

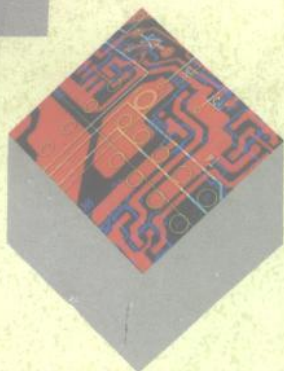
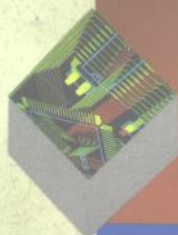
GAODENGXUEYIAOJIAOJIANJICHUJIAOYULICONGSHU

高等学校计算机基础教育丛书

谭浩强 主编

Turbo C

程序设计教程



谭浩强 刘炳文 鲍 泓 编著

人民邮电出版社

2
1

高等学校计算机基础教育丛书

谭浩强 主编

Turbo C 程序设计教程

谭浩强 刘炳文 鲍泓 编著

人民邮电出版社

登记证号(京)143号

图书在版编目(CIP)数据

Turbo C 程序设计教程/谭浩强等编著. —北京:人民邮电出版社,1995.2
(高等学校计算机基础教育丛书)
ISBN 7-115-05492-4

I. T… II. 谭… III. C语言—程序设计 IV. TP312C

内 容 提 要

Turbo C 具有良好的使用环境,是在微型计算机上广泛使用的一种 C 版本。本书比较全面地、系统地讲解了 Turbo C 的语法规则以及用 Turbo C 进行程序设计的方法。特别是第十二章详细地介绍了 Turbo C 的程序设计环境,对学习和使用 C 语言进行程序设计很有实用价值。

本书是以国家教委工科计算机基础课程教学指导委员会制定的《高级语言程序设计》的基本要求为依据,同时参照国家教委考试中心制定的《全国计算机等级考试大纲》(第二级)编写的。本书概念清晰,取材广泛,叙述通俗易懂,并配有丰富的例题与习题,特别适合于初学者使用。该书是一本较好的 C 语言教科书。

本书可作为高等学校教材以及计算机培训班教材,也可供自学使用。

高等学校计算机基础教育丛书

谭浩强 主编

Turbo C 程序设计教程

谭浩强 刘炳文 鲍泓 编著

责任编辑 赵桂珍

*

人民邮电出版社出版发行

北京朝阳门内南竹杆胡同111号

北京市朝阳区展望印刷厂印刷

新华书店总店科技发行所经销

*

开本:787×1092 1/16 1995年2月 第一版

印张:28.75 1995年2月 北京第1次印刷

字数:710千字 印数:1—6 000册

ISBN 7-115-05492-4/TP·149

定价:25.00元

前 言

C 语言是近年来兴起的一种新型语言,在国内已被广泛应用,不少计算机应用人员和计算机的初学者都希望学习 C 语言。我曾先后主编过《C 程序设计》(清华大学出版社出版)和《C 语言程序设计教程》(高等教育出版社出版),介绍 C 语言以及如何用 C 语言进行程序设计,受到读者欢迎。

目前在微型计算机系统上使用的 C 语言编译程序有许多种不同的版本。其中 Turbo C 是一种流行较广、使用较多的版本。Turbo C 提供一个集成环境,使源程序的建立、编译、连接以及运行通过菜单选择来完成,使用起来十分方便,尤其对初学者来说更具有其优越性。虽然 Turbo C 的语法规则与标准 C 基本相同,但它的运行环境与其它 C 编译有较大不同,语法规则和标准函数库也与标准 C 略有差异。考虑到 Turbo C 目前是微机上使用的主流 C 版本之一,有必要针对 Turbo C 的特点单独编写出版一本 C 教材,以促进 Turbo C 的推广使用。

本书的大部分内容是介绍 C 语言的语法规则和编程方法,这部分内容是具有普遍性的,即:除了个别规定以外,绝大多数内容也适合于其他版本的 C 编译。第十二章专门介绍了 Turbo C 的运行环境,附录给出 Turbo C 的标准库函数,这部分内容是具有特殊性的,只适合于 Turbo C。一个 Turbo C 的程序如果改在其它 C 编译环境下编译运行,要注意其差别,应做必要的修改。

本书是为初学者而编写的,可以做为高等学校的教材和计算机培训班的教材,也可供自学参考。

本书由谭浩强教授根据课程要求确定全书内容和编写大纲,并担任主编。刘炳文副教授编写了第一章至第五章、第十二章及第十一章的 §11.1 节,鲍泓副教授编写第六章至第十一章,谭浩强教授对全书各章内容进行了深入细致的修改,并最后定稿。由于编写时间紧张,本书定会存在一些缺点或不足之处,热诚希望得到广大读者和专家的批评指正。

编著者

《高等学校计算机基础教育丛书》序

当前我们面临着计算机普及的又一次高潮。这一次普及的特点是向一切领域、各个层次的人群普及,也就是全方位、多层次的普及。现在,计算机是作为文化来普及,作为工具来使用的。这一高潮的到来不是偶然的,它是我国计算机应用发展的必然产物。实践证明,只有大力发展计算机应用,才能促进计算机产业的发展。只有把计算机技术与各行各业的各个专业领域结合起来,才能打开计算机应用的局面,才能从根本上推动计算机产业的发展。

计算机的应用是分层次的,计算机的教育也是分层次的。人们可以按以下几个层次循序渐进地学习计算机知识。第一个层次是:计算机的初步知识和计算机的操作使用(包括 DOS 操作命令、汉字输入、字表处理、数据库的简单操作等)。这是最初级的层次。第二个层次是:高级语言程序设计。达到这一层次要求的人员应该能用一种高级语言(如 BASIC、FORTRAN、PASCAL、C)或一种数据库语言(如 dBASE 或 FoxBASE)熟练地编写程序,而程序设计是计算机应用人员的基本功,是计算机基础教育的基础和重点。第三个层次是:进一步学习计算机硬件知识。达到这一层次要求的人员应具有开发应用软件或硬件系统的初步开发能力。第四个层次是结合本专业领域对计算机应用的需要进行学习。

国家教委工科计算机基础课程教学指导委员会据此制定了五门计算机课程的基本要求。这五门课是:《微机系统应用基础》(属第一层次)、《高级语言程序设计》(属第二层次,可选 BASIC、FORTRAN、PASCAL 和 C 语言之一)、《微机原理与应用》(属第三层次的偏硬的方向,使学生具有开发微机硬件系统的初步能力)、《软件开发技术》和《计算机软件基础》(这两门课属第三层次的偏软的方向,使学生具有开发应用软件的初步能力)。以上课程的设置不仅适用于工科,对其它专业也具有参考意义。目前,多数高校的非计算机专业已按以上方案开设了计算机课程。

为了进一步普及计算机教育,国家教委考试中心也已推出面向社会的“全国计算机等级考试”,设搞一、二、三个级别的考试,相当于上面的前三个层次的内容。

以上的安排为广大计算机的初学者和初、中级计算机应用人员提供了很好的教学体系。

根据广大读者要求,我们参照国家教委工科计算机基础课程教学指导委员会制定的计算机课程基本要求和国家教委考试中心公布的《全国计算机等级考试大纲》,编写了这套《高等学校计算机基础教育丛书》。本丛书的读者对象是高等学校非计算机专业的学生,也可供其它计算机爱好者以及计算机应用人员学习参考。本丛书包括三个层次的各门课程所用的教材,将陆续配套出齐。

本丛书的特点是:实用性强、内容新颖、通俗易懂、概念清晰。我们邀请高等学校中有丰富教学经验的老师作为丛书的作者。相信这套丛书的出版会受到高校和社会的欢迎。

由于计算机科学技术发展很快,需要学习的内容在不断变化,因此上面介绍的几个层次的

内容以及课程的设置也会随着形势的发展而变化。我们将根据计算机科学技术的发展和社会的需要,及时修改补充丛书的内容。

我们希望得到各位专家以及广大读者的支持与帮助,请及时给我们提出宝贵的批评意见,以使丛书的内容不断完善。

谭治强

教授計算技術的大師
普及現代科技之巨擘

敬頌譚浩強教授創杰成就

宋健

一九九五年一月

國務委員宋健題詞

祝賀譚浩強教授著作創世界發行記錄

光榮的記錄
傑出的業績

一九九五年九月

盧嘉錫



全國人民代表大會副委員長盧嘉錫題詞

祝譚浩強教授的著作創科技在世界記錄

巨大的貢獻 杰出的成就

李沛瑤
一九九五年二月

全國人民代表大會副委員長李沛瑤題詞

目 录

第一章 概述	(1)
§ 1.1 C 语言发展简况	(1)
§ 1.2 C 语言的特点	(2)
§ 1.3 用 Turbo C 运行 C 程序	(3)
习题.....	(8)
第二章 Turbo C 程序设计基础	(9)
§ 2.1 C 标识符与源程序的构成	(9)
2.1.1 Turbo C 标识符	(9)
2.1.2 简单的 Turbo C 程序	(10)
§ 2.2 Turbo C 数据类型	(12)
§ 2.3 基本数据类型	(13)
2.3.1 数值数据的表示形式	(13)
2.3.2 字符数据的表示形式	(14)
2.3.3 数据的存储	(15)
2.3.4 带符号的数据类型与无符号的数据类型	(16)
§ 2.4 常量与变量	(18)
2.4.1 常量的种类	(18)
2.4.2 常量的表示	(19)
2.4.3 变量及其赋值	(23)
2.4.4 变量的定义和初始化	(24)
§ 2.5 运算符与表达式	(25)
§ 2.6 算术运算符与算术表达式	(28)
2.6.1 二元算术运算符	(28)
2.6.2 一元算术运算符	(30)
2.6.3 使用算术表达式有关问题的说明	(32)
§ 2.7 关系运算与逻辑运算	(34)
2.7.1 关系运算符与关系表达式	(34)
2.7.2 逻辑运算符和逻辑表达式	(35)
2.7.3 条件运算符和条件表达式	(37)
§ 2.8 逗号运算符与逗号表达式	(38)
§ 2.9 类型转换	(39)
2.9.1 隐式类型转换	(39)
2.9.2 显式类型转换	(42)

习题	(43)
第三章 数据输入输出与程序基本结构	(44)
§ 3.1 Turbo C 语句	(44)
§ 3.2 标准输入输出	(46)
3.2.1 字符输入输出函数	(47)
3.2.2 格式输出函数 printf	(50)
3.2.3 格式输入函数 scanf	(56)
§ 3.3 顺序结构	(63)
§ 3.4 选择结构	(66)
3.4.1 if 语句	(66)
3.4.2 switch 语句	(72)
§ 3.5 循环结构	(76)
3.5.1 while 循环	(77)
3.5.2 do—while 循环	(86)
3.5.3 for 循环	(89)
3.5.4 循环的嵌套与退出	(94)
§ 3.6 程序举例	(98)
习题	(106)
第四章 模块化程序设计	(109)
§ 4.1 C 程序与函数	(109)
§ 4.2 函数的定义与声明	(112)
4.2.1 函数的定义	(112)
4.2.2 函数的声明	(118)
§ 4.3 函数的调用	(119)
4.3.1 概述	(119)
4.3.2 值传递	(123)
4.3.3 嵌套调用	(126)
4.3.4 递归调用	(128)
§ 4.4 Turbo C 库函数	(132)
§ 4.5 变量的存储	(133)
4.5.1 动态变量	(134)
4.5.2 静态变量	(138)
4.5.3 外部变量	(140)
4.5.4 存储类别小结	(147)
§ 4.6 编译预处理	(148)
4.6.1 宏替换	(149)
4.6.2 文件包含	(159)
4.6.3 条件编译	(160)
习题	(165)
第五章 数组	(169)

§ 5.1 数组的定义·····	(169)
5.1.1 一维数组的定义·····	(169)
5.1.2 二维数组和多维数组的定义·····	(170)
§ 5.2 数组的初始化·····	(172)
5.2.1 一维数组的初始化·····	(172)
5.2.2 二维数组和多维数组的初始化·····	(173)
§ 5.3 数组元素的引用·····	(175)
5.3.1 一维数组的引用·····	(175)
5.3.2 二维数组和多维数组的引用·····	(176)
§ 5.4 数组作为函数参数·····	(178)
5.4.1 数组元素作为函数参数·····	(178)
5.4.2 数组名作为函数参数·····	(180)
5.4.3 用多维数组作函数参数·····	(183)
§ 5.5 字符数组和字符串·····	(184)
5.5.1 字符数组的定义和初始化·····	(184)
5.5.2 字符串及其存储·····	(186)
5.5.3 字符串的输入输出·····	(188)
5.5.4 字符串函数·····	(190)
5.5.5 二维字符数组·····	(194)
5.5.6 字符数组应用举例·····	(196)
§ 5.6 程序举例·····	(199)
习题·····	(204)
第六章 结构体、共用体及自定义类型 ·····	(205)
§ 6.1 结构体类型数据·····	(205)
6.1.1 概述·····	(205)
6.1.2 定义结构体类型与定义结构体变量·····	(207)
6.1.3 定义结构体类型内的结构体类型成员·····	(210)
6.1.4 结构体变量的初始化·····	(211)
6.1.5 结构体变量的引用·····	(213)
6.1.6 结构体数组·····	(215)
§ 6.2 结构体在函数间的传递·····	(220)
6.2.1 结构体成员在函数间的传递·····	(220)
6.2.2 整个结构体在函数间的传递·····	(221)
6.2.3 返回结构体类型值的函数·····	(224)
§ 6.3 共用体类型数据·····	(227)
6.3.1 共用体的概念·····	(227)
6.3.2 定义共用体类型变量·····	(228)
6.3.3 共用体变量的引用·····	(229)
§ 6.4 枚举类型数据·····	(232)
6.4.1 定义枚举常量和变量·····	(232)

6.4.2 枚举变量的初始化和赋值·····	(232)
6.4.3 枚举值的运算和引用·····	(233)
§ 6.5 用 typedef 定义类型名·····	(234)
习题·····	(238)
第七章 指针基础 ·····	(239)
§ 7.1 指针概述·····	(239)
7.1.1 指针与地址·····	(239)
7.1.2 定义指针变量·····	(240)
7.1.3 指针变量的初始化·····	(242)
7.1.4 指针变量的引用·····	(242)
7.1.5 指针运算·····	(243)
§ 7.2 指针与数组·····	(245)
7.2.1 指针与数组的等价性·····	(245)
7.2.2 数组下标法与指针变量法的特点·····	(248)
7.2.3 字符串的指针表示法·····	(250)
§ 7.3 指针作为函数参数·····	(252)
7.3.1 传递地址常量与传递指针变量·····	(252)
7.3.2 数组的指针作为函数参数·····	(254)
7.3.3 字符串的指针作为函数参数·····	(255)
§ 7.4 返回指针值的函数——指针型函数·····	(257)
§ 7.5 指针数组·····	(258)
7.5.1 指针数组概述·····	(258)
7.5.2 用指针数组处理多维数组数据·····	(259)
7.5.3 用字符指针数组处理字符串组·····	(260)
7.5.4 指针数组作为函数参数·····	(263)
7.5.5 main 函数中的参数·····	(264)
习题·····	(265)
第八章 指针和动态存储分配 ·····	(269)
§ 8.1 指向指针的指针·····	(269)
8.1.1 多级间址的概念·····	(269)
8.1.2 指向指针的指针与指针数组的等价性·····	(270)
§ 8.2 指向函数的指针·····	(271)
8.2.1 函数指针的概念·····	(271)
8.2.2 用函数指针变量调用函数·····	(272)
8.2.3 函数指针作为函数参数·····	(274)
§ 8.3 指向结构体的指针·····	(276)
8.3.1 结构体指针变量的定义·····	(276)
8.3.2 用结构体指针变量访问结构体成员·····	(277)
8.3.3 指向结构体数组的指针·····	(278)
8.3.4 用指向结构体的指针作函数参数·····	(280)

8.3.5 结构体指针型函数.....	(281)
§ 8.4 指向 void 类型的指针	(282)
§ 8.5 动态存储分配.....	(284)
8.5.1 为什么要动态存储分配.....	(284)
8.5.2 用 malloc 函数分配内存空间	(285)
8.5.3 确定被分配存储空间的大小.....	(285)
8.5.4 对 malloc 返回地址的赋值与检测	(286)
8.5.5 访问被分配存储块.....	(287)
8.5.6 用 free 函数释放存储块	(287)
8.5.7 特殊的存储分配函数.....	(287)
§ 8.6 用指针实现链表.....	(288)
8.6.1 链表概述.....	(288)
8.6.2 链表的建立.....	(290)
8.6.3 链表的遍历与查找.....	(291)
8.6.4 链表的插入与删除.....	(292)
习题.....	(298)
第九章 输入输出及文件	(299)
§ 9.1 概述.....	(299)
§ 9.2 C 标准设备文件.....	(300)
9.2.1 C 文件的概念.....	(300)
9.2.2 二进制流和字符流.....	(301)
9.2.3 缓冲与非缓冲文件系统.....	(302)
§ 9.3 标准磁盘输入输出——缓冲文件系统.....	(303)
9.3.1 文件类型指针(FILE 指针)	(303)
9.3.2 文件的打开与关闭.....	(304)
9.3.3 文件缓冲区的控制.....	(306)
9.3.4 文件的读和写.....	(308)
9.3.5 文件状态检测.....	(315)
9.3.6 文件的定位与随机存取.....	(316)
§ 9.4 系统级输入输出——非缓冲文件系统.....	(318)
9.4.1 非缓冲文件系统的特点.....	(318)
9.4.2 系统级文件的操作.....	(319)
§ 9.5 输入输出换向和管道.....	(322)
9.5.1 输入输出换向.....	(322)
9.5.2 管道.....	(323)
9.5.3 C 程序的输入输出换向和管道	(324)
9.5.4 标准错误输出定向.....	(326)
习题.....	(327)
第十章 位运算	(329)
§ 10.1 概述	(329)

§ 10.2 位序列的输出	(332)
§ 10.3 位运算符及其使用方法	(334)
10.3.1 按位取反运算	(334)
10.3.2 按位“与”运算	(336)
10.3.3 按位“或”运算	(338)
10.3.4 按位“异或”运算	(339)
10.3.5 左移运算	(341)
10.3.6 右移运算	(343)
§ 10.4 位运算应用举例	(345)
§ 10.5 位段	(348)
10.5.1 位段的概念和定义方法	(348)
10.5.2 位段的引用方法	(351)
习题	(353)
第十一章 系统资源的利用及 C 程序开发实例	(355)
§ 11.1 DOS 中断与系统功能调用	(355)
11.1.1 中断概述	(355)
11.1.2 访问 ROM—BIOS 系统	(356)
11.1.3 利用 DOS 访问系统功能	(360)
§ 11.2 屏幕编辑器	(363)
11.2.1 屏幕编辑器原理分析	(363)
11.2.2 屏幕编辑程序主要模块的说明	(364)
11.2.3 完整的屏幕编辑子系统	(367)
§ 11.3 绘图程序	(380)
11.3.1 显示方法和调色板	(380)
11.3.2 画点	(381)
11.3.3 画线	(383)
11.3.4 画矩形和填充矩形	(385)
11.3.5 画圆和填充圆	(385)
第十二章 Turbo C2.0 程序设计环境	(388)
§ 12.1 Turbo C2.0 软件包	(388)
12.1.1 Turbo C2.0 的特点	(388)
12.1.2 新增加的实用工具和函数	(389)
§ 12.2 Turbo C2.0 系统的安装	(392)
§ 12.3 Turbo C 的启动与退出	(394)
12.3.1 Turbo C 的启动	(394)
12.3.2 退出 Turbo C	(395)
12.3.3 联机帮助	(396)
§ 12.4 窗口	(396)
§ 12.5 菜单与热键	(398)
12.5.1 打开菜单	(398)

12.5.2 热键	(399)
§ 12.6 Turbo C 菜单命令	(401)
§ 12.7 文本编辑	(409)
12.7.1 简单编辑操作	(409)
12.7.2 块操作	(410)
12.7.3 查找与替换	(411)
12.7.4 其他编辑操作	(413)
§ 12.8 程序调试	(414)
12.8.1 Turbo C 调试器	(414)
12.8.2 程序调试举例	(419)
§ 12.9 在 DOS 下建立可执行程序	(423)
12.9.1 TCC 命令	(424)
12.9.2 TCC 命令选择项	(426)
附录 I 常用字符与 ASCII 代码对照表	
附录 II Turbo C2.0 关键字	(432)
附录 III Turbo C2.0 常用库函数	(433)
附录 IV Turbo C2.0 头文件	(439)

第一章 概 述

§ 1.1 C 语言发展简况

C 语言是一种结构化、模块化、编译式通用程序设计语言,具有较好的可移植性,可用来编写各种软件(包括系统软件和应用软件),特别是系统软件。各种程序设计任务几乎都可以用 C 语言来完成。

早期的系统软件(如操作系统等,包括 UNIX)大多是用汇编语言编写的。由于汇编语言依赖于计算机硬件,程序的可读性和可移植性都比较差。为了提高可读性和可移植性,最好改用高级语言,但一般高级语言难以实现汇编语言的某些功能(汇编语言可以直接对硬件进行操作,如对内存地址的操作、位操作等)。人们希望能找到一种具有一般高级语言特性,又具有低级语言特性的语言,集它们的优点于一身。于是,在这种情况下,C 语言应运而生了。

C 语言是在 B 语言的基础上发展起来的。1960 年推出的 ALGOL60 是一种面向问题的高级语言,它离硬件比较远,不适合用来编写系统程序。1963 年,英国剑桥大学推出了 CPL (Combined Programming Language)语言。CPL 语言在 ALGOL60 的基础上接近硬件一些,但规模较大,难以实现。1967 年,英国剑桥大学的 Martin Richards 对 CPL 语言作了简化,推出了 BCPL (Basic Combined Programming Language)语言。1970 年美国贝尔实验室的 Ken Thompson 以 BCPL 语言为基础,又作了进一步简化,设计出了很简单而且很接近硬件的 B 语言(取 BCPL 的第一个字母),并用 B 语言编写了第一个 UNIX 操作系统,在 PDP-7 上实现。1971 年在 PDP-11/20 上实现了 B 语言,并编写了 UNIX 操作系统。但是,B 过于简单,功能有限。1972 到 1973 年间,贝尔实验室的 D. M. Ritchie 在 B 语言的基础上设计了 C 语言(取 BCPL 的第二个字母)。C 语言既保持了 BCPL 和 B 语言的优点(精练,接近硬件),又克服了它们的缺点(过于简单,数据无类型等)。最初的 C 语言只是为描述和实现 UNIX 操作系统提供一种工作语言而设计的。1973 年,K. Thompson 和 D. M. Ritchie 两人合作把 UNIX 的 90%以上用 C 语言改写,即 UNIX 第 5 版。

此后,C 语言作了多次改进,但主要还是在贝尔实验室内部使用。直到 1975 年 UNIX 第 6 版公布后,C 语言的优点才引起人们的普遍注意。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》,使 C 移植到其他机器时所需做的工作大为简化,这也推动了 UNIX 操作系统迅速在各种机器上实现,如 VAX,AT&T 等计算机系统都相继开发了 UNIX。随着 UNIX 的日益广泛使用,C 语言也得到迅速推广。C 语言和 UNIX 可以说是一对孪生兄弟,在发展过程中相辅相成。1978 年以后,C 语言已先后移植到大、中、小、微型机上,独立于 UNIX 和 PDP。现在,C 语言已风靡全世界,成为应用最广泛的几种计算机语言之一。