



21 世纪大学本科
计算机专业系列教材

王挺 周会平 贾丽丽 徐锡山 编著

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

C++ 程序设计

- 根据教育部高教司主持评审的《中国计算机科学与技术学科教程 2002》组织编写
- 与美国 ACM 和 IEEE/CS *Computing Curricula 2001* 同步



清华大学出版社

内 容 简 介

本书参照美国 ACM 和 IEEE/CS CC2001 以及《中国计算机科学与技术学科教程 2002》的要求,面向大学计算机专业本科教学的需要,系统地介绍 C++ 程序设计语言的语法、语义和语用,使读者掌握结构化程序设计方法和面向对象程序设计方法。全书分为程序设计基础和面向对象程序设计两部分。第一部分主要介绍类型、运算符与表达式、输入与输出、控制结构、函数、数组、指针、结构、联合、枚举、链表等程序设计的基本概念和结构化程序设计方法;第二部分主要介绍类与对象、类和对对象的使用、操作符重载、继承、多态性、文件和流、异常和模板等面向对象程序设计的基本概念和程序设计方法。

本书注重知识的系统性和连贯性,在内容上注意与后续课程的衔接,强调严密的逻辑思维,突出程序设计方法的教学。

本书适合作为高等院校程序设计基础或高级语言程序设计课程的教材,也可供广大自学人员参考。

版权所有,翻印必究。举报电话:010-62782989 13901104297 13801310933

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。

本书防伪标签采用清华大学核研院专有核径迹膜防伪技术,用户可通过在图案表面涂抹清水,图案消失,水干后图案复现;或将表面膜揭下,放在白纸上用彩笔涂抹,图案在白纸上再现的方法识别真伪。

图书在版编目(CIP)数据

C++ 程序设计/王挺等编著. —北京:清华大学出版社,2005.1

(21 世纪大学本科计算机专业系列教材)

ISBN 7-302-09823-9

I. C… II. 王… III. C 语言—程序设计—高等学校—教材 IV. TP312

中国版本图书馆 CIP 数据核字(2004)第 111836 号

出 版 者: 清华大学出版社

地 址: 北京清华大学学研大厦

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

邮 编: 100084

社 总 机: 010-62770175

客户服务: 010-62776969

责任编辑: 张瑞庆

封面设计: 孟繁聪

印 装 者: 北京嘉实印刷有限公司

发 行 者: 新华书店总店北京发行所

开 本: 185×230 印张: 29.75 字数: 646 千字

版 次: 2005 年 1 月第 1 版 2005 年 1 月第 1 次印刷

书 号: ISBN 7-302-09823-9/TP·6776

印 数: 1~5000

定 价: 36.00 元

本书如存在文字不清、漏印以及缺页、倒页、脱页等印装质量问题,请与清华大学出版社出版部联系调换。联系电话:(010)62770175-3103 或(010)62795704

21 世纪大学本科计算机专业系列教材编委会

名誉主任：陈火旺

主 任：李晓明

副 主 任：钱德沛 焦金生

委 员：(按姓氏笔画为序)

马殿富	王志英	王晓东	宁 洪	刘 辰
孙茂松	李大友	李仲麟	吴朝晖	何炎祥
宋方敏	张大方	张长海	周兴社	侯文永
袁开榜	钱乐秋	黄国兴	蒋宗礼	曾 明
廖明宏	樊孝忠			

秘 书：张瑞庆

本书责任编辑：钱乐秋

PREFACE

21 世纪是知识经济的时代,是人才竞争的时代。随着 21 世纪的到来,人类已步入信息社会,信息产业正成为全球经济的主导产业。计算机科学与技术信息产业中占据了最重要的地位,这就对培养 21 世纪高素质创新型计算机专业人才提出了迫切的要求。

为了培养高素质创新型人才,必须建立高水平的教学计划和课程体系。在 20 多年跟踪分析 ACM 和 IEEE 计算机课程体系的基础上,紧跟计算机科学与技术的发展潮流,及时制定并修正教学计划和课程体系是尤其重要的。计算机科学与技术的发展对高水平人才的要求,需要我们从总体上优化课程结构,精炼教学内容,拓宽专业基础,加强教学实践,特别注重综合素质的培养,形成“基础课程精深,专业课程宽新”的格局。

为了适应计算机科学与技术学科发展和计算机教学计划的需要,要采取多种措施鼓励长期从事计算机教学和科技前沿研究的专家教授积极参与计算机专业教材的编著和更新,在教材中及时反映学科前沿的研究成果与发展趋势,以高水平的科研促进教材建设。同时适当引进国外先进的原版教材。

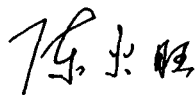
为了提高教学质量,需要不断改革教学方法与手段,倡导因材施教,强调知识的总结、梳理、推演和挖掘,通过加快教案的不断更新,使学生掌握教材中未及时反映的学科发展新动向,进一步拓广视野。教学与科研相结合是培养学生实践能力的有效途径。高水平的科研可以为教学提供最先进的高新技术平台和创造性的工作环境,使学生得以接触最先进的计算机理论、技术和环境。高水平的科研还可以为高水平人才的素质教育提供良好的物质基础。学生在课题研究中不但能了解科学研究的艰辛和科研工作者的奉献精神,而且能熏陶和培养良好的科研作风,锻炼和培养攻关能力和协作精神。

进入 21 世纪,我国高等教育进入了前所未有的大发展时期,时代的进步与发展对高等教育质量提出了更高、更新的要求。2001 年 8 月,教育部颁发了《关于加强高等学校本科教学工作,提高教学质量的若干意见》。文件指出,本科教育是高等教育的主体和基础,抓好本科教学是提高整个高等教育质量的重点和关键。随着高等教育的普及和高等学校的扩招,在校大学本科计算机专业学生的人数将大量上升,对适合 21 世纪大学本科计算

机科学与技术学科课程体系要求的,并且适合中国学生学习的计算机专业教材的需求量也将急剧增加。为此,中国计算机学会和清华大学出版社共同规划了面向全国高等院校计算机专业本科生的《21 世纪大学本科计算机专业系列教材》。本系列教材借鉴美国 ACM 和 IEEE/CS 最新制定的 *Computing Curricula 2001* (简称 CC2001) 课程体系,反映当代计算机科学与技术学科水平和计算机科学技术的新发展、新技术,并且结合中国计算机教育改革成果和中国国情。

中国计算机学会教育专业委员会和全国高等学校计算机教育研究会,在清华大学出版社的大力支持下,跟踪分析 CC2001,并结合中国计算机科学与技术学科的发展现状和计算机教育的改革成果,研究出了《中国计算机科学与技术学科教程 2002》(*China Computing Curricula 2002*, 简称 CCC2002),该项研究成果对中国高等学校计算机科学与技术学科教育的改革和发展具有重要的参考价值和积极的推动作用。

《21 世纪大学本科计算机专业系列教材》正是借鉴美国 ACM 和 IEEE/CS CC2001 课程体系,依据 CCC2002 基本要求组织编写的计算机专业教材。相信通过这套教材的编写和出版,能够在内容和形式上显著地提高我国计算机专业教材的整体水平,继而提高我国大学本科计算机专业的教学质量,培养出符合时代发展要求的具有较强国际竞争力的高素质创新型计算机人才。



中国工程院院士

国防科学技术大学教授

21 世纪大学本科计算机专业系列教材编委会名誉主任

2002 年 7 月

前言

FOREWORD

本教材参照美国 ACM 和 IEEE/CS CC2001 以及《中国计算机科学与技术学科教程 2002》(简称 CCC2002)的基本要求,试图通过教学,使学生掌握结构化程序设计方法和面向对象程序设计方法,掌握 C++ 程序设计语言的语法、语义和语用,能够熟练运用 C++ 语言解决一般问题。此外,通过课程学习,使学生掌握学习高级程序设计语言的一般方法,养成良好的程序设计风格,对软件工程有初步的认识。本教材虽然以讲授 C++ 为主,但并非单纯讲授一门语言,而是更加强调对程序设计方法的掌握和程序设计风格的养成,为学生今后继续学习其他高级程序设计课程打下牢固的基础。

本书的编写参照 CC2001 和 CCC2002 体系,覆盖或涉及其中程序设计基础(PF)、程序设计语言(PL)、算法和复杂性(AL)3 个领域(area),以及 11 个知识单元(unit)。由于部分知识单元的内容将在其他课程(如数据结构、编译原理等)中详细介绍,因此本教材并未深入讲解。

本教材覆盖或涉及 CC2001 和 CCC2002 的如下知识:

- ◆ 程序设计基础(PF)
 - 程序设计基本结构
 - 算法和问题求解
 - 基本数据结构
 - 递归
 - 事件驱动程序设计
- ◆ 程序设计语言(PL)
 - 程序设计语言概论
 - 声明和类型
 - 抽象机制
 - 面向对象程序设计
- ◆ 算法和复杂性(AL)
 - 算法策略

■ 基本算法

本书参考了国外著名高校教材,结合国内高校教学的需要和学生的特点,力求形成如下特色。

(1) 注重知识的系统性和连贯性。本教材面向大学本科计算机专业的学生,参照 CC2001 和 CCC2002 的要求,在教学内容上注意与后续课程(如数据结构、算法设计与分析、编译原理、软件工程等)的衔接。

(2) 突出程序设计方法,强调严密的逻辑思维。本教材并非单纯讲授 C++ 程序设计语言,而是在介绍 C++ 程序设计语言的基础上,系统地讲解程序设计方法(包括结构化程序设计方法和面向对象程序设计方法,特别是后者)。程序设计方法是本书的讲授重点。因此,在教材的示例程序中,一般都配有分析、解释和经验总结等,帮助读者领会设计的方法和思想。

(3) 注重实践能力的培养。本教材提供丰富的典型例题,每章之后配有难易和综合程度各不相同的习题。教材中还适当地穿插介绍一些编程技巧和软件设计经验,使学生能够从实践的角度更好地学习和掌握 C++ 程序设计方法。

(4) 注重程序设计风格的养成。良好的程序设计风格是保证软件质量的基础。本教材除了有专门章节介绍程序设计风格外,全书始终突出对程序设计风格的要求,并介绍了从分析、设计到编程如何实现良好程序设计风格的方法。本教材的示例程序即按照这种方法实现,在变量命名、源程序格式等方面都贯彻统一的风格,成为学生模仿的样板。

C++ 是当今最流行的一种高效实用的高级程序设计语言,应用十分广泛。它也是一门复杂的语言,与 C 语言兼容,既支持结构化的程序设计方法,也支持面向对象的程序设计方法,因而成为编程人员最广泛使用的工具。学好 C++, 很容易学习其他程序设计语言,C++ 架起了通向强大、易用、真正的应用软件开发的桥梁。本书共分为两大部分。第一部分,第 1 章至第 9 章是基础部分,主要介绍 C++ 程序设计语言、程序结构和结构化程序设计基础;第二部分,第 10 章至第 18 章,即面向对象程序设计部分,它建立在 C++ 程序设计基础之上,讲述了面向对象程序设计方法。书中带“*”的章节为选讲内容,可以根据实际情况取舍。

本书是作者根据多年教学实践的经验编写而成,适合作为大学本科计算机专业和非计算机专业的程序设计基础或高级语言程序设计课程的教材,也可供广大读者自学参考。复旦大学钱乐秋教授认真审阅了本书的全部书稿,并提出了很多宝贵意见,在此表示诚挚的谢意。由于水平有限,书中可能存在许多缺点和错误,恳请广大读者批评指正。

作 者

国防科技大学计算机学院

2004 年 11 月

目 录

CONTENTS

第 1 章 计算机与程序设计	1
1.1 计算机系统概述	1
1.1.1 什么是计算机系统	1
1.1.2 计算机硬件	2
1.1.3 计算机软件	3
1.2 程序设计基本概念	4
1.2.1 问题求解过程	4
1.2.2 算法与程序	5
1.2.3 程序设计语言	8
1.3 程序设计方法	9
1.3.1 结构化程序设计	9
1.3.2 面向对象程序设计	10
1.4 C 与 C++	11
1.4.1 C 语言	11
1.4.2 C++ 语言	13
1.5 C++ 编程简介	13
1.5.1 C++ 编程的典型过程	13
1.5.2 一个简单的 C++ 程序	14
1.6 程序设计风格	16
习题	18
第 2 章 数据类型、运算符与表达式	19
2.1 C++ 的字符集和关键字	19
2.1.1 字符集	19

2.1.2	标识符	20
2.1.3	关键字	20
2.2	基本数据类型	21
2.2.1	整型	21
2.2.2	浮点型数据	22
2.2.3	字符型数据	22
2.2.4	布尔类型	23
2.3	变量与常量	23
2.3.1	常量	23
2.3.2	变量	27
2.3.3	符号常量	28
2.4	类型转换	29
2.5	运算符和表达式	30
2.5.1	算术运算	31
2.5.2	关系运算	33
2.5.3	逻辑运算	33
2.5.4	位运算	34
2.5.5	逗号运算	36
2.5.6	赋值运算	37
2.5.7	特殊运算符	38
习题		38
第3章	输入和输出	40
3.1	C++ 的输入和输出	40
3.2	标准输入与输出函数	41
3.2.1	字符输入函数 getchar	41
3.2.2	字符输出函数 putchar	41
3.3	格式化输入输出	42
3.3.1	格式化输入函数 scanf	42
3.3.2	格式化输出函数 printf	44
3.3.3	格式化输入输出函数应用举例	47
3.4	用流进行输入输出	53
3.4.1	通过 cout 流输出数据	53
3.4.2	通过 cin 流输入数据	55

3.5 流操纵算子.....	56
3.5.1 设置整数基数的流操纵算子	56
3.5.2 设置浮点数精度的流操纵算子	57
3.5.3 设置域宽的流操纵算子	59
习题	60
第4章 控制结构	62
4.1 程序的基本控制结构.....	62
4.2 顺序结构.....	63
4.2.1 表达式语句	63
4.2.2 复合语句	64
4.3 选择结构.....	65
4.3.1 if 选择结构	65
4.3.2 if-else 选择结构	66
4.3.3 switch 选择结构	72
4.4 循环结构.....	77
4.4.1 while 循环语句	77
4.4.2 do-while 循环语句	80
4.4.3 for 循环语句.....	81
4.5 控制转移语句.....	85
4.5.1 break 语句	86
4.5.2 continue 语句	87
4.5.3 goto 语句	89
4.5.4 return 语句	90
4.6 结构化程序设计方法.....	90
4.6.1 结构化程序设计思想	90
4.6.2 结构化程序设计原则	93
4.6.3 结构化程序设计示例	94
4.6.4 再谈程序设计风格	99
习题.....	100
第5章 函数	104
5.1 模块化程序设计	104
5.2 预定义函数的使用	105

5.3	函数定义与函数原型	106
5.3.1	函数定义	106
5.3.2	函数原型	108
5.4	函数的调用	109
5.4.1	函数调用的概念	109
5.4.2	参数传递	112
5.4.3	默认参数	114
5.4.4	内联函数	115
5.5	函数重载	117
5.6	存储类别	118
5.6.1	变量的存储特性	119
5.6.2	自动变量	119
* 5.6.3	寄存器变量	120
* 5.6.4	外部变量	120
5.6.5	静态变量	121
5.7	作用域	123
5.8	递归函数	126
5.8.1	递归的概念	126
5.8.2	递归函数应用举例	128
5.9	预处理指令	132
5.9.1	#include 指令	132
5.9.2	#define 指令	133
* 5.9.3	#if、#else、#endif、#ifdef、#ifndef 指令	135
习题		135
第 6 章	数组	138
6.1	一维数组	139
6.1.1	一维数组的声明与初始化	139
6.1.2	数组的应用	140
6.1.3	数组作为函数参数	143
6.2	多维数组	149
6.2.1	多维数组的定义与初始化	149
6.2.2	应用示例	151
6.3	字符串	154

6.3.1 字符与字符串概念.....	154
6.3.2 字符串与字符数组.....	155
6.3.3 字符串的输入和输出.....	157
习题.....	158
第 7 章 指针	160
7.1 指针的定义	161
7.2 指针的运算	162
7.3 指针与数组	166
7.4 字符指针与字符数组	170
7.4.1 字符串函数.....	170
7.4.2 字符串函数的应用.....	176
7.5 指针与 const 限定符	179
7.5.1 基本概念.....	179
* 7.5.2 用 const 限定指针	180
7.6 指针和引用	184
7.7 动态内存分配	185
7.7.1 基本概念.....	185
7.7.2 动态分配内存的应用.....	188
* 7.8 函数指针	191
7.8.1 函数指针的定义	191
7.8.2 函数指针的使用	192
习题.....	194
第 8 章 结构、联合、枚举	197
8.1 结构	197
8.1.1 结构的定义.....	198
8.1.2 结构变量成员的引用.....	200
8.2 结构的使用	201
8.2.1 结构与函数.....	201
8.2.2 结构与数组.....	206
8.2.3 结构与指针.....	209
* 8.2.4 位段.....	213
8.3 联合	215

8.3.1	联合和联合变量定义	216
8.3.2	联合的使用	217
8.4	枚举	219
8.4.1	枚举和枚举型变量的定义	219
8.4.2	枚举类型变量的赋值和使用	220
	习题	221
第9章	链表	223
9.1	链表的基本概念	223
9.2	单向链表	224
9.2.1	单向链表的定义	224
9.2.2	单向链表的操作	225
*9.3	双向链表	235
9.3.1	双向链表的定义	235
9.3.2	双向链表的操作	236
	习题	243
第10章	面向对象程序设计基本概念	245
10.1	面向对象语言和面向对象方法	245
10.2	类、对象和消息	248
10.2.1	类和对象	248
10.2.2	消息	250
10.3	面向对象程序设计的特点	251
10.4	面向对象程序的结构	253
	习题	255
第11章	类与对象	257
11.1	数据抽象的概念	258
11.2	抽象数据类型	259
11.2.1	封装与信息隐藏	260
11.2.2	接口与实现的分离	260
11.2.3	用结构实现用户定义类型：栈	261
11.2.4	用类实现抽象数据类型：栈	265
11.3	类和对象的定义	270

11.3.1	数据成员	270
11.3.2	成员函数	272
11.3.3	访问控制	273
11.3.4	静态成员	274
11.3.5	对象的建立	278
11.4	构造函数	280
11.4.1	构造函数的作用	280
11.4.2	构造函数执行的时机	281
11.4.3	构造函数重载	281
11.4.4	默认构造函数	282
11.4.5	复制构造函数	284
11.5	析构函数	285
11.5.1	析构函数的作用	286
11.5.2	析构函数执行的时机	286
	习题	288
第 12 章	类和对象的使用	290
12.1	类的复合	290
12.2	this 指针	294
12.3	const 特性	297
12.4	友元函数和友元类	302
12.4.1	友元函数	302
12.4.2	友元类	304
	习题	305
第 13 章	运算符重载	307
13.1	运算符重载的概念	308
13.1.1	运算符重载的意义	308
13.1.2	运算符重载的限制	309
13.2	运算符成员函数与友元函数	310
13.3	单目运算符重载	311
13.4	重载流插入和流提取运算符	316
13.5	双目运算符重载	319
13.6	赋值运算符重载	322

* 13.7 类型之间的转换	325
习题	334

第 14 章 继承

14.1 继承和派生的概念	338
14.2 继承的定义	339
14.2.1 派生类和基类	339
14.2.2 继承的方式	340
14.2.3 类的层次	341
14.2.4 在派生类中重定义基类的函数	342
14.2.5 派生类和基类的转换	347
14.3 类指针	347
14.4 继承中的构造函数和析构函数	352
* 14.5 多重继承	356
14.6 软件渐增式开发	361
14.6.1 复合与继承	361
14.6.2 示例	362
习题	372

第 15 章 多态性

15.1 多态性的概念	374
15.1.1 静态绑定和动态绑定	375
15.1.2 多态性的意义	375
15.2 虚函数	376
15.3 抽象基类和纯虚函数	380
15.3.1 纯虚函数	380
15.3.2 抽象类和具体类	381
* 15.4 虚析构函数	385
15.5 软件渐增式开发	390
习题	397

第 16 章 文件和流

16.1 基本概念	399
16.2 通过 FILE 结构进行文件操作	401

16.2.1 打开、建立和关闭文件	401
16.2.2 写文件	403
16.2.3 读文件	406
16.3 通过文件流进行文件操作	413
16.3.1 打开、建立文件	413
16.3.2 写文件	415
16.3.3 读文件	416
习题	422
第 17 章 异常	424
17.1 异常处理的意义	424
17.2 异常处理基础	425
17.3 异常的抛出和传播	428
17.4 异常的捕获和处理	432
习题	435
第 18 章 模板	437
18.1 类属机制	437
18.2 函数模板	438
18.2.1 函数模板的定义	438
18.2.2 使用函数模板	440
18.3 类模板	444
18.3.1 类模板的定义	445
18.3.2 使用类模板	446
习题	448
附录	450
附录 A C++ 运算符的优先级和结合性	450
附录 B ASCII 字符集	452
参考文献	454

第 1 章

计算机与程序设计

学习内容

本章介绍计算机与程序的基本概念。主要包括：

- ◆ 计算机系统的基本组成和工作原理。
- ◆ 程序设计的基本过程和方法。
- ◆ 程序设计语言的基本概念及其发展历史。
- ◆ 结构化程序设计方法和面向对象程序设计方法概述。
- ◆ C 与 C++ 语言。
- ◆ 典型的 C++ 编程环境、开发过程和简单的 C++ 程序结构。
- ◆ 程序设计风格。

学习目标

- ◆ 了解计算机系统、程序设计语言、程序设计的基本概念。
- ◆ 通过简单示例掌握 C++ 程序的结构。
- ◆ 明确程序设计风格的意义。

1.1 计算机系统概述

1.1.1 什么是计算机系统

计算机系统(computer system)是按人的要求接收和存储信息,自动进行数据处理和计算,并输出结果信息的机器系统。计算机是脑力的延伸和扩充,是近代科学的重大成就之一。计算机系统由硬件系统和软件系统组成。前者是借助电、磁、光、机械等原理构成的各种物理部件的有机组合,是系统赖以工作的实体,如 CPU、显示器、内存、硬盘和键盘等;后者是各种程序和文档,用于指挥全系统按指定的要求进行工作,程序是对计算任务的处理对象和处理规则的描述,而文档是与软件研制、维护和使用有关的资料。随着计算机技术的迅速发展,硬件的成本在不断地下降,促使了个人计算机的广泛应用和普及。这