



根据国家教育部最新颁布的
中等职业学校计算机专业教学大纲编写

Windows

C语言 程序设计

全国中等职业学校通用教材 / 计算机专业

新需求 新标准 新目标

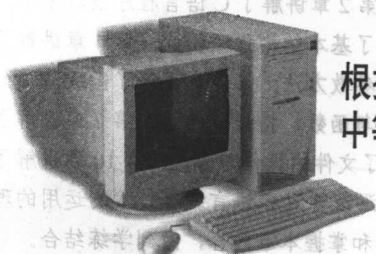
Turbo C

曾令明 李 飞 / 编著



电子科技大学出版社
Dianzikejidaixuechubanshe

要 容 内



根据国家教育部最新颁布的
中等职业学校计算机专业教学大纲编写

C语言程序设计

全国中等职业学校通用教材 / 计算机专业

曾令明 李 飞 / 编著

出 版 社：电子科技大学出版社
 责任编辑：王 斌
 封面设计：王 斌
 印 刷：四川成都印刷厂
 开 本：787mm×1092mm 1/16
 印 张：12
 字 数：380千字
 定 价：12.00元
 2006年7月第1版
 2006年7月第1次印刷
 号：ISBN 7-81114-100-2/TP·46



电子科技大学出版社
Dianzikejidaxuechubanshe

内 容 提 要

全书内容包括:第1章讲解了程序设计的基本概念;第2章讲解了C语言程序设计基础,包括数据类型、变量与常量、运算符和表达式;第3章讲解了基本结构程序设计;第4章讲解了数组,介绍一维数组、二维数组、字符数组和数组作为函数参数方法;第5章讲解了函数与预处理命令;第6章讲解了指针,讲解地址和指针的关系、指针与函数、指针与数组;第7章讲解了结构体与共用体;第8章讲解了位运算与位段;第9章讲解了文件的操作与管理;第10章讲解了课程设计,将C语言应用技术通过几个综合设计来表现,加深读者对C语言编程技术的运用的理解和掌握。每章后附有习题及上机实验,使读者能加深理解和掌握本章内容,做到学练结合。

本书可作为中等职业学校、中等专科学校、技工学校、高等职业学校(三年中专+两年大专)、成人高校以及民办高校、电脑学校计算机及应用相关专业的通用教材,还可作为计算机爱好者的自学参考书。

图书在版编目(CIP)数据

C语言程序设计 / 曾令明, 李飞编著. —成都: 电子科技大学出版社, 2006.7

(中等职业学校系列教程)

ISBN 7-81114-190-6

I. C … II. ①曾… ②李… III. C语言—程序设计—专业学校—教材 IV. TP312

中国版本图书馆CIP数据核字(2006)第074880号

C 语言程序设计

曾令明 李 飞 编著

出 版: 电子科技大学出版社 (成都建设北路二段四号 邮编: 610054)

责任编辑: 黄礼玲

发 行: 新华书店经销

印 刷: 四川省南方印务有限公司

开 本: 787mm×1092mm 1/16 印张: 15 字数: 380千字

版 次: 2006年7月第一版

印 次: 2006年7月第一次印刷

书 号: ISBN 7-81114-190-6/TP·46

定 价: 17.80元

■ 版权所有 侵权必究 ■

◆ 读者服务电话: (028) 83243088

◆ 本书如有缺页、破损、装订错误, 请寄回印刷厂调换。

导 言

众所周知,计算机是 20 世纪最伟大的发明,计算机技术的应用,极大地推动了人类社会的飞跃发展。

如今,遍及世界的计算机及其互联网,把全世界不同肤色、不同语言的人们紧密地连接了起来,使偌大的地球世界变成了一个“村落”,人们亲昵地称之为“地球村”。无论是在大洋彼岸,还是远在扶桑之国,亲友们通过互连网络和多媒体技术,顷刻间就能面对面地问候和交流。计算机这个人类的宠儿,正以无穷的魅力引领着人们向着更加美好的未来飞奔。

中国,世界的泱泱大国,不仅有秀丽的山川,更有丰富的资源,而且人口众多。改革开放以来,中国经济的高速增长为世界瞩目,但是,作为当今高新技术的核心,计算机技术在中国的发展仍在扬鞭奋蹄。

为了与世界的经济、科技接轨,大力推进计算机技术的普及、推广与应用,已成为神州大地当务之急。进入 21 世纪以来,党和政府在所倡导的“科教兴国”中,采取了许多积极的举措,大力培养 21 世纪新型职业技术人才就是其中重要的一项。

我们为贯彻执行《中共中央国务院关于深化教育改革全面推进素质教育的决定》精神,积极推进中央的《面向 21 世纪教育振兴行动计划》。针对中等职业教育的特点,提出在中等职业教育中,不仅应有全新的教学理念,更应有符合要求的“全新教材”。

我们对“全新教材”的理解是:首先要全面贯彻素质教育,要从社会的实际需要出发,强化对学生创新精神和实践能力的培养,大胆在教材中融入最先进的教材理念,并通过教材诱发学生积极的学习热情。

在认真对中等职业教育进行了分析研究后,严格按照国家《中等职业教育计算机专业教学大纲》的要求,邀请了在一线从事计算机教学与科研的著名计算机教育专家,并与之鼎力合作全力打造了适应于当前中职计算机教学的一套全新教材。

首期推出的有:

- | | |
|---------------------------|---------------|
| •《计算机应用基础》 | •《计算机网络应用基础》 |
| •《C 语言程序设计》 | •《局域网组网与维护》 |
| •《Internet 应用基础》 | •《计算机组装与维护》 |
| •《Photoshop 7.0 图形图像处理基础》 | •《网页设计与制作》 |
| •《Visual FoxPro 程序设计》 | •《常用工具软件应用基础》 |

本套教材经过专家们审定认为是目前同类教材中较具活力的范本。专家们热评:“该套教材取材新颖,能紧跟计算机技术发展的前沿;在重点讲解计算机专业知识基础的同时,又充分凸显了计算机系统的特点,尤其对计算机的操作与应用,注入了全新的教材理念;教材

中以新型计算机的高度智能化和图解、图示为主线,结合大量的应用实例,辅以来源于实践的思考练习题,使该套教材在整体上十分生动活泼。”

总体来说,本套教材具有如下的特色和亮点:

一、全新取材。计算机技术是众多实用性学科中,升级换代最快的一个门类。本套教材不仅将计算机硬件中的双核技术,第三代互联网技术和多种前端接口技术有机地融入教材中,而对于软件中的最新操作系统,办公软件的高端版本都一一引入,作为教材的核心内容。

二、基础第一。任何教材无疑都应把该学科的知识基础作为平台,只有搭建起稳固的“基础知识平台”,并使之尽可能地完美,才能使读者在寓教于学中轻松地吸收该教材的精髓。在本套教材中,作者按照计算机技术的特点,层次清晰地阐述了知识中的“重点”、“难点”和“盲点”,使读者在学习中能从容地抓重点、突难点、解盲点。可以说这一点是本套教材极为重要的亮点。

三、结构清晰。一本教材或图书的结构,就像人体的脉络,人体的脉络如果紊乱,就意味着身体有了病变。同样,对于一本教材或图书,如果其结构散乱不清,庞杂无序,读者将产生阅读困难,自然也就无从掌握该教材或图书中的知识体系。本套教材脉络清楚,书中的章、节、目组合有序,让“教”者与“学”者都趣味盎然。

四、案例生动。中等职业教育的要求是“技能”和“实践”的高度统一,因此,中等职业教育教材无疑应把实用性放在首位。本套教材融入了许多极富实用性的案例,并围绕案例阐述和剖析知识原理,使读者能够得到学以致用效果。

五、语言流畅。出版教材或图书的目的,无非就是要让读者真情地去阅读。因此,教材或图书的可读性往往成为判别其优劣的重要指标。读者在阅读时,第一感觉就是跃然纸上的文字,书中优美的文字,丰富的语汇,强烈的语言节奏感将会使读者手不释卷、流连忘返,并且迸发出巨大的阅读情趣。本套书作者由于长期从事计算机基础教学,不仅具有丰富的实践经验,更在多年来的计算机类教材或图书编撰中总结出一套人文化的写作方法,使本套书在阅读上极富文学性。

此外,本套书版式别致,封面、装帧、印刷都极为考究,加之图书定价实惠,可以预测本套书将成为中等职业教育计算机类图书中的一大亮点。

当然,出版者也愿意去接受激烈市场竞争的洗礼,为本套书再版时提供更为完美的修订依据。

中等职业教育教材编写委员会

目 录

第 1 章 概述	1
1.1 程序设计的基本概念	1
1.1.1 程序	1
1.1.2 程序设计	1
1.1.3 程序设计语言	1
1.2 C 语言简介	3
1.2.1 C 语言的产生及发展	3
1.2.2 C 语言的特点	3
1.2.3 C 程序组成及书写格式	4
1.2.4 C 程序上机过程	5
本章小结	6
习题	6
上机实验	7
第 2 章 数据类型、运算符与表达式	8
2.1 关键字与标志符	8
2.1.1 关键字	8
2.1.2 标志符	8
2.2 数据类型	9
2.2.1 C 的数据类型	9
2.2.2 C 的基本数据类型	9
2.3 常量	11
2.3.1 整型常量	11
2.3.2 实型常量	12
2.3.3 字符型常量	13
2.3.4 符号常量	15
2.4 变量	16
2.4.1 变量的定义	16
2.4.2 变量的基本操作	16
2.4.3 整型变量	17
2.4.4 实型变量	17

2.4.5 字符变量	17
2.5 运算符与表达式	19
2.5.1 运算符概述	19
2.5.2 算术运算符与算术表达式	19
2.5.3 赋值运算符与赋值表达式	20
2.5.4 表达式中的类型转换	21
2.5.5 逗号运算符与逗号表达式	23
本章小结	23
习题	23
上机实验	26
第 3 章 基本结构程序设计	27
3.1 算法	27
3.1.1 算法概念	27
3.1.2 算法的描述	28
3.2 顺序结构	32
3.2.1 语句	32
3.2.2 数据的输入输出	34
3.3 选择结构	40
3.3.1 关系运算与逻辑运算	41
3.3.2 if 语句	43
3.3.3 switch 语句	47
3.4 循环结构	49
3.4.1 while 语句	49
3.4.2 do-while 语句	50
3.4.3 for 语句	52
3.4.4 循环的嵌套	54
3.4.5 break 语句与 continue 语句	55
本章小结	58
习题	58
上机实验	64
第 4 章 数组	65
4.1 数组概述	65
4.1.1 数组的引入	65
4.1.2 数组的概念	65
4.2 一维数组	66
4.2.1 一维数组的定义	66

4.2.2 一维数组的引用	67
4.2.3 一维数组的初始化	68
4.2.4 一维数组应用举例	69
4.3 二维数组	70
4.3.1 二维数组的定义	70
4.3.2 二维数组的引用	71
4.3.3 二维数组的初始化	72
4.3.4 二维数组应用举例	72
4.4 字符数组	74
4.4.1 字符数组的定义	74
4.4.2 字符数组的初始化	74
4.4.3 字符串	75
4.4.4 字符数组的引用	75
4.4.5 字符数组的输入和输出	75
4.4.6 字符串处理函数	77
4.4.7 字符数组应用举例	81
本章小结	81
习题	82
上机实验	85
第 5 章 函数与预处理命令	86
5.1 模块化程序设计与函数	86
5.1.1 模块化程序设计的概念	86
5.1.2 函数分类	86
5.2 函数的定义与调用	87
5.2.1 函数定义	87
5.2.2 函数调用	89
5.3 函数的嵌套与递归	95
5.3.1 函数的嵌套调用	95
5.3.2 函数的递归	96
5.4 数组作为函数参数	98
5.4.1 数组元素作函数实参	98
5.4.2 数组名作函数参数	99
5.5 变量的作用域	100
5.5.1 局部变量	100
5.5.2 全局变量	102
5.6 静态存储变量与动态存储变量	104

5.6.1 静态存储变量	104
5.6.2 动态存储变量	105
5.7 预处理命令	108
5.7.1 宏定义	108
5.7.2 文件包含	115
本章小结	116
习题	116
上机实验	120
第 6 章 指针	122
6.1 地址和指针的概念	122
6.2 指针变量的定义与使用	123
6.2.1 指针变量的定义	123
6.2.2 指针变量的使用	123
6.3 指针与函数	126
6.3.1 指针作函数的参数	126
6.3.2 返回值为指针的函数	127
6.3.3 指向函数的指针	128
6.4 指针与数组	130
6.4.1 指向一维数组的指针	130
6.4.2 指向多维数组的指针	138
6.4.3 指针与字符串	142
6.4.4 指针数组和指向指针的指针	145
本章小结	150
习题	150
上机实验	153
第 7 章 结构体与共用体	154
7.1 结构体概述	154
7.1.1 结构体的引入	154
7.1.2 结构体类型的定义	154
7.2 结构体变量	156
7.2.1 结构体变量的定义	156
7.2.2 结构体变量的引用	158
7.2.3 结构体变量的初始化	160
7.2.4 结构体变量作函数参数	160
7.3 结构体数组	162
7.3.1 结构体数组的定义	162

7.3.2 结构体数组的初始化	162
7.3.3 结构体数组元素的引用	162
7.3.4 结构体数组应用举例	163
7.4 结构体与指针	164
7.4.1 指向结构体变量的指针	164
7.4.2 链表	166
7.4.3 指向结构体数组的指针	172
7.4.4 用指向结构体的指针作函数参数	173
7.5 共用体	174
7.5.1 共用体类型的定义	174
7.5.2 共用体变量的定义	175
7.5.3 共用体变量的引用	176
7.6 枚举类型	176
7.7 用 typedef 定义类型	178
本章小结	179
习题	179
上机实验	183
第 8 章 位运算与位段	184
8.1 位运算	184
8.1.1 按位与运算	184
8.1.2 按位或运算	185
8.1.3 按位异或运算	185
8.1.4 求反运算	185
8.1.5 左移运算	185
8.1.6 右移运算	185
8.1.7 位运算赋值运算符	186
8.2 位段	186
本章小结	187
习题	187
上机实验	188
第 9 章 文件	189
9.1 文件概述	189
9.1.1 文件的定义	189
9.1.2 文件的分类	189
9.2 文件指针	191
9.3 文件的打开与关闭	191

9.3.1 文件打开函数 (fopen)	191
9.3.2 文件关闭函数 (fclose)	193
9.4 文件的读写	193
9.4.1 字符读写函数 fgetc 和 fputc	193
9.4.2 字符串读写函数 fgets 和 fputs	197
9.4.3 数据块读写函数 fread 和 fwrite	198
9.4.4 格式化读写函数 fscanf 和 fprintf	200
9.5 文件的定位	202
9.5.1 fseek 函数	202
9.5.2 rewind 函数	203
9.5.3 ftell 函数	204
9.6 文件检测函数	205
9.6.1 feof 函数	205
9.6.2 ferror 函数	205
9.6.3 clearerr 函数	205
本章小结	205
习题	205
上机实验	207
第 10 章 课程设计	208
10.1 课程设计的步骤	208
10.2 电子通信录	208
10.2.1 设计目的	208
10.2.2 设计要求	209
10.2.3 算法分析	210
10.3 学生成绩管理系统	211
10.3.1 设计目的	211
10.3.2 设计要求	211
10.3.3 算法分析	214
附录	217
附录 A 常用字符与 ASCII 代码对照表	217
附录 B 运算符的优先级和结合性	218
习题参考答案和上机实验参考源程序	219

第1章 概述

学习目标:

- 了解程序设计的基本概念
- 了解程序设计语言的分类
- 熟悉 C 语言程序组成

随着科学技术的迅猛发展,计算机技术日新月异,计算机程序设计语言也层出不穷。那么,什么是程序设计语言?应该学哪一种程序设计语言?如何进行程序设计?这些都是程序设计初学者首先遇到的问题。

不论是什么样的计算机程序设计语言,其程序设计的基本方法是相同的。本书作为程序设计的入门教材,将以 C 语言程序设计为主线,介绍程序设计的基本概念和基本方法,讲述 C 语言的语法规则和实用的 C 程序设计技术。作为全书的开篇,本章就程序设计的基本知识作概括性讨论,同时还介绍了 C 语言的总体特征。需要说明的是,有些概念和方法要随着后续各章的深入学习才会有深刻的理解,读者不必在一时不懂的问题上停滞不前。

1.1 程序设计的基本概念

1.1.1 程序

从计算机诞生起,就有了程序。计算机通过执行程序完成其工作,如计算、控制、文字处理、图形处理、网络通信等。所谓程序,就是按照具体要求产生的适合计算机处理的指令序列。我国颁布的《计算机软件保护条例》对程序的概念给出了更为精确的描述:“程序是指为了得到某种结果而可以由计算机等具有信息处理能力的装置执行的代码化指令序列,或者可被自动地转化成代码化指令序列的符号化序列,或者符号化语句序列。”这就是说,程序要有目的性和可执行性。

1.1.2 程序设计

程序设计是指完成程序的过程,通常包含以下几个步骤:

- (1) 提出问题,明确程序要达到的要求。
- (2) 确定数据结构和所采用的算法。
- (3) 编制程序。
- (4) 调试程序。
- (5) 整理并写出文档。

1.1.3 程序设计语言

语言是用于传达信息的表示方法、约定和规则的集合,是人们交流信息的工具和媒介。

而程序设计语言是实现程序设计，以便于人与计算机进行信息交流的工具和媒介。

程序设计语言一般分为低级语言和高级语言两大类，其中低级语言包括机器语言和汇编语言。

1. 机器语言

机器语言是以二进制形式表示的机器指令的集合，是计算机系统不需要翻译就可以直接识别和执行的程序设计语言。用机器语言编写程序，程序人员必须熟悉机器指令的二进制代码，记忆指令代码能完成的操作，还应指出这一操作对象的位置，即记忆指令的操作码和地址码。由于机器语言程序是直接在计算机上执行，所以效率比较高，能充分发挥计算机的高速计算的能力。在计算机发展的初期，人们都是使用机器语言直接编制程序。但机器语言不易记忆和理解且缺乏直观性，所以用机器语言编制程序的难度很大。

2. 汇编语言

汇编语言用助记符来表示指令的操作码和操作对象，用标号和符号来代替地址、常量和变量。例如，“ADD AX,BX;”代表两个寄存器数相加的功能。用汇编语言编写的程序，较直观、易理解。但计算机却不能识别和直接运行汇编语言程序，必须由汇编程序将汇编语言程序“翻译”成机器语言程序后才能识别并运行。此外，不同 CPU 的指令系统对应的汇编语言不同。如：单板机、单片机、微处理器等，随机器型号、类型的不同，各自的汇编语言也不同。这就是说，用汇编语言编写的程序缺乏通用性。

3. 高级语言

为了方便使用，程序设计语言朝着接近人们熟悉、习惯的自然语言和数学语言描述的高级化方向发展，形成了各种各样、丰富多彩的程序设计高级语言。

高级语言是一类面向问题的程序设计语言，且独立于计算机的硬件，其表达方式接近于被描述的问题，易于人们的理解和掌握。用高级语言编写程序，可简化程序编制和测试，其通用性和可移植性好。目前，计算机高级语言虽然很多，据统计已经有好几百种，但广泛应用的却仅有十几种，他们有各自的特点和使用范围。如：BASIC 语言，是一类普及性的会话语言；FORTRAN 语言，多用于科学及工程计算；COBOL 语言，多用于商业事务处理和金融业；PASCAL 语言，它有利于结构化程序设计；C 语言，常用于软件的开发；PROLOG 语言，多用于人工智能；而当前流行的，还有面向对象的程序设计语言 C++ 和 Java 等。

用高级语言编写的程序称为源程序，源程序不能被计算机直接执行，必须将他们“翻译”成具体机器的机器语言程序才能执行。C 语言源程序的运行过程如图 1-1 所示。

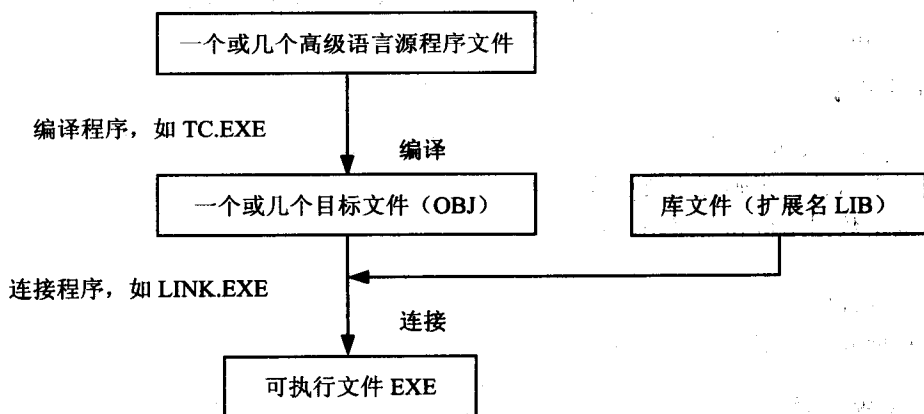


图 1-1 C 语言源程序的运行过程

高级语言“翻译”成机器语言有两种方式：一种是“解释”，一种是“编译”。C语言属于编译型语言。编译的原理就是由编译程序把源程序编译、连接成可执行文件，然后由机器直接执行。

1.2 C语言简介

1.2.1 C语言的产生及发展

C语言是由美国人D.M.Ritchie于1972年在B语言的基础上开发出来的，主要用于编写UNIX操作系统。后来C语言进行了多次改进，1977年出现了可移植的C语言编译程序，使得用C语言编写的UNIX系统可在各种计算机上使用。随着UNIX的广泛应用，C语言得到了普及和推广，并最终独立于UNIX而成为世界上应用最广泛的计算机语言之一。

在C语言的推广过程中，出现了多种版本。这些版本虽然是类似的，但通常是不兼容的。为了解决这些版本的兼容性问题，20世纪80年代，美国国家标准化协会(ANSI)制订了C语言标准，称为ANSI C。目前流行的各种版本的C语言都是以它为基础的，如Turbo C, Microsoft C, Quick C等。这些版本的C语言虽然基本部分相同，但也有些差异，读者可以参考相关手册。本书的叙述是以ANSI C为基础，所列程序是在Turbo C或VC++ 6.0上调试通过的。

1.2.2 C语言的特点

C语言发展迅速，成为最受欢迎的语言之一，主要是因为它具有强大的功能。许多著名的系统软件，如DBASE IV、UNIX都是用C语言编写的。归纳起来C语言具有下列特点：

1. C语言是高级语言中的低级语言

C语言把高级语言的基本结构和语句与汇编语言的功能结合了起来。它可以像汇编语言一样对位、字节和地址进行操作。C语言的这种特性使它成为程序设计语言中的全能冠军，既可以用来设计应用软件，也可以用来设计系统软件。

2. C语言是结构化语言

结构化语言的显著特征是代码和数据的分离，能够把执行某个特殊任务的代码和数据从程序中分离出来，使整个程序由一个个模块组成。用结构化语言编写的程序层次清晰，便于使用、维护以及调试。

用C语言编写的程序由函数组成，每个函数完成一个特定的功能，由函数作为模块来实现程序的模块化。同时C语言具有多种循环、条件语句控制程序流向，从而使程序完全结构化。

3. C语言功能齐全

C语言运算符丰富，还可以把括号、赋值、强制类型转换等作为运算符处理。C语言具有各种各样的数据类型，并引入了指针概念，具有构造新的数据类型的机制，使程序效率更高。另外C语言也具有强大的图形功能，支持多种显示器和驱动器。

4. C语言简洁、灵活

C语言仅有32个具有特定意义的关键字，9种控制语句，程序书写上去掉了一切不必要的成分，因此语句很简洁。与其他语言相比，编译程序对语法检查比较宽松，程序员使用起来也就更灵活，写出的程序效率也更高。

1.2.3 C 程序组成及书写格式

我们先分析下面两个简单的 C 语言源程序（简称 C 程序），然后总结出 C 程序的组成及书写格式。

例 1-1 在屏幕显示一行文字：“This is a simple C program.”。

完成该任务的 C 程序如下：

```
/*例 1-1*/  
#include <stdio.h>  
main()  
{  
    printf("This is a simple C program.\n");  
}
```

这个程序很简单，但它是一个完整的 C 程序。第 1 行是注释，说明这是例 1-1，它不参与程序的编译和执行，只是便于阅读。第 2 行是 C 预处理程序的一条指令，这条指令通知预处理程序把标准输入输出头文件（stdio.h）中的内容包括到程序中来（关于预处理和头文件在后续章节有介绍，这里读者可以简单地把头文件看成许多函数的集合，在程序中用到了这些函数中的一个或多个）。第 3 行中的 main() 表示程序中的“主函数”，C 程序的执行均是由执行 main 函数开始的。第 4 行与第 6 行表示 main 的函数体范围，左花括号“{”表示函数体的开始，“}”表示函数体的结束。第 5 行是 main() 的函数体，此函数体只有一条语句，它调用函数 printf()。printf() 函数是编译系统提供的、包含在头文件 stdio.h 中的标准库函数，功能是将双引号内的字符串输出到屏幕上，“\n”表示输出后换行。

例 1-2 由键盘输入两个整数，然后在屏幕上输出其中较大者。

完成该任务的 C 程序如下：

```
#include <stdio.h>  
void main()  
{  
    int x, y, z;  
    printf("Please input two numbers:\n");  
    scanf("%d %d", &x, &y);  
    z = max(x, y);  
    printf("maxmum=%d", z);  
}  
int max(int a, int b)  
{  
    int c;  
    if(a>b)  
        c = a;  
    else  
        c = b;  
    return c;  
}
```

该程序除了主函数 `main` 和 `printf` 外,增加了函数 `scanf` 和 `max`。其中 `scanf` 函数同 `printf` 一样是由编译系统提供的库函数,包含在标准输入输出头文件 (`stdio.h`) 中。`max` 函数是用户自定义的函数,它的作用是比较两个整数的大小,并返回较大的那个整数。程序中定义并使用了变量 `x`、`y`、`z`、`a`、`b` 和 `c` (关于变量类型、定义详见第 2 章)。程序还使用了结构化控制语句 `if-else` (详见第 3 章),增加了程序的判断功能。程序中输出的提示信息“Please input two numbers:”和“`maxmum=`”,增加了人机交互性。使用了赋值语句,如“`c = a`”,它将等号右边的数赋给左边。

通过上面的例子可以总结出 C 程序的组成和书写格式:

(1) C 程序由函数组成,函数是 C 程序的基本单位。每个程序有且仅有一个主函数 `main()`。可以调用标准库函数,也可以定义和调用自定义函数。当要调用标准库函数时,需要在程序前面用预处理命令包含相应的头文件。

(2) 程序的执行均是由执行 `main` 函数开始的,最后又回到 `main` 函数结束程序。中间可能因调用其他函数而跳出主函数。

(3) 函数由两部分组成:函数首部和函数体。函数首部确定函数名并说明函数类型、函数的形式参数名及其类型;函数体由一对“{”和“}”括起来的若干语句构成。

(4) 语句可以分为变量定义语句和执行语句。变量定义语句定义变量,如“`int x, y, z;`”和“`int c;`”;执行语句完成函数的功能,如“`z = max(x, y);`”。每条语句必须有分号作为结束标志。一行可写几条语句,一条语句也可以分行书写。

(5) C 语言区分大小写字母。一般用小写字母书写程序,大写字母作常量。

(6) 注释以“`/*`”开头,以“`*/`”结尾。注释文字可以是任意字符,如汉字、拼音、英文等。注释可以放在程序的任何位置,以提高程序的可读性。它不参与程序的编译和运行,是可有可无的。

(7) 在书写程序时为了便于阅读、维护和修改,最好一行一条语句,遇到嵌套语句向后缩进,必要时在语句后面加上注释。

1.2.4 C 程序上机过程

学习编程,上机是非常重要的。在计算机上编程并运行,从结果中了解程序的运行过程,进而了解程序的组成,比只看书要有效得多。另外,读者可以对已有的程序做各种修改,进而了解语言更多的知识,也可以向别人学到很多东西。上机可以提高读者对编程的兴趣,也可以提高自信心,当看到自己编写的程序在计算机上正确地运行时,读者会感到自己的创造力。

这里我们简要介绍用 Turbo C 2.0 这种比较流行的 C 语言编程环境进行编程的基本步骤,使读者能够用它进行简单的 C 程序开发。如果读者希望对它有更深入的了解,请查看其帮助文档和用户手册。

1. 启动 Turbo C 2.0

进入命令提示符窗口,在 Turbo C 2.0 的安装目录下,键入命令 `TC` 并回车,就进入 Turbo C 2.0 的集成环境,如图 1-2 所示。这时可以按 `F10` 键将光标移到顶部的主菜单行,然后可以用键盘上的“←”、“→”、“↑”和“↓”键将光标移动到需要的菜单命令上,按回车选中该命令。

2. 创建、编辑 C 程序

使用“File”菜单,可进行各种有关程序文件的操作,如创建新程序文件、打开已有的程序文件、存储程序和改变工作路径等。打开已有文件(用“Load”命令)或创建新文件(用

“New”命令)后,就可以输入或修改源程序了。要保存源程序,则应选择“File”下的“Save”命令。若当前文件是 NONAME.C,保存文件时最好换一个有意义的文件名。编辑区有已经打开的文件时,使用 Edit 菜单也可以进入编辑状态。



图 1-2 Turbo C 2.0 主界面

3. 编译、连接源程序

在 Turbo C 2.0 环境下,源程序的编译和连接可一起完成,有两种方法:直接按“F9”键和选择“Compile”菜单下的“Make EXE file”命令。也可以先对源程序进行编译,然后再连接。即先选择“Compile”菜单下的“Compile to OBJ”命令,编译通过后再选择“Compile”菜单下的“Make EXE file”命令。

如果编译和连接过程中产生警告(Warning)或出错(Error)信息,窗口中会显示警告和错误信息的数量,此时按任意键,这些错误信息会显示在屏幕下部的信息(Message)窗口中,必须纠正这些错误,对源程序进行修改,再重新编译、连接。

4. 运行程序

源程序编译、连接后就可以运行了。有两种方式:按“Ctrl+F9”键和选择“Run”菜单下的“Run”命令。

程序运行完毕后,将返回到 Turbo C 主屏幕。要查看程序运行后在屏幕上显示的结果,可以按“Alt+F5”键,看完后,再按任意键回到 Turbo C 主屏幕。

本章小结

本章主要是让读者对程序设计和 C 语言有个初步印象,为进一步深入学习打下基础。

习 题

一、选择题

1. 用 C 语言编写的代码程序()。

- A. 可立即执行
- B. 是一个源程序
- C. 经过编译即可执行
- D. 经过编译解释才能执行

2. 以下叙述中正确的是()。