



普通高等教育“十三五”规划教材
应用型特色规划教材

C语言程序设计案例教程

C Programming Language

© 肖利群 石 彬 主编

中国工信出版集团



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
<http://www.phei.com.cn>

普通高等教育“十三五”规划教材
应用型特色规划教材

C 语言程序设计案例教程

肖利群 石 彬 主编

電子工業出版社
Publishing House of Electronics Industry
北京 · BEIJING

内 容 简 介

C 语言是一门入门语言,是学习其他计算机语言的基础。本书共 10 章,内容包括:初识 C 语言,C 语言案例概述,基本数据类型、运算符和表达式,数据的输入/输出,结构化程序设计,数组,函数,自定义数据类型,指针,文件。本书内容的讲解注重理论联系实际,以教师工资管理系统案例贯穿 C 语言各知识点的讲解,并在附录中给出教师工资管理系统的完整代码。本书包含精选的习题,学生通过练习不仅能掌握 C 语言相关理论知识,还能进一步训练编程能力。

本书可作为大学本科、高职高专学生“C 语言程序设计”课程的教学用书,同时也可作为参加计算机等级考试的人员和自学者的参考书。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有,侵权必究。

图书在版编目(CIP)数据

C 语言程序设计案例教程 / 肖利群, 石彬主编. —北京: 电子工业出版社, 2018.9

ISBN 978-7-121-34746-7

I. ①C… II. ①肖… ②石… III. ①C 语言—程序设计—高等学校—教材 IV. ①TP312.8

中国版本图书馆 CIP 数据核字(2018)第 159485 号

策划编辑: 戴晨辰

责任编辑: 戴晨辰

印 刷: 涿州市京南印刷厂

装 订: 涿州市京南印刷厂

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编: 100036

开 本: 787×1092 1/16 印张: 19.25 字数: 493 千字

版 次: 2018 年 9 月第 1 版

印 次: 2018 年 9 月第 1 次印刷

定 价: 49.80 元

凡所购买电子工业出版社图书有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,联系及邮购电话:(010)88254888, 88258888。

质量投诉请发邮件至 zltz@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

本书咨询联系方式: dcc@phei.com.cn。

前言

Preface

C 语言是一门通用的计算机编程语言,它功能强大、使用灵活、应用广泛、目标程序效率高、可移植性好,适用于编写系统软件,深受编程人员欢迎。

作者自 2014 年起进行教学改革,变“单元制”教学为“理实一体”教学,更加注重知识点的学习及学生实践能力的培养。因此,通过多年的教学,作者积累了大量的“案例+项目”实训经验,特别是在“C 语言程序设计”课程中积累了较多的优秀案例、教学成果。

本书根据教学改革的变化进一步调整、完善了 C 语言的知识点和体系,以便更好地适应项目教学的授课要求。本书的编写目标是:集理论教学和实训教学于一体,针对计算机学科交叉、实践性强等特点,改变现有专业教材将各类知识点拆分、独立讲解的现状,编写理论讲解与实训案例相结合的教材,供计算机科学与技术、软件工程、网络工程、物联网、数字媒体等专业的学生使用。本书主要有以下特点。

(1)涉及理论、概念等内容讲解时,注重穿插学习方法的介绍,结合应用型本科学生的特点,关注内容的实用性和综合性。删减以往同类图书中较刻板的理论知识点,将更多的学时和内容重点放在程序设计方法、设计技能及设计过程的阐述上。

(2)突出“理实一体”的教学方法和过程展示,以教师工资管理系统案例贯穿 C 语言各知识点的讲解。将 C 语言的基本语法与知识点内嵌在案例中,并站在学生的角度,由浅入深、由表及里地罗列与讲解 C 语言相关知识,使学生循序渐进地学习 C 语言。

(3)语言通俗易懂,难理解之处都配有图示。案例配有完整可用的程序代码,以便帮助学生在知识学习的同时,逐步提高编程能力。

(4)在章节编排上力求循序渐进,在语言描述上力求准确、易懂,在案例实现上力求实用。

(5)包含配套视频、源代码、教学课件等资源,读者可登录华信教育资源网(www.hxedu.com.cn)注册后免费下载。为帮助学生更好地学习相关知识点,本书还提供在线答疑(可从华信教育资源网获取答疑地址)。

本书由肖利群、石彬主编,第 3、4、5、6 章由肖利群编写,第 1、2、7、8、9、10 章由石彬编写,还要感谢李志、李爱华、谢治军、杨开林、郑丽娟、张德发、王邦千等老师对本书付出的辛勤劳动。本书在编写过程中,得到了四川工商学院领导的大力支持,同时得到了许多同行、专家的指导与帮助,在此对他们表示衷心的感谢。

本书难免存在不足之处,欢迎各位专家和广大读者提出宝贵意见。

作者

2018 年 7 月于四川工商学院



Contents

第1章 初识C语言	1
1.1 C语言的发展历史和特点	1
1.1.1 C语言的起源与发展	1
1.1.2 C语言的特点	2
1.2 C语言程序的基本结构	4
1.2.1 第一个C语言程序: Hello world!	4
1.2.2 基本结构	4
1.3 C语言的集成开发环境	5
1.3.1 主流开发工具介绍	5
1.3.2 VC++ 6.0 环境介绍	6
1.3.3 C语言程序的编译运行	8
1.4 本章小结	12
1.5 习题	12
第2章 C语言案例概述	14
2.1 案例功能描述	14
2.1.1 输入记录模块	14
2.1.2 查询记录模块	15
2.1.3 更新记录模块	15
2.1.4 输出记录模块	15
2.2 案例总体设计	15
2.2.1 功能模块设计	15
2.2.2 数据结构设计	18
2.2.3 函数功能描述	19
2.3 案例运行结果	20
2.4 本章小结	24
第3章 基本数据类型、运算符和表达式	25
3.1 C语言的字符集和词汇	25

3.1.1	C 语言的字符集	25
3.1.2	C 语言的词汇	26
3.2	常量和变量	28
3.2.1	常量	28
3.2.2	变量	29
3.3	数据类型	30
3.3.1	整型数据	31
3.3.2	实型数据	33
3.3.3	字符型数据	35
3.4	运算符与表达式	39
3.4.1	算术运算符与算术表达式	40
3.4.2	自增、自减运算符与表达式	40
3.4.3	关系运算符与关系表达式	41
3.4.4	逻辑运算符与逻辑表达式	42
3.4.5	赋值运算符与赋值表达式	43
3.4.6	逗号运算符与逗号表达式	44
3.4.7	条件运算符与条件表达式	45
3.4.8	sizeof 运算符与 sizeof 表达式	46
3.5	数据类型的转换	47
3.6	本章小结	48
3.7	习题	49
第 4 章	数据的输入/输出	54
4.1	输入/输出概述	54
4.2	非格式化字符的输入/输出	54
4.3	格式化数据的输出	55
4.4	格式化数据的输入	57
4.5	本章小结	59
4.6	习题	59
第 5 章	结构化程序设计	66
5.1	算法	66
5.1.1	算法的概念	66
5.1.2	结构化程序设计的三种基本结构	67
5.1.3	流程图	67
5.2	if 分支语句	69
5.2.1	if 语句中的条件表示	70
5.2.2	if 语句的三种形式	71
5.2.3	复合语句在分支语句中的应用	75
5.2.4	if 语句的嵌套	76

5.2.5	条件运算符与条件表达式	78
5.3	switch 分支语句	79
5.3.1	switch 语句	79
5.3.2	分支结构程序举例	83
5.4	循环结构	86
5.4.1	while 语句	86
5.4.2	do...while 语句	91
5.4.3	for 语句	93
5.4.4	break 语句和 continue 语句	99
5.4.5	多重循环结构	101
5.4.6	循环结构程序举例	104
5.5	本章小结	110
5.6	习题	111
第 6 章	数组	138
6.1	一维数组	138
6.1.1	一维数组的定义和初始化	138
6.1.2	一维数组的引用	142
6.1.3	一维数组程序举例	144
6.2	二维数组	150
6.2.1	二维数组的定义和初始化	150
6.2.2	二维数组的引用	153
6.2.3	二维数组程序举例	153
6.3	字符数组	156
6.3.1	字符数组的定义	156
6.3.2	字符数组的初始化	156
6.3.3	字符数组的赋值	157
6.4	字符串	158
6.4.1	字符串常量	158
6.4.2	利用字符串对字符数组初始化	158
6.4.3	字符数组与字符串的输入、输出	159
6.4.4	字符串处理函数	163
6.4.5	字符串程序举例	165
6.5	本章小结	168
6.6	习题	168
第 7 章	函数	179
7.1	函数的定义与调用	179
7.1.1	函数的分类	180
7.1.2	函数的定义	180

7.1.3	函数的调用	182
7.2	函数的参数传递	183
7.3	函数的调用方式	185
7.3.1	函数的嵌套调用	186
7.3.2	函数的递归调用	186
7.4	变量的作用域	189
7.4.1	变量的存储类型	189
7.4.2	全局变量与局部变量	190
7.5	编译预处理	192
7.5.1	文件包含	192
7.5.2	宏定义与替换	193
7.5.3	条件编译	195
7.6	本章小结	195
7.7	习题	196
第8章	自定义数据类型	204
8.1	结构体	204
8.1.1	结构体的定义	204
8.1.2	结构体变量的定义与初始化	205
8.1.3	结构体变量的引用	207
8.2	结构体数组	208
8.2.1	结构体数组的定义与初始化	208
8.2.2	结构体数组元素的引用	209
8.3	共用体	210
8.3.1	共用体的定义	210
8.3.2	共用体变量的定义与初始化	212
8.3.3	共用体变量的引用	212
8.4	本章小结	214
8.5	习题	215
第9章	指针	222
9.1	指针的概念	222
9.1.1	指针与指针变量	222
9.1.2	指针变量的引用	224
9.2	指针的运算	225
9.3	指针与数组	227
9.3.1	指向一维数组的指针	227
9.3.2	指向二维数组的指针	230
9.3.3	字符指针	232
9.4	指针与函数	233

9.4.1	指针和数组名作为函数参数	233
9.4.2	指针作为函数的返回值	240
9.5	链表	243
9.5.1	链表的概念	243
9.5.2	链表的基本操作	245
9.6	本章小结	250
9.7	习题	251
第 10 章	文件	261
10.1	文件概述	261
10.1.1	文件的概念	261
10.1.2	缓冲文件系统与非缓冲文件系统	262
10.1.3	文件指针	262
10.2	文件的打开与关闭	263
10.2.1	打开文件函数	263
10.2.2	关闭文件函数	264
10.3	文件的读与写	264
10.3.1	文件的写函数	265
10.3.2	文件的读函数	268
10.4	其他相关函数	273
10.5	本章小结	276
10.6	习题	276
附录 A	ASCII 码表	282
附录 B	运算符的优先级和结合性	283
附录 C	常用 ANSI C 标准库函数	284
附录 D	教师工资管理系统完整代码	290

第1章 初识C语言

C语言是一门诞生较早的面向过程的高级程序设计语言。从诞生开始，由于其有着其他结构化程序设计语言所没有的优点，而深受广大编程人员的喜爱，并得到广泛使用。在C语言的基础上进行扩展，又衍生出C++、C#等面向对象的程序设计语言。编程初学者多以C语言作为计算机编程学习的第一门语言。

本书将为读者揭开计算机编程的神秘面纱。本章作为本书的第1章，将针对C语言的发展历史和特点、C语言程序的基本结构、C语言程序的编译环境等内容进行详细讲解。

学习目标

- 了解C语言的发展历史和特点
- 掌握C语言程序的基本结构
- 掌握C语言程序的编写方式
- 熟悉开发工具VC++ 6.0的使用方法

1.1 C语言的发展历史和特点

1.1.1 C语言的起源与发展

在C语言诞生之前，计算机系统软件主要是用汇编语言编写的，但汇编语言编写的程序可读性和可移植性较差，已有的高级语言还没有对硬件直接访问的功能。因此，人们急需一种高级语言，使用它可以编写出可读性和可移植性强的程序，又能够直接操作硬件。这就为C语言的诞生提供了条件。

1963年，剑桥大学在ALGOL(Algorithmic Language)的基础上开发出具有处理硬件能力的CPL(Combined Programming Language)。

1967年，剑桥大学的Martin Richards对CPL进行简化，于是产生BCPL(Basic Combined Programming Language)。

1970年，美国贝尔实验室的Ken Thompson以BCPL为基础，设计出简单且接近硬件的B语言(取BCPL的首字母)。

1972年，美国贝尔实验室的D. M. Ritchie在B语言的基础上最终设计出了一种新的语言，他将BCPL的第2个字母作为这种语言的名称，这就是C语言。

随着计算机的发展和普及，C语言受到越来越多的编程人员的喜爱，已经成为使用人数最多的结构化程序设计语言之一。随后，C语言出现了多个版本，不同的版本之间存在一些不一致的地方。因此，美国国家标准学会(ANSI)为C语言制定了一套标准，即ANSI C标准。

1989 年, ANSI 通过的 C 语言标准称为 C89。

1990 年, ISO 组织批准了 ANSI C 成为国际标准, 于是 ISO C (又称 C90) 诞生了。ISO C (C90) 和 ANSI C (C89) 在技术上完全一样。

随后, ISO 在 1994 年、1996 年分别出版了 C90 的技术勘误文档, 更正了一些印刷错误, 并在 1995 年通过了 C90 的技术补充, 对 C90 进行了微小的扩充, 经过扩充后的 ISO C 称为 C95。

1999 年, ANSI 和 ISO 又通过了最新版本的 C 语言标准和技术勘误文档, 该标准称为 C99。

1.1.2 C 语言的特点

C 语言具有如下特点。

(1) 简洁、紧凑、灵活

C 语言的核心内容很少, 只有 32 个关键字, 9 种控制语句; 程序书写格式自由, 压缩了一切不必要的成分。

(2) 表达方式简练、实用

C 语言有一套强有力的运算符, 共 44 个, 可以构造出多种形式的表达式。使用一个表达式就可以实现其他语言需多条语句才能实现的功能。

(3) 具有丰富的数据类型

数据类型越多, 数据的表达能力就越强。C 语言具有现代语言的各种数据类型, 如字符型、整型、实型、数组、指针、结构体和共用体等。可以实现如链表、堆栈、队列、树等各种复杂的数据结构。其中指针的使用可使参数的传递更加简单、迅速, 节省内存。

(4) 具有低级语言的特点

C 语言具有与汇编语言相近的功能和描述方法, 如地址运算、二进制数位运算等, 对硬件端口等资源直接操作, 可充分使用计算机资源。C 语言既具有高级语言便于学习和掌握的特点, 又具有机器语言或汇编语言对硬件的操作能力。因此, C 语言既可以作为系统描述语言, 又可以作为通用的程序设计语言。

(5) 是一种结构化语言, 适合大型程序的模块化设计

C 语言提供编写结构化程序的基本控制语句, 如 if...else 语句、switch 语句、while 语句、do...while 语句等。C 语言程序是函数的集合, 函数是构成 C 语言程序的基本单位, 每个函数具有独立的功能, 函数之间通过参数传递数据。除了用户编写的函数, 不同的编译系统、操作系统还提供大量的库函数供用户使用, 如输入/输出函数、数学函数、字符串处理函数等, 灵活使用库函数可以简化程序设计。

(6) 各种版本的编译系统都提供预处理命令和预处理程序

预处理扩展了 C 语言的功能, 提高了程序的可移植性, 为大型程序的调试提供方便。

(7) 可移植性好

程序从一个环境不经改动或稍加改动就可移植到另一个完全不同的环境中运行。这是因为系统库函数和预处理程序将可能出现的与机器有关的因素与源程序隔离开, 使在不同的 C 语言编译系统之间重新定义有关内容变得容易。

(8) 生成的目标代码质量高

用C语言编写得到的目标代码的运行效率仅比用汇编语言编写的低10%到20%，可充分发挥机器的效率。

(9) 语法限制少，程序设计自由度高

C语言程序在运行时不做如数组下标越界和变量类型兼容性等检查，而是由程序员自己保证程序的正确性。C语言几乎允许所有数据类型的转换，字符型和整型可以自由混合使用，所有类型均可作为逻辑型，还可自己定义新的类型，将某类型强制转换为指定的类型。实际上，这使得程序员有了更大的自主性，可编写出更加灵活、优质的程序。但这也给初学者的学习增加了一定的难度，只有在熟练掌握C语言程序设计后，才能体会到其灵活的特性。

通过上述介绍，我们已经了解了C语言的若干特点。C语言虽然是一种优秀的计算机程序设计语言，但也存在以下缺点，了解这些缺点，才能在实际使用中扬长避短。

(1) C语言程序的错误更隐蔽

C语言的灵活性使得程序员在编写程序时更容易出错，且C语言的编译器不会检查类似的错误。C语言与汇编语言类似，需要在程序运行时才能发现这些逻辑错误。还有一些错误需要程序员重视，例如，将比较运算符“==”写成赋值运算符“=”，在语法上没有错，但这种逻辑错误不易被发现，要找出来往往十分费时。

(2) C语言程序有时难以理解

C语言语法成分相对简单，是一种小型语言。但是，其数据类型多，运算符丰富且结合性多样，使得理解起来有一定的难度。有关运算符和结合性，人们最常说的一句话是“先乘除，后加减，同级运算从左到右”，但是，C语言的使用远比这要复杂。发明C语言时，为了减少字符输入，程序设计常比较简洁，这也使得C语言程序有时难以理解。

(3) C语言程序有时难以修改

考虑到程序规模的大型化或巨型化，现代编程语言通常会提供“类”、“包”之类的语言特性，这样的特性可以将程序分解成更加易于管理的模块。然而，C语言缺少这样的特性，维护大型程序显得比较困难。

早期的计算机语言有BASIC语言、Fortran语言、ALGOL、COBOL和Pascal语言等，如今除非是在既有的软件系统使用这些语言，其他时候很少使用。

现在的软件开发常使用C++语言和Java语言，在Web应用软件开发时则会使用JSP语言和PHP语言等。随着面向对象技术的广泛普及，Java语言受到更多人的青睐，这是由于Java语言具有编程效率高，可降低软件开发成本，不需要考虑存储的分配与回收等特点，编写出来的程序更具有健壮性，但也需在一定程度上付出运行效率的代价。C++语言介于C语言和Java语言之间，也是面向对象的计算机语言，具有编程效率高和运行速度快的特点。

C语言是一种过程性的语言，职业程序员或软件开发人员应该认真学习该语言。这是因为，C语言可以代替机器语言或汇编语言编写运行速度快的程序；对于单片机应用、嵌入式系统和通信软件等是不可替代的；C语言的指针与计算机硬件的地址类似，是了解计算本质的钥匙；通过C语言的存储分配函数，可以深入了解计算机存储分配的原理。

1.2 C 语言程序的基本结构

1.2.1 第一个 C 语言程序：Hello world!

学习一门程序设计语言往往都是从屏幕上输出简单的信息开始的，一个输出简单信息的 C 语言程序如案例 1-1 所示。

【案例 1-1】 编写程序，在屏幕上输出 “Hello world!”。

```
#include<stdio.h>
void main()
{
    printf("Hello world!\n");
}
```

程序运行后的结果如图 1-1 所示。

1.2.2 基本结构

C 语言程序由一个或多个函数组成，每个函数是完成一定功能的一段 C 语言语句，如案例 1-1 中的 `main()` 函数。因此，构成 C 语言程序的基本单位是函数，在构成 C 语言程序的所有函数中必须有且只有一个 `main()` 函数，即主函数。也就是说，一个 C 语言程序必须由至少一个函数组成，这个函数就是主函数。程序运行总是从主函数开始，在主函数结束。

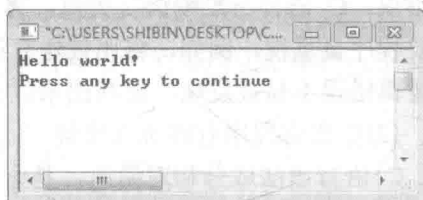


图 1-1 程序运行后的结果

在 C 语言程序中，除主函数外，其他函数的函数名由程序员确定，称为自定义函数。每个函数由函数类型、函数名和函数名后圆括号内的参数组成函数首部，其后在花括号中由若干条功能语句构成函数体。在最简单的情况下，函数的格式如下：

```
函数类型 函数名(参数)
{
    函数体
}
```

其中，圆括号内的参数既可以有一个也可以有多个，还可以没有，但没有参数时，圆括号不能省略；函数体由若干条 C 语句构成，每条 C 语句以分号表示结束，而不以换行表示一条语句结束，多数程序员都习惯在一条语句结束的地方换行，以增强程序的可读性。

自定义函数不能像主函数那样独立运行，它只能由主函数或其他函数调用运行。所谓调用，就是函数暂时中断本函数的执行，转去执行被调用函数的功能语句。当被调用函数执行完或遇到 `return` 语句时，必须返回原先中断执行的函数，继续执行该函数后面的语句。在这个过程中会发生原函数调用被调函数和被调函数返回原函数的情况，可见，函数之间就是互相调用和返回的关系，如案例 1-2 所示。

【案例 1-2】 函数之间的调用与返回。

```
#include<stdio.h>
int add(int x,int y)
{
    return x+y;
}
int main()
{
    int a=50,b=40;
    int c;
    c=add(a,b); /* 调用 add() 函数*/
    printf("a+b=%d\n",c);
    return 0;
}
```

在案例 1-2 中定义了一个两数相加的 `add()` 函数，其参数是 `x` 和 `y`，称为形式参数。当主函数执行到“`c=add(a,b);`”时，中断主函数的执行，去执行 `add()` 函数，`add()` 函数中实现 `x` 与 `y` 相加，并由 `return` 将两个数相加的结果带回到调用它的函数中，接着再执行主函数中的语句，程序执行后的结果是输出“`a+b=90`”。

总之，C 语言程序是由一个或多个函数组成的，其中必须有且仅有一个名为 `main()` 的主函数；程序的执行从主函数开始，在主函数结束，其他函数通过调用来执行；非主函数之间可以相互调用，但不能调用主函数；函数体是一段完成某个功能的 C 语句，每条语句由一个分号作为结束标志。

1.3 C 语言的集成开发环境

1.3.1 主流开发工具介绍

C 语言是一门计算机高级语言，用 C 语言编写的程序称为“源程序”，但计算机不能直接执行，因为计算机只能识别和执行由 0 和 1 组成的二进制的指令。为了使计算机能够执行以 C 为扩展名的源程序，必须先编译源程序，将源程序翻译成以 `.obj` 为扩展名的二进制形式的目标程序，然后将目标程序与系统的函数库和其他目标程序链接起来，形成以 `.exe` 为扩展名的可执行程序。编译和链接的过程需要专用的编译工具，主要的编译工具有如下几种。

1. VC++ 6.0 工具

Microsoft Visual C++ 6.0，简称 VC++ 6.0，是微软于 1998 年推出的一款 C/C++ 编译器。其集成了 MFC 6.0，包含标准版 (Standard Edition)、专业版 (Professional Edition) 与企业版 (Enterprise Edition)。发行至今一直被广泛用于大大小小的项目开发。它不但具有程序框架自动生成、类管理灵活方便、代码编写和界面设计集成交互操作、可开发多种程序等优点，通过设置还可生成程序框架支持数据库接口、OLE 2.0、WinSock 网络。

2. Dev-C++工具

Dev-C++是一个 Windows 环境下的适合于初学者使用的轻量级 C/C++集成开发环境 (IDE)，它是一款自由软件，遵守 GPL 许可协议分发源代码。

Dev-C++使用 MingW64/TDM-GCC 编译器，遵循 C++ 11 标准，同时兼容 C++ 98 标准。开发环境包括多页面窗口、工程编辑器及调试器等，在工程编辑器中集合了编辑器、编译器、链接程序和执行程序，提供高亮度语法显示，以减少编辑错误，还有完善的调试功能，适合初学者与编程高手的不同需求，是学习 C 或 C++的首选开发工具。

3. Code::Blocks 工具

Code::Blocks 是一个开放源代码的全功能跨平台 C/C++集成开发环境，它最大的特点是支持通过插件的方式对 IDE 自身功能进行扩展，这使得 Code::Blocks 具有很强的灵活性，方便程序员使用。

Code::Blocks 本身并不包含编译器和调试器，它仅提供了一些基本的工具，用来帮助程序员从命令行中解放出来，使程序员享受更友好的代码编辑界面。不过，后期 Code::Blocks 的发行版本以插件的形式提供了编译和调试功能。

4. Turbo C 2.0 工具

Turbo C 2.0 不仅是一个快捷、高效的编译程序，同时还有一个易学、易用的集成开发环境。使用 Turbo C 2.0 无须独立编辑、编译和链接程序，就能建立并运行 C 语言程序。因为这些功能都组合在 Turbo C 2.0 的集成开发环境内，并且可以通过一个简单的主屏幕使用这些功能。但由于 Turbo C 2.0 不支持鼠标操作，对新的 Windows 系统兼容性不是很好，现在已经很少使用。

1.3.2 VC++ 6.0 环境介绍

全国计算机等级考试的考试大纲要求 C 语言的编辑环境是 VC++ 6.0，计算机专业的多数初学者也把 VC++ 6.0 作为编辑环境，这是由于 VC++ 6.0 在 Windows 下运行友好，学生更易于上手操作。本书选取 VC++ 6.0 作为 C 语言程序的开发工具。

1. VC++ 6.0 的安装

VC++ 6.0 是微软于 1998 年推出的一款 C/C++编译器，如今主流的硬件环境都能满足安装要求，对于软件系统，Windows 系统也能很好地支持。VC++ 6.0 在 Windows 7 系统中的安装步骤如下。

双击安装光盘中的 Setup.exe，启动安装向导，如图 1-2 所示。

单击“下一步”按钮，进入许可协议界面，选中“接受协议”后继续单击“下一步”按钮，进入“产品号和用户 ID”界面，如图 1-3 所示。输入相关信息后单击“下一步”按钮，在打开界面的“自定义-服务器安装程序选项”中选择“安装 Visual C++ 6.0 中文企业版(I)”，再单击“下一步”按钮，进入“选择公用安装文件夹”界面，如图 1-4 所示。单击“浏览”按钮设置安装路径后，单击“下一步”按钮，打开如图 1-5 所示的“Visual C++ 6.0 Enterprise 安装程序”对话框，选择“Typical”安装方式后开始复制相关文件进行安装。

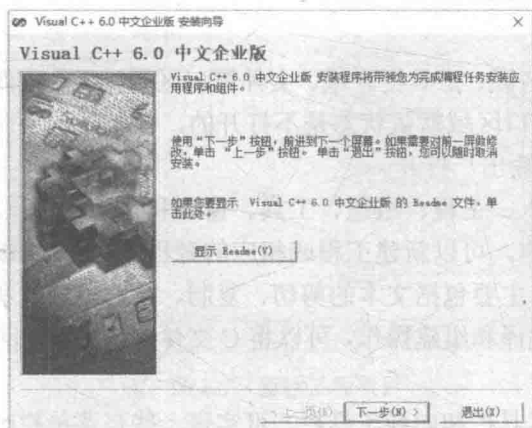


图 1-2 安装向导



图 1-3 “产品号和用户 ID”界面



图 1-4 “选择公用安装文件夹”界面

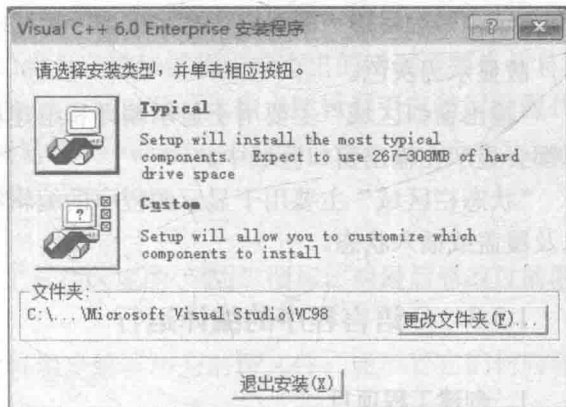


图 1-5 “Visual C++ 6.0 Enterprise 安装程序”对话框

安装完成后在系统的开始菜单中展开“Microsoft Visual C++ 6.0”, 单击其中的“Microsoft Visual C++ 6.0”项就可以启动 VC++ 6.0, 其主界面如图 1-6 所示。

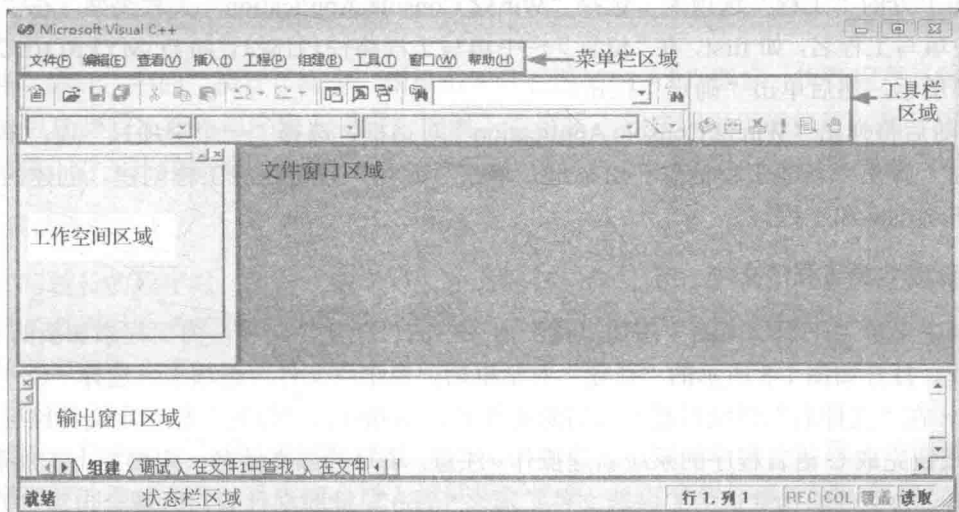


图 1-6 VC++ 6.0 主界面

2. VC++ 6.0 的环境介绍

在图 1-6 所示的主界面中,主要包括菜单栏区域、工具栏区域、文件窗口区域、工作空间区域、输出窗口区域和状态栏区域,其中输出窗口区域默认状态是不打开的,在编译程序时会自动打开,也可以在“查看”菜单项中选中“输出”打开。

“菜单栏区域”包括文件、编辑、查看、插入、工程、组建、工具、窗口和帮助菜单项。“文件”菜单项主要用于与文件及工程相关的操作,可以新建工程或相应的源程序文件。“编辑”菜单项用于对 C 语言程序的各种编辑操作,主要包括文本的剪切、复制、粘贴、查找和替换等。“组建”菜单项主要用于 C 语言程序的编译和组建操作,可以将 C 文件编译组建成可以执行的程序。

“工具栏区域”默认打开标准工具栏、编译工具栏和向导工具栏,可实现一些与菜单栏区域相同的操作,如文件的新建、保存和编辑操作,程序的编译与组建操作等。

“文件窗口区域”主要用于输入 C 语言程序。当前没有打开任何文件,故不能输入程序。

“工作空间区域”主要用于显示当前工程的文件及函数构成。当前没有打开任何工程或项目,故显示为灰色。

“输出窗口区域”主要用于显示编译和组建程序的结果。在编译或组建时,正确或错误信息都会显示在输出窗口区域中。

“状态栏区域”主要用于显示源程序的编辑状态,包括源程序的编辑位置,所在行和列,以及覆盖或插入状态。

1.3.3 C 语言程序的编译运行

1. 创建工程项目

用 Visual C++ 6.0 系统建立 C 语言应用程序,首先要创建一个工程项目(project),用来存放 C 语言程序的所有信息。创建一个工程项目的操作步骤如下。

① 进入 Visual C++ 6.0 环境后,选择主菜单“文件”中的“新建”选项,在弹出的对话框中单击上方的“工程”选项卡,选择“Win32 Console Application”工程类型,在“工程名称”栏中填写工程名,如 first,在“位置”栏中填写工程路径(目录),如 D:\MYPROJECT\first,如图 1-7 所示,然后单击“确定”按钮。

② 随后将弹出“Win32 Console Application”对话框,选择“一个空项目”项,单击“完成”按钮。弹出“新建工程信息”对话框,单击“确定”按钮完成工程创建。创建的工作区文件为 first.dsw 和工程。

2. 新建 C 语言程序文件

选择主菜单“工程”中的“添加工程”命令中的“新建”选项,为工程添加新的 C 语言程序文件。打开如图 1-8 所示的“新建”对话框后,单击“文件”选项卡,选择“C++ Source File”项,在“文件名”栏填写新添加的源文件名,如 first.c,“位置”栏指定文件路径,单击“确定”按钮完成 C 语言程序的系统新建操作。注意:填写的源文件名一定要加上扩展名“.c”,否则系统会为文件添加默认的 C++源文件扩展名“.cpp”。此时文件窗口区域会自动打开 first.c 源程序,可对该程序进行编辑。