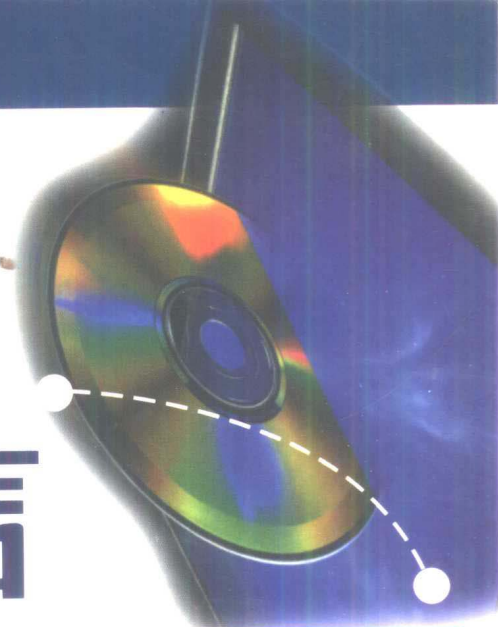


◎ 伍晓宇 辛 勇 编著



语言



与 研 究 开 发 实 践



航空工业出版社

C/C++语言与项目开发实践

伍晓宇 辛勇 编著

航空工业出版社

内 容 提 要

本书包括四个部分：第一部分：C 程序设计；第二部分：C++程序设计；第三部分：Visual C++ 可视化程序设计；第四部分：研究开发专题。

前两部分主要是标准 C/C++ 程序设计的学习，介绍的重点放在语言本身的掌握和使用上，尽量避免较复杂的算法，本书的特点在于强调 C++ 非面向对象内容的学习和正确理解，可使读者顺利向各种面向对象概念进行平稳的过渡。

可视化程序设计部分详细讨论了 Windows 程序工作原理及其消息处理机制、Wizard 技术的应用、文档/视窗模型以及各种常用程序设计技术。

第四部分收集了作者近年来在科研工作中的一些研究开发实例，经过简化后总结于本书中，可供科研开发人员及师生参考。例如深入分析了 C/C++ 程序与其相应汇编代码之间的关系，使读者能从更深层次上理解和掌握 C/C++ 的精髓。作者提出了应用 C/C++ 程序自动生成汇编代码这一技术可提高开发工作的效率和程序的稳定性。最后以浮动许可证设计为题，详细讨论了 TCP/IP 网络的 Socket 编程技术和以 Windows 操作系统为平台的网络加密系统的设计技术。

本书可作为大学理工科专业本科生教材，也可供有关工程技术人员和科研开发人员参考。

图书在版编目 (CIP) 数据

C/C++ 语言与研究开发实践 / 伍晓宇, 辛勇编著. — 北京: 航空工业出版社, 1999

ISBN 7-80134-552-5

I. C… II. ①伍… ②辛… III. C 语言-程序设计 IV. T
P312

中国版本图书馆 CIP 数据核字 (1999) 67885 号

航空工业出版社出版发行

(北京市安定门外小关东里 14 号 100029)

河北省香河印刷厂印刷 全国各地新华书店经售

2000 年 1 月第 1 版

2000 年 1 月第 1 次印刷

开本: 787×1092 1/16 . 印张: 18 字数: 448 千字

印数: 1—3000

定价: 28.80 元

序

近年来，在国内各高校的理工科类专业广泛设置了 C 语言及其相关课程的教学。但随着计算机软件的飞速发展，原有的 C 语言教学体系已不能适应需要，极有必要对其教学内容进行调整和完善。作为 C/C++ 语言的教材，这本书是我所见到的书中比较适用的著作之一。因为本书有下列几个特点：

1. 紧密结合当前高等教育改革的形势要求，为学生增加授课学时以外的动手机会，本书收集了作者近年来在科研工作中的一些研究开发实例，经过简化后作为学生科研开发实践的研究专题，也可作为选修课的教学内容，使学生在掌握 C/C++ 的基本知识的基础上，能够尝试着应用这些基本知识去分析和解决各种实际工程问题。

2. 作者针对近年来在 C/C++ 教学过程中所存在的问题，加强了 C++ 非面向对象内容的学习和正确理解，以便使学生较顺利地完成 C++ 面向对象概念的过渡。

3. 为避免课堂学习与将来实际应用之间的脱节，并针对目前的可视化程序设计技术非常流行的形势，本书增加了相关的学习内容，并较好地将它与 C/C++ 的基本内容衔接，完成从传统设计技术向可视化设计技术的过渡。

4. 书中深入分析了 C/C++ 程序与其相应汇编代码之间的关系，使读者能从更深层次上理解和掌握 C/C++ 语言精髓，作者提出了汇编代码设计的 C/C++ 辅助技术，在实际开发工作中具有重要意义。

5. 在计算机网络和 Windows 操作系统非常流行的今天，过去长期使用的各种软件保护技术已不能适应实际工作的需要，作者设计的浮动许可证加密系统，以 TCP/IP 网络协议和 Windows 操作系统为应用平台，同时也可以很容易地移动到 UNIX 等其它操作系统中，因此，该系统具有广泛的适用性。

我相信本书的出版将会对同“C/C++ 程序设计”有关的教学和研究与开发工作起到一定促进作用。我也深知作者诚挚地欢迎读者对本书提出批评与建议。

谨为之序。

杨叔之

中国科学院院士

华中理工大学教授

一九九九年十一月十一日

前 言

近年来,在国内各高校的许多专业,尤其是理工科类专业广泛设置了C语言课程的教学。这是因为在实际工作中C语言已不仅为计算机专业工作者所使用,而且开始为广大非计算机专业出身的工程技术人员所喜爱和使用,成为理工科各专业必备的工具。学习C语言已成为广大计算机应用人员和初学者的迫切要求。

但是,随着计算机软件技术的发展,特别是可视化程序设计技术的广泛应用,原有的C语言教学体系已不能适应需要,应该对其教学内容进行修改、调整和充实,并加入C++和C/C++可视化程序技术等内容。

目前,高等教育改革要求压缩课堂授课学时,增加授课学时以外的动手机会,使学生不仅能够完整全面地掌握C/C++的基本知识,同时能够初步应用这些基本知识尝试去分析和解决各种实际工程问题。

针对以上问题,本书包括以下几个部分:

第一部分: C 程序设计

本部分是全书的基础,作者把介绍的重点放在语言本身的掌握和使用上,尽量避免较复杂的算法,因为算法本身不属于语言学习的范畴。该部分几乎所有的实例都是短小和简练的,力求在初步掌握基本知识的基础上,使学生上机时能尽快通过实例调试,以增强学习的信心和兴趣。

本书对目前国内某些C语言教材的一些不确切观点进行了修正,例如,在PC机上关于short、int型数据是否有区别,register变量是否能起实际作用等。另外,对于过去教学中没有提及的一些问题进行了探讨,如PC机上指针通常的实际长度等。本书还指出了Turbo C和Visual C++在处理个别问题上的不同点。

第二部分: C++程序设计

以往我们在C++程序设计学习过程中,往往强调C++面向对象技术内容的一面,而忽略它的非面向对象技术部分。作者认为C++非面向对象技术的内容是学习其面向对象技术内容的基础。对于C++程序设计学习过程中的困难,常常是因为对某些非面向对象特性不能较好理解造成的。本书强调C++非面向对象内容的学习和正确理解,例如强调结构体是类的子集。应在正确理解C++结构体的基础上,顺利向类的概念进行平稳的过渡,这样可以收到较好的学习效果。

第三部分: Visual C++可视化程序设计

可视化程序设计是近年来非常流行的程序开发技术,由于它大大提高了设计效率和程序的稳定性,因此得到了广泛应用。但目前在各高校的C/C++语言教学中,它一般不属于课堂教学的范畴,这就造成了课堂学习与学生将来实际应用之间的脱节。因此,在本书中我们将该部分的基本内容加上。

可视化程序设计与传统的程序设计在基本概念上有很大的区别，本部分的目的在于帮助读者在学习标准的 C/C++ 基本知识的基础上，完成从传统设计技术向可视化设计技术的过渡。

第四部分：研究开发专题

该部分收集了作者近年来在科研工作中的一些研究开发实例，经过简化后总结于本书中，可供科研开发人员参考，也可作为学生在完成 C/C++ 的课程学习之后，进一步学习提高之用。

第 2 4 章“C/C++ 程序深入剖析”从编译系统的角度，对 C/C++ 执行程序的内存映象结构进行了分析，并以 Turbo C 和 Borland C++ 编译器为实例，深入分析了 C/C++ 程序与其相应汇编代码之间的关系，使读者能从更深层次上理解和掌握 C/C++ 语言的精髓。在该章中，作者提出了应用 C/C++ 程序自动生成汇编代码这一技术可大大提高开发工作的效率和程序的稳定性。

第 2 5 章“PC 机存储器型卡程序设计”充分利用第 24 章的成果，将其应用于电路板卡的程序设计工作中。

第 2 6 章浮动许可证加密系统(Floating License)设计，以作者为新加坡国立大学所属某软件公司开发的网络浮动加密系统为原型，详细讨论了 TCP/IP 网络的 Socket 编程技术、计算机 ID 号的构造、浮动许可证工作原型和设计的基本技术等。读者在学习基本原理的基础上，可以开发自己的系统。

本书由深圳大学组织编写，其中第一、二、四部分由深圳大学伍晓宇编写，第三部分由南昌大学辛勇编写，全书由伍晓宇统编和修改，由南昌大学杨国泰教授审稿。作者十分感谢华中理工大学同窗学友李建军博士和吴波教授为本书所提出的许多建设性的意见。另外，深圳大学学生刘斌、胡勇、冯俊秋、储昕、奚荣斌和张顺娣等在参加作者指导的科研实践和毕业设计中作了不少与本书有关的工作，在此向他们表示感谢。

由于作者水平有限，肯定会有不少缺点或错误，希望得到专家和读者批评指正。

伍晓宇 辛勇

1999 年 9 月

第一部分 C 程序设计**1**

第1章 C语言概述	2
1.1 C语言来源与特点	2
1.2 C语言的结构	3
第2章 常量、变量、运算符与表达式	5
2.1 C语言的基本数据类型	5
2.2 常量	7
2.3 变量	9
2.4 各类数值型数据之间的混合运算	14
2.5 运算符与表达式	15
第3章 程序控制语句	21
3.1 条件分支语句	21
3.2 循环控制语句	22
3.3 开关语句与跳转语句	24
第4章 数组	28
4.1 一维数组	28
4.2 二维数组	30
4.3 字符数组	31
第5章 指针	33
5.1 定义与引用	33
5.2 一维数组与指针的关系	34
5.3 字符串指针	35
5.4 指针内存分配	36
5.5 多维数组与指针的关系	38
5.6 指针数组和指向指针的指针	40
第6章 函数	43
6.1 函数的定义	43
6.2 函数的说明	44

6.3 函数的调用	45
6.4 递归调用	49
6.5 命令行参数	51
6.6 函数指针	52
6.7 函数变参	54
第7章 结构体	56
7.1 结构体的定义	56
7.2 结构体的引用	57
7.3 结构体的初始化	58
7.4 指向结构体的指针	59
7.5 共用体（联合体）	61
7.6 枚举类型	63
7.7 自定义类型	64
7.8 位段	66
第8章 编译预处理	68
8.1 宏定义	68
8.2 文件包含处理	70
8.3 条件编译	70
第9章 文件 I/O	72
9.1 文件的操作模式	72
9.2 文件与设备的读写	73

第二部分 C++程序设计

81

第10章 C++的非面向对象特性(C的扩展)	82
10.1 C++的输入和输出	82
10.2 new 和 delete	82
10.3 const	83
10.4 注解	84
10.5 缺省的函数形式参数	84
10.6 强制类型转换	84

10.7 引用(reference)	85
10.8 函数原型(说明)	87
10.9 结构体(struct)与共用体(union)	87
第11章 类	94
11.1 类与对象(变量)	94
11.2 静态成员	100
第12章 派生类	103
12.1 一般派生	103
12.2 虚函数	109
12.3 虚基类	112
第13章 友元	114
13.1 友元函数	114
13.2 友元类	114
第14章 运算符重载	116
14.1 一元运算符	116
14.2 二元运算符	117
第15章 流库	122
15.1 流类的结构	122
15.2 设备操作符	123
15.3 ostream 和 istream 成员函数的应用	124
15.4 文件管理	126
第16章 模板	132
16.1 函数模板	132
16.2 类模板	133
第三部分 Visual C++可视化程序设计	137
第17章 Visual C++可视化编程的基本思想	138
17.1 MFC 编程特点	138

17.2 VC 工程文件的构成	138
17.3 AppWizard 工具	139
17.4 ClassWizard 工具	140
17.5 Resource Editor 工具	142
第 18 章 MFC 程序结构分析	144
18.1 Windows 程序工作原理	144
18.2 建立应用程序	144
18.3 程序结构剖析	145
18.4 显示文本	148
18.5 显示图形	149
18.6 显示位图	150
18.7 文档的串行化	151
18.8 直接显示数据	152
18.9 保持显示数据	152
第 19 章 消息处理过程	155
19.1 消息处理机制	155
19.2 鼠标消息	155
19.3 键盘消息	156
19.4 窗口消息	156
19.5 其它消息	158
19.6 自定义消息	160
第 20 章 菜单	163
20.1 菜单的处理机制	163
20.2 菜单项的增加	163
20.3 菜单项的更新控制	165
第 21 章 对话框	167
21.1 生成对话框资源	167
21.2 对话框的数据交换	167
第 22 章 常用按键	170
22.1 进度条	170

22.2	滑动条	171
22.3	Up-down 控制	173
22.4	工具条	174
22.5	状态条	175
22.6	部件库	177
22.7	ActiveX 控件简介	183
第 23 章	多任务编程初步	187
23.1	多进程的启动	187
23.2	多线程的启动	187
23.3	优先级的设定	188
23.4	任务间通信与同步	190
第四部分 专题应用研究		195
第 24 章	C/C++ 程序深入剖析	196
24.1	C 程序对应汇编代码分析	196
24.2	C++ 程序对应汇编代码分析	208
24.3	C/C++ 语言辅助汇编程序设计	217
第 25 章	PC 机存储器型卡程序设计	230
25.1	存储型卡驻留程序的基本格式	230
25.2	测试电路及其写入程序	231
25.3	ROM 文件的准备	232
第 26 章	浮动许可证加密系统	235
26.1	计算机 ID 号的构造	235
26.2	Floating License 核心技术	238
26.3	License 状态的管理	255
26.4	网络防火墙与远程监控	272
附录: ASCII 字符代码		274
参 考 文 献		278

第一部分

C 程序设计

第 1 章 C 语言概述

第 2 章 常量、变量、运算符与表达式

第 3 章 程序控制语句

第 4 章 数组

第 5 章 指针

第 6 章 函数

第 7 章 结构体

第 8 章 编译预处理

第 9 章 文件 I/O

第 1 章 C 语言概述

1.1 C 语言来源与特点

一、来源

C 语言是 UNIX 操作系统的主程序设计语言, 最初 UNIX 是用汇编语言写的, 存在: 1) 移植性差; 2) 开发效率低; 3) 难读等问题。因此, 人们希望找到一种既像汇编语言一样能对硬件进行直接操作, 同时又具有高级语言的优点(包括可读性好, 移植性强等)。这类语言介于高级语言和低级语言之间, 称为中级语言。

C 语言是一种中级语言。它的发展过程大体如下:

1970 年贝尔实验室的 K. Thompson 设计了 B 语言, 并用它来改写 UNIX 系统。

1972 年贝尔实验室的 D. M. Ritchie 针对 B 语言过于简单、功能不强和生成的解释代码运算速度慢的弱点, 在 B 语言的基础上设计了 C 语言。

1973 年两人合作用 C 语言重写了 UNIX 操作系统, 原来的 UNIX 系统是他们 1969 年用汇编语言合写的。

现在 UNIX 系统 90% 的代码是 C, 广泛用于各种工作站和小型机, 如 HP、Sun、RISC/6000 等, 并产生了很多变种, 如 Aegis、AIX 等。

70 年代后期微型机迅速发展, C 语言开始移植到非 UNIX 系统上, 已独立于 UNIX。

1983 年, 美国国家标准协会制定了 C 语言标准 —— ANSI C。

目前在 PC 机上比较流行的 C 编译系统有:

MSC, Visual C++

Turbo C 2.0, 3.0. Turbo C++, Borland C++

High C++

Watcom C

二、特点

优点:

1. 表达能力强, 设计自由度大。既能完成普通高级语言的功能, 又能对计算机硬件进行直接访问, 而且能直接对物理地址进行操作或进行字位级操作, 几乎没有限制。

2. 代码紧凑、高效、书写简洁。

源代码中, 仅有四种基本数据类型, 32 个关键字。

目标代码中, 执行效率仅比汇编程序低 10~20%, 编译代码小。

3. 良好的移植性。

缺点:

安全性差,表现在数据和指针没有屏障;类型转换比较随便;数组越界及变量类型一致性不作语法上的严格检查。

但要辩证地看待上述问题,应强调程序员自己去保证程序的正确性。以下是几种常用计算机语言的应用范围。

Basic	——	初学者
Fortran	——	科学计算
Pascal	——	学生
COBOL	——	银行、金融部门
C	——	程序员

1.2 C 语言的结构

/* [例 1-1] 最简单的 C 程序之一, 求两整数和*/

/* sampl_1.c */

void main()

{ int a, b, c;

a=1; b=2;

c=a+b;

printf("c=%d\n", c);

}

运行结果:

c=3

/* [例 1-2] 从键盘输入两整数, 并且显示较大者*/

/* sampl_2.c */

#include<stdio.h> /* 头文件 stdio.h, 它包含标准输入输出库函数的等信息的说明

int max(int x, int y); /* 函数 max() 的说明(申明) */

/* 主函数定义 */

void main()

{ int a, b, c; /* {} 内为函数体, 本行为其中变量定义部分, 以下为执行部分*/

/* 调用 ANSI 标准库函数 scanf, 键盘输入变量 a、b 值*/

scanf("%d,%d",&a,&b);

c=max(a,b); /* 调用自编函数 max, 其中 a、b 为实参*/

printf("max=%d", c); /* 调用 ANSI 标准库函数 printf, 显示变量 c 的值*/

}

/* 自编函数 max 的定义 */

int max(int x, int y) /*函数说明部分:函数类型 int, 函数名 max, 函数形参 x, y 及其类型*/

{ int z; /* {} 范围内为函数体, 本行为其中的变量定义部分 */

```
if(x>y) z=x;      /* 以下为执行部分 */  
else z=y;  
return z;         /* 返回函数值 */  
}
```

运行结果:

3,4 (从键盘上输入两整数)

max=4

通过以上例子可以看出:

1. C 程序是由函数构成的

至少包含一个主函数(main 函数), 也可以包含一个 main 函数和若干个子函数。

函数包括两种:

1) 库函数。它包括 ANSI 标准函数和非 ANSI 标准函数。为了保证程序的良好移植性, 应尽可能使用标准函数。

2) 自编函数。它包括一个 main 函数和若干个子函数。

2. 一个函数的定义由函数的说明部分和函数体两部分组成

1) 函数的说明部分包括函数类型、函数名、函数参数(形参)名和形式参数类型等。

2) 函数体是函数说明部分后的大括号 {.....} 之间的部分。它一般包括: a) 变量定义;

b) 执行部分; c) 返回函数值。

此外, 在程序的开头还常常有自编函数的(说明)申明以及说明库函数的头文件等。

3. 关于书写格式

1) main 函数位置任意, 但习惯上放在其它函数前面。

2) 每条语句的最后必须有一个分号。

3) 一行内可以写几个语句, 一个语句也可以分写在多行上。

4) 程序的注释部分应括在 /* / 之间, / 和 * 之间不允许留有空格; 注释部分允许出现在程序中的各个位置上, 但在双引号 "" 内无效。

5) C 语言标识符的大小写是不同的(或称大小写敏感)。

第2章 常量、变量、运算符与表达式

2.1 C语言的基本数据类型

C语言的基本类型可以分为：

1. 字符型。它以 char 表示。

表 2-1

数据类型	Windows 表示	字节数	位数	取值范围
char	CHAR	1	8	-128~+127
short	SHORT	2	16	-32768~+32767
int	INT	4	32	-2147483648~+2147483647
long	LONG	4	32	-2147483648~+2147483647
unsigned char	UCHAR	1	8	0~+256
unsigned short	USHORT	2	16	0~+65535
unsigned int	UINT	4	32	0~+4294967295
unsigned long	ULONG	4	32	0~+4294967295
float	FLOAT	4	32	-1.0E38~+1.0E38 (8位有效数字)
double	DOUBLE	8	64	-1.0E308~+1.0E308 (16位有效数字)

2. 整型。它又可分为：

短整型，以 short 或 short int 表示；

普通整型，以 int 表示；

长整型，以 long 或 long int 表示。

在 PC 机上的 DOS 操作系统中，一般无 int 与 short 型的区别，两者是一致的，但对于 PC 机上的 Windows 系统，或者其它机种上的各种操作系统，两者一般是有区别的。

对于字符型和整型，可以将其存储单元中全部二进位(bit)用作存放数本身，而不包括符号，构成无符号字符型和无符号整型。包括：

```
unsigned char
unsigned short
unsigned int
unsigned long
```

3. 浮点型(实型)。它又可分为:

单精度型。它以 float 表示;

双精度型。它以 double 表示。

4. 无值型(void)

ANSI 标准没有具体规定以上各类数据类型所占内存字节数, 各种机器或操作系统处理上有所不同。表 2-1 是 PC 机上 Windows 系统(Win 3.x, Win95/97/98)下, 各种数据类型的有关信息列表。可以用标准库函数 sizeof 计算数据类型的长度进行验证, 参见以下实例。

/* [例 2-1] 计算各种基本数据类型的字节长度*/

/* samp2_1.c */

#include<stdio.h>

#include<windows.h>

void main()

```
{    printf( "Size of char  : %d\n",    sizeof(char) );
    printf( "Size of short : %d\n",    sizeof(short) );
    printf( "Size of int   : %d\n",    sizeof(int) );
    printf( "Size of long  : %d\n",    sizeof(long) );
    printf( "Size of float : %d\n",    sizeof(float) );
    printf( "Size of double: %d\n\n",  sizeof(double) );
```

/* 以下求 Windows 表示法的数据类型长度, 若在 Turbo C 等 DOS 操作系统上的编译器中编译时, 需将下列部分删除或注释 */

```
    printf( "Size of CHAR  : %d\n",    sizeof(CHAR) );
    printf( "Size of SHORT : %d\n",    sizeof(SHORT) );
    printf( "Size of INT   : %d\n",    sizeof(INT) );
    printf( "Size of LONG  : %d\n",    sizeof(LONG) );
    printf( "Size of FLOAT : %d\n",    sizeof(FLOAT) );
    printf( "Size of DOUBLE: %d\n\n",  sizeof(DOUBLE) );
}
```

以上程序应用 Visual C++ 编译后, 运行可得以下结果:

```
Size of char  : 1
Size of short : 2
Size of int   : 4
Size of long  : 4
Size of float : 4
```