

C语言 程序设计

任秀娟 张震 宿宏毅◎主编



北京理工大学出版社

BEIJING INSTITUTE OF TECHNOLOGY PRESS

北京理工大学出版社

C 语言程序设计

主 编：任秀娟 张 震 宿宏毅
副主编：裴 霞 徐兴习 王树声
参 编：马小婧 宗月萍 许学江 朱桂花 李 杰
主 审：唐西光 焦新伟

版权专有 侵权必究

图书在版编目 (CIP) 数据

C 语言程序设计/任秀娟, 张震, 宿宏毅主编. —北京: 北京理工大学出版社, 2016. 12

ISBN 978 - 7 - 5682 - 3155 - 8

I. ①C… II. ①任… ②张… ③宿… III. ①C 语言 - 程序设计 IV. ①TP312.8

中国版本图书馆 CIP 数据核字 (2016) 第 234930 号

出版发行 / 北京理工大学出版社有限责任公司

社 址 / 北京市海淀区中关村南大街 5 号

邮 编 / 100081

电 话 / (010) 68914775 (总编室)

(010) 82562903 (教材售后服务热线)

(010) 68948351 (其他图书服务热线)

网 址 / <http://www.bitpress.com.cn>

经 销 / 全国各地新华书店

印 刷 / 三河市华骏印务包装有限公司

开 本 / 787 毫米 × 1092 毫米 1/16

印 张 / 19.25

字 数 / 446 千字

版 次 / 2016 年 12 月第 1 版 2016 年 12 月第 1 次印刷

定 价 / 52.00 元

责任编辑 / 王玲玲

文案编辑 / 王玲玲

责任校对 / 周瑞红

责任印制 / 李志强

图书出现印装质量问题, 请拨打售后服务热线, 本社负责调换

前 言

本书以高等院校教学改革的要求为指导思想,根据高等院校人才培养方案、学生能力和课程体系等相关教学要求,结合作者多年从事计算机教学改革的经验,并在广泛征求用人单位专家及相关院校一线教师意见的基础上编写而成。

根据高等院校改革的发展规律,以及高层次技术技能人才的培养目标,本书从高等教育的特点出发,采用学生易于接受的表达方式进行教学设计,以培养学生程序设计能力为目标,以程序设计为主线,重点讲解程序设计的思想和方法。因此,本书在教学设计和内容组织上具有以下特点:

1. 本书特别注重教材的可读性和可用性,每章均以学习目标开头,以解决问题为目的,介绍相关的语言知识,注重实用,由浅入深,强化知识点、算法、编程方法与技巧,每章的结尾都安排了小结并配有多种题型的习题,以方便学生掌握编程思想和方法。

2. 为了便于学生学习和掌握编程技巧,本书以专门章节列举了C语言编程过程中容易出现的问题,以醒目的方式给出了常用语法规则和常见错误提示,帮助学生学习和理解教学内容。

3. 考虑到高等院校学生的学习特点和认知规律,本书的编写是将基本概念和基本理论融入实例中进行讲解,从而将抽象问题具体化,将复杂的理论简单化,便于学生学习和理解。

4. 本书例题丰富,通过学与练的紧密结合,实现学有所悟,练有所思,从而培养学生的多向思维能力和自主学习的习惯。

本书由任秀娟老师、张震老师、宿宏毅老师担任主编,裴霞老师、徐兴习老师、王树声老师担任副主编,参与编写的还有马小婧老师、宗月萍老师、许学江老师、朱桂花老师、李杰老师,其中宿宏毅老师编写第1章,任秀娟老师编写第2章、第3章,裴霞老师编写第4章,徐兴习老师编写第5章,王树声老师编写第6章,张震老师编写第7章,马小婧老师编写第8章,宗月萍老师编写第9章,许学江老师编写第10章,朱桂花老师编写第11章,李杰老师编写第12章。

本书由唐西光老师、焦新伟老师主审。

本书可作为高等院校、成人教育、函授计算机等相关专业的程序设计入门教材,也可作为全国计算机等级考试的辅导教材,还可供相关领域的工程技术人员学习参考。



本书在编写过程中充分汲取了本行业多位专家提出的许多宝贵意见和建议，并结合人才培养目标对教材进行了多次修正和完善，在此向各位专家和老师们一并表示感谢！

由于作者水平有限，书中难免有不当之处，恳请广大读者批评指正，以便再版时改进。

编者

目 录

第1章 C语言概述	1
1.1 C语言概述	1
1.1.1 C语言的特点	2
1.1.2 C语言源程序的结构特点	2
1.1.3 书写程序时应遵循的规则	3
1.1.4 C语言的字符集	3
1.1.5 C语言词汇	3
1.2 编制并运行一个简单程序	5
1.2.1 编制并运行程序的“四步曲”	5
1.2.2 工程 (Project) 以及工程工作区 (Project Workspace)	5
1.2.3 启动并进入 VC6.0 的集成开发环境	6
1.2.4 创建工程并输入源程序代码	7
1.2.5 不创建工程, 直接输入源程序代码	11
1.2.6 编译、连接而后运行程序	11
1.2.7 及时备份自己的创作	12
1.2.8 将自己设计的产品提供给他人使用	13
1.3 VC6.0 开发环境使用参考	13
1.3.1 VC6.0 常用菜单命令项	13
1.3.2 VC6.0 的主要工作窗口	15
习题	17
第2章 C语言常见错误	20
第3章 C语言程序基础	25
3.1 C语言的数据类型	25
3.2 常量与变量	27
3.2.1 常量和符号常量	27
3.2.2 变量	28
3.3 整型数据	28
3.3.1 整型常量的表示方法	28



3.3.2	整型变量	29
3.4	实型数据	33
3.4.1	实型常量的表示方法	33
3.4.2	实型变量	34
3.4.3	实型常数的类型	36
3.5	字符型数据	36
3.5.1	字符常量	36
3.5.2	转义字符	36
3.5.3	字符变量	37
3.5.4	字符数据在内存中的存储形式及使用方法	37
3.5.5	字符串常量	39
3.6	变量赋初值	39
3.7	各类数值型数据之间的混合运算	40
3.8	算术运算符和算术表达式	41
3.8.1	C 语言运算符简介	42
3.8.2	算术运算符和算术表达式	42
3.9	赋值运算符和赋值表达式	45
3.10	关系运算符和表达式	47
3.10.1	关系运算符及其优先次序	47
3.10.2	关系表达式	47
3.11	逻辑运算符和表达式	49
3.11.1	逻辑运算符及其优先次序	49
3.11.2	逻辑运算的值	49
3.11.3	逻辑表达式	50
3.12	逗号运算符和逗号表达式	51
3.13	条件运算符和条件表达式	52
3.14	小结	53
3.14.1	C 语言的数据类型	53
3.14.2	基本类型的分类及特点	54
3.14.3	常量后缀	54
3.14.4	常量类型	54
3.14.5	数据类型转换	54
3.14.6	运算符优先级和结合性	54
3.14.7	表达式	55



习题	55
第4章 顺序程序设计	61
4.1 C 语句概述	61
4.1.1 控制语句	61
4.1.2 表达式语句	62
4.1.3 复合语句	63
4.2 数据的输入和输出	63
4.2.1 单个字符的输出——putchar() 函数	64
4.2.2 单个字符的输入——getchar() 函数	64
4.2.3 puts 函数 (字符串、字符数组中字符串输出函数)	65
4.2.4 gets 函数 (字符串输入函数)	65
4.2.5 printf 函数 (格式输出函数)	65
4.2.6 scanf 函数 (格式输入函数)	68
4.3 顺序结构程序设计举例	70
习题	72
第5章 选择结构程序设计	78
5.1 if 语句	78
5.2 switch 语句	82
5.3 选择结构程序设计举例	84
习题	86
第6章 循环结构程序设计	92
6.1 while 语句 (当型循环)	92
6.2 do-while 语句 (直到型循环)	95
6.3 for 语句 (当型循环)	96
6.4 break 和 continue 语句	99
6.5 应用举例	100
习题	101
第7章 数组	108
7.1 一维数组的定义和引用	108
7.1.1 一维数组的定义方式	108
7.1.2 一维数组元素的引用	109
7.1.3 一维数组的初始化	111
7.1.4 一维数组程序举例	112
7.2 二维数组的定义和引用	113



7.2.1	二维数组的定义	113
7.2.2	二维数组元素的引用	114
7.2.3	二维数组的初始化	115
7.2.4	二维数组程序举例	116
7.3	字符数组	117
7.3.1	字符数组的定义	117
7.3.2	字符数组的初始化	117
7.3.3	字符数组的引用	118
7.3.4	字符串和字符串结束标志	118
7.3.5	字符数组的输入/输出	119
7.3.6	字符串处理函数	120
7.4	程序举例	123
	习题	126
第8章	函数	137
8.1	概述	137
8.2	函数定义的一般形式	139
8.3	函数的参数和函数的值	141
8.3.1	形式参数和实际参数	141
8.3.2	函数的返回值	142
8.4	函数的调用	142
8.4.1	函数调用的一般形式	142
8.4.2	函数调用的方式	143
8.4.3	被调用函数的声明和函数原型	144
8.5	函数的嵌套调用	145
8.6	函数的递归调用	146
8.7	数组作为函数参数	149
8.8	局部变量和全局变量	154
8.8.1	局部变量	155
8.8.2	全局变量	156
8.9	变量的存储类别	158
8.9.1	静态存储方式与动态存储方式	158
8.9.2	auto 变量	158
8.9.3	用 static 声明局部变量	159
8.9.4	register 变量	160



8.9.5 用 extern 声明外部变量	160
习题	161
第9章 预处理命令	169
9.1 概述	169
9.2 宏定义	170
9.2.1 无参宏定义	170
9.2.2 带参宏定义	173
9.3 文件包含	178
9.4 条件编译	178
9.5 本章小结	181
习题	181
第10章 指针	187
10.1 地址指针的基本概念	187
10.2 变量的指针和指向变量的指针变量	188
10.2.1 定义一个指针变量	189
10.2.2 指针变量的引用	189
10.2.3 指针变量作为函数参数	193
10.2.4 指针变量几个问题的进一步说明	196
10.3 数组指针和指向数组的指针变量	199
10.3.1 指向数组元素的指针	200
10.3.2 通过指针引用数组元素	201
10.3.3 数组名作函数参数	203
10.3.4 指向多维数组的指针和指针变量	210
10.4 字符串的指针指向字符串的指针变量	213
10.4.1 字符串的表示形式	213
10.4.2 使用字符串指针变量与字符数组的区别	216
10.5 函数指针变量	217
10.6 指针型函数	218
10.7 指针数组和指向指针的指针	220
10.7.1 指针数组的概念	220
10.7.2 指向指针的指针	223
10.7.3 main 函数的参数	224
10.8 有关指针的数据类型和指针运算的小结	226
10.8.1 有关指针的数据类型的小结	226



10.8.2 指针运算的小结	226
习题	227
第 11 章 结构体与共用体	240
11.1 定义一个结构的一般形式	240
11.2 结构类型变量的说明	241
11.3 结构变量成员的表示方法	243
11.4 结构变量的赋值	244
11.5 结构变量的初始化	244
11.6 结构数组的定义	245
11.7 结构指针变量的说明和使用	247
11.7.1 指向结构变量的指针	247
11.7.2 指向结构数组的指针	249
11.7.3 结构指针变量作函数参数	250
11.8 共用体类型	251
11.8.1 共用体的定义	251
11.8.2 共用体变量的引用	252
11.9 动态存储分配	255
11.10 枚举类型	257
11.10.1 枚举类型的定义和枚举变量的说明	257
11.10.2 枚举类型变量的赋值和使用	258
11.11 类型定义符 typedef	259
习题	260
第 12 章 位运算与文件	267
12.1 位运算及其运算符	267
12.2 文件	268
12.2.1 C 语言文件概述	268
12.2.2 文件的打开与关闭	269
12.2.3 文件的读写	271
12.2.3 出错检测	276
习题	277
附录 I 常用字符与 ASCII 码对照表	279
附录 II C 语言中的关键字	280
附录 III 运算符优先级和结合性	281
附录 IV Turbo C (V2.0) 编译错误信息	282



附录V 常用库函数及其标题文件	288
参考文献	293

C语言概述

C语言是继FORTRAN和ALGOL之后，出现的一种高级程序设计语言。它是在B语言的基础上，吸收了ALGOL和PASCAL的一些特点，并加入了一些新的特性，如指针、结构体、共用体、枚举类型、宏定义、位运算等，使其成为一种通用性强、效率高、可移植性好的语言。C语言广泛应用于系统软件、嵌入式系统、数据库系统、网络系统等领域。

本章学习目标

1. 了解C语言的发展历史、特点、应用领域及与其他语言的区别。
2. 掌握C语言的基本数据类型、运算符、表达式、语句、函数、数组、指针、结构体、共用体、枚举类型、宏定义、位运算等。
3. 了解C语言与其他语言的区别，掌握C语言的特点。
4. 了解C语言在系统软件、嵌入式系统、数据库系统、网络系统等领域的应用。

本章重点、难点

重点：C语言的基本数据类型、运算符、表达式、语句、函数、数组、指针、结构体、共用体、枚举类型、宏定义、位运算等。

难点：C语言与其他语言的区别，掌握C语言的特点。

1.1 C语言概述

C语言是在20世纪70年代初由美国贝尔实验室的D.M. Ritchie和B.W. Kernighan等人开发的一种通用的高级程序设计语言。它是在B语言的基础上，吸收了ALGOL和PASCAL的一些特点，并加入了一些新的特性，如指针、结构体、共用体、枚举类型、宏定义、位运算等，使其成为一种通用性强、效率高、可移植性好的语言。C语言广泛应用于系统软件、嵌入式系统、数据库系统、网络系统等领域。

从20世纪80年代开始，C语言在嵌入式系统领域得到了广泛的应用。这是因为C语言具有体积小、效率高、可移植性好等特点，能够满足嵌入式系统对资源的要求。同时，C语言还具有丰富的库函数，能够提供强大的功能。因此，C语言成为嵌入式系统开发的首选语言。

随着C语言的发展，它也在不断地吸收其他语言的特点，使其功能更加强大。例如，C语言引入了结构体、共用体、枚举类型、宏定义、位运算等特性，使其能够处理更复杂的数据结构和操作。此外，C语言还引入了指针，使其能够直接操作内存，提高了程序的效率。这些特性的引入，使得C语言在系统软件、嵌入式系统、数据库系统、网络系统等领域得到了广泛的应用。

第 1 章

C 语言概述

C 语言是编程语言中较为流行的一种。随着计算机的普及和发展, C 语言在各个领域的应用越来越广泛。几乎各类计算机都支持 C 语言的开发环境, 这为 C 语言的普及与应用奠定了基础。本章主要介绍 C 语言的发展过程、特点及一些基本的概念, 以及 C 语言程序的开发过程和 VC6.0 的开发环境的使用。

本章学习目标:

1. 了解 C 语言的发展史、特点、C 语言和 C++ 语言的区别。
2. 掌握 C 语言的关键字、字符集和简单输入/输出函数的特点。
3. 掌握 C 语言程序的结构特点、书写特点及注意事项。
4. 掌握 C 语言开发环境的使用。

本章重点、难点:

1. C 语言程序的结构特点、书写特点及注意事项。
2. C 语言开发环境的使用。

1.1 C 语言概述

C 语言是在 20 世纪 70 年代初问世的。美国电话电报公司 (AT&T) 贝尔实验室于 1978 年正式发表了 C 语言。同时, B. W. Kernighan 和 D. M. Ritchie 合著了著名的《THE C PROGRAMMING LANGUAGE》一书。此书通常简称为 K&R, 也有人称之为 K&R 标准。但是, 在《K&R》中并没有定义一个完整的标准 C 语言, 后来由美国国家标准协会 (American National Standards Institute) 在此基础上制定了 C 语言标准, 于 1983 年发表, 通常称之为 ANSI C。

从 2008 年 4 月开始, 全国计算机等级考试已经全面停止了 Turbo C2.0 (简称 TC) 软件的使用, 所有参加二级 C 语言、三级信息技术、网络技术和数据库技术上机考试的考生, 都要在 Visual C++6.0 (简称 VC6.0) 环境下调试运行 C 程序。众所周知, VC 软件主要用来开发 C++ 程序的, 如何在 VC 下调试运行 C 程序, 对于众多程序设计者来说, 确实是一个新的话题。

VC 与 TC 的差异, 主要体现在程序设计语言 C++ 与 C 的关系上。其实, 了解计算机程序设计的人都知道, C++ 是 C 语言的升级版, VC 是 Windows 进行 C++ 程序设计的主要软件, VC 完全兼容了 C 的功能, 也就是说, 在 VC 下是完全能够进行 C 语言程序开发的。



1.1.1 C 语言的特点

C 语言简洁、紧凑，使用方便、灵活。ANSI C 只有 32 个关键字，见表 1-1。

表 1-1

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

在 C 语言中，有 9 种控制语句，程序书写自由，主要用小写字母表示，压缩了一切不必要的成分。

Turbo C 扩充了 11 个关键字：

asm _cs _ds _es _ss cdecl far huge interrupt near pascal

注意：

- ①在 C 语言中，关键字都是小写的。
 - ②运算符丰富，共有 34 种。C 语言把括号、赋值、逗号等都作为运算符处理，从而使其运算类型极为丰富，可以实现其他高级语言难以实现的运算。
 - ③数据结构类型丰富。
 - ④具有结构化的控制语句。
 - ⑤语法限制不太严格，程序设计自由度大。
 - ⑥C 语言允许直接访问物理地址，能进行位（bit）操作，能实现汇编语言的大部分功能，可以直接对硬件进行操作。因此有人把它称为中级语言。
 - ⑦生成目标代码质量高，程序执行效率高。
 - ⑧与汇编语言相比，用 C 语言写的程序可移植性好。
- 但是，C 语言对程序员要求也高。虽然程序员用 C 语言写程序会感到限制少、灵活性大、功能强，但与其他高级语言相比，学习 C 语言要困难一些。

1.1.2 C 语言源程序的结构特点

- ①一个 C 语言源程序可以由一个或多个源文件组成。
- ②每个源文件可由一个或多个函数组成。
- ③一个源程序不论由多少个文件组成，都有一个且只能有一个 main 函数，即主函数。
- ④源程序中可以有预处理命令（include 命令仅为其中的一种）。预处理命令通常放在源文件或源程序的最前面。
- ⑤每一个说明、每一个语句都必须以分号结尾。但预处理命令、函数头和花括号“{”



之后不能加分号。

⑥标识符、关键字之间必须至少加一个空格以示间隔。若已有明显的间隔符，也可不再加空格来间隔。

1.1.3 书写程序时应遵循的规则

从书写清晰，便于阅读、理解、维护的角度出发，在书写程序时应遵循以下规则：

①一个说明或一个语句占一行。

②用 {} 括起来的部分，通常表示程序的某一层结构。{} 一般与该结构语句的第一个字母对齐，并单独占一行。

③低层次的语句或说明可比高一层次的语句或说明缩进若干格后书写，以便看起来更加清晰，增加程序的可读性。

在编程时应力求遵循这些规则，以养成良好的编程风格。

1.1.4 C语言的字符集

字符是组成语言的最基本的元素。C语言字符集由字母、数字、空格、标点和特殊字符组成。在字符常量、字符串常量和注释中，还可以使用汉字或其他可表示的图形符号。

1. 字母

小写字母 a~z，共 26 个。

大写字母 A~Z，共 26 个。

2. 数字

0~9，共 10 个。

3. 空白符

空格符、制表符、换行符等统称为空白符。空白符只在字符常量和字符串常量中起作用，在其他地方出现时，只起间隔作用，编译程序时可对它们忽略不计。因此，在程序中使用空白符与否，对程序的编译不产生影响，但在程序中适当的地方使用空白符将增加程序的清晰性和可读性。

4. 标点和特殊字符

1.1.5 C语言词汇

在C语言中使用的词汇分为六类：标识符、关键字、运算符、分隔符、常量、注释符等。

1. 标识符

在程序中使用的变量名、函数名、标号等统称为标识符。除库函数的函数名由系统定义外，其余都由用户自定义。C语言规定，标识符只能是由字母 (A~Z, a~z)、数字 (0~9)、下划线 (_) 组成的字符串，并且其第一个字符必须是字母或下划线。

以下标识符是合法的：

a, x, x3, BOOK_1, sum5

以下标识符是非法的：

3s 以数字开头



s * T 出现非法字符 *

-3x 以减号开头

bowy -1 出现非法字符 - (减号)

在使用标识符时,还必须注意以下几点:

①标准 C 语言不限制标识符的长度,但它受各种版本的 C 语言编译系统限制,同时也受到具体机器的限制。例如,在某版本 C 语言中规定标识符前八位有效,当两个标识符前八位相同时,则被认为是同一个标识符。

②在标识符中,大小写是有区别的。例如,BOOK 和 book 是两个不同的标识符。

③标识符虽然可由程序员随意定义,但标识符是用于标识某个量的符号的,因此,命名应尽量有相应的意义,以便于阅读理解,做到“顾名思义”。

2. 关键字

关键字是由 C 语言规定的具有特定意义的字符串,通常也称为保留字。用户定义的标识符不应与关键字相同。C 语言的关键字分为以下几类:

①类型说明符。用于定义、说明变量、函数或其他数据结构的类型。

②语句定义符。用于表示一个语句的功能。

③预处理命令字。用于表示一个预处理命令。

3. 运算符

C 语言中含有相当丰富的运算符。运算符与变量、函数一起组成表达式,表示各种运算功能。运算符由一个或多个字符组成。

4. 分隔符

在 C 语言中采用的分隔符有逗号和空格两种。逗号主要用在类型说明和函数参数表中,分隔各个变量。空格多用于语句各单词之间,作间隔符。在关键字、标识符之间必须要有一个以上的空格符作间隔,否则将会出现语法错误,例如,把“int a”写成“inta”,C 编译器会把“inta”当成一个标识符处理,其结果必然出错。

5. 常量

C 语言中使用的常量可分为数字常量、字符常量、字符串常量、符号常量、转义字符等多种。在后面章节中将专门进行介绍。

6. 注释符

C 语言的注释符有两种:

第一种是以“/*”开头并以“*/”结尾的串。在“/*”和“*/”之间的内容即为注释。当同一注释内容出现在多行上时,则使用这种注释方法。

第二种是以“//”开头的串,在“//”后面的内容即为注释。当注释的内容很短,就出现在一行上时,一般使用这种注释方法,当然,也可以使用第一种注释方法。

程序编译时,不对注释做任何处理。注释可出现在程序中的任何位置。注释用来向用户提示或解释程序的意义。在调试程序过程中,暂不使用的语句也可用注释符括起来,使翻译跳过,不做处理,待调试结束后再去掉注释符。



1.2 编制并运行一个简单程序

1.2.1 编制并运行程序的“四步曲”

本节用 VC6.0 先来编制一个最简单的程序，让它运行（执行）并得出结果，以此作为了解 VC6.0 的开端。这个程序的功能仅仅是向屏幕上输出一个字符串“Hello World”。程序虽短，但与编制运行大程序的整个过程是相同的，都包含着如下“四步曲”：

①编辑（把程序代码输入，交给计算机）。

②编译（成为目标程序文件 .obj）。编译就是把高级语言变成计算机可以识别的二进制语言，计算机只认识 1 和 0，编译程序把人们熟悉的语言换成二进制的。编译程序把一个源程序翻译成目标程序的工作过程分为五个阶段：词法分析；语法分析；语义检查和中间代码生成；代码优化；目标代码生成。主要是进行词法分析和语法分析，又称为源程序分析，分析过程中发现有语法错误时，给出提示信息。

③连接（成为可执行程序文件 .exe）。连接是将编译产生的 .obj 文件和系统库连接装配成一个可以执行的程序。由于在实际操作中可以直接单击“Build”，从源程序产生可执行程序，可能有人就会质疑：为何要将源程序翻译成可执行文件的过程分为编译和连接两个独立的步骤？不是多此一举吗？之所以这样做，主要是因为：在一个较大的复杂项目中，有很多人共同完成一个项目（每个人可能承担其中的一部分模块），其中有的模块可能是用汇编语言写的，有的模块可能是用 VC 写的，有的模块可能是用 VB 写的，有的模块可能是购买的（不是源程序模块，而是目标代码）或已有的标准库模块，因此，各类源程序都需要先各自编译成目标程序文件（机器指令代码），再通过连接程序将这些目标程序文件连接装配成可执行文件。

④运行（可执行程序文件）。

上述四个步骤中，其中第一步的编辑工作是最繁杂而又必须由人工在计算机上细致地完成的，其余几个步骤则相对简单，基本上由计算机来自动完成。

1.2.2 工程（Project）以及工程工作区（Project Workspace）

在开始编程之前，必须先了解工程（Project）的概念。工程又称为项目，它有两种含义：一种是指最终生成的应用程序，另一种则是指为了创建这个应用程序所需的全部文件的集合，包括各种源程序、资源文件和文档等。绝大多数较新的开发工具都利用工程来对软件开发过程进行管理。

用 VC6.0 编写并处理的任何程序都与工程有关（都要创建一个与其相关的工程），而每一个工程又总与一个工程工作区相关联。工作区是对工程概念的扩展。一个工程的目标是生成一个应用程序，但很多大型软件往往需要同时开发数个应用程序，VC 开发环境允许用户在一个工作区内添加数个工程，其中有一个是活动的（缺省的），每个工程都可以独立进行编译、连接和调试。

实际上，VC6.0 是通过工程工作区来组织工程及其各相关元素的，就像是一个工作间（对应于一个独立的文件夹，或称子目录），以后程序所涉及的所有文件、资源等元素都将放入这一工作间中，从而使各个工程之间互不干扰，使编程工作更有条理、更具模块化。最