

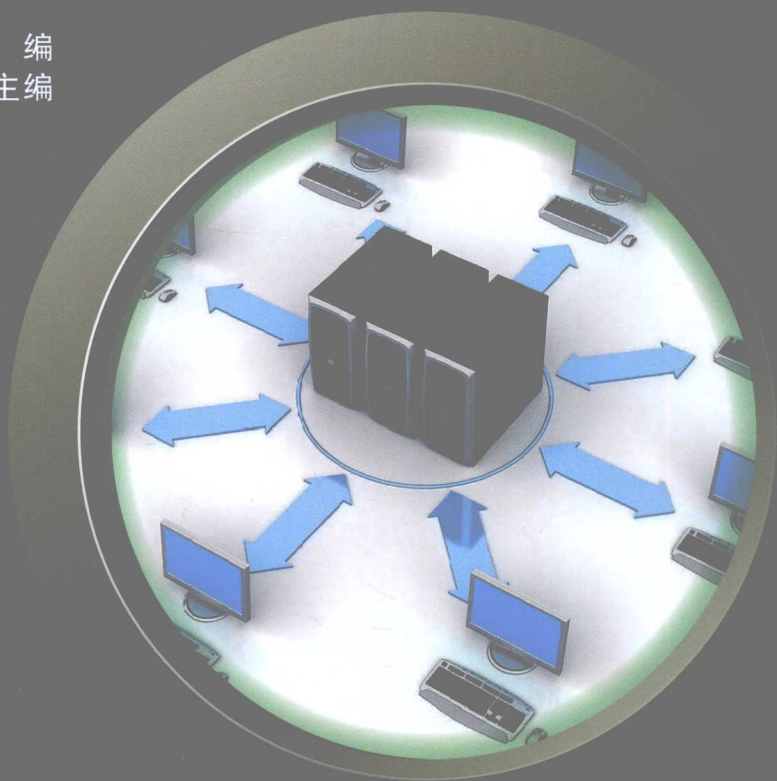


高等院校计算机应用技术系列教材

C 语言程序设计

——Visual C++ 6.0

傅 力 主 编
李志达 沈 卓 副主编



清华大学出版社 • 北京交通大学出版社

高等学校计算机应用技术系列教材

C 语言程序设计

——Visual C++ 6.0

傅 力 主 编
李志达 沈 卓 副主编

清华大学出版社
北京交通大学出版社

·北京·

内 容 简 介

高级语言程序设计是计算机教学的基础课程,本书较为详尽地介绍了C语言及其程序设计方法,着重培养读程序、编程序两种能力,以及初步运用C语言进行大型程序和工程软件设计的能力。书中使用Visual C++ 6.0和Turbo C 2.0两种教学软件,以Visual C++ 6.0为主进行教学,程序采用汉字和英文两种显示和输入方式。

书中内容主要包括:C语言的形成与发展、程序的基本概念、C程序的运行方法、基本数据类型、运算符与表达式、可执行语句与程序执行流程、数组、指针、结构与联合、结构的位成员、不完全类型、系统函数与自定义函数、函数的递归调用、文件操作函数、存储类说明符、类型限定符、翻译单元与源程序、存在期与可见性、预处理命令等内容,详尽地给出了C语言程序设计的一个应用编程实例。作为算法应用和选讲内容,介绍了用C语言实现的常用排序、查找算法等。此外书中还给出了三套全国计算机等级考试二级C语言程序设计笔试模拟试卷,习题和模拟题给出了相应的参考答案。

本书可作为高等院校本科、专科教材,也可作为计算机爱好者自学用书。C语言在工程领域,特别是在嵌入式系统领域的应用非常广泛,因而本书也是工程技术人员学习C语言程序设计的参考书。目前,国家计算机考试将C语言列为二级考试,本书亦可作为二级考试C语言部分的参考用书。

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语言程序设计: Visual C++ 6.0/傅力主编. —北京:清华大学出版社;北京交通大学出版社,2009.1

ISBN 978-7-81123-496-1

I. C… II. 傅… III. C语言-程序设计 IV. TP312

中国版本图书馆CIP数据核字(2008)第211430号

责任编辑:郭东青

出版发行:清华大学出版社 邮编:100084 电话:010-62776969

北京交通大学出版社 邮编:100044 电话:010-51686414

印刷者:北京东光印刷厂

经 销:全国新华书店

开 本:185×260 印张:14.75 字数:368千字

版 次:2009年1月第1版 2009年1月第1次印刷

书 号:ISBN 978-7-81123-496-1/TP·459

印 数:1~4 000册 定价:23.00元

本书如有质量问题,请向北京交通大学出版社质监组反映。对您的意见和批评,我们表示欢迎和感谢。

投诉电话:010-51686043,51686008;传真:010-62225406;E-mail:press@bjtu.edu.cn。

前 言

本书是作者在多年教学实践的基础上，结合 C 语言发展应用现状编写而成。书中内容循序渐进，从具体样例入手，逐步给出规范化的语法，便于理解掌握；对于编译、内存等概念，采用类比的方法讲解，便于初学者入门和深入学习；书中还附有大量图例，使抽象问题具体化。对于例题的讲解，尽量采用简单语义来阐述复杂的语法，并采用一种语义，多种语法的讲解方式，使复杂内容通俗易懂，适于不同层次人员学习。大型应用实例结合日常应用，实用性较强。习题和等级考试模拟题经过课堂讲授和精选，并给出了参考答案，便于自学。

多年从事 C 语言教学，笔者感到使用 Turbo C 2.0 或 Turbo C++ 3.0 教学（16 位 C 编译系统），由于属于命令提示符环境，独立于 Windows 操作，教学效率不高，不失为教学瓶颈，为提高教学效率，有必要采用新的教学环境。Visual C++ 6.0（32 位 C 编译系统）和 Turbo C 2.0 是计算机不同发展时期出现的软件，书中以 Visual C++ 6.0 为主流教学软件，Windows 环境，且无需辅助软件，程序可直接采用汉字显示和输入方式运行，比较而言教学效率更高，更易于学习；Turbo C 2.0 属于早期教学软件，为了遵从教学习惯，本书沿用了此软件。书中分别阐述了 C 语言在两种编译环境中的不同点，有助于教学软件向 Visual C++ 6.0 过渡过程中的教学。

本书内容较为系统、全面，共分 8 章，其中对概念的阐述，注意了软件技术的发展现状。作为高级语言来学习，1~5 章属于基本内容。第 1 章介绍了 C 语言和 C 标准的形成发展现状，以及程序设计的基本概念，并介绍了在计算机上建立和运行一个 C 程序的具体方法；第 2 章阐述了 C 语言的基本数据类型、运算符和表达式等内容，并在附录中给出了运算符的优先级和结合性总表，其中还讲述了程序设计中“副作用”的概念和实例，避免初学者走入学习误区；第 3 章阐述了可执行语句和程序的执行流程，该章节内容的学习有助于增加程序设计的灵活性；第 4 章是对第 2 章中数据类型内容的进一步扩充，是本书学习的难点和重点，详细介绍了枚举类型、数组类型、结构类型、指针类型、联合（或共用体）类型、typedef 声明的使用方法，数据库表和链表等；第 5 章阐述了自定义函数的概念和方法，并对系统函数进行了小结，该章也对前述章节的内容进行了总结，使 C 语言语法学习进一步规范，同时预处理程序和预处理命令有关内容也归入此章讲解，这一章节也是本书学习的

难点和重点；第 6 章对文件操作函数进行了讲解和小结，讲述了使用 C 语言建立和读写文件的方法；第 7 章给出了 C 语言程序设计的一个具体应用实例，介绍了用 C 语言进行软件设计的一般方法，对实例进行了较为详细的分析和介绍，并给出了完整的程序清单；第 8 章介绍了排序、查找的概念和常用算法，以及评价算法好坏的一般方法，对各算法进行了具体实现和评价，与前述章节相比，讲授时可以作为选学内容来学习。三套全国计算机等级考试二级 C 语言程序设计笔试模拟试卷，给出了相应的参考答案，也可选择相应题目，作为课后习题使用。

书中带有“*”的章节，初次学习时，可以暂不讲授，作为进一步学习之用，或者放在课程后期讲授。本书应用于理工类本科教学时，讲授学时和实践学时一般总体控制在 70 学时左右，可酌情适当延长或缩短学时数。

本课程的学习，目的在于使学生全面掌握 C 语言程序设计的基本内容，掌握典型的程序设计形式，并初步学会运用 C 语言进行大型程序和工程软件设计的方法。同时为进一步学习面向对象编程语言 Visual C++，利用 Visual C++ 进行可视化程序设计，打下良好的基础。

书中内容若有不当之处，敬请读者批评指正。

作者

2008 年 11 月

目 录

第1章 C语言程序设计概述	1
1.1 C语言的形成与发展	1
1.2 程序的基本概念	2
1.2.1 数据结构	2
1.2.2 算法	3
1.2.3 C程序的基本成分	3
1.2.4 程序设计的基本步骤	4
1.3 运行一个C程序	4
1.3.1 一个C程序	4
1.3.2 编辑C程序	5
1.3.3 运行C程序	6
本章小结	7
习题	7
第2章 基本数据类型与表达式	8
2.1 基本数据类型	8
2.1.1 数据类型	8
2.1.2 整型数据	8
2.1.2.1 整型数据	8
2.1.2.2 整型常量	9
2.1.2.3 整型变量	9
2.1.2.4 整型常量的类型与取值范围	10
2.1.3 实型数据	11
2.1.3.1 实型数据	11
2.1.3.2 实型常量	12
2.1.3.3 实型变量	12
2.1.4 字符型数据	13
2.1.4.1 字符常量	13
2.1.4.2 字符变量	13
2.1.4.3 字符串常量	14
2.2 运算符及其表达式	14
2.2.1 算术运算符及其表达式	15

2.2.1.1 算术运算符	15
2.2.1.2 算术表达式	15
2.2.2 赋值运算符及其表达式	17
2.2.2.1 赋值运算符	17
2.2.2.2 赋值表达式	17
2.2.3 关系运算符及其表达式	19
2.2.3.1 关系运算符	19
2.2.3.2 关系表达式	19
2.2.4 逻辑运算符及其表达式	20
2.2.4.1 逻辑运算符	20
2.2.4.2 逻辑表达式	20
2.2.5 条件运算符及其表达式	21
2.2.6 逗号运算符及其表达式	22
2.2.6.1 逗号运算符	22
2.2.6.2 逗号表达式	22
*2.2.7 按位运算符及其表达式	22
2.2.7.1 按位运算符	22
2.2.7.2 按位运算表达式	23
*2.2.8 一些常用的计算函数	24
本章小结	27
习题	27
第3章 程序执行流程	29
3.1 顺序执行流程	29
3.2 选择执行流程	29
3.2.1 if 语句	29
3.2.2 switch 语句	31
3.3 循环执行流程	33
3.3.1 while 语句	33
3.3.2 do-while 语句	34
3.3.3 for 语句	34
3.4 跳转执行流程	37
3.4.1 break 语句	37
3.4.2 continue 语句	37
3.4.3 goto 语句和加标号的语句	38
本章小结	38
习题	38
第4章 数据类型的扩展	40
4.1 枚举类型	40

4.2	数组	43
4.2.1	一维数组	43
4.2.2	多维数组	44
4.2.3	字符数组与字符串	45
4.3	结构	49
4.3.1	结构与结构变量	49
4.3.2	结构数组	52
4.4	指针	53
4.4.1	指针变量	54
4.4.2	指针与结构	57
*4.4.3	指针数组和指向指针的指针	59
4.5	联合	62
*4.6	结构的位成员	63
*4.7	不完全类型	64
4.8	类型说明符的别名	65
	本章小结	66
	习题	67
第5章	函数	68
5.1	系统函数与自定义函数	68
5.1.1	系统函数	68
5.1.1.1	系统函数小结	68
5.1.1.2	printf 函数	70
5.1.1.3	scanf 函数	73
5.1.2	函数的定义与调用	75
5.1.2.1	变量声明小结	75
5.1.2.2	函数的定义与调用	78
*5.1.3	指向函数的指针	82
5.2	函数的参数与数据类型	84
5.2.1	函数调用与函数参数	84
5.2.2	基本数据类型数据作为函数参数	85
5.2.3	数组数据作为函数参数	86
5.2.4	结构或联合数据作为函数参数	90
5.2.5	指针类型数据作为函数参数	92
5.3	存储类说明符	94
5.3.1	源文件和源程序	94
5.3.2	标识符的存在期和可见性	96
5.3.3	存储类说明符	98
5.3.3.1	用于外部级变量声明的存储类说明符	99

5.3.3.2 用于内部级变量声明的存储类说明符	101
5.3.3.3 在函数声明中使用存储类说明符	102
*5.3.4 命名空间	102
*5.4 类型限定符	103
*5.5 函数的递归调用	104
5.6 预处理程序和预处理命令	108
5.6.1 宏定义的使用	109
5.6.2 文件包含命令	111
*5.6.3 条件编译命令	112
*5.7 一些与字符串有关的常用系统函数	116
本章小结	122
习题	122
第6章 文件操作函数	124
6.1 文件读写实例	125
6.1.1 文件的建立	125
6.1.2 文件的读写	130
6.2 常用的文件操作函数	136
本章小结	139
习题	139
第7章 C语言程序设计应用实例——袖珍电子词典上的名片程序	140
7.1 程序总体结构	140
7.2 权限验证和主菜单模块	141
7.3 录入名片模块	142
7.4 查找名片模块	143
7.5 程序清单及注释	144
本章小结	149
习题	149
*第8章 数据的排序与查找	150
8.1 排序	150
8.1.1 算法好坏的评价方法	150
8.1.2 排序的概念	151
8.1.3 冒泡排序	152
8.1.4 其他排序方法	157
8.1.4.1 直接插入排序	157
8.1.4.2 选择排序	159
8.1.4.3 快速排序	160
*8.1.4.4 二路归并排序	162

8.2 查找	166
本章小结	169
习题	170
附录A 模拟试卷及相关答案	171
全国计算机等级考试二级C语言程序设计笔试模拟试卷(一)	171
全国计算机等级考试二级C语言程序设计笔试模拟试卷(二)	181
全国计算机等级考试二级C语言程序设计笔试模拟试卷(三)	191
部分习题及模拟题参考答案	201
第2章习题参考答案	201
第3章习题参考答案	201
第4章习题参考答案	205
第5章习题参考答案	209
第6章习题参考答案	211
第8章习题参考答案	212
全国计算机等级考试二级C语言程序设计笔试模拟试卷(一)参考答案	217
全国计算机等级考试二级C语言程序设计笔试模拟试卷(二)参考答案	217
全国计算机等级考试二级C语言程序设计笔试模拟试卷(三)参考答案	218
附录B 相关知识	219
运算符的优先级和结合性	219
ASCII 字符编码表	221
C 语言语法概要	222
参考文献	226

第1章 C语言程序设计概述

1.1 C语言的形成与发展

C程序设计语言作为一种高级语言，最初是随着UNIX操作系统的开发需要而出现的。1969年，Kenneth Thompson在PDP-7小型计算机上研发UNIX，他是美国电话电报公司(AT&T)所属贝尔实验室(Bell labs)的成员。当时的UNIX是用汇编语言写的，由于发展的需要，Thompson试图用高级语言来实现，这样1970年他写了B语言，这是一种没有类型的语言，不能编译成可执行代码，产生的代码执行速度较慢，不适于开发操作系统。1971年，与Thompson在同一个实验室的Dennis M. Ritchie开始在PDP-11上开发B语言的编译程序，它能够直接产生可执行代码，这种语言“新B”最终被称做C，它对B进行了改进，引入了诸如数据类型等新概念。1972年在用C重写UNIX时，由于在解决怎样运行基本的合作进程，即如何在进程之间进行切换的问题上遇到了困难，Ritchie在C中加入了“结构”这一类型，并使编译程序能够支持，这样到了1973年初，C语言的开发已基本完成。1978年，Brian W. Kernighan和Dennis M. Ritchie合著了《C程序设计语言》一书，该书成了事实上的C标准。

从设计思想上来看，程序设计语言ALGOL对C语言的形成有很大影响。ALGOL是一个命令式计算机语言的家族，最早在20世纪50年代中期就已开发，有三个主要分支：ALGOL 58、ALGOL 60和ALGOL 68，它后来成为报告算法的实际标准。ALGOL语言不是为某种特定机器而设计的，是纯粹面向描述计算过程的语言，主要用于科学计算，比较抽象，几乎不能用于具体的编程任务。1963年伦敦大学和剑桥大学合作，结合ALGOL 60的设计思想，进一步开发了CPL(Combined Programming Language)语言，它的目的是用于具体的编程任务，然而由于体积庞大、难以学习和执行，没有流行起来。为了改进CPL，1967年Martin Richards在剑桥大学设计了BCPL(Basic Combined Programming Language)语言，它对CPL进行了简化，但体积仍然有点大，而且比较抽象。1970年Thompson写的B语言，是CPL的又一个较大的简化版，而Ritchie对C的开发，进一步改进了B，其中对数据类型的引入，直接受到了ALGOL 68的影响。

1983年，美国国家标准协会ANSI成立X3J11委员会，着手制定一个与机器无关的，无二义性的C语言标准。1989年12月，该委员会公布了正式的C语言标准ANSI X3.159-1989。1990年，该标准被国际标准化组织ISO采纳为国际标准ANSI/ISO 9899-1990。之后，C标准经过多次修订，于1999年，ANSI与ISO又发布了新版C语言标准ANSI/INCITS/ISO/IEC 9899-1999，该标准详细说明了C语言程序的形式和解释规则。包括：C程序的表示方法；C语言的语法和限定；C语言的语义规则。新版标准发布之后，于2001年9月和2004年11月，

ISO 又两次发布了对该标准的修订，对其中的打印错误和不确切内容作了更正。2004年7月，ISO 还发布了支持嵌入式处理器的扩展 C 程序设计语言标准 ISO/IEC TR 18037:2004，该标准在 ANSI/INCITS/ISO/IEC 9899-1999 标准的基础上，进行了一系列的扩展，用于支持嵌入式处理器所具有的共同特征。其内容包括对处理器定点运算、内存空间、基本 I/O 的支持，以及对 ANSI/INCITS/ISO/IEC 9899-1999 标准的相应变更，这使得 C 语言在嵌入式系统领域的应用得以规范化。由于 C 语言的应用，特别是在网络编程中所带来的问题，造成软件自身弱点的暴露，C 标准依然在不断的修订中，譬如 ISO/IEC TR 24731-1:2007 标准在 2007 年 8 月发布，用于对 C 系统函数库进行扩展和改进，这使得新版 C 程序具有更高的安全性。

C 语言数据类型多样，易于实现各种数据结构；且运算符丰富，有多样的可执行语句，能够形成各种表达式和不同的程序执行流程，程序设计灵活；还可进行涉及计算机硬件的软件开发等。具有语言简洁、使用方便、目标程序执行效率高、可移植性好的特点，因而在各类计算机及不同的操作系统下得以广泛的应用。C 语言在工程领域，特别是在嵌入式系统领域的应用尤其广泛，它也是设计操作系统的基本语言。作为高级语言，它被典型地用于课堂讲授。而对于目前流行的面向对象程序设计语言 C++ 而言，对它的学习起到了良好的基础作用。

目前教学上经常使用的 C 语言程序设计集成开发环境有 Microsoft Visual C++ 6.0 和 Turbo C 2.0 等，环境中包括了 C 语言编译程序。本书使用这两种软件来进行具体的 C 语言程序设计讲解。

1.2 程序的基本概念

高级语言源程序是用于完成具体编程任务的语句序列，用 C 语言编写的程序是该类源程序的一种具体形式。关于计算机程序，人们在长期的编程实践中总结出这样的观点：程序 = 数据结构 + 算法。那么什么是数据结构？什么是算法呢？

1.2.1 数据结构

程序设计语言的基本功能是描述数据和对数据的运算。数据是对客观事物的符号表示，在计算机科学中是指所有能输入到计算机中并被计算机程序处理的符号的总称。如在数值计算问题中，计算机处理的整数或者实数都是数值型数据。数据的基本单位是数据元素。数据元素可以是简单的，如一个整数；也可以是复杂的。复杂的数据元素可以由若干个数据项组成，数据项是数据不可分割的最小单位。如一个描述学生的数据元素可以包含学号、姓名、系名等成员，每一个成员即为一个数据项。

在任何问题中，数据元素都不是孤立存在的，在它们之间存在着某种关系，这种数据元素相互之间的关系被称为结构。数据结构是指具有结构的数据元素的集合，它是数据元素的有限集，和在该有限集上的关系的集合。例如，含有 26 个英文字母的字母表： (A, B, \dots, Z) 是一个线性表结构，表中的数据元素都是英文字母。学生花名册也可以视为一个线性表结构，表中的每一行可以视为一个数据元素，代表一个学生的情况。线性表中的数据元素之间存在着一对一的关系，即除了第一个和最后一个数据元素之外，其他数据元素都是首尾相接的，如上述英文字母表中第一个元素 A 的前面没有元素，后面有一个元素 B；中间的每个元素之

前都有且仅有一个元素，之后也有且仅有一个元素；最后一个元素 Z 前面有一个 Y 元素，之后没有元素。线性表是一种线性结构。

1.2.2 算法

一般来讲，算法是一个明确定义的可执行指令的有限序列，用来完成某一任务，是对任务完成步骤的一种描述，该任务从一个初始状态依次进行下去，通过一连串明确定义的连续状态，最后终结在一个结束状态。流程图是用图形表述的算法，图 1.1 用流程图表述了计算 10 个数平均值的算法，可以分为 4 步：①输入 10 个数；②计算 10 个数的平均值；③输出显示平均值；④结束算法。数学公式是用数学语言表述的算法，数学公式 $S = \pi r^2$ 表述了计算圆面积的算法，其输入为 r ，输出为 S 。计算机程序是用程序语言表述的算法。

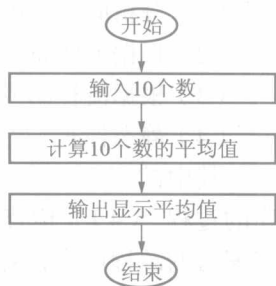


图 1.1 算法的流程图

算法具有五种特性：①有穷性：即算法必须在执行有穷步骤后结束；②确定性：是指每一条指令表示一个或多个确定的无二义性的操作，使算法对于相同的输入只能得到相同的输出；③输入：有 0 个或多个输入；④输出：有 1 个或多个输出，输出量与输入量具有某种特定的关系；⑤有效性（或可行性）：要求算法的指令必须具有可执行性，并能得到确定的结果。

通过算法可以建立和操作数据结构。例如，通过算法建立一个花名册，即一个学生线性表，然后通过算法可以登录、查询该花名册，等等。

1.2.3 C程序的基本成分

C 语言在描述数据时，引入了数据类型的概念，数据类型是建立数据结构的基础；而对数据的运算过程或者说对数据的操作过程，主要是通过函数来描述的。

数据类型是一个值的集合和定义在这个值集上的一组操作的总称。如整数类型，其值集可以定义为区间 $[-\text{maxint}, \text{maxint}]$ ，定义在其上的一组操作可以是：加、减、乘、整除和求余数等。其中 maxint 代表该区间最大的整数，由于计算机存储空间的有穷性，该值不能定义为无穷大。程序中的每个变量、常量或表达式，都有一个它所属的确定的数据类型。常量、变量和表达式的概念类似于数学中的概念，但程序设计中的概念与数学中的概念一定程度上又有区别，它们的详细定义将在第 2 章中介绍。

C 语言的基本成分包括常量、变量、表达式、语句和函数等。函数包括了若干语句，每一条语句都包含了一定的功能，用于使计算机完成一定的操作，如由表达式构成的语句可以完成相应的运算。其他程序设计语言中还有子程序概念，实际上 C 语言中函数的概念就包括了子程序的概念。这里函数的概念与数学中函数的概念既类似又有区别，我们将在第 5 章中对函数进行较完整的介绍。从整体上看，C 程序主要由若干函数构成，而函数由若干 C 语句构成。

1.3 节的例 1.1 是一个简单的 C 程序，其主要成分是一个 main 函数，大括号内包括两条语句，每条语句用“;”结束，这两条语句是 main 函数的组成部分。程序执行时，将按由上到下的顺序依次执行这两条语句，完成显示“你好!”的功能。

1.2.4 程序设计的基本步骤

程序设计是设计、编制和检查调试程序的过程。程序设计的一般步骤包括：

- (1) 分析问题并确定解题思路；
- (2) 根据解题思路画出流程图或结构图；
- (3) 根据流程图或结构图编制源程序；
- (4) 静态检查，上机调试，修改源程序，最后确定源程序。

其中的问题是指需要利用计算机解决的实际问题。(1)(2)为设计过程，(3)为编制程序的过程，(4)为检查调试排错的过程。静态检查是通过阅读程序来发现错误；上机调试是将编写的源程序在计算机上进行编译，生成可执行代码，运行可执行代码，在上述过程中发现错误、改正错误的过程。结构图用于大型程序设计，它表述了程序结构，表述了各程序模块间的关系，其中对于程序模块用其名字来标记，关于程序模块第7章中还要做进一步的叙述。图1.2是图书管理系统的简化结构图。



图 1.2 图书管理系统的简化结构图

1.3 运行一个 C 程序

1.3.1 一个 C 程序

本书将以 C 程序设计为例，介绍高级语言结构化程序设计的基本方法。

【例 1.1】编写一个程序 aaa.c，其内容用于在计算机上显示“你好！”。将其保存在 C 盘根目录下，然后再将其转化为可执行文件并运行之。

```
#include "stdio.h"
main()
{ printf("你好! \n");
  getchar();
}
```

该源程序中，main 为主函数，每一个 C 程序都包含一个主函数。用 {} 扩起来的部分是主函数的函数体，它决定了主函数所要完成的操作。这里主函数体内包括两条语句，语句用“;”结束。计算机执行程序时，一般运行的第一条语句是主函数函数体中的声明语句后面的第一条可执行语句，关于声明语句将在后续章节中介绍。这里主函数中没有声明语句，运行的第一条语句是 printf("你好!\n");，其中 printf("你好!\n") 为一格式输出函数，是系统函数，在 C 语言编译环境中已定义好，用户可以直接拿来使用。该语句为函数调用语句，执行时将程序的执行控制从主函数传递给被调用的 printf 系统函数，该系统函数执行完后，程序执行控制再返回给主函数，主函数接着执行下一条语句 getchar();。printf 函数中的双引号及其中的汉字或字符为该函数自变量，双引号中的内容将被原样输出到屏幕；\n 为回车换行符，表示屏幕上当前行显示完“你好！”后，光标定位在下一行行首。getchar() 也是系统函

数，为**字符输入函数**，它暂停程序的执行，等待从键盘接收一个字符，该函数没有自变量。函数体中的语句执行完后，程序结束。这样本程序完成的操作是：首先在屏幕上显示“你好！”，接着光标定位在下一行行首并暂停程序的执行，然后当用户从键盘上输入任一字符后，程序结束。除了主函数外编程者还可以根据需要自定义若干其他函数，这将在后续章节中介绍。

`#include"stdio.h"` 为文件包含命令，是 `#include` 类**预处理命令**，此命令在该源程序编译前由预处理程序处理，预处理程序用磁盘文件 `stdio.h` 中的 C 语句取代该预处理命令，即将 `stdio.h` 文件的内容包含到本程序中，预处理命令以 `#` 开始。C 语言程序设计集成开发环境中包括了预处理程序。`stdio.h` 是扩展名为 `.h` 的文件，在 C 语言中叫做**头文件**，在 C 语言程序设计集成开发环境中已建立好，其中有关于 `printf` 函数和 `getchar` 函数的声明语句。由于该程序使用了 `printf` 和 `getchar` 等系统函数，这些**系统函数应先声明后使用**，因而程序前部须使用这条对 `stdio.h` 文件的包含命令。

编制出的 C 语言源程序将以文件形式存储在磁盘上，这里源程序文件的名字是 `aaa.c`，其中 `.c` 是 C 语言源程序文件的扩展名，C 语言源程序文件也叫做**C 程序源文件**。源程序需经过编译程序编译，进一步生成相应的可执行程序（扩展名为 `.exe` 的文件），计算机才能执行，这一过程可以使用 Microsoft Visual C++ 6.0 或 Turbo C 2.0 来完成。Microsoft Visual C++ 6.0 软件的安装方法与一般的微软软件安装方法相同，首先将 Microsoft Visual Studio 6.0 Enterprise 安装光盘插入光驱，然后在“我的电脑”中双击该光驱图标，打开后再双击 `Setup` 图标，按屏幕提示安装即可；Turbo C 2.0 软件的安装比较简单，只要将该软件拷贝或解压到 `C:\TURBOC2` 文件夹下即可。

1.3.2 编辑C程序

编制出的源程序，首先需通过键盘输入到计算机中，这是编辑建立 C 程序源文件的过程。

在 Windows 操作系统中，使用 Microsoft Visual C++ 6.0 编辑 C 语言源程序的操作方法是：“开始”菜单 → 程序选项 → Microsoft Visual Studio 6.0 → Microsoft Visual C++ 6.0 → Microsoft Visual C++ 6.0 窗口界面 → “文件”菜单 → 新建选项 → “文件”标签 → C++ Source File → 在“文件”文本框中输入文件名 `aaa.c` → 在“目录”文本框中输入磁盘保存路径 `C:\` → 确定按钮 → 在 `C:\aaa.c` 编辑窗口中输入例 1.1 的 C 源程序 → 保存。打开后的 Microsoft Visual C++ 6.0 窗口界面如图 1.3 所示。从磁盘上打开 C 语言源程序文件的方法同一般微软软件方法，这里不再赘述。

使用 Turbo C 2.0 编辑 C 语言源程序的操作方法是：选择 `C:\TURBOC2` 文件夹 → 双击 `TC` 图标 → Turbo C 2.0 窗口界面 → 按 `Enter` 键选定 `File` 菜单 → 移动光标键选定 `New` 选项 → `Edit` 窗口中出现闪动的光标 → 从键盘输入例 1.1 的 C 源程序 → 按 `F10` 功能键使光标移至菜单行 → 移动光标键选定 `File` 菜单 → `Save` 选项 → 输入磁盘保存路径和文件名 `C:\aaa.c` → 按 `Enter` 键确定。打开后的 Turbo C 2.0 窗口界面如图 1.4 所示，窗口最下面一行为七种功能键的用法。从磁盘上打开 C 语言源程序文件的方法为：选择 `File` → `Load` 选项，在出现的文本框中输入文件路径和文件名，按 `Enter` 键即可。

Turbo C 2.0 只能显示英文，且不能使用鼠标进行操作，源程序中的“你好”可以改写为“Hello”。

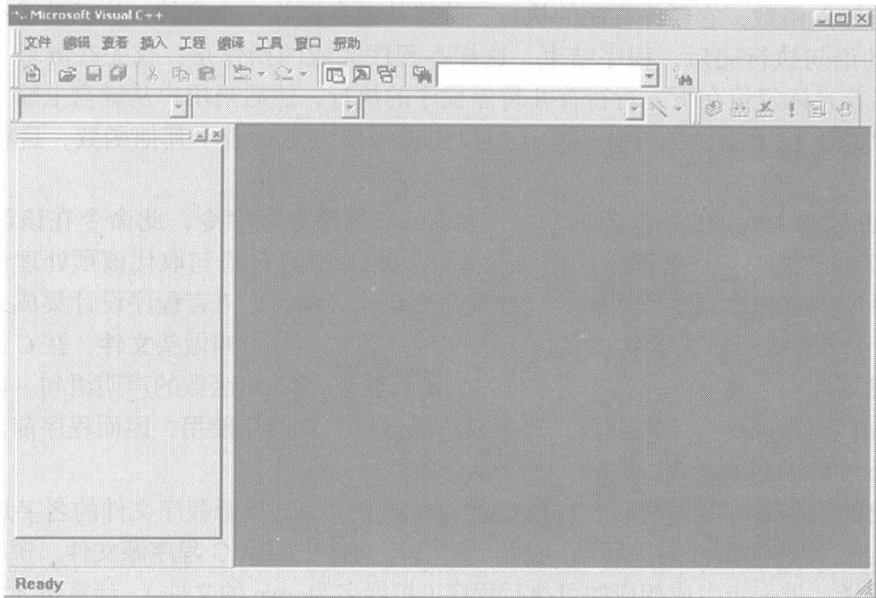


图 1.3 Microsoft Visual C++ 6.0 窗口界面

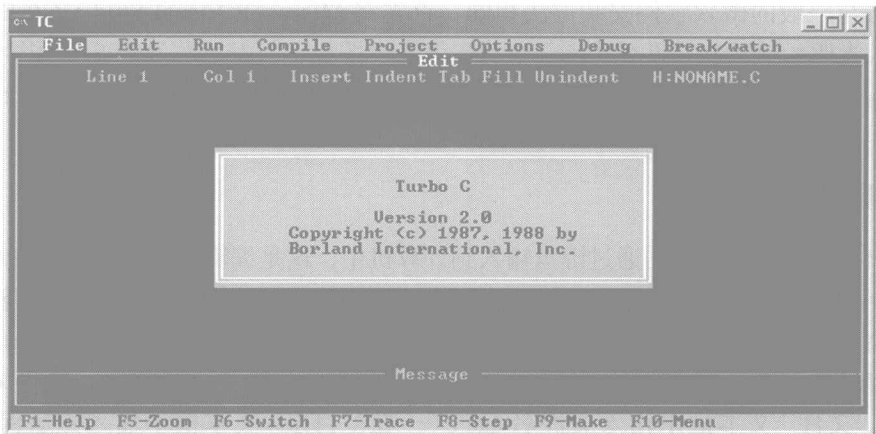


图 1.4 Turbo C 2.0 窗口界面

1.3.3 运行C程序

上述编辑好的源程序经过“构建”，将生成可执行文件。“构建”包括：对 .c 源文件的预处理和编译，以及将生成的 .obj 文件与系统程序（如 C 语言库函数）进行链接，生成可执行的 .exe 程序文件。其中预处理过程，由前面提到的预处理程序来完成。生成后的可执行程序，就可以直接“运行”了。

在 Microsoft Visual C++ 6.0 中的操作方法为：①编译菜单 → 构建 aaa.exe → 是；②编译菜单 → 执行 aaa.exe。构建或运行完后的程序，如果有错误，直接在 aaa.c 源程序编辑窗口中修改即可，修改后再重复上述过程，直至程序正确为止。在构建完成后，语法错误会显示在编辑窗口下部的窗口中，用鼠标单击相应的错误信息，在编辑窗口中将有箭头指示错误的位置。

在 Turbo C 2.0 中的操作方法为：① Compile 菜单 → Make EXE file C:AAA.EXE 选项；② Run 菜单 → Run 选项。程序运行完后，如果要看运行结果，可以按 Alt + F5 组合键，看完后再按任意键即可回到 Edit 窗口。构建后的程序，如果有错误，下部 Message 窗口中会有提示信息，移动光标到相应的错误信息，Edit 窗口中对应的错误将点亮，按 F6 键可以回到 Edit 窗口，在 Edit 窗口中进一步修改源程序，然后再重复上述过程，直到程序正确为止。

学习 C 语言程序设计，涉及两个概念，一个是“编译程序”，另一个是“内存”，这两个概念自始至终都在使用，因而有必要搞清楚。这里我们用类比的方法，加深一下对“编译程序”的理解，“内存”的概念在后续章节中还要进一步论述。我们来看一下这样一个过程，小李想请 Johnson 做一件事情，小李是中国人不懂英语，Johnson 是英国人不懂汉语，于是小李请小何来做翻译，小何既懂汉语又懂英语。小李把办事的具体步骤用汉语写在纸上，给了小何，小何将其翻译成英语写在了另一张纸上，然后他把写英文的纸交给了 Johnson，Johnson 看懂了意思并将其记忆在脑中，然后帮小李完成了这件事情。这里涉及小李、小何、Johnson、写汉语的纸、写英文的纸，可以将上述过程粗略地类比为我们上机运行程序的过程，小李是编程序的人，写汉语的纸是源程序，小何是编译程序，写英文的纸是可执行程序，Johnson 是计算机，而“内存”就是 Johnson 脑中的记忆区域。

本章小结

本章阐述了 C 语言和 C 标准的形成、发展和现状。介绍了 C 语言的特点和应用及教学上经常使用的 C 语言程序设计集成开发环境。阐述了程序的基本概念和 C 程序的基本成分。介绍了程序设计的一般步骤，详述了一个 C 程序的简单实例，及其在 Turbo C 2.0 和 Microsoft Visual C++ 6.0 两种环境中的编辑、构建和运行方法。

习 题

1. 什么是数据结构？什么是算法？它们之间有什么关系？
2. 请写出程序设计的基本步骤。
3. C 程序设计语言的基本成分有哪些？从程序结构上看，C 程序的主要构成成分是哪一种？计算机执行程序时，执行的第一条语句是哪一条？
4. 什么是数据类型？试举例说明。
5. 使一个 C 语言源程序成为可执行程序，并运行该可执行程序，一般需经过哪些步骤？