

中等职业学校计算机系列教材

zhongdeng zhiye xuexiao jisuanji xilie jiaocai

C语言程序设计

钟廷志 赵洪波 编著



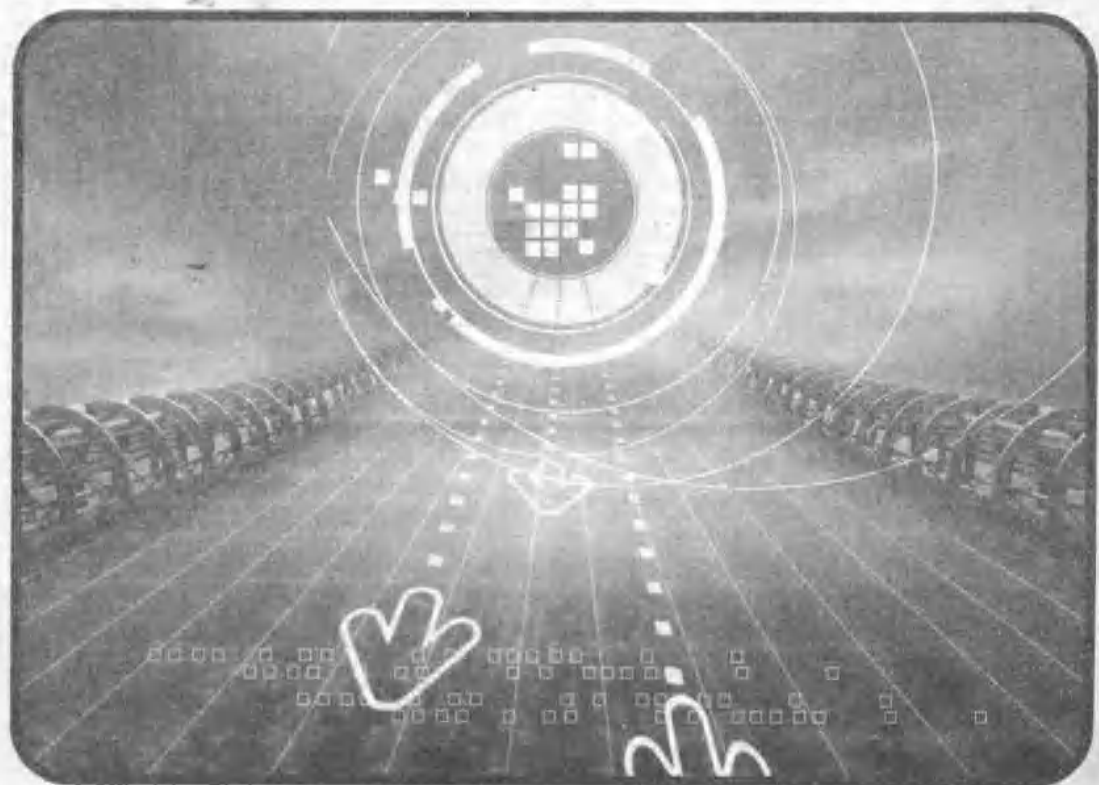
人民邮电出版社
POSTS & TELECOM PRESS

中等职业学校计算机系列教材

zhongdeng zhiye xuexiao jisuanji xilie jiaocai

C语言程序设计

钟廷志 赵洪波 编著



人民邮电出版社

图书在版编目 (CIP) 数据

C 语言程序设计/钟廷志, 赵洪波编著. —北京: 人民邮电出版社, 2004.6

(中等职业学校计算机系列教材)

ISBN 7-115-12153-2

I. C... II. ①钟... ②赵... 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 (2004) 第 048544 号

内容提要

本书详细介绍了 C 语言编程的基础知识和操作方法, 帮助学生具备使用 Turbo C 进行 C 语言程序设计的能力。全书共 11 章, 内容主要包括 Turbo C 的安装、启动、集成开发环境、三大程序结构、函数、数组、指针、预处理、共用体、结构体、文件以及 C++简介。在每章的最后均设有习题, 学生能够通过练习巩固本章所学知识。

本书适合作中等职业学校“C 语言程序设计”课程的教材, 也可作为 C 语言初学者的自学参考书。

中等职业学校计算机系列教材

C 语言程序设计

-
- ◆ 编 著 钟廷志 赵洪波
策 划 廖 霞 舒 凯
责任编辑 王文娟
 - ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号
邮编 100061 电子函件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
读者热线 010-67132692
北京汉魂图文设计有限公司制作
北京朝阳展望印刷厂印刷
新华书店总店北京发行所经销
 - ◆ 开本: 787×1092 1/16
印张: 16.25 2004 年 6 月第 1 版
字数: 390 千字 2004 年 6 月北京第 1 次印刷

ISBN7-115-12153-2/TP·3892

定价: 22.00 元

本书如有印装质量问题, 请与本社联系 电话: (010) 67129223

《中等职业学校计算机系列教材》编委会

(按姓氏笔画排列, 排名不分先后)

主任: 吴文虎

副主任: 马 騅 吴必尊 吴玉琨

吴甚其 周察金 梁金强

委员: 王计多 龙天才 任 毅 刘玉山 刘载兴

何文生 何长健 吴振峰 张孝剑 李 红

李任春 李智伟 杨代行 杨国新 杨速章

苏 清 邹 铃 陈 浩 陈 勃 陈禹甸

陈健勇 房志刚 林 光 侯穗萍 胡爱毛

郭红彬 税启兵 蒲少琴 赖伟忠 戴文兵

本书编委: 刘国亮 侯穗萍 康英健

序

中等职业教育是我国职业教育的重要组成部分。中等职业教育的培养目标定位于“具有综合职业能力强，在生产、服务、技术和管理第一线工作的高素质的劳动者和初中级专门人才”。

中等职业教育课程改革是为了适应市场经济发展的需要，适应课程模块化和综合化改革的需要，是为了适应实行一本多纲，满足不同学制、不同专业和不同办学条件的需要。

为了适应中等职业教育课程改革的发展，我们组织编写了本套教材。在编写过程中，我们参照了教育部职业教育与成人教育司制订的《中等职业学校计算机及应用专业教学指导方案》及劳动部职业技能鉴定中心制订的《全国计算机高新技术考试技能培训和鉴定标准》，并仔细研究了已出版的中职教材，去粗取精，全面兼顾了中职学生就业和考级的需要。

为了使本套教材能更好地适应不同地区教学的需要，我们选择了4个省市——北京、广东、湖南、四川进行了实地调研，走访了近100所中职学校，与约300名一线的中职老师进行了面对面的交流。通过座谈，我们更深刻地了解了中等职业学校的教学现状，以及师生们对教材内容、形式等方面的要求。

本套《中等职业学校计算机系列教材》第一批有30种，包括21种教材和9种配套的“上机指导与练习”。本套教材在写作风格上分为两类：

- 软件操作类。此类教材都与一个（或几个）实用软件或具体的操作技术相对应，如 Photoshop、Flash、3ds max 等，实践性很强。对于这类教材我们采用“任务驱动、案例教学”的方式编写，目的是提高学生的学习兴趣，使学生在积极主动地解决问题的过程中掌握所学知识。
- 理论教学类。此类教材需要讲授的理论知识较多，有比较完整的体系结构，操作性稍弱。对于这类教材，我们采用“传统教材+典型案例”的方式编写，力求在理论知识“够用为度”的基础上，使学生学到更实用的知识和技能。

为了方便教学，我们免费为选用本套教材的老师提供教学辅助光盘，光盘内容包括：

- 部分理论教学类课程的 PowerPoint 多媒体课件。
- 老师备课用的素材，包括本书目录的电子文档，按章提供的“本章学习目标”、“功能简介”、“案例小结”、“本章总结”等的电子文档。
- 按章提供教材上所有的习题答案。
- 按章提供所有实例制作过程中用到的素材。书中需要引用这些素材时会有相应的叙述文字，如“打开教学辅助光盘中的图片‘4-2.jpg’”。
- 按章提供所有实例的制作结果，包括程序源代码。
- 提供2套模拟测试题及答案，供老师考试使用。

本套书出版后，我们会在人民邮电出版社的网站（<http://www.ptpress.com.cn>）上开辟专门的讨论区，请作者与老师、同学们直接交流。在教材使用中老师们有什么意见或建议也可直接跟我们联系，联系电话是 010-67171429，电子邮件地址是 wangwenjuan@ptpress.com.cn。

编 者

2004年4月

前 言

本书是为中等职业学校编写的配套教材，根据教育部 2001 年颁布的《中等职业学校计算机及应用专业教学基本要求》编写，同时参考了《全国计算机信息高新技术考试技能培训和鉴定标准》中的程序员级考试大纲。

本书面向中等职业学校的学生，在内容的安排上尽量贴近应用，在叙述上尽量做到通俗易懂，全书循序渐进地向学生讲授如何使用 C 语言进行程序设计。既方便教师讲授，又便于学生理解掌握。

本书以 Turbo C 为蓝本，详细介绍了使用 Turbo C 进行 C 语言编程的基础知识和操作方法，帮助学生建立起 C 语言编程的思想，熟练掌握其编程的方法。

本书共 11 章：

- 第 1-2 章为基础概念部分，主要介绍 Turbo C 的安装、启动、集成开发环境、数据类型、运算符、表达式等基础知识。
- 第 3-5 章介绍 3 种基本程序结构，详细地讲述了顺序结构、选择结构和循环结构的设计方法。
- 第 6 章为函数部分，主要讲述了函数的定义和调用方法、函数参数的传递方法、变量的作用范围等知识。
- 第 7 章为数组部分，主要讲述了一维、二维数组的相关运用，以及有关字符串处理的几个函数。
- 第 8 章为指针部分，详细讲解了一维指针的使用，一维指针和一维数组的关系，简单讲解了二维指针的相关知识。
- 第 9 章主要讲述了编译预处理、结构体和共用体，其中以结构体为重点。
- 第 10-11 章分别简述了文件和 C++ 的相关知识。

书中各章都配有相应的习题，以利于学生对本章内容进行巩固。为了便于教学，每章还给出了大量实例，让学生在学习基本理论知识的同时，能够提高动手实践能力。本书另外配有一本《C 语言程序设计上机指导与练习》，使学生能够通过上机实验真正掌握相关知识点，具备运用 C 语言编写简单程序的能力。

教师一般可用 40 个学时讲解本教材内容，然后配合《C 语言程序设计上机指导与练习》一书，辅以 40 个学时的上机时间，即可较好地完成教学任务，总的讲课时间约为 80 个课时。两本教材中的内容也可以满足扩充至 120 课时所需的内容要求，教师在实际授课过程中可以根据需要对学时进行适当的调整。

本书适合作为中等职业学校计算机及应用专业以及其他相关专业的教材，也可作为各类计算机培训学校的教学用书，以及全国计算机等级考试的辅导用书，还可供计算机爱好者参考使用。

由于作者水平有限，疏漏之处敬请各位老师和同学指正。

作者

2004 年 4 月

目 录

| | |
|---------------------------------|----|
| 第 1 章 C 语言概述 | 1 |
| 1.1 C 语言简史 | 1 |
| 1.2 C 语言的特点 | 1 |
| 1.3 C 程序的组成 | 3 |
| 1.4 C 程序的编译、链接和运行 | 6 |
| 1.5 小结 | 8 |
| 1.6 习题 | 9 |
| 第 2 章 数据类型、运算符与表达式 | 10 |
| 2.1 常量与变量 | 10 |
| 2.1.1 常量 | 10 |
| 2.1.2 变量 | 11 |
| 2.2 基本数据类型 | 13 |
| 2.2.1 划分数据类型的必要性 | 13 |
| 2.2.2 整型数据 | 14 |
| 2.2.3 实型数据 | 17 |
| 2.2.4 字符型数据 | 19 |
| 2.2.5 类型的混合运算 | 23 |
| 2.3 运算符与表达式 | 26 |
| 2.3.1 算术运算符与算术表达式 | 27 |
| 2.3.2 赋值运算符与赋值表达式 | 28 |
| 2.3.3 关系运算符与关系表达式 | 29 |
| 2.3.4 逻辑运算符与逻辑表达式 | 30 |
| 2.3.5 其他运算符 | 32 |
| 2.3.6 运算符的优先级和结合性 | 35 |
| 2.4 小结 | 38 |
| 2.5 习题 | 39 |
| 第 3 章 顺序程序设计 | 44 |
| 3.1 语句 | 44 |
| 3.2 数据的输入与输出 | 47 |
| 3.2.1 字符输入/输出函数 | 47 |
| 3.2.2 格式化输入/输出函数 | 50 |
| 3.3 顺序结构程序设计 | 61 |
| 3.4 小结 | 64 |
| 3.5 习题 | 67 |

| | |
|---|------------|
| 第 4 章 选择程序设计 | 74 |
| 4.1 概述 | 74 |
| 4.2 条件选择语句——if 语句 | 74 |
| 4.2.1 if 语句的第 1 种形式——if 结构 | 74 |
| 4.2.2 if 语句的第 2 种形式——if-else 结构 | 76 |
| 4.2.3 if 语句的第 3 种形式——if-else if-else 结构 | 78 |
| 4.3 if 语句的嵌套使用（选学） | 80 |
| 4.3.1 在 if 语句中嵌套具有 else 子句的 if 语句 | 80 |
| 4.3.2 在 if 语句中嵌套没有 else 子句的 if 语句 | 83 |
| 4.4 多分支选择语句——switch 结构 | 85 |
| 4.4.1 switch 结构的基本形式 | 85 |
| 4.4.2 带 break 语句的 switch 语句 | 87 |
| 4.4.3 switch 语句的嵌套（选学） | 89 |
| 4.4.4 if 语句和 switch 语句的相互转换 | 89 |
| 4.5 选择程序设计综合实例 | 89 |
| 4.6 小结 | 94 |
| 4.7 习题 | 95 |
| 第 5 章 循环程序设计 | 100 |
| 5.1 概述 | 100 |
| 5.2 while 和 do-while 循环 | 100 |
| 5.2.1 while 循环 | 100 |
| 5.2.2 do-while 循环 | 103 |
| 5.2.3 while 循环和 do-while 循环的比较 | 105 |
| 5.3 for 循环 | 106 |
| 5.3.1 for 循环的一般形式 | 106 |
| 5.3.2 for 循环的特殊形式（选学） | 109 |
| 5.4 循环的嵌套 | 110 |
| 5.5 break 语句与 continue 语句 | 113 |
| 5.5.1 break 语句 | 113 |
| 5.5.2 continue 语句 | 114 |
| 5.6 for 循环与 while 循环的转换 | 115 |
| 5.7 典型例题评析 | 117 |
| 5.8 小结 | 122 |
| 5.9 习题 | 122 |
| 第 6 章 函数 | 127 |
| 6.1 概述 | 127 |
| 6.2 函数的定义 | 129 |
| 6.2.1 无参函数的定义 | 129 |
| 6.2.2 有参函数的定义 | 129 |

| | | |
|------------|---------------------------|------------|
| 6.3 | 函数的参数和返回值..... | 131 |
| 6.3.1 | 函数的参数..... | 131 |
| 6.3.2 | 函数的返回值..... | 135 |
| 6.4 | 函数的调用..... | 138 |
| 6.4.1 | 函数调用的一般形式..... | 138 |
| 6.4.2 | 函数调用的方式..... | 139 |
| 6.4.3 | 函数的声明..... | 140 |
| 6.5 | 变量的作用范围..... | 143 |
| 6.5.1 | 局部变量..... | 143 |
| 6.5.2 | 全局变量..... | 143 |
| 6.6 | 变量的存储类别(选学)..... | 145 |
| 6.6.1 | 自动变量 auto..... | 146 |
| 6.6.2 | 静型变量 static..... | 146 |
| 6.6.3 | 寄存器变量 register..... | 148 |
| 6.6.4 | 外部变量 extern..... | 148 |
| 6.7 | 典型实例剖析..... | 148 |
| 6.8 | 小结..... | 152 |
| 6.9 | 习题..... | 152 |
| 第7章 | 数组..... | 157 |
| 7.1 | 概述..... | 157 |
| 7.2 | 一维数组..... | 157 |
| 7.2.1 | 一维数组的定义..... | 157 |
| 7.2.2 | 一维数组的引用..... | 158 |
| 7.2.3 | 一维数组中元素的地址..... | 160 |
| 7.2.4 | 一维数组的初始化..... | 160 |
| 7.3 | 字符串..... | 164 |
| 7.3.1 | 字符串的输入函数 scanf、gets..... | 165 |
| 7.3.2 | 字符串的输出函数 printf、puts..... | 166 |
| 7.3.3 | 求字符串长度的函数 strlen..... | 167 |
| 7.3.4 | 字符串的复制函数 strcpy..... | 167 |
| 7.3.5 | 字符串的比较函数 strcmp..... | 168 |
| 7.3.6 | 字符函数的综合应用..... | 168 |
| 7.4 | 二维数组(选学)..... | 170 |
| 7.4.1 | 二维数组概述..... | 170 |
| 7.4.2 | 二维数组的定义..... | 170 |
| 7.4.3 | 二维数组中各元素的引用..... | 171 |
| 7.4.4 | 二维数组元素的地址..... | 171 |
| 7.4.5 | 二维数组的初始化..... | 173 |
| 7.5 | 数组在函数中的运用..... | 176 |
| 7.6 | 典型例题剖析..... | 179 |

| | | |
|---------------|----------------------------|------------|
| 7.7 | 小结 | 182 |
| 7.8 | 习题 | 183 |
| 第 8 章 | 指针 | 186 |
| 8.1 | 概述 | 186 |
| 8.2 | 指针的定义 | 186 |
| 8.2.1 | 地址的概念 | 187 |
| 8.2.2 | 指针的定义 | 187 |
| 8.2.3 | 指针变量的操作 | 188 |
| 8.2.4 | 指针用法小结 | 192 |
| 8.3 | 指针变量作函数参数 | 193 |
| 8.4 | 指针在数组中的运用 | 196 |
| 8.5 | 指针其他用法 (选学) | 200 |
| 8.6 | 典型实例剖析 | 202 |
| 8.7 | 小结 | 205 |
| 8.8 | 习题 | 206 |
| 第 9 章 | 编译预处理、结构体和共同体 | 210 |
| 9.1 | 概述 | 210 |
| 9.2 | 编译预处理 | 210 |
| 9.2.1 | 宏定义 | 211 |
| 9.2.2 | 文件包含与条件编译 | 214 |
| 9.3 | 结构体 | 215 |
| 9.3.1 | 结构体的定义 | 215 |
| 9.3.2 | 定义结构体变量 | 217 |
| 9.3.3 | 结构体变量初始化 | 219 |
| 9.3.4 | 结构体变量成员的访问 | 221 |
| 9.4 | 共同体 (选学) | 223 |
| 9.5 | 位运算符与位运算 (选学) | 225 |
| 9.6 | 小结 | 226 |
| 9.7 | 习题 | 228 |
| 第 10 章 | 文件 (选学) | 231 |
| 10.1 | 文件概述 | 231 |
| 10.2 | 文件指针 | 231 |
| 10.3 | 文件的打开 | 232 |
| 10.4 | 文件的关闭 | 234 |
| 10.5 | 文件的读写 | 234 |
| 10.6 | 其他一些函数 | 237 |
| 10.7 | 小结 | 238 |
| 10.8 | 习题 | 238 |

| | |
|------------------------------------|------------|
| 第 11 章 C++程序设计简介 (选学) | 239 |
| 11.1 C++的发展历程..... | 239 |
| 11.2 C++的特点..... | 239 |
| 11.3 C++与 C 的联系..... | 241 |
| 11.4 小结..... | 243 |
| 11.5 习题..... | 243 |
| 附录 常用资料 | 244 |
| F.1 C 语言的关键字..... | 244 |
| F.2 双目算术运算中两边运算量类型转换规律..... | 244 |
| F.3 运算符的优先级和结合性..... | 244 |
| F.4 C 库函数..... | 245 |
| F.5 常用字符与 ASCII 码对照..... | 248 |

第1章 C 语言概述

计算机语言经历了较长的发展阶段，各类语言的出现为计算机技术的发展提供了强大的动力。但是，当今时代，影响最大的还是各类高级语言，其中 C 语言便是典型代表。本章首先向读者介绍有关 C 语言的一般知识。

本章学习目标

- 了解 C 语言的历史和发展过程。
- 了解 C 语言的特点及 C 程序的组成部分。
- 了解 C 程序的编译、链接过程。

1.1 C 语言简史

C 语言于上个世纪 70 年代诞生于美国的贝尔实验室。在此之前，人们编写系统软件主要是使用汇编语言。由于汇编语言编写的程序依赖于计算机硬件，其可读性和可移植性都比较差。而高级语言的可读性和可移植性虽然较汇编语言好，但一般高级语言又不具备低级语言能够直观地对硬件实现控制和操作，程序执行速度快的特点。在这种情况下，人们迫切需要一种既具有一般高级语言特性，又具有低级语言特性的语言。于是 C 语言就应运而生了。

由于 C 语言既具有高级语言的特点又具有低级语言的特点，因此迅速普及，成为当今最有发展前途的计算机高级语言之一。C 语言既可以用来编写系统软件，也可以用来编写应用软件。例如美国的导弹防御系统就是用 C 语言编写而成的。现在，C 语言广泛地应用在机械、建筑和电子等行业，编写各类应用软件。

C 语言的发展历程如下。

- C 语言是美国贝尔实验室的 D.M.Ritchie 在 B 语言的基础上设计出来的，此时的 C 语言只是为了描述和实现 UNIX 操作系统的一种工作语言。在一段时间里，C 语言还只在贝尔实验室内部使用。
- 1975 年，UNIX 第 6 版公布后，C 语言突出的优点引起人们的普遍注意。
- 1977 年出现了可移植的 C 语言。
- 1978 年 UNIX 第 7 版的 C 语言成为后来被广泛使用的 C 语言版本的基础，被称为标准 C。
- 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 进行发展和扩充，并制定了新的标准，称为 ANSI C。
- 1990 年，国际标准化组织 ISO 制定了 ISO C 标准，目前流行的 C 语言编译系统都是以它为标准的。

1972 年以来，C 语言几经修改和发展，出现多个版本。在微型机上使用的有 Microsoft C、Turbo C、Quick C 以及 Borland C 等，它们的不同版本又略有差异。目前最流行的版本是 Microsoft C 和 Turbo C。本书的叙述基本上以 ANSI C 为基础，上机练习则以 Turbo C 环境为基础。

1.2 C 语言的特点

C 语言兼有汇编语言和高级语言的优点，既适合于开发系统软件，也适合于编写应用程



序。被广泛应用于事务处理、科学计算、工业控制、数据库技术等领域。

C语言之所以能存在和发展,并具有强大的生命力,这都要归功于其鲜明的特点。这些特点是多方面的,归纳如下。

(1) C语言是结构化的语言

C语言程序有3种基本结构:

- 顺序结构。
- 选择结构。
- 循环结构。

而由这3种基本结构组成的程序可以解决许多复杂的问题。C语言还具有结构化的控制语句,如if...else语句、while语句、switch语句以及for语句等,使用这些语句可以方便地控制程序的流程。因此,C语言是理想的结构化语言,符合现代编程风格的要求。

(2) C语言是模块化的语言

C语言主要用于编写系统软件和应用软件。一个系统软件的开发需要很多人经过几年的时间才能完成。一般来说,一个较大的系统程序往往被分为若干个模块,每一个模块用来实现特定的功能。

在C语言中,用函数作为程序的模块单位,便于实现程序的模块化。在程序设计时,将一些常用的功能模块编写成函数,放在函数库中供其它函数调用。模块化的特点可以大大减少重复编程。程序设计时,只要善于利用函数,就可减少劳动量、提高编程效率。

(3) 语言简洁、紧凑,使用方便、灵活

C语言一共只有32个关键字(如表1-1所示)和9种控制语句,程序书写形式自由,主要用小写字母表示。在一般语言中的许多结构单元在C语言中都通过库函数调用来完成,库函数可根据需要方便地扩充,压缩了一切不必要的程序组成部分。

表 1-1 ANSIC 的关键字

| | | | | |
|----------|---------|--------|----------|--------|
| 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 |
| volatile | while | | | |

(4) 程序可移植性好

C语言程序便于移植,目前C语言在许多计算机上的实现大都是由C语言编译移植得到的,不同机器上的编译程序大约有80%的代码是公共的。程序不做任何修改就可用于各种型号的计算机和各种操作系统。

(5) 数据结构丰富,具有现代化语言的各种数据结构

C语言的基本数据类型有整型(如int、unsigned int等)、实型(如float、double等)以及字符型(char)等。在此基础上还可创建各种构造数据类型,如数组、指针、结构体和共用体等。使用C语言还能用来实现复杂的数据结构,如链表、树等。这样丰富的数据结构无疑极大地增强了C语言的功能。



(6) C 语言运算符丰富、代码效率高

C 语言共有 34 种运算符, 使用各种运算符可以实现在其他高级语言中难以实现的运算。在代码质量上, C 语言可与汇编语言媲美, 其代码效率仅比用汇编语言编写的程序的代码低 10%~20%。

总之, C 语言以其突出的优点, 吸引着越来越多的使用者的关注。目前, 国内外使用和研究 C 语言的人正在迅猛增加, 优秀的 C 语言版本和配套的工具软件不断出现。现在, C 语言不仅为计算机专业工作者所使用, 而且为广大的计算机应用人员所喜爱和使用。



虽然 C 语言的优点很多, 但并不是完美无缺, 仍然存在一些不足, 例如运算符的优先级太多, 不便于记忆; 语法限制不太严格, 程序设计自由度大; 类型检验太弱, 不同类型数据转换比较随便, 因此不太安全等。

1.3 C 程序的组成

程序是在特定语言环境下为解决某一具体问题而编写的代码集合。下面先介绍几个简单的 C 程序, 从中分析 C 程序的组成及其特性。

【例1-1】 在屏幕上显示 I love you, China!。

```
#include<stdio.h>
void main()
{
    printf("I love you,China!");
}
```

本程序的作用是在屏幕上输出以下信息:

```
I love you,China!
```

本程序的说明如下。

- “main()”表示主函数, 大括号“{}”括起来的部分称为函数体。每个程序中都必须有且仅有一个 main 函数。
- 语句是组成程序的最小结构单元。主函数的函数体中只包括一个输出语句: printf() 是 C 语言的输出函数; 双引号内的内容按原样输出; “\n”是换行符, 即在输出“I love you,China!”后光标换到下一行。
- 每个语句最后必须有一个分号。

【例1-2】 比较两个变量值的大小。

```
#include<stdio.h>
void main()
{
    int a,b,c;           /*定义变量 a,b,c 为整型变量*/
    a=20;b=15;          /*给变量 a,b 赋初值*/
    c=a>b?a:b;          /*比较变量 a,b 的大小, 将较大的数赋给变量 c*/
    printf("%d\n",c);   /*输出比较结果*/
}
```

程序运行结果为:



本程序的作用是比较整数 a 和 b 的大小, 说明如下。

- 程序中 “/*……*/” 包围的部分为注释 (如果在 $c++$ 的环境下运行 c 程序, 其注释部分可用 “//” 代替 “/*……*/”, 把注释语句放在 “//” 之后即可)。注释部分便于阅读程序的人理解程序员的设计意图, 并不影响程序的编译和运行。注释可以放在程序中的任何位置。
- C 语言中使用变量作为数据组织的基本形式, 变量的基本操作包括变量的定义、变量的赋值以及变量的运算等。
- 程序第 4 行定义了 a 、 b 和 c 这 3 个变量, 并指定它们为整型 (int) 变量; 程序第 5 行是两个赋值语句, 为变量 a 和 b 设定初始值分别为 20 和 15; 程序第 6 行将变量 a 与 b 中较大的数赋给变量 c 。

【例1-3】 使用函数对两个数据进行加减运算。

```
#include<stdio.h>
void main()
{
    int add(int x,int y);
    int mul(int x,int y); /*对被调用函数的声明*/
    int a,b,sum,count; /*声明部分,定义变量*/
    a=20;b=35; /*为a,b赋初值*/
    sum=add(a,b); /*调用函数add,将得到的值赋给sum*/
    count=mul(a,b); /*调用函数mul,将得到的值赋给count*/
    printf("sum=%d,count=%d\n",\
        sum,count); /*输出sum,count的值*/
}
int add(int x,int y) /*定义add函数,函数值为整型,形式参数x和y为整型*/
{
    int z; /*add函数中的声明部分,定义本函数用到的变量z为整型*/
    z=x+y;
    return(z); /*将z的值返回,通过add带回调用处*/
}
int mul(int x,int y) /*定义mul函数*/
{
    int w; /*定义mul函数中用到的变量w为整型*/
    w=x*y;
    return(w); /*返回w的值*/
}
```

运行结果为:

```
sum=55,count=700
```

这个程序的目的是显示两个整数 a 、 b 的和与积, 说明如下。

- 函数是构成 C 程序的基本单位。本程序包含 3 个函数: 主函数 $main()$ 、函数 $add()$ 和 $mul()$ 。函数 $add()$ 的作用是求 a 、 b 之和, 函数 $mul()$ 的作用是求两个



数的乘积。

- 在 C 程序中，一个函数可以使用函数调用的方法访问另一个函数。程序中的第 6 行和第 7 行为 `main()` 函数对 `add()` 函数和 `mul()` 函数的调用。在调用函数时将实际参数 `a` 和 `b` 的值分别传送给函数 `add()` 和 `mul()` 中的形式参数 “`x`” 和 “`y`”，然后执行函数相应的运算，并将结果通过 `return` 语句返回给主函数。

通过以上几个程序，可以看出 C 语言的以下结构特点。

(1) 语句是 C 程序的最小构成单位

C 程序中每个语句以一个 “;” (分号) 结束。分号是 C 程序语句必不可少的组成部分。

如【例 1-3】中第 7 行 “`a=20; b=35;`”，如写成 “`a=20 b=35`”，编译时在 “`a=20`” 后面未发现分号，就把下一行 “`b=35`” 也作为上一行的语句的一部分，系统将显示程序有语法错误。

(2) `main()` 函数是程序执行的起点

一个 C 程序总是从 `main()` 函数开始执行的，而不论 `main()` 函数在整个程序中的位置如何。`main()` 函数可以放在程序的最前面，也可以放在程序最后，或放在一些函数之间。

(3) C 程序书写格式自由

C 程序的一行内可以书写几个语句。如【例 1-3】中第 7 行 “`a=20;b=35;`”，它包含 “`a=20;`” 和 “`b=35;`” 两个语句。一个语句也可以写在多行上，用 “\” 作续行符。如【例 1-3】中第 10 行。C 程序没有行号，不像其他语言（如 COBOL、Fortran 等）那样严格规定书写格式。

(4) 可在 C 程序的任何部分添加注释语句，以提高程序的可读性

C 语言注释由 “`/*`” 开始，至 “`*/`” 结束。注释可分为若干行，但不允许嵌套。注释使程序变得清晰，增加了程序的可读性，便于阅读和理解。给程序加上注释是一个良好的编程习惯。注释部分不会对程序本身的运行产生影响。

(5) 函数是 C 程序的基本组成单位

函数是构成 C 语言程序的基本单元。一个 C 源程序中有且仅有一个 `main()` 函数，还可以包含任意多个其他函数。程序运行时，通过函数调用的形式来实现函数之间的信息传递。被调用的函数可以是系统提供的库函数（如 `printf()` 函数），也可以是用户根据需要自己编写的函数（如【例 1-3】中的 `add()` 和 `mul()` 函数）。各函数的书写位置无关紧要。

(6) 函数是 C 程序模块化结构的重要体现

根据【例 1-3】可以归纳出 C 语言程序的一般形式：

```
main()
{
    变量定义      (声明部分)
    调用函数 fun1
    调用函数 fun2
    .....
}
fun 1()
{
    变量定义
    调用函数 fun3
```




```

.....
    程序段
}
.....

fun1{
}

fun2{
}

fun3{
}

fun4{
}

fun5{
}

fun6{
}

fun7{
    变量定义
    调用函数 fun7
    程序段
}

```

其中 fun1、fun2 为用户自定义函数。



C语言把许多常用的函数功能模块通过库函数的形式提供给用户，C语言的函数库十分丰富。函数库是由C语言开发者编制，用户需要时直接调用即可。灵活使用库函数，可以提高编程效率，使C程序本身规模减少，编译程序简单，很容易在各种机器上实现，同时还增强程序的可移植性。

1.4 C程序的编译、链接和运行

在了解了C语言的基本知识后，下面接着介绍运行C语言程序的相关问题。本节介绍目前使用最广泛的C语言程序开发软件——Turbo C的基本知识。Turbo C是Borland公司开发的一个适用于微型计算机，同时集程序编辑、编译、链接和调试为一体的C编译系统。它具有良好的用户界面和丰富的库函数，同时还有运行速度快、效率高和功能强等优点，使用非常方便。

一个C语言程序要能够最终实现既定的功能，需要依次经历以下基本环节。

- 编辑：使用C语言编写程序代码，创建源文件。
- 编译：在C程序的编译过程中，可以查出程序中的语法错误。编译器将程序转换为机器代码后即可生成目标程序（.obj）。
- 链接：C程序是模块化设计程序，一个C程序可能由多个程序设计者分工合作编写。最后需要将库函数以及其他目标程序链接为一个整体，生成可执行文件（.exe）。
- 运行：运行源文件经过编译链接后生成的可执行文件（.exe）即可获得正确的结果。

下面介绍使用Turbo C开发C语言程序的一般方法和步骤。

(1) 启动 Turbo C

启动Turbo C后，将打开Turbo C集成开发环境，计算机屏幕上出现Turbo C主窗口，这也是Turbo C的图形用户界面。该界面由主菜单、编辑窗口、信息窗口和功能键提示行4部分组成，如图1-1所示。

(2) 主菜单

主菜单包含下面8个下拉菜单。

- **【File】**：文件操作，载入或保存文件，管理目录，调入DOS和退出Turbo C系统。