



普通高等教育“十一五”国家级规划教材

孙素燕 靖定国 主编

C语言程序设计 案例化教程

21世纪
计算机
科学
与
技术
实
践
型
教
程

丛书主编 陈明



清华大学出版社

校企合作教材

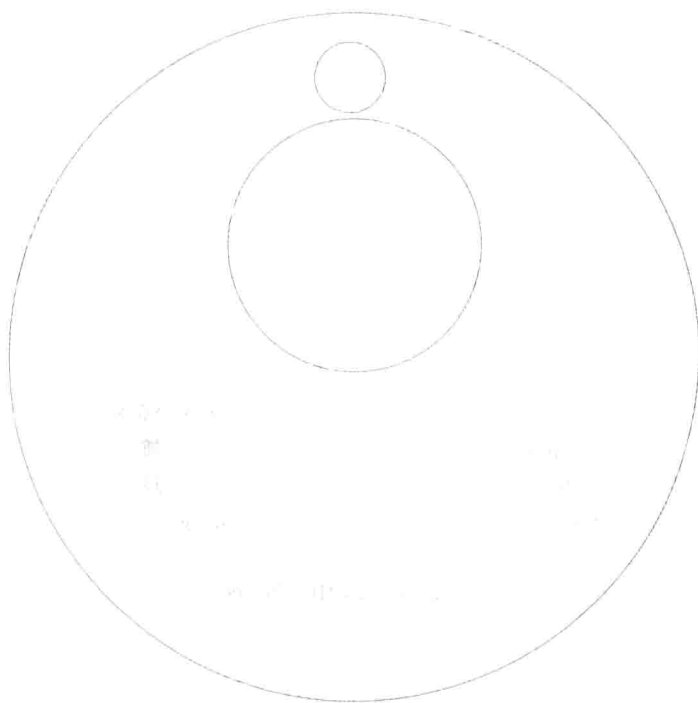


普通高等教育“



孙素燕 靖定国 主 编
杨 琴 汪 军 副主编

C语言程序设计 案例化教程



21世纪
计 算 机 科 学 与 技 术
实 践 型 教 程

丛书主编 陈明

清华大学出版社
北 京

内 容 简 介

本书作为校企合作教材,是程序设计教材的创新,实现了从以计算机知识为主线的体系结构向以企业能力训练为主线的体系结构转变,把程序设计的学习从语法知识学习提高到解决问题的能力培养上。全书共设 11 个项目。项目 1 介绍了 C 语言的开发环境;项目 2 介绍了 C 语言的基本语法;项目 3、项目 4、项目 5 介绍了程序流程控制结构;项目 6 介绍了数组;项目 7 介绍了函数;项目 8 介绍了指针;项目 9 介绍了编译预处理的概念;项目 10 介绍了文件;项目 11 是课程实训,通过实训案例让学生学会编写简单的 C 程序软件。

本书可作为应用型院校程序设计课程的教材,又可作为等级考试及计算机培训班的教材或参考书,也可作为计算机相关专业的程序设计课程用书。

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 13701121933

图书在版编目(CIP)数据

C 语言程序设计案例化教程/孙素燕,靖定国主编. —北京:清华大学出版社,2014

21 世纪计算机科学与技术实践型教程

ISBN 978-7-302-37342-1

I. ①C… II. ①孙… ②靖… III. ①C 语言—程序设计—高等学校—教材 IV. ①TP312

中国版本图书馆 CIP 数据核字(2014)第 160213 号

责任编辑:谢琛 薛阳

封面设计:何凤霞

责任校对:白蕾

责任印制:刘海龙

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座 邮 编:100084

社总机:010-62770175 邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质量反馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

课件下载: <http://www.tup.com.cn>, 010-62795954

印装者:北京国马印刷厂

经 销:全国新华书店

开 本:185mm×260mm 印 张:15.5

字 数:359 千字

版 次:2014 年 8 月第 1 版

印 次:2014 年 8 月第 1 次印刷

印 数:1~2500

定 价:29.00 元

产品编号:059206-01

前 言

当前,“工学结合”是应用型教育培养模式改革的重要切入点和出发点,而“校企合作”则对应用型教育的模式具有积极的导向作用。为了适应培养创业型复合人才的需要,进一步完善和补充 C 语言程序设计系列的教材,特编写了适用于应用型学生特色的《C 语言程序设计案例教程》。

C 语言是一种在世界范围内被普遍采用的优秀的程序设计语言,是现在最流行的结构化程序设计语言之一,它具有语法简单、使用灵活方便、功能丰富、表达力强、便于大型程序开发、便于编写、可移植性好等优点。由于 C 语言引入了反映计算机硬件特性的机制,这使得 C 语言是一种既能编写系统软件,又适合编写应用软件的高级语言。

本书共设 11 章,通过场景导入引出问题,然后详细讲解用来解决问题的知识点,最后回到场景中解决问题,以此主线引导全文。全书主要内容如下。

第 1 章: C 语言概述,主要包括 C 语言的发展和特点、VC++ 6.0 环境介绍等。

第 2 章: 数据描述与基本操作,主要包括 C 数据类型、运算符及表达式等。

第 3 章: 顺序结构程序设计,主要包括 C 语句、数据的输入和输出、顺序结构设计等。

第 4 章: 选择结构程序设计,主要包括选择结构语句和选择结构程序示例等。

第 5 章: 循环结构程序设计,主要包括 while 循环、do-while 循环、for 循环、循环的嵌套、continue 和 break 语句以及循环结构程序设计示例等。

第 6 章: 数组,主要包括一维数组、二维数组的定义及应用等。

第 7 章: 函数,主要包括函数定义、函数间的参数传递、函数调用、变量的作用域等。

第 8 章: 指针,主要包括指针的定义、指针变量、指针与数组、字符数组与字符指针的区别以及程序示例等。

第 9 章: 编译预处理,主要包括文件包含、有参数和无参数的宏定义、条件编译等。

第 10 章: 文件,主要包括文件和文件指针的定义、文件的基本操作等。

第 11 章: 课程实训,主要是对前面各项目知识点的应用,即用 C 语言编写公司员工信息管理系统。读者可以自己尝试一下编写程序。

本书的编写具有以下三个主要特点。

(1) 以突出培养创业型复合人才为目标,用丰富的版块合理安排全文,突出实用性和

可操作性。

(2) 以企业实际案例为依托,紧紧围绕“场景导入”→“知识讲解”→“回到场景”→“拓展训练”这一主线进行编写,强化能力训练。

(3) 以工作过程为导向,全面展开案例实施的全过程,提炼技术要点,即学即用,面向就业。

本书主要面向应用型技术院校,既可作为高职高专院校程序设计课程的教材和教学参考书,又可作为等级考试及计算机培训班的教材或参考书,也可作为计算机相关专业的程序设计课程用书。

本书配有免费的多媒体课件、教案、程序源代码和习题参考答案供广大教师和读者使用,旨在为教师授课、读者学习提供方便;为满足等级考试的需要,本书还赠送全国计算机等级考试题库和模拟软件。需要者可从清华大学出版社网站(www.tup.tsinghua.edu.cn)下载。

本书由硅湖职业技术学院孙素燕和靖定国担任主编,硅湖职业技术学院杨琴和昆山环科计算机有限公司汪军担任副主编,其中第1章、第2章、第9章、第11章及附录由孙素燕编写,第3~5章、第10章由靖定国编写,第6~8章由杨琴编写,相关企业案例由汪军提供。全书由孙素燕统稿、定稿。参与本书编写和资料整理的还有何光明、卢振侠、王珊珊、周海霞、石雅琴、张居晓、朱贵喜、张华明、李佐勇等。

本书在编写过程中得到了学院领导的鼓励与支持,得到了计算机教研室全体老师的帮助和指导,在此向他们表示衷心的感谢!同时,还得到了昆山环科计算机有限公司的大力支持,在此一并表示感谢!

由于编者水平有限,加之时间仓促,书中难免存在不足之处,敬请广大读者批评指正。联系信箱: ssy9914@usl.edu.cn。

编 者

2014年6月

目 录

第 1 章 C 语言概述	1
1.1 C 语言的发展与特点	1
1.1.1 C 语言的发展历程	1
1.1.2 C 语言的特点	2
1.2 C 语言程序的格式和构成	3
1.2.1 C 语言程序的格式说明	3
1.2.2 C 语言程序的构成及编译	4
1.2.3 C 语言程序的基本要求	5
1.3 C 语言的开发环境	5
1.3.1 Visual C++ 6.0 的安装	6
1.3.2 进入 Visual C++ 6.0 开发环境	8
1.3.3 Visual C++ 6.0 重要菜单命令介绍	10
1.4 拓展训练	10
1.5 知识链接	11
第 2 章 数据描述与基本操作	17
2.1 场景导入	17
2.2 数据类型	18
2.2.1 常量与变量	18
2.2.2 整型数据	22
2.2.3 实型数据	23
2.2.4 字符型数据	24
2.3 运算符与表达式	25
2.3.1 算术运算符与算术表达式	25
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	31
2.3.4 逻辑运算符与逻辑表达式	32
2.4 回到场景	35

2.5	拓展训练	36
2.6	知识链接	38
2.6.1	条件运算符与条件表达式	38
2.6.2	逗号运算符与逗号表达式	39
2.6.3	不同类型数据间的混合运算	40
第3章	顺序结构程序设计	42
3.1	场景导入	42
3.2	C 语句概述	43
3.2.1	表达式语句	43
3.2.2	函数调用语句	44
3.2.3	流程控制语句	44
3.2.4	空语句	44
3.2.5	复合语句	45
3.3	赋值语句	46
3.4	格式输入与输出	46
3.4.1	格式输出函数 printf 函数	47
3.4.2	格式输入函数 scanf 函数	49
3.5	顺序结构程序设计	51
3.5.1	程序设计中的三种基本结构	51
3.5.2	顺序结构程序设计举例	53
3.6	回到场景	55
3.7	拓展训练	56
3.8	知识链接	58
3.8.1	putchar 函数	58
3.8.2	getchar 函数	59
3.8.3	getch 函数	60
3.8.4	puts 函数	60
3.8.5	gets 函数	61
第4章	选择结构程序设计	62
4.1	场景导入	62
4.2	if 语句	63
4.2.1	选择结构概述	63
4.2.2	if 语句的三种形式	63
4.3	switch 语句	66
4.3.1	switch 语句概述	66
4.3.2	switch 语句程序设计举例	67

4.4	选择结构程序设计举例	68
4.5	回到场景	72
4.6	拓展训练	76
4.7	知识链接	78
4.7.1	if 语句的嵌套	78
4.7.2	条件运算符	81
第 5 章	循环结构程序设计	84
5.1	场景导入	84
5.2	while 语句构成的循环	85
5.2.1	循环结构概述	85
5.2.2	while 循环结构	85
5.2.3	while 循环结构举例	86
5.3	do-while 语句构成的循环	87
5.3.1	do-while 循环结构	87
5.3.2	do-while 循环结构举例	87
5.3.3	while 和 do-while 循环的比较	88
5.4	for 语句构成的循环	89
5.4.1	for 循环结构	89
5.4.2	for 循环结构举例	90
5.4.3	几种循环的比较	90
5.5	break 语句和 continue 语句	91
5.5.1	break 语句	91
5.5.2	continue 语句	92
5.5.3	break 语句和 continue 语句的区别	93
5.6	循环结构程序设计举例	94
5.7	回到场景	97
5.8	拓展训练	98
5.9	知识链接	102
5.9.1	for 循环的变型	102
5.9.2	循环的嵌套	103
5.9.3	goto 语句构成的循环	105
第 6 章	数组	108
6.1	场景导入	108
6.2	一维数组	109
6.2.1	一维数组的定义与引用	109
6.2.2	一维数组的初始化	109

6.3	一维数组应用举例	110
6.4	二维数组	113
6.4.1	二维数组的定义与引用	113
6.4.2	二维数组的初始化	114
6.5	二维数组应用举例	114
6.6	回到场景	116
6.7	拓展训练	117
6.8	知识链接	121
6.8.1	多维数组	121
6.8.2	字符数组的定义和初始化	121
6.8.3	字符串	122
6.8.4	字符串的输入与输出	123
第7章	函数	125
7.1	场景导入	125
7.2	函数定义	126
7.3	函数参数和函数的返回值	127
7.3.1	形式参数和实际参数	127
7.3.2	函数的返回值	127
7.4	函数的参数传递方式	128
7.4.1	普通变量作为函数参数	128
7.4.2	数组作为函数参数	129
7.5	函数的调用	130
7.5.1	函数调用的一般形式	130
7.5.2	函数调用方式	130
7.6	函数的嵌套调用	131
7.7	变量的作用域	133
7.7.1	局部变量	133
7.7.2	全局变量	133
7.8	回到场景	134
7.9	拓展训练	135
7.10	知识链接	139
7.10.1	数组作为函数参数	139
7.10.2	函数的递归调用	140
7.10.3	变量的作用域	141
7.10.4	变量的存储类型	143
7.10.5	函数的作用域	143

第 8 章 指针	145
8.1 场景导入	145
8.2 指针的概念	146
8.2.1 指针变量的定义	146
8.2.2 指针变量赋值与初始化	147
8.2.3 指针变量应用	148
8.2.4 指针变量作为函数参数	150
8.3 指针与数组	151
8.3.1 指向数组元素的指针变量	151
8.3.2 通过指针引用数组元素	152
8.3.3 数组名作函数参数	154
8.4 指针与字符串	156
8.4.1 字符串的表示形式	156
8.4.2 字符指针变量与字符数组	158
8.5 回到场景	159
8.6 拓展训练	160
8.7 知识链接	163
8.7.1 二维数组的指针	163
8.7.2 指向多维数组的指针和指针变量	167
8.7.3 指针型函数	169
8.7.4 指向指针的指针	172
第 9 章 编译预处理	174
9.1 场景导入	174
9.2 宏定义	175
9.2.1 无参数宏定义	175
9.2.2 有参数宏定义	178
9.3 文件包含	180
9.4 条件编译	183
9.5 回到场景	186
9.6 拓展训练	188
9.7 知识链接	190
第 10 章 文件	193
10.1 场景导入	193
10.2 C 语言中文件的概念	194
10.2.1 文件的概念与分类	194

10.2.2	文件的处理方式	195
10.2.3	文件类型指针	195
10.3	文件的打开与关闭	196
10.3.1	文件的打开	196
10.3.2	文件的关闭	197
10.4	文件的读写	199
10.4.1	字符读写函数	199
10.4.2	字符串读写函数	200
10.5	文件的定位与检测	202
10.5.1	文件定位函数	202
10.5.2	文件出错检测函数	204
10.6	回到场景	205
10.7	拓展训练	206
10.8	知识链接	209
10.8.1	文件类型指针的定义	209
10.8.2	数据块读写函数	209
10.8.3	格式化读写函数	210
第 11 章	课程实训	211
11.1	项目案例	212
11.2	拓展训练	221
11.3	知识链接	222
附录 A	常用 ASCII 代码对照表	226
附录 B	C 语言中的关键字	227
附录 C	运算符的优先级和结合性	228
附录 D	常用库函数	230
参考文献		236

第 1 章 C 语言概述



知识要点:

- (1) C 语言的发展及特点。
- (2) C 语言程序的格式和构成。
- (3) C 语言的开发环境介绍。



技能目标:

- (1) 掌握 Visual C++ 6.0 的安装方法。
- (2) 了解 C 语言的发展及特点。
- (3) 熟悉 Visual C++ 6.0 开发环境中主要菜单项的使用方法。
- (4) 掌握 C 语言源程序的编译流程。

1.1 C 语言的发展与特点

1.1.1 C 语言的发展历程

C 语言是在一台使用 UNIX 操作系统的 DEC PDP-11 计算机上首次实现的,由 Dennis Ritchie 设计完成。C 语言是由一种早期的编程语言 BCPL 发展演变而来,BCPL 目前在欧洲还在使用。由于 Martin Richards 改进了 BCPL,从而促进了 Ken Thompson 所设计的 B 语言的发展,最终促使 20 世纪 70 年代 C 语言的问世,并一直沿用至今。

后来,人们对 C 语言又做了很多次改进,但主要还是局限于在贝尔实验室使用。直到 1975 年 UNIX 第 6 版公布后,C 语言的突出优点才引起人们的普遍关注。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 语言移植到其他机器时所需做的工作大大简化,这也推动了 UNIX 操作系统迅速地在各种机器上的实现。例如 VAX、AT&T 等计算机系统都迅速相继开发了 UNIX。随着 UNIX 使用的日益广泛,C 语言也迅速得到推广。C 语言与 UNIX 可以说是孪生兄弟,在发展过程中相辅相成。1978 年以后,C 语言已先后移植到大型、中型、小型及微型计算机上,并独立于 UNIX。现在 C 语言已风靡全世界,成为世界上应用最广的几种计算机语言之一。

随着微型计算机的日益普及,出现了许多 C 语言版本。但由于没有统一的标准,使得这些 C 语言之间出现了一些不一致的地方。为了改变这种情况,美国国家标准协会为 C 语言制定了一套 ANSI 标准,成为现行的 C 语言标准。

除了系统软件外,C语言还成功地用于数值计算、文字处理、数据库、计算机网络和多媒体等。C语言所呈现的高级语言强有力的表达能力和效率,使得它成为在计算机程序设计实践中做出了重大贡献的一种语言。

目前,在计算机界比较流行的C语言编译系统有Microsoft C、Turbo C、Borland C等。尽管上述各种编译系统存在着一些差异,但是它们的基本部分还是相同的,本书的上机环境统一采用Microsoft Visual C++ 6.0。

1.1.2 C语言的特点

C语言是目前应用最广泛的高级程序设计语言之一,该语言有很多较为突出的优点,不过它并不是完美的,还有一些缺点。

1. C语言的优点

1) 语言描述简洁、灵活、高效

C语言只有32个标准的关键字和9种控制语句,并且以易读易写的小写字母为基础,压缩了一切不必要的成分,使程序书写规整紧凑。

2) 有丰富的数据类型

C语言数据类型丰富,不仅具有4种最基本的数据类型(char、int、float、double),还有多种组合类型(数组和结构)以及复杂的导出类型,并允许使用简单的组合结构构造复杂的数据类型或直接由用户自己定义数据类型。

3) 运算符丰富

C语言提供了45种标准运算符和多种获取表达式值的方法,并提供了与地址密切相关的指针及其运算符,运算符不仅具有优先级的概念,还有结合性的概念。因此,灵活使用各种运算符和表达式,不仅可以简化程序,还可以实现在其他语言中难以实现的运算。

4) C语言具有固定的标识符

C语言的标识符主要用来表示常量、变量、函数和类型等的名字,它们只是起标识作用的一种符号。C语言共有32个固定的标识符,它们都用小写字母表示:

int	long	double	float	short
char	const	static	do	if
while	else	goto	break	continue
unsigned	switch	default	struct	union
enum	sizeof	void	auto	case
extern	register	return	typedef	volatile
define	include			

5) 提供了功能齐全的函数库

C语言标准函数库提供了功能极强的各类函数库,例如串、数组、结构乃至图形的处理等,只需调用一下库函数即可实现,为编程者提供了极大的方便。

6) 具有结构化的控制语句

C语言提供了9种控制语句实现三种结构(顺序、分支和循环结构),如if-else、while、

switch、for 等,并用函数作为程序模块,是理想的结构化程序设计语言。C 程序由具有独立功能的函数构成,函数定义是平行、独立的,但函数调用可以是嵌套调用和递归调用,通过函数调用可以实现复杂的程序功能,多种存储属性的数据可共享同一段内存,从而保证了模块化的程序设计风格。另外,对于复杂的源程序文件,可以分割成多个较小的源文件,分别编译和调试,最后组装、链接,得到可执行的目标程序文件。

7) 具有良好的通用性和程序的移植性

在 C 语言中,与硬件有关的操作都是通过调用系统提供的库函数实现的,这使得 C 语言具有很好的通用性,程序能容易地从一种计算机环境移植到另一种计算机环境中。

8) 生成目标代码质量高,程序执行效率高

C 语言生成目标代码的效率与汇编语言相比,一般低 10%~20%。C 语言也可以像汇编语言一样对位、字节和地址,甚至对硬件进行直接操作。换句话说,C 语言既具有汇编语言的强大功能,又没有汇编语言的难度,特别适合进行底层开发。

9) 语法限制不严格

C 语言程序设计的自由度大,例如,对数组下标越界不做检查,各种类型的变量可以通用等。

2. C 语言的缺点

当然,C 语言的缺陷还是十分明显的。C 语言缺乏一致公认的标准,表现在语法限制不太严谨,运算符的优先级和结合性比较复杂,不容易记忆。C 语言对数据类型缺乏一致性的检测,对数组、结构体等类型的整体运算存在一定的限制。C 语言在程序设计方面灵活性有余,而安全性和可靠性不足。

对于上述问题,计算机科学家和工程技术人员正在不断寻求解决的办法,并一直在不断地提出一些改进方案以进一步提高 C 语言的应用能力。

1.2 C 语言程序的格式和构成

首先通过讨论一个具体的 C 语言的实例,帮助读者对 C 语言有一个初步的认识,同时也有助于对 C 语言的格式和构成特点进行剖析。

1.2.1 C 语言程序的格式说明

【例 1-1】 设有如下程序:

```
#include <stdio.h>
int add(int x, int y)
{
    int z;
    z=x+y;                /* 计算和 */
    return(z);
}
```

```

main()                                /* 主函数 */
{
    int x, y, z;                       /* 定义变量 */
    scanf("%d%d", &x, &y);            /* 从键盘上输入两个变量值 */
    z=add(x, y);                       /* 调用 add 函数, 计算两个数的和 */
    printf("The result is %d\n", z);   /* 输出 */
}

```

运行结果如图 1-1 所示。

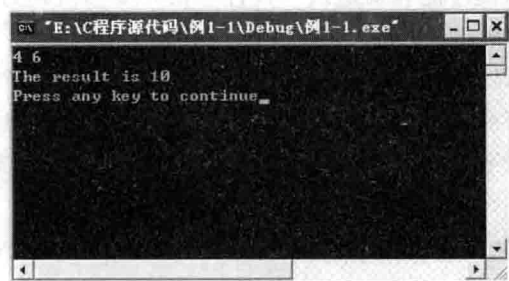


图 1-1 例 1-1 运行结果

通常一个 C 语言源程序由一个主函数和若干个其他函数组成;一个函数由若干条语句组成;每条语句又由标识符、运算符等组成。本示例程序共包括两个函数,一个是主函数 main(),另一个是 add()函数,用来求出两个整数的和。在执行时,先由 scanf()函数从键盘上输入两个数值,将输入的数传入 add()函数中,在 add()函数中计算两个数的和,并用 return 语句将和作为函数的返回值。这样就完成了将两个整数相加的功能。

1.2.2 C 语言程序的构成及编译

1. C 语言程序的构成

一般 C 语言程序的构成如下:

```

函数类型 函数名(参数类型 参数名)    /* 自定义函数 */
{
    函数体;
}
main()
{
    变量定义部分;
    语句执行部分;                     /* 语句执行部分包括调用自己定义的函数 */
}

```

2. C 语言的编译过程

1) 编译

编译就是将已经编好的源程序翻译成二进制的目标代码,如果有错误,会给出错误提

示,这时应该重新进入编辑环境,对源程序进行修改,直到通过编译为止。编译后的文件扩展名为 obj。经过编译的二进制代码不能直接执行,因为一个程序的各个模块往往是单独编译的,必须把经过编译的各个模块的目标代码与系统提供的标准模块进行连接后才能够执行。

2) 连接

经过连接后,源程序成为可执行的文件,它是计算机能够直接运行的文件。文件的扩展名为 exe。

3) 执行

直接在操作系统的命令行环境下输入文件名就可以执行经过编译连接的可执行程序。

整个过程可以用图 1-2 来表示。

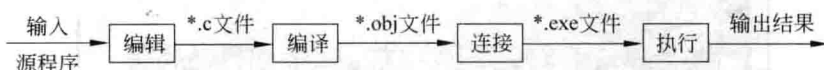


图 1-2 程序的编译、连接和执行过程

具体程序的运行过程详见 1.5 节。

1.2.3 C 语言程序的基本要求

通常,C 语言程序有以下要求。

(1) 程序至少要有有一个 main 函数,用户也可以自己根据需要设计函数,像上面的 add() 函数。

(2) 函数由两部分组成。第一部分是函数的说明部分,如函数的名称、函数的返回值类型、函数的参数及类型。第二部分是函数的实现部分,包括变量定义和执行语句。

(3) C 程序总是从 main 函数开始执行,不论 main 函数在程序的什么地方,也就是说,可以将 main 函数放在任何位置。

(4) C 程序的书写比较自由,可以在一行上写若干条语句,也可以在多行上写一条语句。但每条语句后面都要有一个分号,这个分号是必不可少的。这一点必须注意。

(5) C 语言中没有专门的输入、输出语句。输入和输出是通过 scanf 和 printf 两个库函数实现的。

(6) C 程序中可以用“/* ... */”对任何部分进行注释,好的程序都应该有必要的注释以提高程序的可读性。

(7) 为了提高程序的可读性、可维护性和扩充性,应尽可能按照标准的编码方式进行源代码的编写。

1.3 C 语言的开发环境

微软公司提供了一个支持可视化编程的集成开发环境: Visual Studio (又名 Developer Studio)。Developer Studio 是一个通用的应用程序集成开发环境,它不仅支持

Visual C++ ,还支持 Visual Basic、Visual J++、Visual InterDev 等 Microsoft 系列开发工具。

1.3.1 Visual C++ 6.0 的安装

Visual C++ 6.0 的安装过程如下。

(1) 在 CD-ROM 驱动器中放入 Visual C++ 6.0 安装盘,双击安装文件 setup.exe,出现如图 1-3 所示的界面。

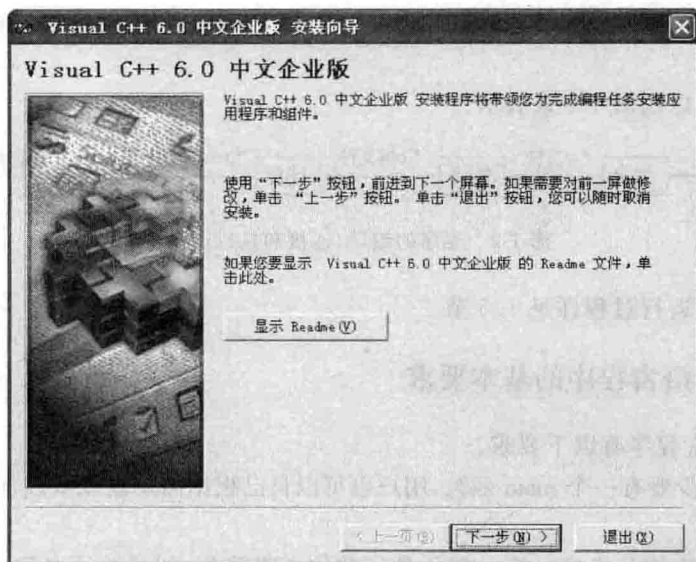


图 1-3 安装步骤 1

(2) 根据安装向导的提示一直安装下去,进入如图 1-4 所示的“程序正在安装”的界面。



图 1-4 安装步骤 2