



普通高等教育“十一五”国家级规划教材

高等学校计算机基础教育教材精选



C 程序设计教程 (第2版)



崔武子 赵重敏 李 青 编著

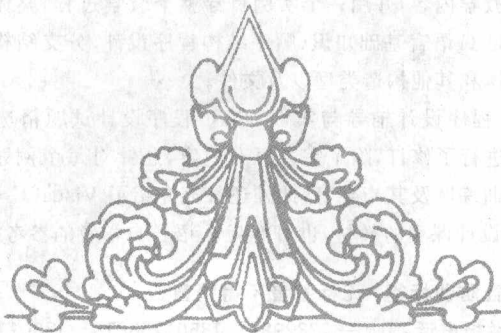
清华大学出版社



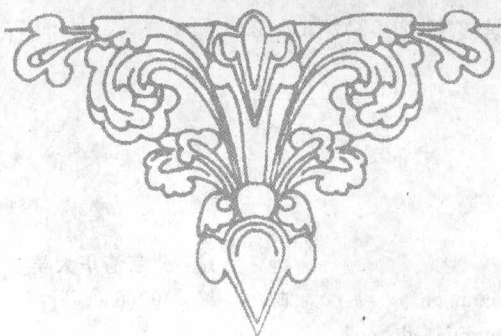


普通高等教育“十一五”国家级规划教材

高等学校计算机基础教育教材精选



C 程序设计教程 (第2版)



崔武子 赵重敏 李 青 著

清华大学出版社
北京

内 容 简 介

本书是以 C 程序设计零起点读者作为主要对象的程序设计教程,2003 年 7 月出版了第 1 版,本次再版则进一步强化了编程能力。本书已被纳入普通高等教育“十一五”国家级规划教材。

本书用例题组织所有的教学内容,并用一个实例贯穿整个教学过程,整体内容编排独特,组织形式新颖。全书共分 10 章,分别是 C 语言基础知识、顺序结构程序设计、分支结构程序设计、循环结构程序设计、数组、指针、函数、结构体和其他构造类型以及文件。

本书配备了辅助教材《C 程序设计辅导与实训》和《C 程序设计试题精选》以及内容生动的电子教案。本书在第 1 版的基础上进行了修订,除了完善原有内容,还针对每个例题增加了知识要点的介绍,在各章之后补充了本章上机训练题及其指导等,此外还补充介绍了 Visual C++ 6.0 环境。

本书是高等院校 C 程序设计课程的教材,也可作为 C 语言自学者的参考书。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13501256678 13801310933

图书在版编目(CIP)数据

C 程序设计教程/崔武子,赵重敏,李青编著. —2 版. —北京:清华大学出版社,2007.6
(高等学校计算机基础教育教材精选)
ISBN 978-7-302-14955-2

I. C… II. ①崔… ②赵… ③李… 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 044236 号

责任编辑:谢 琛

责任校对:白 蕾

责任印制:何 芊

出版发行:清华大学出版社

<http://www.tup.com.cn>

c-service@tup.tsinghua.edu.cn

社总机:010-62770175

投稿咨询:010-62772015

地 址:北京清华大学学研大厦 A 座

邮 编:100084

邮购热线:010-62786544

客户服务:010-62776969

印刷者:北京市世界知识印刷厂

装订者:三河市李旗庄少明装订厂

经 销:全国新华书店

开 本:185×260 印 张:22.75

版 次:2007 年 6 月第 2 版

印 数:1~5000

定 价:29.00 元

字 数:522 千字

印 次:2007 年 6 月第 1 次印刷

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题,请与清华大学出版社出版部联系调换。联系电话:010-62770177 转 3103 产品编号:024926-01

出版说明

——高等学校计算机基础教育教材精选——

在教育部关于高等学校计算机基础教育三层次方案的指导下,我国高等学校的计算机基础教育事业蓬勃发展。经过多年的教学改革与实践,全国很多学校在计算机基础教育这一领域中积累了大量宝贵的经验,取得了许多可喜的成果。

随着科教兴国战略的实施及社会信息化进程的加快,目前我国的高等教育事业正面临着新的发展机遇,但同时也必须面对新的挑战。这些都对高等学校的计算机基础教育提出了更高的要求。为了适应教学改革的需要,进一步推动我国高等学校计算机基础教育事业的发展,我们在全国各高等学校精心挖掘和遴选了一批经过教学实践检验的优秀教学成果,编辑出版了这套教材。教材的选题范围涵盖了计算机基础教育的三个层次,包括面向各高校开设的计算机必修课、选修课,以及与各类专业相结合的计算机课程。

为了保证出版质量,同时更好地适应教学需求,本套教材将采取开放的体系和滚动出版的方式(即成熟一本、出版一本,并保持不断更新)。坚持宁缺毋滥的原则,力求反映我国高等学校计算机基础教育的最新成果,使本套丛书无论在技术质量上还是出版质量上均成为真正的“精选”。

清华大学出版社一直致力于计算机教育用书的出版工作,在计算机基础教育领域出版了许多优秀的教材。本套教材的出版将进一步丰富和扩大我社在这一领域的选题范围、层次和深度,以适应高校计算机基础教育课程层次化、多样化的趋势,从而更好地满足各学校由于师资和生源水平、专业领域等的差异而产生的不同需求。我们热切期望全国广大教师能够积极参与到本套丛书的编写工作中来,把自己的教学成果与全国的同行们分享;同时也欢迎广大读者对本套教材提出宝贵意见,以便我们改进工作,为读者提供更好的服务。

我们的电子邮件地址是 jiaoh@tup.tsinghua.edu.cn。联系人:焦虹。

清华大学出版社

前言

C 程序设计教程(第 2 版)

本书是以 C 程序设计零起点读者作为主要对象编写的程序设计教程。本书内容编排独特,组织形式新颖,能使读者在较短的时间内掌握 C 程序设计的精华。本书是高等院校 C 程序设计课程的教材,也可作为自学者的参考书。

本书第 1 版特点:

(1) 每章内容分成基础部分和提高部分。考虑到 C 的语法现象众多、初学者往往难以接受,书中将每章的内容分成了基础和两个部分。将常识性的、基础类的、必须掌握的内容放在基础部分中;将具有扩展性的、提高性的、专业性的内容安排在提高部分中。通过基础部分的学习,能够掌握最基本的语法,初步建立程序设计的思维方式和编写一般程序的能力,同时可培养学生的学习兴趣。即使因学时不足跳过提高部分,也不至于影响后续内容的学习。

(2) 所有教学内容用例题组织。在基本遵循 C 语言教学体系的情况下,将所有教学内容用例题组织。即根据要介绍的内容精心编写相应的例题,将大量的、正确的、规范的程序介绍给学生,在讲解例题的过程中,使学生学习语法、了解概念、掌握算法,做到在解决实际问题中讲授语法,而不是为了教语法而举例。

(3) 涉及算法的例题增设编程点拨,部分算法提供多种解法。针对学生“读程序容易,编程序难”的情况,书中凡涉及算法的例题,在给出其完整的程序之前,都增设了编程点拨,有些算法还提供了多种解法。

(4) 分阶段介绍程序调试方法。为了培养学生调试程序、排除错误的能力,教材中分阶段通过具体例题介绍了调试程序的方法,使用的环境是 Turbo C 2.0。

(5) 讲授指针和函数时不涉及新算法。指针和函数是 C 语言中的重点和难点,为了使能够顺利接受新概念,将有关算法内容尽量安排在这两章之前,避免学生在接受指针和函数概念的同时,又要理解新算法。这样做不仅容易突破难点,而且有利于巩固已学过的知识。

(6) 用实例贯穿整个教学过程。为了使能够尽早接触应用程序的编写过程,在提高部分中,提供了贯穿整个教学过程的一个实例,并随着讲授内容的增多,逐步补充和完善其程序的功能。

(7) 习题形式新颖,提供单号习题答案。与教材内容相对应,各章习题也分为基础和两个部分。为了逐步提高学生的编程能力,还精心编写了形式新颖的习题。书中单号习题提供参考答案,以方便学生自测和教师布置作业。

(8) 配备含有电子教案等内容的课件。该课件包括电子教案、全书所有例题和实例的源程序。为了减轻教师备课的负担,将基础部分中的所有内容和提高部分中的部分内容,制作成生动的电子教案,通过演示可使读者尽早了解本课程的基本目标。

(9) 配套出版了辅助教材。《C 程序设计辅导与实训》内容既兼顾了随教学进度安排的阶段实训,也考虑到全部教学内容完成后的集中实训。实训部分以本教材中的实例作为样例;辅导部分给出了大量例题的解析、训练题和参考答案,因此是本教材的最佳教学辅助资料。而《C 程序设计试题精选》提供了 1000 道练习题,包括选择题、填空题和上机题。选择题按等级考试“四选一”的标准形式;每道填空题则根据等级考试要求最多含有 3 个需要填写的空格;上机题有改错题、补充完整题和编程题 3 种类型。

本书第 2 版主要修订内容:

(1) 在各章开头针对每道例题添加了知识要点。

(2) 在各章之后添加了上机训练内容,每个训练题均分为题目、目标、步骤、提示和扩展。

(3) 补充介绍 Visual C++ 6.0 环境。

(4) 精简内容,修改或更换部分例题,例题中所有变量均预先赋初值,规范了变量定义形式。

(5) 适当添加讨论题,促使互动教学。

(6) 将例题、习题、讨论题、上机训练题重新统筹安排。

(7) 在附录中添加一个含关键字、运算符、库函数的索引。

使用建议:

(1) 必学基础部分。基础部分是学生必须掌握的知识,但在教学过程中教师可将部分例题留给学生自学。

(2) 选学提高部分。书中的提高部分是为了帮助读者更上一层楼,教师可以根据实际情况,选择其中部分内容进行介绍(标有“*”的例题有一定难度)。为了提高学生的编程和调试能力,建议教师指导学生其中的实例程序。

(3) 单、双号习题成对做。单号习题提供参考答案,双号习题则在类型上与前一单号习题相同,侧重点也接近。基础部分中提供的习题都是最基本的,题量也不多,建议读者全部做完,提高部分中的习题可根据情况选做(标有“*”的习题有一定难度)。

(4) 选作上机训练题中的扩展题。在完成训练题的基础上可根据不同层次的学生情况,选作扩展题。

本书由崔武子主编,赵重敏和李青参与编写,齐华山和李红豫承担了部分工作。本书由鲍有文教授审阅。在编写本书的过程中,得到谭浩强教授和高林教授的热情关怀和指导。刘莹老师对第 1 版书提出许多宝贵意见,在此一并表示真挚的感谢。

限于作者水平,书中难免有错误和疏漏之处,恳请读者批评和指正。

作者电子邮件地址:ldtwuzi@163.com

作者

2007 年 2 月

目录

C 程序设计教程(第 2 版)

第 1 章 C 语言基础知识	1
1.1 C 语言概述	2
1.1.1 什么是 C 语言	2
1.1.2 C 程序形式和程序执行过程	3
1.2 简单 C 程序与上机步骤	4
1.2.1 简单 C 程序	4
1.2.2 上机步骤	5
1.3 数据类型	7
1.4 常量与变量	8
1.4.1 常量与变量的概念	8
1.4.2 整型常量与变量	10
1.4.3 实型常量与变量	12
1.4.4 字符型常量与变量	13
1.5 运算符和表达式	16
1.5.1 算术运算符和表达式	16
1.5.2 赋值运算符和表达式	18
1.5.3 逗号运算符和表达式	20
1.5.4 自加、自减运算符	20
1.6 提高部分	21
1.6.1 Turbo C 2.0 集成环境	21
1.6.2 Visual C++ 6.0 集成环境	24
1.6.3 赋值运算符的进一步讨论	28
1.6.4 自加、自减运算符的进一步讨论	29
1.6.5 整型数据类型的进一步讨论	31
1.7 上机训练	33
习题 1	34
基础部分	34
提高部分	35

第 2 章 顺序结构程序设计	37
2.1 结构化程序设计的基本结构	38
2.1.1 语句的概念	38
2.1.2 三种基本结构	38
2.2 赋值语句	40
2.3 输入输出语句	41
2.3.1 格式输入输出函数	41
2.3.2 字符输入输出函数	45
2.4 提高部分	46
2.4.1 输入输出函数的进一步讨论	46
2.4.2 实例的部分程序	50
2.5 上机训练	51
习题 2	53
基础部分	53
提高部分	53
第 3 章 分支结构程序设计	55
3.1 关系运算符和关系表达式	56
3.1.1 关系运算符	56
3.1.2 关系表达式	56
3.2 逻辑运算符和逻辑表达式	57
3.2.1 逻辑运算符	57
3.2.2 逻辑表达式	58
3.3 if 语句	59
3.3.1 if 语句的一般形式	60
3.3.2 if 语句的嵌套	67
3.4 switch 语句	69
3.5 提高部分	72
3.5.1 if、switch 语句的进一步讨论	72
3.5.2 条件运算符和表达式	79
3.5.3 实例的部分程序	80
3.6 上机训练	82
习题 3	84
基础部分	84
提高部分	86
第 4 章 循环结构程序设计	87
4.1 for 语句	88

4.2	while 语句	97
4.3	do-while 语句	100
4.4	break 语句和 continue 语句	103
4.4.1	循环体中使用 break 语句	103
4.4.2	循环体中使用 continue 语句	105
4.5	循环语句的嵌套	105
4.6	提高部分	109
4.6.1	for 语句的应用	109
4.6.2	三种循环的对比	112
4.6.3	goto 语句以及用 goto 语句构成的循环	116
4.6.4	实例的部分程序	117
4.7	上机训练	120
	习题 4	123
	基础部分	123
	提高部分	126
第 5 章	数组	128
5.1	一维数组	129
5.1.1	一维数组的定义和引用	129
5.1.2	一维数组的初始化	132
5.1.3	一维数组程序举例	135
5.2	字符串	141
5.2.1	字符串的概念和字符串的输入输出	141
5.2.2	字符串处理函数	143
5.3	二维数组	146
5.4	提高部分	149
5.4.1	数组的补充程序举例	149
5.4.2	实例的部分程序	158
5.5	上机训练	160
	习题 5	163
	基础部分	163
	提高部分	164
第 6 章	指针	166
6.1	变量的地址和指针变量的概念	167
6.2	指针变量的定义和引用	167
6.3	指针和一维数组	170
6.3.1	使指针变量指向一维数组	171

6.3.2	对指针的算术运算	172
6.4	指针和字符串	175
6.4.1	通过字符数组名引用字符串	175
6.4.2	通过指针变量引用字符串	176
6.5	提高部分	178
6.5.1	指针的进一步讨论	178
6.5.2	指针和二维数组	180
6.5.3	实例的部分程序	183
6.6	上机训练	193
习题 6		196
基础部分		196
提高部分		197
第 7 章 函数 198		
7.1	函数的引例	199
7.2	函数的定义与调用	202
7.2.1	函数的定义	202
7.2.2	函数的调用	203
7.2.3	函数的调用过程	208
7.2.4	函数的返回值	211
7.2.5	被调函数的原型说明	215
7.3	函数的嵌套调用	216
7.4	数组做实参	217
7.4.1	一维数组名做实参	217
7.4.2	二维数组名做实参	220
7.5	变量的存储类别	221
7.5.1	内部变量和外部变量	221
7.5.2	动态存储变量和静态存储变量	223
7.6	提高部分	224
7.6.1	函数的递归调用	224
7.6.2	带参数的 main 函数	227
7.6.3	指向函数的指针	231
7.6.4	变量存储类别的进一步讨论	233
7.6.5	预处理命令	235
7.6.6	实例的部分程序	238
7.7	上机训练	241
习题 7		244
基础部分		244

提高部分	247
第 8 章 结构体和其他构造类型	249
8.1 结构体类型变量的定义和使用	250
8.1.1 结构体类型的概念和声明	250
8.1.2 结构体类型变量的使用	251
8.2 结构体和函数调用	257
8.3 提高部分	261
8.3.1 结构体的进一步讨论	261
8.3.2 链表	264
8.3.3 共用体	272
8.3.4 实例的部分程序	274
8.4 上机训练	277
习题 8	279
基础部分	279
提高部分	280
第 9 章 位运算	282
9.1 移位运算符	282
9.2 按位与、或、异或运算符	284
9.3 按位取反运算符	285
9.4 提高部分	286
9.4.1 位运算的复合赋值运算符	286
9.4.2 负整型数据的位运算	286
9.4.3 无符号整型数据的位运算	287
9.4.4 不同类型数据之间的位运算	287
习题 9	287
基础部分	287
提高部分	288
第 10 章 文件	289
10.1 文件的概述	290
10.2 文件的基本操作	291
10.3 提高部分	299
10.3.1 文件读写操作的进一步讨论	299
10.3.2 文件的定位操作	305
10.3.3 实例的部分程序	309
10.4 上机训练	312

习题 10	314
基础部分	314
提高部分	315
附录 A C 语言关键字	316
附录 B 常用字符与 ASCII 代码对照表	317
附录 C 运算符的优先级和结合方向	319
附录 D 常用 C 库函数	320
附录 E 常见错误、警告信息表	324
附录 F 关键字、运算符、库函数索引	326
附录 G 单号题参考答案	329
参考文献	348

本章将介绍的内容

基础部分：

- C 程序的基本概念和上机步骤。
- 各种数据类型的常量和变量。
- 算术运算、赋值运算和逗号运算。

提高部分：

- Turbo C 2.0 和 Visual C++ 6.0 集成环境。
- 进一步学习赋值运算符和自加、自减运算符。
- 进一步学习整型数据类型。

各例题的知识要点

例 1.1 C 程序形式和程序执行过程。

例 1.2 主函数、函数体和输出控制。

例 1.3 换行控制。

例 1.4 数据类型的正确选用。

例 1.5 常量和变量的概念。

例 1.6 合法与非法的变量名。

例 1.7 整型数据的各种进制输出。

例 1.8 整型变量的定义和数据的溢出现象。

例 1.9 实型常量的不同输出形式。

例 1.10 实型变量的定义和有效数字。

例 1.11 常规字符不同的格式输出以及字符常量的算术运算。

例 1.12 特殊字符的输出和转义字符的概念。

例 1.13 字符型变量的定义和赋值。

例 1.14 将代数式转化为 C 语言表达式。

例 1.15 强制类型转换。

例 1.16 逗号表达式。

例 1.17 自加、自减运算符。

(以下为提高部分例题)

- 例 1.18 复合赋值运算符。
- 例 1.19 由自加、自减运算符构成的表达式。
- 例 1.20 复杂形式的逗号表达式。
- 例 1.21 整型常量的各种进制表示法。
- 例 1.22 各种整型数据的输出。
- 例 1.23 用图理解基本型整数的变化情况。

1.1 C 语言概述

1.1.1 什么是 C 语言

人和人之间交换信息需要借助于语言工具,人和计算机交换信息也同样要用语言工具。我们将后一种语言称为计算机语言。用计算机语言编写的代码称为程序。

随着计算机技术的发展,计算机语言逐步得到完善。最初使用的计算机语言是用二进制代码表达的语言——机器语言。后来采用与机器语言相对应的助记符表达的语言——汇编语言。虽然用这两种语言编写的程序执行效率高,但程序代码很长,又都依赖于具体的计算机,因此编码、调试、阅读程序都很困难,通用性也差。我们称这两种语言为低级语言。现在使用最广的计算机语言是高级语言——更接近于人们自然语言的表达语言。高级语言独立于机器,编码相对简短,可读性强。FORTRAN、QBASIC、Pascal、COBOL、C 等都是高级语言。用高级语言编写的程序叫做源程序。由于计算机只能识别 0 和 1,因此源程序必须通过编译和连接后,才能被计算机执行。

C 语言比其他高级语言功能更强,它既具有高级语言的功能也具有低级语言的许多功能,即具有双重性,因此有的书把 C 语言称为中级语言。C 语言是由附录 A 中列出的 32 个关键字再加上语法规则构成的。

要得到 C 程序的运行结果,首先将源程序输入计算机(在计算机上输入或修改源程序的过程叫做编辑),然后还要把源程序翻译成机器能识别的目标程序,如在 Turbo C 2.0 集成环境中,将编辑后的源文件保存为 e1.c,则通过编译产生的与源文件相对应的目标程序为 e1.obj;但是,目标程序不是可执行文件,不能直接运行,还要把目标程序和系统提供的库函数等连接起来产生可执行文件 e1.exe,这时我们才可以运行程序,并看到运行结果。C 程序的编辑、编译、连接、运行过程可用图 1.1 表示。

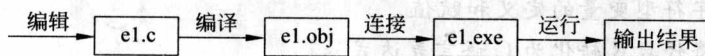


图 1.1 C 程序的编辑、编译、连接、运行过程

C 程序的编辑、编译、连接、运行过程可以在不同的环境中进行,而本书中的所有例题是在 Turbo C 2.0 集成环境下运行通过的。

1.1.2 C 程序形式和程序执行过程

下面举一个 C 语言程序的完整例题,用此例说明 C 程序的一般形式和程序的执行过程。程序中的具体语法规则和其他细节将在后续章节中陆续介绍。

【例 1.1】 编写一个完整的 C 语言程序示例。

【解】 程序如下:

```
#include <stdio.h> /* 包含文件 */
#include <math.h> /* 包含文件 */
int mysum(int m,int n); /* 函数原型说明 */

main( ) /* 主函数首部 */
{ /* 声明部分 */
    double c,y,z; /* 声明部分 */
    c = 4.0; /* 以下 6 行均为语句部分 */
    y = sqrt(c);
    a = 10; b = 20;
    x = mysum(a,b);
    z = x + y;
    printf("z = %lf\n",z);
} /* 主函数到此结束 */

int mysum(int m,int n) /* mysum 函数首部 */
{ /* 声明部分 */
    int k;

    k = m + n; /* 以下 2 行均为语句部分 */
    return k;
}
```

运行结果:

```
z = 32.000000
```

程序说明:

(1) 正如本例所示,C 语言程序是由若干函数构成的,函数中至少包含一个主函数,C 程序从主函数开始执行,主函数名必须是 main。例 1.1 中程序的执行过程如图 1.2 所示,程序按①到⑨的顺序执行。

(2) 程序中用“/* */”括起来的部分是注释部分,该部分对程序的运行无任何作用,注释的目的是方便阅读程序。注释可以出现在程序的任何位置。

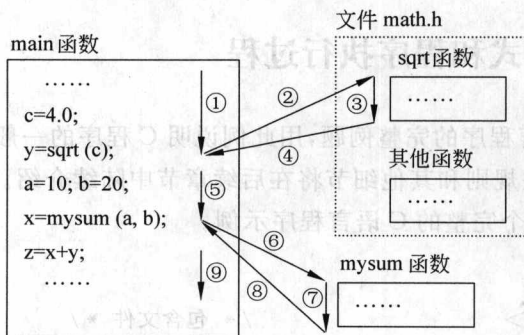


图 1.2 例 1.1 程序的执行过程

1.2 简单 C 程序与上机步骤

本节将给出几个最简单的 C 程序,通过这些例题,介绍 C 语言的基本概念,以及在 Turbo C 2.0 集成环境下的上机步骤。

1.2.1 简单 C 程序

【例 1.2】 编写在屏幕上显示一个句子“Let's study the C language.”的程序。

【解】 程序如下:

```
#include <stdio.h>
main()
{
    printf("Let's study the C language.");
}
```

运行结果:

Let's study the C language.

程序说明:

(1) 程序中 main 是主函数名,每一个 C 程序都必须包含而且只能包含一个主函数。用一对大括号({})括起来的部分是函数体。本例函数体中只有一条语句“printf(“Let's study the C language.”);”,此语句是输出语句,其作用是按原样输出双引号内的字符串“Let's study the C language.”(详见 2.3.1 节)。语句最后的“;”不能丢。

(2) C 语言中英文大小写字母被认为是不同的字符,即 main 不能写成 Main,printf 也不能写成 Printf。若程序中有此类错误,则很难发现。C 语言中的所有关键字都用小写字母。

(3) 程序中的“#include <stdio.h>”称为命令行,有了此行,就可以成功地调用 C

语言标准库中提供的输入、输出函数,所以编写程序时,在程序的第一行都写此行。

【例 1.3】 编写输出两个句子“Let's study the C language.”和“It's interesting.”的程序。

【解】 程序如下:

```
#include <stdio.h>
main( )
{
    printf("Let's study the C language.\n");    /* 输出字符串后换行 */
    printf("It's interesting.\n");
}
```

运行结果:

Let's study the C language.

It's interesting.

程序说明:

(1) 本程序的函数体包括两条输出语句,函数体可以包括任意多条语句。

(2) “\n”是换行符,如果程序中去掉“\n”,输出形式则为:

Let's study the C language. It's interesting.

【讨论题 1.1】 程序中最后一个输出语句中的换行符“\n”的作用是什么?

小结:

(1) C 程序一定有且仅有一个主函数,主函数名必须是 main,其后的圆括号内可以是空的,但圆括号不能省略。

(2) 函数体中可以有多条语句,每条语句都用分号结束。

(3) 注释用“/* …… */”形式,注意,“/”和“*”之间不能加有空格。为了提高程序的可读性,建议加上必要的注释。

(4) C 程序的书写格式比较自由。例如,一行内可以包括多条语句,一条语句也可以写在多行上,每行的内容可以从任何一列开始写。但是建议初学者每行写一条语句,而每行的语句根据需要适当向右缩进几列(参见例 4.19),这对读懂程序和调试程序很有帮助。本书中的所有程序都按最常用的书写格式,请读者参考。

1.2.2 上机步骤

下面在 Turbo C 2.0 集成环境下,给出例 1.2 和例 1.3 的上机步骤。

先介绍例 1.2 的上机步骤。

第 1 步: 在 c 盘 TC 目录下安装 Turbo C 2.0。如果已经安装,则跳过此步。

第 2 步: 进入 Turbo C 集成环境。在 Windows 的资源管理器中,双击 tc.exe 文件可进入 Turbo C 集成环境,按 Alt+Enter 组合键,可使屏幕变成最大,再按一次此组合键,可以恢复原屏幕大小(详见 1.6.1 节)。