中国水利水电出版社网站 www.waterpub.com.cn 下载。此外还提供相关教材中所有程序的源代码，方便教师直接切换到系统环境中教学，提高教学效果。

总之，本套规划教材凝聚了众多长期在教学、科研一线工作的教师及科研人员的教学科研经验和智慧，内容新颖，结构完整，概念清晰，深入浅出，通俗易懂，可读性、可操作性和实用性强。本套规划教材适用于应用型高等院校各专业，也可作为本科院校举办的应用技术专业的课程教材，此外还可作为职业技术学院和民办高校、成人教育的教材以及从事工程应用的技术人员的自学参考资料。

我们感谢该套规划教材的各位作者为教材的出版所做出的贡献，也感谢中国水利水电出版社为选题、立项、编审所做出的努力。我们相信，随着我国高等教育的不断发展和高校教学改革的不断深入，具有示范性并适应应用型人才培养的精品课程教材必将进一步促进我国高等院校教学质量的提高。

我们期待广大读者对本套规划教材提出宝贵意见，以便进一步修订，使该套规划教材不断完善。

21 世纪高等院校规划教材编委会

2004 年 8 月

前 言

C 语言是一种通用的程序设计语言，具有丰富的数据类型。用户可以按一定规则组成构造数据类型的数据，利用构造类型可以实现各种复杂的数据结构。同时，由于其具有简洁灵活、目标程序效率高、可移植性好以及兼有高级语言和低级语言的特点等，因此，C 语言成为计算机专业和各专业理想的程序设计语言之一。

本书基本依据 ANSI C 标准编写，并参考教育部和一些高校计算机专业的 C 语言程序设计教学大纲，对内容进行了精心的选择和组织，以满足各个专业学习及应用计算机的要求。根据作者在教学中的经验，对学生难以理解的部分用比较多的篇幅和图形示例方式进行直观讲解。例题部分强调对基本概念、原理和方法的运用能力。这一部分围绕基本算法，按照循序渐进、覆盖面广、重点突出的原则进行选题，目的是解难释疑，开阔思路。在例题中，按照软件工程思想对空格、空行和注释等几个方面内容进行编排，希望学生建立良好的编程风格，并对经常出现的问题进行分析，启发学生建立正确程序设计概念和方法。

全书共分为 11 章。第 1 章主要介绍了 C 语言的历史沿革，程序设计的基本概念、基本过程及方法。第 2 章介绍了 C 语言的基本数据类型及表达式。第 3 章介绍了 C 语言的顺序结构的程序设计。第 4 章介绍了选择结构的程序设计。第 5 章介绍了循环结构的程序设计。第 6 章介绍了 C 语言的数组这一数据类型。第 7 章介绍了 C 语言的函数。第 8 章介绍了构造数据类型。第 9 章介绍了指针。第 10 章介绍了文件的基本内容。第 11 章介绍了位运算。在每章之后均有一定数量的习题供读者练习。

本书在编写的过程中，力求做到概念准确、内容正确，由浅入深、循序渐进、繁简适中。书中全部例题和习题都已在 Visual C++ 和 Turbo 2.0 编译环境中上机通过。为方便教学，我们为本书制作了电子教案，可在中国水利水电出版社网站上免费下载。与本书配套的还有《C 语言程序设计题解与实验指导》一书，以方便学生上机练习使用。

本书适合作为高等院校各专业本、专科学生的 C 语言程序设计课程教材，同样也可以作为广大计算机爱好者和各行各业学员学习 C 语言程序设计的参考书。

本书由韩春成、张丹彤主编，韩春成编写第 1、3、5、7、9、11 章，张丹彤编写第 2、4、6、8、10 章。最后全书由韩春成统稿。

由于作者水平有限，书中可能存在疏忽和错误之处，恳请读者和同行批评指正。

编者

2006 年 4 月

目 录

序

前言

第 1 章 概述	1
1.1 C 语言概述	1
1.1.1 C 语言的发展及其特点	1
1.1.2 C 语言程序的组成和结构特点	2
1.2 C 语言的基本符号	5
1.2.1 基本符号集	5
1.2.2 标识符	5
1.3 算法及其描述方法	7
1.3.1 算法的概念	7
1.3.2 算法的描述方法	8
1.4 程序设计方法简介	10
本章小结	12
习题	13
第 2 章 C 数据及其表达式	14
2.1 C 数据类型	14
2.2 常量	14
2.2.1 整型常量	15
2.2.2 实型常量	16
2.2.3 字符常量	16
2.2.4 字符串	17
2.2.5 符号常量	18
2.2.6 转义字符	19
2.3 变量	20
2.3.1 变量的概念	21
2.3.2 变量说明	21
2.3.3 变量地址	22
2.3.4 变量的初始化	22
2.3.5 数据类型	23
2.4 运算符和表达式	25
2.4.1 运算符及表达式	25

2.4.2 赋值运算	26
2.4.3 算术运算	27
2.4.4 逗号运算	30
2.4.5 关系运算	31
2.4.6 逻辑运算	32
2.4.7 条件表达式	33
2.4.8 运算符的结合律和优先级	34
本章小结	35
习题	37
第 3 章 顺序结构的程序设计	41
3.1 顺序结构的语句	41
3.1.1 表达式语句、空语句和赋值语句	41
3.1.2 复合语句	42
3.2 字符数据的输入和输出	43
3.2.1 字符输出函数 putchar()	43
3.2.2 字符输入函数 getchar()	44
3.3 输出函数 printf()	46
3.4 输入函数 scanf()	48
本章小结	51
习题	52
第 4 章 选择结构的程序设计	54
4.1 条件选择结构	54
4.1.1 if 语句	54
4.1.2 无条件选择结构	56
4.2 条件分支的嵌套	57
4.3 开关选择结构	60
本章小结	67
习题	68
第 5 章 循环结构的程序设计	70
5.1 用 goto 语句的循环结构	70
5.2 while 语句	71
5.3 do-while 语句	73
5.4 for 语句	75
5.5 循环嵌套结构	78
5.6 循环结构中使用 break 和 continue 语句	80
5.6.1 break 语句	80
5.6.2 continue 语句	81

5.6.3 循环语句小结	82
5.7 程序举例	82
本章小结	84
习题	85
第6章 数组	89
6.1 数组和数组元素	89
6.2 一维数组	90
6.2.1 一维数组的定义和使用	90
6.2.2 一维数组的初始化	92
6.2.3 一维数组程序举例	93
6.3 多维数组	97
6.3.1 二维数组的定义和使用	97
6.3.2 二维数组的初始化	99
6.3.3 二维数组程序举例	101
6.4 字符数组	102
6.4.1 字符数组的定义和使用	103
6.4.2 字符数组的初始化	103
6.4.3 字符串的输入和输出	104
6.4.4 用于字符处理的库函数	106
本章小结	106
习题	108
第7章 C语言函数	113
7.1 C语言程序的组成	113
7.1.1 C语言函数的结构	113
7.1.2 C语言程序的组成	114
7.1.3 存储类型和变量的作用域	115
7.2 函数的参数	118
7.2.1 带参数函数定义的一般形式	118
7.2.2 形式参数和实际参数	119
7.2.3 数组作为函数的参数	120
7.3 函数的调用	125
7.3.1 函数的原型	125
7.3.2 函数调用的一般形式	127
7.3.3 函数的值	128
7.3.4 函数的递归调用	129
7.4 程序编译预处理	133
7.4.1 宏定义	133

7.4.2	文件包含	134
7.4.3	条件编译	135
本章小结		137
习题		139
第 8 章	构造数据类型	143
8.1	结构体	143
8.1.1	结构体类型定义及结构体类型变量说明	143
8.1.2	结构体类型变量的使用	147
8.1.3	结构体变量的初始化	149
8.1.4	结构体数组	150
8.1.5	结构体和函数	152
8.2	共用体	153
8.2.1	共用体类型的定义和共用体变量的说明	153
8.2.2	共用体成员的使用	154
8.3	枚举类型	156
8.3.1	枚举类型的定义和枚举变量的说明	156
8.3.2	枚举类型数据的使用	157
8.4	用 typedef 定义类型	158
本章小结		160
习题		161
第 9 章	指针	165
9.1	指针变量的概念	165
9.1.1	变量和地址	165
9.1.2	指针变量和指针的类型	166
9.2	变量的指针与指针变量	166
9.2.1	指针变量的定义及使用	166
9.2.2	指针变量的初始化	169
9.2.3	指针运算	170
9.3	指针与数组	172
9.3.1	指向数组的指针	172
9.3.2	字符指针与字符数组	176
9.3.3	多级指针及指针数组	180
9.3.4	指针与多维数组	183
9.4	指针与函数	187
9.4.1	函数参数为指针	187
9.4.2	函数的返回值为指针	190
9.4.3	指向函数的指针	192

9.4.4 命令行参数	195
9.5 指针与结构体	198
9.5.1 结构体指针与函数	198
9.5.2 结构体指针与指向结构体数组的指针	200
9.6 链表	202
9.6.1 链表和动态存储分配	202
9.6.2 用包含指针项的结构体变量构成结点	204
9.6.3 用于动态存储分配的函数	207
9.6.4 链表应用举例	208
本章小结	214
习题	216
第 10 章 文件	222
10.1 C 语言文件概述	222
10.1.1 文件概述	222
10.1.2 缓冲文件系统和非缓冲文件系统	223
10.1.3 标准输入输出库函数	223
10.1.4 标准设备文件及 I/O 改向	226
10.2 缓冲型文件输入输出系统	227
10.2.1 文件 (FILE) 类型结构及文件指针	227
10.2.2 文件的打开与关闭	228
10.2.3 文件的读写	229
10.2.4 文件的定位	236
10.2.5 出错的检测	238
本章小结	238
习题	239
第 11 章 位运算	242
11.1 基本位运算符与位运算	242
11.1.1 按位与运算符 (&)	242
11.1.2 按位或运算符 ()	243
11.1.3 按位异或运算符 (^)	244
11.1.4 按位取反运算符 (~)	245
11.2 位移运算符与位移运算	246
11.2.1 左移运算符 (<<)	246
11.2.2 右移运算符 (>>)	247
11.3 位运算的复合赋值运算符	248
11.4 位段	248
11.4.1 位段的定义	248

11.4.2 位段的引用.....250

11.5 位运算应用举例251

本章小结253

习题254

附录 1 基本 ASCII 码表.....256

附录 2 Turbo C 常用库函数.....257

参考文献262

第 1 章 概述

1.1 C 语言概述

1.1.1 C 语言的发展及其特点

1. 程序设计语言的发展

随着计算机的诞生，用于计算机的程序设计语言随即产生，人们利用程序设计语言编制程序以便更好地使用计算机。程序设计语言经历了机器语言、汇编语言、高级语言的不同发展阶段。

机器语言是计算机能够识别和直接执行的二进制语言，用机器语言编写的程序不直观、难懂、易出错。

为了克服机器语言的不足，人们又发明了汇编语言，汇编语言是用助记符来表示机器语言，但计算机不能直接执行，需通过汇编系统将汇编语言翻译成机器语言。机器语言与汇编语言都是面向机器的语言，一般称为低级语言。

随着计算机的发展，人们发明了接近自然语言的程序设计语言，这些语言能直接表达计算式和逻辑式，如 FORTRAN、Pascal、Basic、COBOL、C 以及 C++ 等，一般称为高级语言或算法语言。

2. C 语言简介

C 语言是一种得到广泛重视并被普遍应用的计算机程序设计语言，也是国际公认的最重要的几种通用程序设计语言之一，它既可用于编写系统软件也可用来编写应用软件。

1972 年，C 语言是由贝尔实验室的 Dennis Ritchie 和 Brian Kernighan 根据 Thompson 的 B 语言设计的，而 B 语言又是由一种早期的编程语言 BCPL (Basic Combined Programming Language) 发展演变而来的。

最初的 C 语言是为描述和实现 UNIX 操作系统提供的一种工具语言。但 C 语言并没有被束缚在任何特定的硬件或操作系统上，它具有良好的可移植性。1977 年出现了不依赖于具体机器的 C 语言编译文本——《可移植 C 语言编译程序》，用该程序编写的 UNIX 系统迅速在各种机器上实现，UNIX 系统支持的 C 语言也被移植到相应的计算机上。C 语言和 UNIX 系统在发展过程中相辅相成，得到了广泛应用，使它先后被移植到各种大、中、小、微型计算机上。

以 1978 年发表的第七版本 UNIX 系统中的 C 语言编译程序为基础，B.W.Kernighan 和 D.M.Ritchie 合著了 “The C Programming Language”。这本书中介绍的 C 语言成为后来广泛使用的 C 语言版本的基础，被称为标准 C 语言。1983 年，美国国家标准化协会 (ANSI) 根据 C 语言问世以来的各种版本对 C 语言的发展和扩充制定了新的标准，称为 ANSI C。1990 年，C 语言成为国际标准化组织 (ISO) 通过的标准语言。本书介绍的 C 语言部分以 ANSI C 为基础。书中的程序全部在 Turbo C 2.0 系统中调试通过。

3. C 语言的特点

一种语言之所以能存在和发展,并具有生命力,总是有不同于其他语言的特点。C 语言也是如此,它的特点是多方面的,人们从不同的角度可总结出众多特点,但从全面考虑可归纳为:

(1) C 语言是比较低级的语言。有人把 C 语言称为高级语言中的低级语言,也有人称它是中级语言。它具有许多通常只有汇编语言才具备的功能,如位操作、直接访问物理地址等,这使得 C 语言在进行系统程序设计时显得非常有效,而过去系统软件通常只能用汇编语言编写。事实上,C 语言的许多应用场合是汇编语言的传统领地,现在用 C 语言代替汇编语言,使程序员得以减轻负担,提高效率,而且写出的程序具有更好的可移植性。

C 语言具有更多的接近硬件操作的功能,而不提供直接处理复合对象。如作为整体看待的字符串、数组等操作,这些较高级的功能必须通过调用函数来完成。这看起来是个缺陷,但这种语言规模小,更容易说明,学习起来也快。比如说,C 语言只有 32 个关键字,而一些微机上的 BASIC 语言,关键字多达 100 个以上。

(2) C 语言是结构化的程序设计语言。C 语言的主要结构成分是函数,函数允许一个程序中的各任务分别定义和编码,使程序模块化。C 语言还提供了多种结构化的控制语句,如用于循环的 for、while、do-while 语句,用于判定的 if-else、switch 语句等,十分便于采用自顶向下、逐步细化的结构化程序设计技术。因此,用 C 语言编制的程序容易理解、便于维护。

(3) C 语言具有丰富的运算能力。在 C 语言中除了一般高级语言使用的算术运算及逻辑运算功能外,还具有独特的以二进制位 (bit) 为单位的位与、位或、位非以及移位操作等运算。并且 C 语言具有如 a++、b-- 等单项运算和 +=、-= 等复合运算功能。

(4) C 语言数据类型丰富,具有现代语言的各种数据类型。C 语言的基本数据类型有整型 (int)、浮点型 (float)、字符型 (char)。在此基础上按层次可产生各种构造类型,如数组、指针、结构体、共用体等。同时还提供了用户自定义数据类型。用这些数据类型可以实现复杂的数据结构,如栈、链表、树等。因此,C 语言具有较强的数据处理能力。

(5) C 语言具有预处理能力。在 C 语言中提供了 #include 和 #define 两个预处理命令,实现对外部文件的包含以及对字符串的宏定义。同时还具有 #if ~ #else 等条件编译预处理语句。这些功能的使用提高了软件开发的工作效率并为程序的组织和编译提供了便利。

(6) C 语言可移植性好。目前,C 语言在许多机器上实现,大部分是由 C 语言编译移植得到的。C 编译程序的可移植性,也就是 C 语言程序便于移植。

C 语言的优点很多,但也有一些不足。如语法限制不太严格、类型检验太弱、不同类型数据转换比较随便,这就要求程序员对程序设计的方法和技巧更熟练,以保证程序的正确性。总之,C 语言已成为国内外广泛使用的一种编程语言,并且非常合适用于程序设计语言课程的教学工作中。

1.1.2 C 语言程序的组成和结构特点

C 语言是函数型语言,函数是构成 C 语言程序的基本单位。下面通过一个例子来分析 C 语言程序的组成和结构。

【例 1.1】C 语言程序的组成和结构。

```
main()                /* 主函数 */
{
```

```

int a,b,sum;          /* 定义 a, b 和 sum 三个变量 */
a=3; b=4;             /* 为 a, b 赋值 */
sum=add(a,b);          /* 调用函数 add, 将得到的值赋给变量 sum */
printf("sum=a+b=%d\n",sum); /* 屏幕输出 sum 变量的值 */
}

int add(int x,int y)   /* 定义 add 函数和形式参数 x, y */
{
    int z;             /* 定义 z 变量 */
    z=x+y;             /* 变量 x 与 y 相加的和送给 z */
    return(z);         /* 返回 z 的值送给 add */
}

```

运行结果为:

```
sum=a+b=7
```

从本例中可以看到, C 语言程序有如下特点:

1. C 程序是由函数组成的

C 语言源程序由若干个函数组成, 函数是 C 程序的基本单位。组成程序的若干函数中必须有且仅有一个名为 `main` 的函数。例 1.1 中包含两个函数: `main()` 和 `add()`。因为在 `main()` 函数中调用 `add()` 函数, 所以 `main()` 为主函数, `add()` 为被调用的函数。被调用函数可以是系统提供的库函数 (如例 1.1 中的 `printf()` 函数), 也可以是用户根据需要自己定义的函数 (如例 1.1 中的 `add()` 函数)。一个 C 程序可以包含零个到多个用户自定义函数。

2. C 语言函数由函数首部和函数体两部分组成

(1) 函数的首部。

这部分包括函数名、函数类型、参数名和参数类型。

如例 1.1 中 `add()` 函数的说明部分:

int	add	(int	x,	int	y)
↓	↓	↓	↓	↓	↓
函数类型	函数名	参数类型	参数名	参数类型	参数名

函数名后必须有一对圆括号(), 这是函数的标志, 函数类型是函数返回值的类型。参数类型就是形参类型。形参可以有也可以省略, 形参省略时函数名后的一对圆括号不能省略, 如 `main()` 函数就没有参数。如果有参数, 放在圆括号中, 如 `int add(int x,int y)`。

参数类型的说明也可以放在圆括号外, 是传统的函数说明形式, 如:

```

add(x,y)
int x,y;

```

这种参数类型的说明形式和放在圆括号中的参数说明形式 `int add(int x,int y)` 作用一样。但现代用法已经不用这种格式了。

(2) 函数体。

函数体由函数首部下面最外层的一对大括号内中的内容组成。注意, 一个函数如果有多对大括号, 则最外层的一对中的内容为函数体的范围。

函数体一般包含变量定义 (变量说明) 和执行语句两部分。在例 1.1 中 `main()` 函数中的 “`int a,b,sum;`” 是变量定义部分, 其余是语句执行部分。

3. main()函数

C 程序必须有 main()函数, 习惯上称其为主函数。C 语言程序总是从 main()函数开始执行, 并且在 main()函数中结束, 这与 main()函数在程序中的位置无关, main()函数可以在整个程序的任意位置, 通常把 main()函数放在程序中其他函数的前面。

4. C 程序书写格式自由

C 程序没有行号, 书写格式自由, 一行内可写多条语句, 且语句中的空格和回车符均可忽略不计。一个语句也可以写在多行上, 用“\”作续行符。

5. 程序中的每个语句后必须有一个分号

分号“;”是 C 语句的一部分。例如: “sum=a+b;”中的分号不可少, 即便是程序的最后一条语句也应包含分号。

6. C 语言本身没有输入输出语句

输入和输出的操作是由库函数 scanf()和 printf()等函数来完成的。C 语言对输入输出实行“函数化”。

7. 可以在 C 程序的任何部分加注释, 以提高程序的可读性

注释使程序变得清晰, 能帮助我们阅读和理解程序。给程序加注释是一个良好的编程习惯。C 语言注释部分由“/*”开始, 至“*/”结束, 应括在/* ... */之间, /和*之间不允许留有空格。注释部分允许出现在程序中的任何位置上。注释可为若干行, 但不允许在/* ... */中间又出现/* ... */注释。

下面介绍几个简单的 C 语言程序, 以便对 C 程序结构有进一步的了解。

【例 1.2】最小的 C 程序例。

```
main()
{ }
```

这是一个最小的 C 程序, 什么也不做, 但这是符合 C 语言函数规定的程序, 有 main()组成的函数的首部和两个大括号中无任何内容的函数体, 在计算机上运行时没有错误, 因为函数体是空的, 这个 C 程序什么也不做。

【例 1.3】函数 C 程序例。

```
main()                /*主函数*/
{
    printf("This is a C program.");
}
```

程序的运行结果:

This is a C program.

这个程序的 main()函数的函数体中有一个 printf()输出函数, 在计算机上运行时 printf()按照原样向显示屏幕输出双引号中的字符串, 本程序的字符串是英文句子 This is a C program.。

【例 1.4】编写 C 语言程序求键盘输入的两个数中较小的数, 并且将其输出到屏幕上。

```
main()                /* 主函数 */
{
    int a,b,c;          /* 定义三个变量 a,b 和 c */
    printf("Input two integers: "); /* 输出一行提示信息 */
    scanf("%d,%d",&a,&b); /* 键盘输入两个数值送给 a 和 b 变量 */
    c=min(a,b);         /* 调用 min 函数, 运行结果送给 c 变量 */
}
```



```

printf("min=%d",c);          /* 输出 c 变量的值 */
}
int min(int x,int y)          /* 定义 min 函数, x,y 为形参 */
{
    int z;
    if(x<y) z=x;              /* 条件语句: 如果 x 小于 y 成立, 则 z=x, 否则 z=y */
    else    z=y;
    return(z);                /* 返回 z 的值, 通过 min 带回调用处 */
}

```

当运行上面这个程序时, 首先, 屏幕上显示一条提示信息:

Input two integers:

要求用户从键盘输入两个整数。如果用户输入 3 和 5, 即

Input two integers: 3,5 ✓

这里, 符号✓表示按一下回车键, 以示输入的结束。此时屏幕显示运行结果:

min=3

本程序包括主函数 `main()` 和被调用的函数 `min()`。程序执行 `scanf()` 时, 操作员由键盘输入两个整数值送给 `a` 和 `b` 变量。程序执行 “`c=min(a,b)`” 时, 调用 `min()` 函数, 将 `a` 的值送给 `x`, `b` 的值送给 `y`, 程序转到 `min()` 函数执行, `min()` 函数中的 `if` 语句的作用是将 `x` 变量和 `y` 变量中的较小值赋给 `z` 变量。`return` 语句的作用是将 `z` 变量的值返回给 `min()` 函数同时程序返回主函数执行, `min()` 函数值再送给 `c` 变量。最后 `printf()` 输出 `c` 变量的值到屏幕。

1.2 C 语言的基本符号

1.2.1 基本符号集

C 语言的基本符号是指在 C 程序中可以出现的字符, 主要是由 ASCII 字符集中的字符组成, 包括阿拉伯数字、大小写英文字母、特殊符号、转义字符和键盘符号。这些字符多数是可以见到的, 对于不可见的字符 (如回车键), C 语言规定用转义字符来表示, 转义字符将在本书 2.2 节介绍。C 语言的基本符号具体是以下几个部分:

(1) 阿拉伯数字 10 个: 0、1、2、3、…、9

(2) 大小写英文字母各 26 个: A、B、C、…、Z、a、b、c、…、z。

(3) 下划线: `_`。

(4) 特殊符号, 主要是指运算符和操作符, 它通常是由 1~2 个特殊符号组成:

+、-、*、/、%、<、<=、>、>=、==、!=、&&、||、!、
 ,、&、|、~、=、++、--、?:、<<、>>、()、[、]、.、
 ->、+=、-=、*=、/=、%=、&=、^=、|=、^、#、sizeof。

1.2.2 标识符

标识符是一个在 C 语言中作为名字的字符序列, 用来标识变量名、类型名、数组名、函数名、文件名等, 其实标识符就是一个名字, C 语言规定了标识符的命名规则。C 语言的标识符可分为用户标识符、保留字和预定义标识符三类, 有些教材称保留字为“关键字”。