

新世纪 标准C++程序设计

培训教程

郝阿朋 韩 杰 等编著

- ◆ 从零起步，快速入门
- ◆ 循序渐进，内容全面
- ◆ 示例明确，练习丰富



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

<http://www.phei.com.cn>



新世纪标准 C++程序设计 培训教程

郝阿朋 韩 杰 等编著

電子工業出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 提 要

本书是专为 C++ 语言的初学者而设计的,旨在帮助读者快速地掌握 C++ 语言。本书由浅入深,循序渐进,对一些概念和规则的说明简单明了。全书共分 14 章,全面而系统地介绍了 C++ 语言程序设计的基本概念和方法,包括基本运算、数据类型、程序结构、函数、类与对象、预处理指令和文件处理等内容,重点介绍了类和对象的概念及其设计使用方法。为增强直观性,书中给出了许多完整并带有注释、运行结果和分析的程序例子。所有例子均不涉及复杂的物理和数学背景,算法也很简单,主要用于说明 C++ 语言的语法规则和使用特性。为便于教学和读者练习,每章之前有教学目标 and 重点难点说明,每章之后附有上机练习和习题。书中所有例题均在 Visual C++ 6.0 开发环境中调试通过。

本书可以作为 C++ 语言教学的培训教材,也可作为大专院校计算机及相关专业师生的参考资料。

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

新世纪标准 C++ 程序设计培训教程 / 郝阿朋等编著. —北京: 电子工业出版社, 2002.8
(新世纪电脑培训教程)

ISBN 7-5053-7851-1

I. 新... II. 郝... III. C 语言—程序设计—技术培训—教材 IV. TP312

中国版本图书馆 CIP 数据核字 (2002) 第 054092 号

责任编辑: 祁玉芹

印 刷: 北京鑫正大印刷有限公司

出版发行: 电子工业出版社 <http://www.phei.com.cn>

北京市海淀区万寿路 173 信箱 邮编 100036

经 销: 各地新华书店

开 本: 787×1092 1/16 印张: 25.5 字数: 600 千字 附光盘 1 张

版 次: 2002 年 8 月第 1 版 2002 年 8 月第 1 次印刷

印 数: 6000 册 定价: 38.00 元

凡购买电子工业出版社的图书,如有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系。联系电话: (010)68279077

出版说明

电脑作为一种工具，已经广泛地应用到现代社会的各个领域，正在改变各行各业的生产方式以及人们的生活方式。在进入新世纪之后，不掌握电脑应用技能就跟不上时代，这已成为不争的事实。因此，如何快速、经济地获得使用电脑的知识和技术，并将所学应用于现实生活和实际工作中，就成为新世纪每个人迫切需要解决的新问题。

为适应这种需求，各种电脑应用培训班应运而生，各类学校也将非计算机专业学生的电脑知识和技能教育纳入教学计划，并陆续出台了相应的教学大纲。然而，目前市场上的电脑图书虽然种类繁多，但适合我国国情、学与教两相宜的教材却很少。

《新世纪电脑应用培训教程》丛书就是我们约请熟悉并有丰富电脑教学与培训经验的教师和学者进行深入研讨，并对正在参加培训和将要参加培训的用户进行广泛调查的基础上组织编写的一套系列丛书。

本丛书是为所有从事电脑教学的老师和需要接受电脑应用技能培训或自学人员编写的，可作为各类电脑培训班的培训教材与电脑初、中级用户的自学参考书，也可用做高等学校的公修电脑教材，以及各类职业学校的(就业)专业教材。

本丛书紧密结合培训与课堂教学的特点，针对广大初、中级读者电脑基础知识薄弱的现状，突出基础知识和实践指导方面的内容。此外，在每本书中每章开头明确地指出本章的教学目标和重点、难点，既有助于教师抓住重点确定自己的教学计划，又利于读者课后在实践中自学。

本丛书追求语言严谨、通俗、准确，专业词语全书统一，操作步骤明确且描述方法一致，尽量避免晦涩难懂的语言与容易产生歧义的描述。

电脑的软硬件日新月异，功能非常强大，而电脑培训只有短短几十个学时，教师很难将所有的知识传授给学生。中国有句老话：“师傅引进门，修行在个人”。针对这种情况，本丛书在安排内容时，先讲解一些基础概念和基本功能，为实践和自学打下基础；然后介绍一些在此基础上的典型应用实例，使读者能够在实践中掌握所学内容，不断提高应用水平。如果读者能够正确回答每章最后提出的问题，并能够顺利完成练习，就基本上掌握了这一章所描述的应用技能。这就是本丛书最终要达到的目的。

本丛书采用任务驱动的方式安排内容，而每一种软件的功能都有其限制性。绝大多数

ADS 70/03

情况下，为了完成某一任务，往往需要综合使用多种相关软件。因此，本丛书中的每种书中通常组合多种软件。目前本丛书所涉及到的行业主要有程序设计、网络与数据库的管理与开发、平面与三维设计、网页设计、专业排版、多媒体制作，以及建筑、机械等电脑应用最为密集的行业。涉及的软件也基本上涵盖了目前的各种经典主流软件与流行面虽窄但技术重要的软件。本丛书对于软件版本的选择原则是：紧跟软件更新步伐，以最近半年新近推出的成熟版本为选择的重点；对于兼有中英文版本的软件，尽量舍弃英文版而选用中文版，充分保证图书的技术先进性与应用的普及性。

我们的目标是为所有读者提供读得懂、学得会、用得巧的电脑培训和自学教程，我们期盼着每个阅读本丛书的读者成功。

电子工业出版社

前 言

C++语言是在C语言基础上发展起来的面向对象程序设计语言,保持了C语言的特点。它既支持面向对象程序设计,又支持过程化设计,可用于开发各类系统软件和应用软件。因此深受各类程序设计者的喜爱,并且是大专院校计算机及相关专业必学的一门课程。与其他高级语言相比,C++语言具有不可替代的优点。例如,它支持编程人员在“低级”水平上操作计算机,即可以像汇编语言一样直接与机器硬件接口打交道。它的类和对象功能,为编程人员提供了面向对象设计的编程机制。

本书是专为C++语言的初学者而设计的,旨在帮助读者快捷地掌握C++语言。全书共分14章和两个附录,全面系统地介绍了C++语言程序设计的基本概念和方法,主要内容包括基本运算、数据类型、程序结构、函数、类与对象和文件处理等。

本书在写法上根据初学者的特点,由浅入深,循序渐进,对一些概念和规则的说明简单明了。为增强直观性,书中给出了许多完整的带有注释、运行结果和分析的程序例子。所有例子均不涉及复杂的物理和数学背景,算法也很简单,主要用于说明C++语言的语法规则和使用特性。为便于教学和读者练习,每章之前有教学目标和重点难点说明。本教程在编写时采用任务驱动的方式,每章围绕实例展开。通过简明易学的例子介绍C++程序设计的基础知识,同时每章之后附有上机练习和习题,可以帮助读者巩固所学内容,从而达到自学的目的。作者结合本人的开发经验,在书中融入了一些开发技巧,对读者提高应用水平有一定帮助。书中所有例题均通过Microsoft Visual C++ 6.0(书中简称为Visual C++ 6.0)开发环境中调试。本书的附录B介绍了安装Visual C++ 6.0开发系统的方法。

本书写作的立足点是学习标准C++语言,尽管选择了流行的Visual C++ 6.0作为本书示例和练习的调试运行环境,但是,Visual C++ 6.0不作为本书的重点。

本书可以作为C++教学的培训教材,也可作为广大计算机编程爱好者的自学教材。

本书的第1~12章由郝阿朋同志编写,第13~14章及附录由韩杰同志编写。参加本书编写工作的还有袁丽、秦绪起、林红、卫东、林静、郝曙光、王志远、吕峰、李军、钱炜、孙成军及常明亮等同志。

由于编者水平所限,书中错误和疏漏之处在所难免,希望专家和读者朋友及时指正。

我们的E-mail地址: qiyuqin@sina.com。

作 者
2002年7月

目 录

第 1 章 C++简介.....	1
1.1 C++语言的发展及特点.....	2
1.1.1 C 语言的出现与发展.....	2
1.1.2 面向对象程序设计.....	3
1.1.3 C++语言的特点	3
1.2 C++程序的设计风格.....	4
1.2.1 简单程序示例.....	4
1.2.2 C++程序的书写格式.....	7
1.2.3 main()函数	9
1.2.4 C++程序的基本结构.....	9
1.3 基本符号、标识符和关键字.....	11
1.3.1 基本符号	11
1.3.2 标识符.....	11
1.3.3 关键字.....	12
1.4 程序设计中的几个基本概念.....	13
1.5 使用 Visual C++ 6.0	14
1.5.1 Visual C++ 6.0 简介	14
1.5.2 安装和启动.....	15
1.5.3 输入、编辑、连接和运行 C++程序.....	15
1.5.4 错误处理.....	21
1.5.5 存盘和打开.....	22
1.5.6 退出 Visual C++ 6.0 系统	23
1.6 本章小结.....	23
1.7 上机练习.....	24
1.8 习题.....	24
第 2 章 基本数据类型	25
2.1 常量与变量.....	26
2.1.1 常量.....	26
2.1.2 变量.....	26
2.2 整数类型.....	27
2.2.1 整数类型的取值.....	27

2.2.2 整型常量.....	28
2.2.3 整型变量的使用.....	28
2.3 浮点数类型.....	29
2.3.1 浮点数类型的取值.....	29
2.3.2 浮点型常量.....	30
2.3.3 浮点型变量的使用.....	30
2.4 字符类型.....	31
2.4.1 字符类型的取值.....	31
2.4.2 字符型常量.....	31
2.4.3 字符串常量.....	33
2.4.4 字符型变量的使用.....	33
2.5 cout 输出和 cin 输入.....	34
2.5.1 cout 输出.....	34
2.5.2 cin 输入.....	39
2.6 本章小结.....	41
2.7 上机练习.....	41
2.8 习题.....	42
第 3 章 基本运算.....	45
3.1 算术运算.....	46
3.1.1 二元算术运算.....	46
3.1.2 一元算术运算.....	48
3.1.3 自增和自减运算.....	49
3.2 赋值运算.....	51
3.3 关系运算.....	52
3.4 逻辑运算.....	54
3.5 位运算.....	56
3.6 复合赋值运算.....	60
3.7 其他基本运算.....	62
3.7.1 sizeof 运算.....	62
3.7.2 条件运算.....	64
3.7.3 顺序运算.....	65
3.8 类型转换.....	65
3.8.1 表达式类型转换.....	66
3.8.2 赋值类型转换.....	66
3.8.3 强制类型转换.....	66
3.9 运算符的优先级与结合性.....	67
3.10 本章小结.....	69
3.11 上机练习.....	70

3.12 习题.....	73
第4章 程序结构及语句.....	75
4.1 顺序结构及其语句.....	76
4.1.1 声明语句.....	76
4.1.2 表达式语句.....	76
4.1.3 函数调用语句.....	77
4.1.4 输入输出语句.....	78
4.2 选择结构及其语句.....	79
4.2.1 条件语句.....	79
4.2.2 多路分支语句.....	84
4.3 循环结构及其语句.....	87
4.3.1 for 循环语句.....	87
4.3.2 while 循环语句.....	90
4.3.3 do_while 循环语句.....	92
4.4 辅助控制语句.....	94
4.4.1 无条件转向语句.....	94
4.4.2 跳出语句.....	96
4.4.3 继续语句.....	97
4.5 本章小结.....	98
4.6 上机练习.....	98
4.7 习题.....	101
第5章 数组.....	103
5.1 一维数组.....	104
5.1.1 一维数组的声明.....	104
5.1.2 一维数组的使用.....	105
5.2 二维数组.....	111
5.2.1 二维数组的声明.....	111
5.2.2 二维数组的使用.....	112
5.3 字符数组.....	117
5.3.1 字符数组的声明.....	117
5.3.2 字符数组的使用.....	118
5.4 本章小结.....	122
5.5 上机练习.....	123
5.6 习题.....	125
第6章 指针.....	127
6.1 变量与指针.....	128

6.2	指针变量的声明.....	129
6.3	取地址和取内容运算.....	130
6.3.1	取地址运算.....	130
6.3.2	取内容运算.....	131
6.4	指针变量与数组.....	132
6.4.1	指针变量与一维数组.....	133
6.4.2	指针变量与多维数组.....	135
6.4.3	指针变量的运算.....	137
6.5	字符型指针变量.....	142
6.5.1	字符型指针的声明.....	142
6.5.2	字符型指针变量的操作.....	143
6.6	指针数组.....	146
6.7	本章小结.....	148
6.8	上机练习.....	149
6.9	习题.....	151
第 7 章	结构	153
7.1	结构的基本概念.....	154
7.2	结构类型的定义与结构变量的声明.....	154
7.2.1	结构类型的定义.....	154
7.2.2	结构变量的声明和初始化.....	156
7.3	结构变量的使用.....	157
7.3.1	引用结构变量成员.....	157
7.3.2	结构变量的运算和输入输出.....	158
7.4	结构数组.....	159
7.4.1	结构数组的声明.....	159
7.4.2	结构数组的使用.....	160
7.5	结构指针与结构数组.....	162
7.6	new 和 delete 运算、递归结构和链表.....	165
7.6.1	new 和 delete 运算.....	166
7.6.2	递归结构.....	168
7.6.3	链表.....	171
7.7	本章小结.....	173
7.8	上机练习.....	174
7.9	习题.....	176
第 8 章	联合、枚举和自定义类型	177
8.1	联合类型.....	178
8.1.1	联合类型的定义.....	178

8.1.2	联合变量的声明和初始化.....	179
8.1.3	联合变量的使用.....	180
8.1.4	联合与结构的比较.....	181
8.2	枚举类型.....	183
8.2.1	枚举类型的定义.....	184
8.2.2	枚举变量的声明.....	185
8.2.3	枚举变量的使用.....	186
8.3	自定义类型.....	187
8.4	本章小结.....	189
8.5	上机练习.....	189
8.6	习题.....	190
第9章	函数.....	193
9.1	函数的定义与调用.....	194
9.1.1	函数的定义.....	194
9.1.2	函数的调用.....	196
9.1.3	返回语句与返回值.....	198
9.2	函数的参数传递.....	201
9.2.1	值传递方式.....	202
9.2.2	指针传递方式.....	202
9.2.3	引用传递方式.....	207
9.3	void 类型参数.....	208
9.3.1	void 返回类型.....	208
9.3.2	void 参数类型.....	208
9.4	递归函数设计.....	209
9.5	带默认值参数的函数.....	210
9.6	内联函数.....	212
9.7	重载函数.....	213
9.8	分程序.....	215
9.9	main()函数参数的使用.....	217
9.10	C++库函数.....	218
9.10.1	数学计算函数.....	218
9.10.2	字符分类函数.....	219
9.10.3	字符串处理函数.....	220
9.11	本章小结.....	221
9.12	上机练习.....	222
9.13	习题.....	225

第 10 章 存储类别	227
10.1 存储类别的概念	228
10.2 变量的存在性和可见性	228
10.3 变量的存储类别	230
10.3.1 自动变量	230
10.3.2 寄存器变量	231
10.3.3 静态变量	233
10.3.4 外部变量	234
10.3.5 只读变量	235
10.4 函数的存储类别	237
10.5 在 Visual C++ 6.0 中调试多个源程序文件组成的 C++ 程序	238
10.6 本章小结	242
10.7 上机练习	243
10.8 习题	244
第 11 章 类与对象	245
11.1 类的定义	246
11.2 用类建立对象	248
11.3 内联成员函数	253
11.4 构造函数和析构函数	254
11.4.1 构造函数	254
11.4.2 带参数的构造函数	255
11.4.3 析构函数	258
11.4.4 构造和析构函数的调用	260
11.5 类的封装与访问性控制	260
11.5.1 类的访问性控制	261
11.5.2 类的封装性	264
11.6 类的静态数据成员	264
11.7 空类	268
11.8 类的嵌套	268
11.9 成员函数的重载	269
11.10 运算符的重载	271
11.11 友元类和友元函数	275
11.11.1 友元类	275
11.11.2 友元函数	277
11.12 对象指针	282
11.13 本章小结	283
11.14 上机练习	284

11.15 习题.....	287
第 12 章 类的继承	289
12.1 类的继承性.....	290
12.2 单一继承.....	292
12.2.1 单一继承派生类的定义.....	292
12.2.2 继承模式.....	293
12.2.3 构造函数的调用顺序.....	297
12.2.4 构造函数的参数传递.....	299
12.2.5 析构函数的调用顺序.....	301
12.2.6 保护成员的继承性.....	302
12.3 多继承.....	303
12.4 成员函数的覆盖.....	307
12.4.1 虚函数.....	308
12.4.2 纯虚函数与抽象类.....	310
12.5 本章小结.....	312
12.6 上机练习.....	313
12.7 习题.....	316
第 13 章 输入输出处理	317
13.1 标准设备的输入输出函数 scanf()和 printf()	318
13.1.1 数据输入.....	318
13.1.2 数据输出.....	322
13.2 文件与流式文件的概念.....	327
13.3 流式文件的打开和关闭.....	328
13.3.1 打开文件.....	328
13.3.2 关闭文件.....	329
13.4 流式文件的读写.....	330
13.4.1 字符串读/写操作.....	330
13.4.2 字符读/写操作.....	333
13.4.3 整数读/写操作.....	335
13.4.4 格式化读/写操作.....	337
13.4.5 数据块读/写操作.....	339
13.5 流式文件的定位.....	342
13.6 流式文件与设备文件.....	346
13.7 流式文件的错误检测.....	347
13.8 本章小结.....	349
13.9 上机练习.....	350
13.10 习题.....	353

第 14 章 预处理指令.....355

14.1 预处理指令简介..... 356

14.2 宏替换指令..... 356

14.2.1 简单宏替换..... 356

14.2.2 带参数宏替换..... 358

14.3 文件嵌入指令..... 359

14.4 条件编译指令..... 360

14.5 本章小结..... 362

14.6 上机练习..... 362

14.7 习题..... 362

附录 A C++常用库函数.....363

A1 字符分类函数..... 364

A2 数学运算和类型转换函数..... 365

A3 串和内存操作函数..... 372

A4 输入输出函数..... 376

附录 B Visual C++ 6.0 的安装和启动.....387

第 1 章 C++简介

教学目标:

讲解 C++语言的发展及特点、程序结构、基本符号、标识符和程序设计的有关知识。通过本章的学习,使读者初步掌握使用 Visual C++ 6.0 开发环境,编辑、调试和运行 C++程序。

教学重点与难点:

- ◆ C++语言的发展及特点。
- ◆ C++程序的基本结构。
- ◆ 使用 Visual C++ 6.0 开发环境编辑、调试和运行 C++程序。

1.1 C++语言的发展及特点

C++是在C语言基础上发展起来的面向对象的程序设计语言，它包含了C语言的全部功能和特性，并有所增加，是C语言的超集。C++对C语言的增加主要体现在支持面向对象程序设计上，例如，类和对象是C++为实现面向对象程序设计提供的编程机制。C++是C语言和面向对象程序设计方法相结合的产物。本节简单介绍C++的发展情况。

1.1.1 C语言的出现与发展

C语言是1972年由美国贝尔实验室的D.M.Ritchie设计开发的一种通用程序设计语言，它适合于编写各种系统软件和应用软件。当前C语言仍然是世界上最有影响的程序设计语言之一。目前在我国软件开发领域，C语言仍十分流行。主要原因是，一方面它具有语言简捷、表达力强、代码质量高、可移植性好等特点。其独到之处是兼具“高级语言”的特性与“低级处理”的能力，它能代替汇编语言直接与机器硬件打交道，这是众多其他高级语言所不能的；另一方面，C语言在各类机型(大、中、小、微)上均能使用，尤其在各类微机上十分普及。此外，C语言还有极其雄厚的后源支撑，众多实用的C语言程序库和程序包为用户提供了良好的开发环境。因此，它深受广大程序员的喜爱。但是，C语言也有不足的一面，例如，其语法检查很不严格，因此，程序的可靠性和可维护性不高。

90年代以后，计算机的应用迅速增长，人们对计算机的软硬件期望之高也是空前的，这就使得软件设计要面对以下几方面的难题。

1. 难度急速上升

在软件开发前和开发过程中，系统需求或者不甚明确或者一直在变化中。随着计算机应用深入到不同专业领域，一个成功的软件必定会涉及到很多领域的专业知识，并且随着其所应用专业的发展而发展。因此，在一个软件系统的生命周期(从设计开发一直到被淘汰)中，必然要经历不断修改和完善的过程，这就要求软件系统的可读性和可维护性要好。

2. 软件大型化

开发者和用户都趋向于把软件系统的功能设计的尽可能完善，用户界面要美观友好，这必然使软件的规模越来越大。因此，要求软件设计者能有效地对大型软件进行控制和管理。

3. 软件设计规模化

软件“生产”的手工作坊方式已经逐步转向流水线式的批量生产。软件的规模生产，要求软件设计者之间要相互合作。

4. 软件设计需继承性的发展

现在设计一套软件不再是从头开始，而是在前面软件或类似的软件系统的基础上有继承地发展，这就要求软件的设计具有可继承性。

由于上述原因和硬件费用的不断下降,在一段时期内,出现了软件费用不断上升、软件的可靠性不断下降、管理和控制软件的开发过程变的十分困难,这就是所谓的软件危机。

后来人们逐渐意识到对付软件危机,必须采用先进的程序设计方法和能够体现先进程序设计方法的程序设计语言来解决。

在这种形势下,用C语言编写程序就显得有些不足,例如,可靠性不高、可维护性不好,特别是对开发大型软件(十几万行以上的源程序)难以管理和控制等。

1.1.2 面向对象程序设计

面向对象程序设计(OOP, Object Oriented Programming)是一种比较先进的程序设计方法,其目的是让程序设计者能够理解、控制和管理更大、更复杂的程序。它特别支持自顶向下地进行层次化、模块化程序设计,关键是确立了对象的概念。所谓对象就是一个既包含数据(数据成员),又包含方法(成员函数)被封装的逻辑实体。对象的数据成员描述了事物的状态,对象成员函数描述了事物的行为。对象的成员又分为私有成员和公共成员,私有成员只能由对象的设计者访问使用;对象的公共成员由对象的使用者访问使用,是对象设计者为对象使用者提供的使用接口。这样对象既提供了使用功能,又隐藏和保护了内部的信息。采用面向对象方法进行程序设计,可以使我們更自然更容易地描述客观事物。

1.1.3 C++语言的特点

C++是在C语言基础上,扩充引进了面向对象的概念和功能,而发展起来的一种支持面向对象设计方法的通用程序设计语言。由于C++提供了把对象数据和操作对象数据的代码以类和对象的形式进行封装的机制,并通过派生、继承、重载和覆盖等特性,实现了人们所希望的软件重用技术。使得软件,特别是大型复杂软件的设计和维护变得更有效和更容易,并使软件开发能更自然地反映事物的本来面貌。此外,C++又与C语言完全兼容,它继承了C语言原有的优点。这些使得C++在软件开发领域受到更广泛的欢迎。

提示:

面向对象程序设计方法不依赖具体的语言,即用C也能进行面向对象设计,但是它远远不如用C++更自然、更方便和更有效。

从程序设计者的角度看,C++语言主要具有以下特点。

1. 是一种高效率编译型语言

C++是编译型语言,并且适当地考虑了背景机。因此,其生成的目标程序质量高,运行速度快。

2. 语言简捷、结构清晰

C++程序通常是由若干个函数组成,这一特性支持模块化程序设计。C++语言提供多种顺序语句、循环语句和选择语句支持结构化程序设计。此外,C++语言程序书写格式自由。因此,C++程序简捷清晰、可读性较强。