

21世纪高等教育精品规划教材



C语言程序设计

主编 吴学东 田文成



天津大学出版社
TIANJIN UNIVERSITY PRESS

21 世纪高等教育精品规划教材

C 语言程序设计

主 编	吴子东	田文成	
副主编	张单阳	那一沙	
参 编	郑万明	翟自强	张爱玲
	童锡骏	王淑英	刘 瑶
	那 娜	杨顺起	

 天津大学出版社
TIANJIN UNIVERSITY PRESS

内 容 提 要

本书是根据《全国计算机等级考试二级考试大纲》和《天津市高职接本大纲》中 C 语言程序设计的要求编写的。全书共分 8 章,依次介绍了 C 语言的基本概念、程序的基本控制结构、数组、函数、指针、结构体和共用体、预处理命令和文件。每章用标准的、基本的、典型的实例介绍语句的基本概念和方法。每章后配有丰富的练习题和实验题并给出了习题的参考答案,在实验题中也给出了详细的解题思路和参考程序。

本书既可作为普通高等院校、高职高专的教材,也可供参加计算机等级考试的人员和其他专业技术人员参考。

图书在版编目(CIP)数据

C 语言程序设计/吴子东,田文成主编. —天津:天津大学出版社,2007.9

21 世纪高等教育精品规划教材

ISBN 978-7-5618-2497-9

I . C . . . II . ①吴 . . . ②田 . . . 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(2007)第 107226 号

出版发行	天津大学出版社
出 版 人	杨欢
地 址	天津市卫津路 92 号天津大学内(邮编:300072)
电 话	发行部:022-27403647 邮购部:022-27402742
网 址	www.tjup.com
短信网址	发送“天大”至 916088
印 刷	天津泰宇印务有限公司
经 销	全国各地新华书店
开 本	185mm × 260mm
印 张	18.75
字 数	480 千
版 次	2007 年 9 月第 1 版
印 次	2007 年 9 月第 1 次
印 数	1 - 3,000
定 价	29.00 元

凡购本书,如有缺页、倒页、脱页等质量问题,烦请向我社发行部门联系调换

版权所有 侵权必究

前 言

计算机已经是现代社会必不可缺的重要工具。在诸多的计算机应用领域中,程序设计是学会用计算机解决实际问题的必由之路。学习程序设计就是学习独立开发程序。C语言具有简单易学、功能丰富、表达力强、使用灵活方便、程序执行效率高、可移植性好的特点,既有低级语言可直接操作计算机硬件的能力,又有高级语言的易读特点;既可编写系统软件,又可编写应用软件。学好C语言也是学习高层次的C++和Java等语言必须具备的基础。

本教材就是为学生学习程序设计而编写的,与其他介绍C语言程序设计的教材相比较有以下特点:其一,在文字叙述上采用通俗易懂的语言来描述,对每个语句都采用给出标准格式的方法进行介绍,增加了许多带有总结归纳性的内容;其二,在一些典型例题中重点介绍了程序设计的方法,目的是使读者对程序设计有全面的了解。本书每章还都配备了一定数量的习题和实验题,尤其实验题还都给出了详细的解题思路和参考程序,目的是让读者巩固所学知识。

本教材共分8章。第1章主要介绍C语言的基础知识和基本概念。第2章从结构化程序设计角度讲解了程序的三种基本控制结构(顺序结构、分支结构和循环结构)的标准语法和基本使用方法。第3章介绍了数组的基本概念,包括一维数组、二维数组和字符数组的基本使用方法。第4章全面、重点地介绍了函数的标准格式、函数的调用方法、函数的说明方法以及函数间数据传递的方法。第5章介绍了C语言的重要概念——指针,详细地介绍了指针的基本概念、指针的运算、指针与数组的关系,讨论了指针和数组在函数调用中的几种结合方法,并且介绍了指针数组与函数指针。第6章结构体和共用体,介绍了它们是如何进行类型定义、变量说明以及引用的,还介绍了在C语言中如何进行类型定义,为后续课程打下基础。第7章讨论了预处理命令宏定义和文件包含。第8章讨论了用C语言处理文件的方法,重点介绍了顺序文件的打开方式,读、写方式以及随机文件的使用。

为了保证教材的质量,各章至少有两人参加了撰写,多人参与了意见,以便集思广益,尽量使内容丰富而准确无误。参加编写的人员有天津大学职业技术学院吴子东、那一沙,天津电子信息职业技术学院田文成、张单阳、翟自强,西安理工大学高等技术学院张爱玲,天津商业大学郑万明,天津市行政学院童锡骏,天津青年职业学院王淑英、刘瑶,天津市育才职业中专那娜,天津市广播电视大学杨顺起。

吴子东、田文成副教授作为主编除了参加本书的编写工作外,还主要承担了全书统稿和规范格式的工作,数遍浏览,修改全书。各位参编人员对该书提出了不少宝贵意见和建议。

此外,在本书编写过程中得到了天津大学职教学院和各位参编人员所在单位领导及其基层组织和同志们的帮助和支持,在此表示衷心感谢。在写作过程中还参考了许多专业书籍和

重要资料、文献,在此也一并向著者和提供者表示衷心感谢。

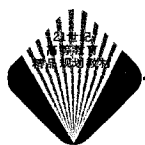
最后还需说明的是,限于作者的水平,时间仓促,书中难免有不足之处,恳请各位专家、任课教师和广大读者批评指正,在此先致以谢意。

作 者

2007年5月8日

目 录

1 C语言基础	(1)
1.1 C语言概述	(1)
1.1.1 C语言的发展与特点	(1)
1.1.2 C语言的基本语法	(3)
1.1.3 C语言程序结构和风格	(4)
1.2 C程序的输入与输出	(6)
1.2.1 格式输入与输出	(7)
1.2.2 字符数据输入与输出	(9)
1.3 C程序的开发过程	(11)
1.3.1 C程序开发的一般步骤	(11)
1.3.2 几种常用编译环境简介	(12)
1.3.3 Turbo C集成环境操作	(13)
1.4 数据类型	(18)
1.4.1 C数据类型概述	(18)
1.4.2 常量	(19)
1.4.3 变量	(20)
1.5 运算符与表达式	(22)
1.5.1 运算符概述	(22)
1.5.2 算术类运算符	(23)
1.5.3 关系运算符与逻辑运算符	(25)
1.5.4 位运算符	(26)
1.5.5 其他基本运算符与专用运算符	(28)
1.5.6 表达式	(30)
1.6 例题详解	(32)
习题	(33)
实验	(38)
2 程序的基本控制结构和算法	(42)
2.1 程序设计的三种基本控制结构	(42)
2.1.1 顺序结构	(42)
2.1.2 选择结构	(43)
2.1.3 循环结构	(43)
2.2 条件语句	(44)
2.2.1 if语句的三种形式	(44)
2.2.2 if语句的嵌套	(46)



2.2.3	switch 语句	(48)
2.3	循环语句	(50)
2.3.1	goto 语句	(50)
2.3.2	for 语句	(51)
2.3.3	while 语句	(52)
2.3.4	do-while 语句	(52)
2.3.5	循环的嵌套	(53)
2.3.6	break 和 continue 语句	(55)
2.4	算法的概念	(56)
2.4.1	算法的描述方法	(56)
2.4.2	算法设计的基本方法	(58)
2.5	例题详解	(62)
习题		(64)
实验		(71)
3	数组	(75)
3.1	一维数组	(75)
3.1.1	一维数组的定义	(75)
3.1.2	一维数组的使用	(76)
3.2	二维数组	(78)
3.2.1	二维数组的定义	(79)
3.2.2	二维数组的使用	(79)
3.3	字符数组	(80)
3.3.1	字符数组的定义	(80)
3.3.2	字符数组的使用	(81)
3.4	例题详解	(84)
习题		(87)
实验		(93)
4	函数	(97)
4.1	函数概述	(97)
4.1.1	函数种类	(97)
4.1.2	用户自定义函数	(98)
4.2	函数参数及返回值	(100)
4.2.1	函数参数	(100)
4.2.2	函数返回值	(102)
4.3	函数的一些用法	(102)
4.3.1	函数调用一般格式	(102)
4.3.2	函数说明	(103)
4.3.3	函数递归调用	(105)

4.3.4 局部变量与全局变量	(106)
4.4 例题详解	(110)
习题	(111)
实验	(118)
5 指针	(122)
5.1 指针与指针变量	(122)
5.1.1 指针与指针变量的概念	(122)
5.1.2 指针变量的定义及初始化	(123)
5.1.3 指针运算	(125)
5.2 指针变量的引用	(128)
5.2.1 用指针变量访问内存	(128)
5.2.2 指针变量作为函数的参数	(129)
5.2.3 指针与数组	(129)
5.2.4 指针与字符串	(132)
5.2.5 指向二维数组的指针	(133)
5.2.6 指针数组	(134)
5.3 函数指针	(135)
5.3.1 函数指针的定义及赋值	(135)
5.3.2 通过函数指针调用函数	(136)
5.4 例题详解	(137)
习题	(140)
实验	(148)
6 结构体和共用体	(153)
6.1 结构体	(153)
6.1.1 结构体类型定义和变量说明	(153)
6.1.2 结构体变量的访问及初始化	(156)
6.2 共用体	(159)
6.2.1 共用体类型定义和变量说明	(159)
6.2.2 共用体变量的引用	(160)
6.3 类型定义	(161)
6.4 例题详解	(161)
习题	(162)
实验	(166)
7 预处理命令	(169)
7.1 宏定义	(169)
7.1.1 宏定义的建立	(169)
7.1.2 宏定义的终止	(173)
7.2 文件包含命令	(174)



7.2.1	文件包含的形式	(174)
7.2.2	文件包含的功能	(175)
7.2.3	文件包含的嵌套	(175)
7.3	例题详解	(176)
习题		(177)
实验		(180)
8	文件	(182)
8.1	文件概述	(182)
8.1.1	文件的基本概念	(182)
8.1.2	文件指针	(183)
8.1.3	标准设备文件的换向	(184)
8.2	文件的打开与关闭	(185)
8.2.1	文件的打开函数 <code>fopen()</code>	(185)
8.2.2	文件的关闭函数 <code>fclose()</code>	(186)
8.3	文件的顺序读、写	(186)
8.3.1	字符读、写函数	(187)
8.3.2	字符串读、写函数	(188)
8.3.3	格式化读、写函数	(189)
8.3.4	数据块读、写函数	(190)
8.4	文件的随机读、写	(192)
8.4.1	文件指针变量的定位	(192)
8.4.2	文件的随机读写	(193)
8.5	文件的检测	(194)
习题		(194)
实验		(201)
附录 1	ASCII 码编码表	(204)
附录 2	C 语言中常用的关键字	(206)
附录 3	C 语言运算符优先级和结合性	(207)
附录 4	C 语言常用库函数	(208)
附录 5	综合练习	(215)
附录 6	习题参考答案	(263)
参考文献		(292)

1 C 语言基础

本章主要介绍 C 语言的由来和发展、特点、基本语法、数据类型、运算符与表达式、程序结构及上机调试步骤等。

1.1 C 语言概述

1.1.1 C 语言的发展与特点

1.1.1.1 C 语言的发展

C 语言的产生可以追溯到 20 世纪 60 年代曾经流行的 ALGOL 60 语言。后来,ALGOL 语言向两路分支发展:一路是面向问题的结构严谨的程序设计语言——Pascal;一路是更为接近硬件的结构化程序设计语言——C 语言,即 ALGOL 语言经过了 CPL(Combined Programming Language——复合程序设计语言,1963 年由英国剑桥大学研制推出)、BCPL(Basic CPL,1967 年由英国剑桥大学 Marting Richards 对 CPL 进行简化)、B 语言(1970 年由美国 Bell 实验室 Ken. Thompson 进一步简化并只用字头 B 命名),再到 C 语言演化过程(1972 年由美国 Bell 实验室 Brian. W. Kennighan 和 Dennis. M. Ritchie 两人充实完善并继之用 C 命名)。C 语言的推出者(K&R)继续合作,用 C 改写了 UNIX 操作系统,并使之运行在 DEC PDP-11 计算机上。

C 语言适合作为系统描述语言,可用于编写系统软件及应用软件。因此 C 语言诞生后,表现出很强的生命力。1977 年推出不依赖于具体机器的 C 语言编译系统。1978 年后 C 语言先后移植到各种大、中、小型计算机和微型机上,并已经独立于特定环境——UNIX 和 PDP 小型机,同时出现了许多 C 语言版本。1983 年夏,美国国家标准化协会(ANSI)依据 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标准——ANSI C。1987 年又完善为 87 ANSI C。目前,微机上广泛使用的 C 编译系统 Turbo C、Microsoft C、Quick C 等都是以此为基础的。但是,不同的编译系统虽然基本语句和基本函数相同,但在使用方法上和某些应用函数(如图形函数)上却有所不同。即使同一编译系统,不同版本也有些差异,因此在使用特定 C 编译系统时需参阅有关手册。本书内容遵从 ANSI C 标准,实际使用的 C 语言编译系统是在微机上广泛使用的 Turbo C 2.0 版本。

C 语言现在已经成为软件产业的主导语言,并仍在继续发展:C++ 就是在 C 基础上扩充了面向对象的程序设计而发展成的一种全新语言,C 成为 C++ 的子集;C++ 则是 C 的超集;VC(Visual C)则是 C++ 可视化编程版本;C# 是微软公司仿 Java 语言对 C++ 进一步改进,完全适合微软 .NET 平台,它是现代、简单、完全面向对象、类型安全的新型语言,学起来也更为简单。

1.1.1.2 C 语言的特点

C 语言发展如此迅速,而且成为最受欢迎的语言之一,主要因为它具有强大的功能。许多著名的系统软件,如 DBASE III PLUS、DBASE IV 都是由 C 语言编写的。用 C 语言加上一些汇编语言子程序,就更能显示 C 语言的优势了,如 PC-DOS、WordStar 等就是用这种方法编写的。归纳起来 C 语言具有下列特点。

(1)C 是中级语言 它把高级语言面向用户的能力与低级语言操纵硬件的能力有效地结合起来,兼有两方面的优点。它较其他高级语言更接近硬件,可以像汇编语言一样对位、字节、字和地址进行操作,运行速度非常快(仅比汇编语言慢),适合编写系统软件和实时控制、图形处理、文字编辑等实用程序;又由于具有高级语言易读、易写、易查错、易维护等特点,也可以编写科学计算、数据处理等应用程序。

(2)C 是结构化语言 结构化语言的显著特点是代码及数据的分隔化,程序各个部分除了必要的信息交流外彼此独立。C 语言以函数形式提供给用户使用或调用,程序逻辑结构由顺序、条件、循环三种基本成分控制程序流向,拒绝乱用 goto 语句,从而使程序完全结构化。这种结构化方式可使程序层次清晰,便于编写、调试、维护和使用。

(3)数据类型和运算符特别丰富 数据类型中基本类型有字符型、整型、实型,还可构造数组、结构和共用等类型,方便地处理链表、树、栈等复杂数据结构,特别是引入了独特的指针概念,可使程序运行效率更高。其运算符多达 44 个,其中包括位运算符、独特的增减 1 运算符和复合赋值运算符,使构成表达式灵活多样,能完成其他高级语言难以表达的运算。

(4)简洁紧凑、移植性好 C 语言仅有 32 个关键字和 9 种控制语句,程序用小写字母书写,形式自由,压缩了一切不必要的成分。C 语言提供了一组预处理语句,为有效地组织和编译程序提供了方便。C 语言没有依赖于硬件的输入/输出语句(该功能是调用系统提供的库函数实现的),因此 C 语言程序基本上不加修改就能从一种机器移植到另一种机器上。C 语言适于多种机型,支持多种显示器和驱动器,也适合于多种操作系统(如 DOS、UNIX 等)。

(5)语法限制不太严格 C 语言对变量的类型不一致性不作语法检查,几乎允许所有类型互相转换,对变量和指针、数组越界没有设置屏障等,这使程序设计自由度大,保持了 C 语言高效灵活的特点,但也容易导致安全性欠缺,程序员不能仅依赖编译程序查出错误,应靠自己保证程序的正确性。这对程序员来说提高了要求,对初学者来说增加了学习难度。

1.1.2 C 语言的基本语法

1.1.2.1 字符集

C 语言的字符集包括英文字母、数字和一些有特定含义的标点符号,任何 C 程序都是由字符集里的符号构成具有一定含义的语句,再由这些语句组成程序。C 的字符集是 ASCII 码字符集的一部分,若使用 C 字符集之外的符号,就会产生出错信息。C 的字符集如下:

- ①英文字母 a~z, A~Z;
- ②数字字符 0~9;
- ③特殊字符(以 ASCII 码为序)有空格! " # % & ' () * + , - . / : ; < = > ? [\] ^ _ { | } ~ (可显示的 ASCII 中只有 \$ @ 未列入);

- ④控制码(以 ASCII 码为序)HT、LF、VT、FF、CR(意义见附录 1)。

1.1.2.2 保留字

保留字也称关键字。就是已被 C 编译系统预先定义的专用标识符(均为小写字母)。它们可作为存储类型符、数据类型符和语句定义符,而不能作其他用途使用(比如不能用作变量名、函数名等的标识符)。C 的保留字如下。

- ①ANSI C 标准定义的关键字共 32 个(按字母顺序):

auto	break	case	char	const	continue	default	do
double	else	enum	extern	float	for	goto	if
int	long	register	return	short	signed	sizeof	static
struct	switch	typedef	union	unsigned	void	while	volatile

- ②其他 C 语言版本(如 Turbo C 2.0、Microsoft C 等)扩充的关键字有:

asm	cs	ds	es	ss	cdecl	far	huge
interrupt	neer	pascal					

- ③预处理中专用词(这些单词属于预处理专用词,不属于关键字,但不要在程序中随便用)有:

define	undef	include	ifdef	endif	line	elif	ifndef
--------	-------	---------	-------	-------	------	------	--------

1.1.2.3 转义字符和格式转换说明符

转义字符由反斜杠(\)后跟一个字母或几个数字组成,有着固定的含义,表示输出相应的控制代码或特殊字符,如:\n 表示回车换行、\r 表示回车、\101 或 \x41 表示字母 A 等等。格式转换说明符由百分号(%)后跟一个符合规定的字母表示输出的格式转换说明符,如:%d 表示按十进制整数输出、%s 表示输出字符串等等。此两种字符参见表 1-1 和表 1-2。



表 1-1 格式转换说明符

格式字符	说 明	格式字符	说 明
d	带符号的十进制整数(正数无符号)	f	小数形式单、双精度数,隐含输出 6 位小数
o	八进制无符号整数	s	控制输出字符串
X,x	十六进制无符号整数	u	无符号十进制整数,仅限输出用
c	仅限单个字符	p	输出指针值(变量地址)
E,e	指数形式的单、双精度数	G,g	选用%f或%e格式中宽度较短的一种格式

备注:表中大写字母是 Turbo C 2.0 所允许的。

表 1-2 转义字符

转义字符	Turbo C 规定	说 明	转义字符	Turbo C 规定	说 明
\n	\X0A	换行,将当前位置移到下一行开头	\a	\X07	响铃报警
\t	\X09	水平制表(跳到下一个 tab 位置)	\\	\X5C	表示反斜杠字符\
\v	\X0B	垂直制表	\'	\X27	表示单引号字符'
\b	\X08	退格,将当前位置左移一格	\"	\X22	表示双引号字符"
\r	\X0D	回车,将当前位置移到下一行开头	\xhh		1到 2 位十六进制代表的字符
\f	\X0C	换页,将当前位置移到下一页开头	\ddd		1到 3 位八进制代表的字符

1.1.2.4 标识符

标识符是指常量、变量、数组、函数、类型、语句标号和文件的名称,这里主要是指用户自定义以上对象的名称。C 语言标识符的定义十分灵活,但必须满足以下规则:

- ①必须使用 C 字符集中的英文字母、数字和下画线,并由字母或下画线开头;
- ②字母大小写不同,所代表的标识符不同(不能使用 C 语言的保留字);
- ③标准长度为 1~32 个字符,而微机中只有前 8 个字符有效,8 个后的字符被忽略。

下面举出几个正确和不正确的标识符:

正确	不正确	正确(用于特定地方)	不正确
smart5	5smart	X_B_T	X·B·T
_decision	bomb?	FLOAT	float
key_board	key.board	A1	a%b

1.1.3 C 语言程序结构和风格

本节首先通过一个简单而又较完整的 C 语言程序介绍 C 语言程序的构成和书写风格。

1.1.3.1 简单的 C 语言程序

从键盘输入一个圆半径(整数)计算并显示圆面积的程序。

```
#include <stdio.h>          /*用预处理命令指定一个包含文件*/
#define PI 3.1416            /*用宏替换命令定义一个常量*/
main()                      /*主函数*/
```

```
{ /*main 函数体开始标记 */
    float area(); /*函数类型说明*/
    int r; /*内部整型变量说明*/
    float a; /*内部实型变量说明*/
    printf("请输入半径 r = "); /*调用 printf 库函数进行提示准备输入*/
    scanf("%d",&r); /*调用 scanf 库函数接受键盘输入*/
    a = area(r); /*调用自定义函数 area 给 a 赋值*/
    printf("圆面积 A = %f\n",a); /*调用 printf 库函数显示计算结果*/
} /*main 函数体结束标记*/

float area(x) /*定义计算圆面积函数 area */
int x; /*函数的形式参数类型说明*/
{ /*area 函数体开始标记*/
    float y; /*内部实型变量说明*/
    y = PI*x*x; /*计算圆面积*/
    return(y); /*返回圆面积值*/
} /*area 函数体结束标记*/
```

上面的程序中每条语句右边都加了注释,这是 C 语言所允许的。这样,即使 C 语言的初学者也基本能够读懂上面的程序。

1.1.3.2 C 为函数式语言

所有 C 程序都是由函数模块构成的,本程序就涉及主函数 main()、计算圆面积函数 area() 和格式化输出输入库函数 printf()、scanf(),从整体上看 C 程序就是函数的集合体。

main()是 C 程序不可少的主函数。它相当于其他语言的主程序,但它同其他函数一样,位置一般可前可后,而程序运行时却总是从主函数的第一条语句开始执行。

函数由说明部分和函数体组成。C 程序对变量和函数必须“先说明,后使用”。关于函数问题将在第 4 章展开介绍。

库函数是 C 语言的标准函数。其定义包含在以 .h 为扩展名的包含文件(也称头文件)中。格式化输入输出库函数是经常使用的两个函数,将在 C 程序的输入与输出中介绍。

1.1.3.3 编译预处理命令

在 C 源程序前一般是以 # 号开头的预处理命令。这些特殊命令表示是在编译 C 源程序之前应该进行的预先处理工作。

(1) # include 文件包含命令。上面程序中引用了包含文件 stdio.h,在编译预处理时就会将包含文件 stdio.h 包含到程序中,在连接时就会将所引用某些目标代码添加到最后形成的目标程序中。

(2) # define 宏定义命令。上面的 area()函数引用了 PI 值,需要先通过该宏定义预处理命令定义宏名 PI。宏名习惯上用大写以区别普通变量。宏的使用提高了程序的可读性和可移植性,修改也方便了。

关于预处理命令(还包括条件编译命令)将在第 7 章进一步介绍。



1.1.3.4 结构化程序设计

C 语言每个函数就是一个模块,顶层模块 `main()` 函数可以调用其他函数模块。按照自顶向下的原则,把问题逐层分解、逐步求精,把大任务变成若干个小任务,直到每个小任务都能用基本结构(顺序、选择、循环)表示终止,使每一个模块都具有单一性、完整性和独立性。这就是结构化程序设计方法。这种方法优点如下。

(1)分散程序设计难点 一个复杂程序分解为若干功能单一、相对独立的模块,而每个模块单独设计、编程、调试,自然就分散了难点。

(2)提高程序的可靠性 由于模块间相对独立,一个函数的错误就不易扩散、蔓延到其他函数中去,因而整个程序的可靠性会提高。

(3)缩短软件开发周期 只需规定函数模块间接口关系,就可以组织多人分别承担任务,联合开发一个复杂程序。

(4)提高软件开发效率 有些函数模块可以直接丰富库函数,可以在多个程序中应用,避免重复开发。

(5)可读性好 模块化程序结构清晰、层次分明,每个模块也短小、简单,容易阅读。

(6)维护方便 修改某个模块不会波及其他模块,扩充功能、增加模块不会涉及整个程序大规模修改。

1.1.3.5 C 程序书写风格

除了规定用小写字母以外,总的来说,C 程序的书写风格是简洁、高效、灵活、结构化。书写 C 程序时注意以下问题:

①C 语言严格区分大小写,关键字必须小写,其他也习惯用小写字母,有些情况下用大写字母区分;

②C 语句没有行号,使用分号“;”作为语句分隔符或语句结束符,一行内可写多条语句,一个语句可分写在多行上,但最好一行写一条语句,并用分号结束(对此需要特别注意);

③大括号{}用于标定一个函数范围或一个复合语句范围,必须前后呼应,配对使用;

④必要时,可在语句后用“/*注释内容*/”格式对语句进行注释,增加程序的可读性,而编译系统将忽略这些注释;

⑤C 语言书写格式相当灵活,但建议采用结构化书写格式,即不同层次的结构最好从不同的起始位置开始书写,同一结构层次的语句按同样字数缩进(输入源程序时可通过 Tab 键、空格键进行控制)。

1.2 C 程序的输入与输出

几乎所有的程序都涉及输入/输出功能,而 C 语言本身不提供输入输出语句,它是靠标准输入输出库函数,即 `putchar()`、`getchar()`、`printf()`、`scanf()` 等函数完成的。这些库函数均以标准输入输出设备(一般为键盘和屏幕)为输入输出对象,函数代码就放在 C 语言编译系统的头文件中。因此,在使用库函数时,要用预处理命令 `#include` 将有关的头文件书写在用户源文件前面,如: `#include <stdio.h>` 或 `#include "stdio.h"`。若无此语句说明,则连接时将会出现错

误(但是若只使用 `printf()`、`scanf()` 函数,可以省略文件预处理命令 `#include <stdio.h>`)。

1.2.1 格式输入与输出

在标准输入输出设备上采用格式化输入输出数据是最常用的功能,C 语言是通过调用格式化输入输出函数而实现的。

1.2.1.1 格式化输出函数 `printf()`

该函数用于向标准输出设备(屏幕)按规定格式输出信息。其调用格式如下:

```
printf("格式控制字符串", 输出项列表);
```

1) 格式控制字符串

格式控制字符串包括两部分内容:一部分是正常字符,这些字符将按原样输出;另一部分是格式转换说明符和转义字符。格式转换说明符以 `%` 开始,后跟符合规定的字符(见表 1-1),用以确定后续输出列表项内容输出格式;转义字符以 `\` 开始,后跟一个或几个数字组成(见表 1-2),表示控制代码或格式转换说明符所占用的字符(如 `%`、`\` 等等)。

格式转换说明符的使用说明:

①输出数据宽度采用 `m.n` 的形式夹在 `%` 和格式转换说明符之间,若不指定则以默认宽度输出。其中 `m` 为输出数据总宽度(若有小数则包含 1 位小数点),`n` 为小数位数。

如果字符串的长度或整型数位数超过说明的宽度,将按其实际长度输出;对浮点数则按实际整数位输出,若小数部分位数超过了小数位说明宽度,则按说明宽度以四舍五入输出;如果用实数表示字符或整型量的输出格式,小数点前后分别表示最大宽度和最小宽度。

②一般输出数据为右对齐,若左对齐则在 `%` 和格式转换说明符之间添加减号。

③若为长整型数据则在 `%` 和格式转换说明符(仅限 `d`、`o`、`x`、`u` 格式)之间加小写字母 `l`(注意小写字母 `l` 和数字 `1` 极易混淆)。

如: `%3d` 表示输出 3 位整型数,不够 3 位右对齐,左补空格; `%9.2f` 表示输出宽度为 9 的浮点数,小数位为 2,整数位为 6,不够 9 位右对齐,左补空格; `%8s` 表示输出 8 个字符的字符串,不够 8 个字符右对齐,左补空格; `%6.9s` 表示显示一个长度不小于 6 且不大于 9 的字符串,若大于 9 则第 9 个字符以后的内容将被删除,不够 6 个字符右对齐,左补空格; `%ld` 表示输出 long 整数; `%lf` 表示输出 double 浮点数; `%-7d` 表示输出 7 位整数左对齐,不够 7 位,右补空格; `%-10s` 表示输出 10 个字符左对齐,不够 10 位,右补空格。

2) 输出项列表

此表指明输出的一系列参数,顺序和个数必须与格式控制字符串说明的格式转换说明符一一对应,各输出项间用逗号分开,否则将会出现错误。

例 1.2.1 `printf()` 函数使用。

```
#include <stdio.h>
#include <string.h>
main()
{ char c, s[20], *p;
  int a = 1234, *i;
  float f = 3.141592653589;
```




```
double x = 0.12345678987654321;
p = "How do you do";
strcpy(s, "Hello, Comrade");
*i = 12;
c = '\x41';

printf("a = %d\n", a);          /*结果输出十进制整数 a = 1234*/
printf("a = %6d\n", a);         /*结果输出 6 位十进制数 a = 1234*/
printf("a = %06d\n", a);        /*结果输出 6 位十进制数 a = 001234*/
printf("a = %2d\n", a);         /*a 超过 2 位, 按实际值输出 a = 1234*/
printf("i = %4d\n", *i);        /*输出 4 位十进制整数*i = 12*/
printf("i = %-4d\n", *i);       /*输出左对齐 4 位十进制整数*i = 12*/
printf("i = %p\n", i);          /*输出地址 i = 02F9(不同系统或机器输出值可
                                能会不同)*/

printf("f = %f\n", f);          /*输出浮点数 f = 3.141593*/
printf("f = 6.4f\n", f);        /*输出 6 位其中小数点后 4 位的浮点数 f =
                                3.1416*/

printf("x = %lf\n", x);          /*输出长浮点数 x = 0.123457*/
printf("x = %18.16lf\n", x);     /*共 18 位, 小数点后 16 位的长浮点数 x =
                                0.1234567898765432*/

printf("c = %c\n", c);          /*输出字符 c = A*/
printf("c = %x\n", c);          /*输出字符的 ASCII 码值 c = 41*/
printf("s[] = %s\n", s);        /*输出数组字符串 s[] = Hello, Comrade*/
printf("s[] = %6.9s\n", s);      /*输出最多 9 个字符的字符串 s[] = Hello, Co*/
printf("s = %p\n", s);          /*输出数组字符串首字符地址 s = 2000, 不同计
                                算机上会不同*/

printf("p = %s\n", p);          /*输出指针字符串 p = How do you do*/
printf("p = %p\n", p);          /*输出指针的值 p = 0194, 地址值在不同计算机
                                上可能不同*/

}
```

1.2.1.2 格式化输入函数 scanf()

该函数用于从标准输入设备(键盘)按规定格式接收输入的信息。其调用格式为:

scanf("格式控制字符串",地址表列);

(1)格式控制字符串 格式控制字符串与 printf 函数中的控制字符串基本相同(表 1-1 标以“仅限输出”的不能用),一般只使用格式转换说明符。若地址表列有多项,可用逗号或空格隔开,一般不出现普通字符。

(2)地址表列 地址表列是需要读入的所有变量的地址,而不是变量本身,这与 printf 函数不同。要特别注意:一般变量名前一定要加地址运算符 &,而字符串数组或字符串指针变量本身就是地址,就不用加 &。其顺序和个数也必须与格式控制字符串的说明对应,各个变量的