



华腾教育
HUA TENG EDUCATION

高等院校计算机基础教育规划教材

C语言程序设计

主编 陈广红 副主编 杨爽 程灵枝



WUHAN UNIVERSITY PRESS

武汉大学出版社

内 容 简 介

本书概念清楚,叙述翔实,采用循序渐进的内容安排,通俗易懂的讲解方法,且书中的实例通俗性、新颖性和实用性强,特别适合读者动手能力的培养。全书分 11 章阐述,主要内容包括:C 语言概述,数据类型、常量和变量,运算符和表达式,程序结构,数组,指针,函数,复杂数据类型,预处理命令和常用库函数,文件,实例演示等。书中完整的例题都在 Turbo C 2.0 和 Visual C++ 6.0 环境下调试通过,便于读者直接上机验证。

本书适合作为高等职业学校、高等专科学校、成人高校、民办高校的教材,也是自学人员学习的首选用书。

图书在版编目(CIP)数据

C 语言程序设计/陈广红主编. —武汉:武汉大学出版社,2006.12

(高等院校计算机基础教育规划教材)

ISBN 7-307-05358-6

I. C… II. 陈… 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 145145 号

责任编辑:齐翠红

出版发行:武汉大学出版社 (430072 武昌 珞珈山)

(电子邮件:wdp4@whu.edu.cn 网址:www.wdp.com.cn)

印刷:北京市昌平百善印刷厂

开本:787×1092 1/16 印张:23.75

字数:585 千字 印数:1~5000 册

版次:2006 年 12 月第 1 版 2006 年 12 月第 1 次印刷

ISBN 7-307-05358-6/TP·225 定价:33.00 元

版权所有,不得翻印;凡购买我社的图书,如有缺页、倒页、脱页等质量问题,请与当地图书销售部门联系调换。

高等院校计算机基础教育规划教材

编委会

主 任：赵云冲

副主任：涂兰敬 王颢媛 杨爽

编 委：（按姓氏笔画排序）

丁青云	王艳梅	王海梅	王 娟
王朝晖	王 辉	方美秀	白海波
邢宇飞	向 蕾	刘少华	刘 辙
孙怀东	孙贤龙	花庆毅	杨希鹏
杨 涛	李运生	李晓洁	李 琴
沈 丹	沈海涛	张国良	张雅辉
陈 曦	金守兵	赵天广	赵红芬
胡 俊	贾云娇	贾 文	钱 勇
徐 楠	殷洪菊	黄 磊	梁 辉
韩丽彦	程灵枝	程灵波	程宗米
廖中华	熊化武		

前言

PREFACE

C 语言是目前最常用的程序设计语言之一,它功能强大、简洁高效、易学易用,适合用来编写各种系统软件和应用软件,常被称为“系统编程语言”。C 语言是程序设计的入门语言,学好 C 语言,有利于更好地学习其他的程序设计语言。

本书的编写体例突出实用性和易学性,将基本的概念和方法的应用放在例题之中,结合实例来进行分析和讲解。同时,借助“说明”和“注意”等教学提示,帮助读者理解教学内容。另外,在每章的课后还附有练习题帮助读者更好地掌握有关的基本概念和方法的运用。

本书分 11 章阐述,主要包括:C 语言概述,数据类型、常量和变量,运算符和表达式,程序结构,数组,指针,函数,复杂数据类型,预处理命令和常用库函数,文件,实例演示等。书中完整的例题都在 Turbo C 2.0 和 Visual C++ 6.0 环境下调试通过,便于读者直接上机验证。

本书适合作为高等职业学校、高等专科学校、成人高校、民办高校的教材,也可作为 C 语言初、中级编程人员学习的首选用书。

本书突出计算机科学与技术学科的特点,强调理论与实践结合。用实例讲解的方法避免枯燥、空洞的理论堆砌,引导读者学会实际应用。由于编者水平有限,书中不足和疏漏之处难免,敬请广大读者不吝赐教。

为了配合本书的教学,华腾教育网站 <http://www.huatengedu.com> 提供了免费的电子教案、课后习题参考答案及程序代码,供读者和教学人员下载使用。

编 者

2006 年 11 月

致 读 者

尊敬的老师、同学们：

感谢您使用武汉大学出版社出版的高等院校计算机基础规划教材。为了今后给您提供更加优秀的教材，我们希望您能在百忙之中，抽出宝贵的时间填写下表并寄回我中心，我们将认真研读您的意见并即时提高我们的教材质量。如果您的来信填写认真，并且所提的意见或建议被我们采纳，我们将根据您的需要赠送文化用品一件。

来信请寄：北京 100080—067 信箱 华腾教育教学与研究中心 邮编：100080

或从华腾教育网 <http://www.huatengedu.com> 下载此表格，填写后发送 E-mail 至 ch@dongfangedu.com.cn。

您需要的文化用品：_____

我们真诚欢迎广大教师来电来信商讨教材的编写和出版事宜。

联系电话：010—62564988 联系人：赵老师

读者反馈表

个 人 信 息	姓名：	专业：	电话：
	我是： ☐ 老师 ☐ 学生		E-mail：
	校名：		系别：
	职务：		地址：
意 见 与 建 议	书名：		书号：
	我感觉本教材做如下修改将会更好：		
	我觉得本中心的网站如果这样建设会更好：		
	我的其他意见与建议：		



其 他 信 息	我系共有学生_____名。																							
	我系领导的姓名、职务及电话： 系主任： 副主任(分管教学)： 副主任(分管后勤)：																							
	我的任课老师的姓名、所任课程及电话： <table border="0"> <tr> <td>老师：</td> <td>任教课程：</td> <td>电话：</td> </tr> <tr> <td>老师：</td> <td>任教课程：</td> <td>电话：</td> </tr> <tr> <td>老师：</td> <td>任教课程：</td> <td>电话：</td> </tr> <tr> <td>老师：</td> <td>任教课程：</td> <td>电话：</td> </tr> <tr> <td>老师：</td> <td>任教课程：</td> <td>电话：</td> </tr> <tr> <td>老师：</td> <td>任教课程：</td> <td>电话：</td> </tr> <tr> <td>老师：</td> <td>任教课程：</td> <td>电话：</td> </tr> </table>			老师：	任教课程：	电话：	老师：	任教课程：	电话：	老师：	任教课程：	电话：	老师：	任教课程：	电话：	老师：	任教课程：	电话：	老师：	任教课程：	电话：	老师：	任教课程：	电话：
	老师：	任教课程：	电话：																					
	老师：	任教课程：	电话：																					
老师：	任教课程：	电话：																						
老师：	任教课程：	电话：																						
老师：	任教课程：	电话：																						
老师：	任教课程：	电话：																						
老师：	任教课程：	电话：																						
我系现有专业： ①_____②_____ ③_____④_____																								
我系开设的计算机课程有： ☐ 计算机应用基础 ☐ 计算机文化基础 ☐ 操作系统 ☐ 计算机组成原理 ☐ 计算机组装与维护 ☐ 计算机网络技术基础 ☐ 多媒体制作 ☐ 软件工程 ☐ 数据结构(C语言描述) ☐ 数据结构(C++描述) ☐ 微机原理与接口技术 ☐ 汇编语言 ☐ 单片机原理 ☐ C语言程序设计 ☐ C++程序设计 ☐ 数据库系统原理 ☐ SQL Server 2000/2005 数据库原理与应用 ☐ Visual Foxpro ☐ PowerBuilder ☐ Oracle ☐ Linux ☐ Delphi ☐ Access ☐ Visual C++ ☐ Visual Basic 程序设计 ☐ ASP ☐ Java 程序设计 ☐ Visual Basic. NET ☐ ASP. NET ☐ AutoCAD ☐ 3ds max ☐ Photoshop ☐ 计算机专业英语 ☐ 电子商务 ☐ 电子商务专业英语 ☐ 数字电路 ☐ 模拟电路 ☐ 电路分析基础 ☐ 人工智能 ☐ 微机控制技术 ☐ Internet 技术与应用 ☐ 计算机网络安全与管理 ☐ 局域网组建与组网工程 ☐ 网页设计与制作 ☐ Web 高级程序设计 ☐ 计算机网络原理及维护																								
我系开设的其他课程有： ①_____②_____ ③_____④_____ ⑤_____⑥_____ ⑦_____⑧_____																								

如本表格空间不够,请另附信纸!



目 录

CONTENTS

第 1 章 C 语言概述	1
1.1 程序设计语言的发展	1
1.2 C 语言的产生及特点	1
1.2.1 C 语言的产生	1
1.2.2 C 语言的特点	2
1.3 C 语言的字符集和标识符	3
1.3.1 字符集	3
1.3.2 标识符	4
1.4 C 程序的基本构成	4
1.4.1 C 程序的组成	4
1.4.2 C 程序的书写格式	5
1.4.3 C 程序的算法	6
1.5 C 程序的编辑、编译和连接	8
1.6 C 语言的集成开发环境	9
1.7 C 语言的应用	11
本章小结	11
习题 1	11
第 2 章 数据类型、常量和变量	13
2.1 数据类型	13
2.2 常量	14
2.2.1 整型常量	14
2.2.2 实型常量	15
2.2.3 字符常量	16
2.2.4 字符串常量	18
2.2.5 符号常量	18
2.2.6 枚举常量	20
2.3 变量	20

2.3.1 变量的定义和初始化	20
2.3.2 整型变量	21
2.3.3 实型变量	23
2.3.4 字符型变量	24
2.3.5 数据类型转换	25
本章小结	27
习题 2	27
第 3 章 运算符和表达式	28
3.1 概述	28
3.1.1 C 语言运算符	28
3.1.2 C 语言表达式	29
3.2 运算符及表达式简介	29
3.2.1 算术运算符及其表达式	29
3.2.2 关系运算符及其表达式	33
3.2.3 逻辑运算符及其表达式	34
3.2.4 位运算符及其表达式	38
3.2.5 赋值运算符及其表达式	41
3.2.6 逗号运算符及其表达式	44
3.2.7 条件运算符及其表达式	45
3.2.8 其他运算符	46
本章小结	48
习题 3	48
第 4 章 程序结构	50
4.1 程序结构概述	50
4.2 顺序结构	51
4.3 选择结构	53
4.3.1 选择结构简介概述	53
4.3.2 简单的 if 结构	54
4.3.3 if-else 语句	56
4.3.4 if-else-if 语句	58
4.3.5 if 语句的嵌套	61
4.3.6 switch 语句	68
4.4 循环结构	73
4.4.1 循环结构简介	73
4.4.2 while 循环语句	73
4.4.3 do-while 循环语句	79
4.4.4 for 循环语句	83
4.4.5 几种循环语句的比较	90

4.4.6 循环语句的嵌套	91
4.5 无条件转向语句	97
4.5.1 break 语句	97
4.5.2 continue 语句	100
4.5.3 goto 语句	102
本章小结	104
习题 4	104
第 5 章 数组	107
5.1 数组概述	107
5.2 一维数组的定义及应用	107
5.2.1 一维数组的定义	107
5.2.2 一维数组的引用	109
5.2.3 一维数组的初始化	110
5.2.4 一维数组的应用	112
5.3 二维数组的定义及应用	117
5.3.1 二维数组的定义	117
5.3.2 二维数组的引用	117
5.3.3 二维数组的初始化	119
5.3.4 二维数组的存储形式	120
5.3.5 二维数组的应用	122
5.4 多维数组的定义及应用	123
5.5 字符数组	125
5.5.1 字符数组的定义	125
5.5.2 字符数组的初始化	125
5.5.3 字符数组元素的引用	127
5.5.4 字符数组的输入输出	127
5.5.5 字符数组程序设计示例	129
本章小结	133
习题 5	133
第 6 章 指针	135
6.1 地址和指针的概念	135
6.2 指向变量的指针变量	137
6.2.1 指针变量的定义	137
6.2.2 指针变量的引用	138
6.2.3 指针变量的运算	141
6.3 指针与数组	143
6.3.1 指向数组元素的指针变量	144
6.3.2 通过指针引用数组元素	145

6.3.3 指向多维数组的指针	148
6.3.4 指针数组	152
6.4 指针与字符串	153
6.4.1 指向字符串的指针变量	153
6.4.2 字符串指针与字符数组的区别	154
本章小结	155
习题 6	155

第 7 章 函数	157
7.1 函数概述	157
7.2 函数的定义	159
7.3 函数的声明和调用	161
7.3.1 函数的声明	161
7.3.2 函数的调用	162
7.4 参数的传递	164
7.4.1 普通值传递	164
7.4.2 地址值传递	167
7.5 嵌套调用和递归调用	170
7.5.1 嵌套调用	170
7.5.2 递归调用	172
7.6 指针与函数	176
7.6.1 指针变量作函数参数	176
7.6.2 返回值为指针值的函数	178
7.6.3 函数的指针和指向函数的指针变量	179
7.7 数组作为函数参数	181
7.8 变量的作用域和存储类别	184
7.8.1 局部变量和全局变量	184
7.8.2 变量的存储类别	188
7.9 外部函数和内部函数	190
7.9.1 内部函数	190
7.9.2 外部函数	191
7.10 命令行参数	192
7.11 实例演示：“极限俱乐部会员信息系统”程序(一)	194
本章小结	207
习题 7	207

第 8 章 复杂数据类型	210
8.1 复杂数据类型概述	210
8.2 结构体类型和结构体变量	210
8.2.1 结构体类型的定义	210

8.2.2 结构体变量的定义	213
8.3 结构体变量的使用和初始化	214
8.3.1 结构体变量成员的引用	214
8.3.2 结构体变量的初始化	217
8.3.3 结构体变量的使用	218
8.4 结构体数据的输入/输出	219
8.5 结构体变量作函数参数	222
8.6 带有位段成员的结构体	224
8.7 链表	226
8.6.1 链表的概念	226
8.6.2 链表的基本操作	227
8.8 共用体类型	233
8.8.1 共用体类型的定义	234
8.8.2 共用体变量的定义	235
8.8.3 共用体变量的引用	236
8.9 枚举类型	238
8.10 实例演示:“极限俱乐部会员信息系统”程序(二)	240
本章小结	253
习题 8	253
第 9 章 预处理命令和常用库函数	256
9.1 预处理命令	256
9.1.1 文件包含命令	256
9.1.2 宏定义	259
9.1.3 条件编译	263
9.2 常用输入/输出函数	265
9.3 常用字符串函数	271
9.4 动态内存分配函数	276
本章小结	278
习题 9	278
第 10 章 文件	280
10.1 文件的基本概念概述	280
10.1.1 文件的概念及分类	280
10.1.2 文件缓冲区	281
10.1.3 设备文件	282
10.2 文件类型及其指针	282
10.3 文件的打开和关闭	283
10.3.1 文件打开函数	283
10.3.2 文件关闭函数	286

10.4 文件的输入和输出	287
10.4.1 字符读/写函数	287
10.4.2 字符串读/写函数	289
10.4.3 格式化读/写函数	291
10.4.4 块读/写函数	293
10.5 文件的定位函数	297
10.5.1 rewind()函数	297
10.5.2 fseek()函数	298
10.6 出错检测函数	299
10.6.1 ftell()函数	299
10.6.2 ferror()函数和 Clearerr()函数	300
本章小结	302
习题 10	302
 第 11 章 实例演示	 304
11.1 “极限俱乐部会员信息系统”程序(三)	304
11.2 “极限俱乐部会员信息系统”程序(四)	319
11.3 “教学成绩管理系统”程序	338
附录 I C 语言中的关键字	357
附录 II 运算符和结合性	358
附录 III C 语言库函数	359

第 1 章 C 语言概述

本章要点

- ◆ 程序设计语言的发展
- ◆ C 语言程序的格式和组成
- ◆ 编译、连接的目的
- ◆ Turbo C 环境的基本用法

1.1 程序设计语言的发展

在日常的工作、生产、学习、生活中,人们利用计算机解决各种问题,特别是在解决数值计算问题时,一般而言,总是要编制程序。程序表达了人们解决问题的某种思路,也反映了需要计算机进行的一系列操作。而程序设计语言就是人们用于编制程序的语言,它也是人与计算机进行交互的工具。为了使计算机能够直接理解并执行人所需要它进行的操作,计算机科学工作者为计算机配制了一套它能识别的语言——机器语言(也称机器指令集)。每台计算机出厂时,厂家都为它配备一套机器语言,不同的计算机,其机器语言通常是不同的。有了机器语言,人们就可以用机器语言编写程序,然后输入计算机,计算机就通过运行程序来体现人们的意图,即计算或处理相应的问题。

由于机器语言是面向具体机器的,所以编写的程序缺乏通用性,编写过程繁琐复杂,易出错,错了又不易查找和修改,而且编出的程序可读性极差。另外,繁杂的机器代码程序很难记忆,未受过专门训练的人不易掌握,这严重阻碍了计算机的应用和发展。于是,人们又在机器语言的基础上研制出汇编语言。

但汇编语言实质上是机器语言的符号化形式,仍属于面向机器的低级语言,其程序的通用性和可读性较差,特别是要想编制解决生产、科学计算等领域实际问题所需要的程序时,必须要具备计算机硬件设备、机器指令和寄存器方面的知识,并能深入了解所要解决问题的实质,然后用相应的机器指令或符号化的机器指令来实现。可想而知,这样的程序设计工作量非常大,而生产率非常低下,因此程序设计语言仍需改进。经过计算机科学工作者的不懈努力,从 20 世纪 50 年代末开始,研制了一批中、高级语言。其中,20 世纪 50 年代末和 60 年代初研制出的高级程序设计语言 BASIC、FORTRAN、AL_GOL60 和 COBOL 被称为计算机语言发展史上的四个里程碑。

1.2 C 语言的产生及特点

1.2.1 C 语言的产生

20 世纪 60 年代,随着计算机科学的迅速发展,高级程序设计语言 FORTRAN、AL_GOL60 等得到了广泛的应用。然而,还缺少一种可以用来开发操作系统和编译程序等系统程序的高级语言,人们只能使用机器语言或汇编语言来编写这些程序。但机器语言和汇编语言存在着不可移植、可读性差、研制软件效率不高缺点,给编程带来很多不便。于是,在

20 世纪 70 年代初,C 语言应运而生了。

1978 年,由美国电话电报公司(AT&T)贝尔实验室正式发表了 C 语言。同时由 B. W. Kernighan 和 D. M. Ritchie 合著了著名的《C 语言程序设计》一书。后来由美国国家标准协会(American National Standards Institute)在此基础上制定了一个 C 语言标准,于 1983 年发表,通常称之为 ANSI C。

目前,从微型计算机到大型计算机都支持 C 编译程序。C 编译程序不仅能在 UNIX 操作系统下运行,而且能在 DOS、Windows 和 Linux 操作系统下运行。由于 C 语言本身具有的优越性,它已经成为在各种计算机上,从系统软件设计到工程应用程序开发都能使用的一种高级程序设计语言。

1.2.2 C 语言的特点

C 语言之所以广泛流行且深受用户青睐,是因为它以其简洁、紧凑、灵活、表达能力强、产生的目标代码质量高和可移植性好等为基本特色,归纳起来有以下主要特点:

1. C 语言具有结构化的控制语句

C 语言提供了结构化程序所必需的基本控制语句,如条件判断语句和循环语句等,实现了对逻辑流的有效控制。C 语言的源程序由函数组成,每个函数各自独立,把函数作为模块化设计的基本单位。C 语言的源文件可以分割成多个源程序,进行单独编译后可连接生成可执行文件,为开发大型软件提供了极大的方便。C 语言提供了多种存储属性,通过对数据的存储域控制提高了程序的可靠性。

2. C 语言具有丰富的数据结构类型

C 语言除提供整型、实型、字符型等基本数据类型外,还提供了用基本数据类型构造出的各种复杂的数据结构,如数组、结构、联合等。C 语言还提供了与地址密切相关的指针类型。此外,用户还可以根据需要自定义数据类型。

3. C 语言具有丰富的运算符

C 语言提供了多达 34 种运算符,运算能力十分丰富,它把括号、逗号、赋值等都作为运算符来处理,从而使 C 的运算类型极为丰富,可以实现其他高级语言难以实现的运算。丰富的数据类型与丰富的运算符相结合,使 C 语言的表达力更具灵活性,同时也提高了执行效率。

4. C 语言简洁、紧凑,使用方便、灵活

程序书写自由,共有 9 种控制语句。ANSI C 规定 C 语言共有 32 个关键字。关键字就是已被编程语言本身使用的标识符,不能作变量名、函数名等其他用途。

C 语言的关键字详见附录 I。

注意:在 C 语言中,关键字都是小写的。

不同的实现对 C 语言的关键字有不同的扩充,Turbo C 扩充了 11 个关键字,如表 1-1 所示。

表 1-1 Turbo C 扩充的关键字

asm	_cs	_ds	_es	_ss	cdecl
far	huge	interrupt	near	pascal	

5. 中级语言

我们通常称面向问题的语言为高级语言,而面向机器的语言为低级语言,C 语言既具有高级语言的功能,又具有低级语言的许多功能。C 语言允许直接访问物理地址,能进行位(bit)操作,实现汇编语言的大部分功能,直接对硬件进行操作。由于 C 语言的这种双重性,使它既是成功的系统描述语言,又是通用的程序设计语言,所以有人称它为中级语言。

6. 与汇编语言相比,可移植性好

可移植性是指程序可以从一个环境不加或稍加改动就可移到另一个完全不同的环境上运行。对汇编语言而言,由于它只面向特定的机器,所以根本不可移植。而一些高级语言(比如 FORTRAN)其编译程序也不可移植,而只能根据国际标准重新实现。但 C 语言在许多机器上的实现是通过将 C 编译程序移植得到的。

7. 功能强大

高级语言是否适用于编写系统软件,除了语言表达能力之外,还有一个很大的因素是该语言的代码质量。一般说来,语言越低级其代码质量就越高,但编程的工作量也越大。由于 C 语言具有低级语言的一些功能,所以,生成目标代码质量高,程序执行效率高。现在许多系统软件都用 C 语言来描述,从而大大提高了编程效率。

1.3 C 语言的字符集和标识符

1.3.1 字符集

C 程序允许出现的所有基本字符的组合称为 C 语言的字符集,主要分为下列四类。

1. 大小写英文字母

A,B,C,...,Z,a,b,c,...,z。

2. 数字

0,1,2,3,4,5,6,7,8,9。

3. 键盘符号

键盘符号详见表 1-2。

表 1-2 键盘符号

符号	含义	符号	含义	符号	含义
~	波浪号	)	右圆括号	:	冒号
`	重音号	_	下划线	;	分号
!	叹号	-	减号	"	双引号
@	a 圈号	+	加号	'	单引号
#	井号	=	等号	<	小于号
\$	美元号		或符号	>	大于号
%	百分号	\	反斜杠	,	逗号
^	异或号	{	左花括号	.	小数点

续表

符号	含义	符号	含义	符号	含义
&	与符号	}	右方括号	?	问号
*	星号	[	左方括号	/	(正)斜杠
(	左圆括号	]	右方括号		空格符号

注意:空格符、制表符、换行符等统称为空白符。空白符只在字符常量和字符串常量中起作用。在其他地方出现时,只起间隔作用,编译程序对它们忽略不计。因此在程序中使用空白符与否,对程序的编译不发生影响,但在程序中适当的地方使用空白符将增加程序的清晰性和可读性。

1.3.2 标识符

C 语言程序中出现的任何对象一般都有一个“名字”,这些对象有函数、变量、符号常量、数组名、数据类型、宏以及存储属性等。用标识符来给对象取“名字”,例如某变量名为 data,某个函数名为 allod,某常量名为 MAXLENGTH 等。

在选择作为“名字”使用的标识符时,要注意以下几点:

(1)标识符只能是字母(A~Z,a~z)、数字(0~9)、下划线(_)组成的字符串,第一个字符必须是字母或下划线。

(2)下划线“_”也起一个字母的作用,它用来帮助分隔长描述名的各部分,例如 interest-todata 可以写成 interest_to_data。

(3)大、小写字母含意不同,如 VELOCITY、velocity 和 Velocity 是三个完全不同的标识符。

(4)标准 C 不限制标识符的长度,但一般版本的 C 语言编译系统规定只有前 8 个字符有意义。如标识符 honorific 和 honorificab,编译系统会把它们看作是同一个标识符,即认为是 honorifi。

(5)根据 C 语言的习惯规定,变量名、函数名等用小写字母表示,而符号常量全用大写字母表示,函数名和外部变量由 6 个字符组成,系统变量由下划线“_”起头构成。

(6)根据一般程序设计的经验,标识符的选择原则应是“常用取简”、“专用取繁”,一般能表示其含意即可,不宜太长,通常在 6 个字符之间均能适应各种系统。

(7)C 语言源程序的文件名选择不属于 C 语言,而属于操作系统。大多数 C 语言编译系统均要求所有 C 语言源代码文件必须以后缀“.c”结束,也就是说 C 语言源程序的文件属性为“.c”。

1.4 C 程序的基本构成

1.4.1 C 程序的组成

一个 C 语言程序可由下面不同的部分组合而成:

- (1)文件包含部分;
- (2)预处理部分;
- (3)变量说明部分;
- (4)函数原型声明部分;

(5)主函数部分;

(6)函数定义部分。

关于程序的结构说明:

(1)并不是所有的C程序都必须包含上面的6个部分,一个最简单的C程序可以只包含文件包含部分和主函数部分两部分。

(2)每一个C程序文件的后缀为“.c”,并且每一个C源程序都必须有且仅有一个主函数,主函数的组成形式为:

```
main()
```

```
{
```

```
    变量说明部分;
```

```
    程序语句部分;
```

```
}
```

注意:每一条C语言的语句都由分号结束。

下面以一个简单的C程序为例来理解C语言程序的组成。

【例1-1】输入三个整数,输出其中的最大数。

```
#include<stdio.h>
```

```
int max(int a,int b,int c)/*定义max函数*/
```

```
{
```

```
    int big;
```

```
    big=a;
```

```
    if(b>big)big=b;
```

```
    if(c>big)big=c;
```

```
    return big;
```

```
}
```

```
void main()/*定义main函数*/
```

```
{
```

```
    int x,y,z,s;
```

```
    printf("输入三个整数:\n");
```

```
    scanf("%d,%d,%d",&x,&y,&z);
```

```
    s=max(x,y,z);/*调用max函数,返回值为三个数中的最大数*/
```

```
    printf("最大的数为:%d\n",s);
```

```
}
```

程序运行结果如图1-1所示:

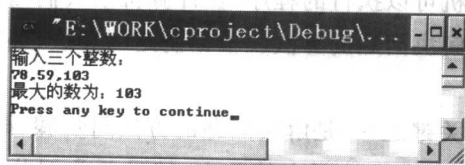


图1-1 【例1-1】程序运行结果