

高等学校教材·计算机科学与技术

可赠送课件

jsjc@tup.tsinghua.edu.cn

C++程序设计 与应用开发

朱振元 朱 承 编著



清华大学出版社

高等学校教材·计算机科学与技术

C++程序设计与应用开发

朱振元 朱 承 编著

清华大学出版社

北 京

内 容 简 介

本书分为两部分,第一部分全面、系统地介绍了面向对象程序设计语言C++的基本概念、基本语法和编程方法,第二部分结合开发工具C++ Builder介绍了可视化的开发方法。书中通过一系列综合开发实例的分析,将C++中的基本语言成分及重要概念融入到Windows应用程序的开发之中。

本书语言通俗易懂、条理清晰、应用性强,可作为高等院校计算机专业的教科书,也可作为应用程序开发人员及电脑爱好者的技术参考书。

版权所有,翻印必究。举报电话:010-62782989 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++ 程序设计与应用开发/朱振元,朱承编著. —北京:清华大学出版社,2005.2

(高等学校教材·计算机科学与技术)

ISBN 7-302-10340-2

I. C… II. ①朱… ②朱… 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(2005)第003384号

出 版 者:清华大学出版社

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

社总机:010-62770175

地 址:北京清华大学学研大厦

邮 编:100084

客户服务:010-62776969

责任编辑:闫红梅

封面设计:王 永

印 刷 者:清华大学印刷厂

装 订 者:北京鑫海金澳胶印有限公司

发 行 者:新华书店总店北京发行所

开 本:185×260 印张:18 字数:442千字

版 次:2005年2月第1版 2005年2月第1次印刷

书 号:ISBN 7-302-10340-2/TP·7039

印 数:1~5000

定 价:23.00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

高等学校教材·计算机

编审委员会成员

(按地区排序)

清华大学

周立柱 教授
覃 征 教授
王建民 教授
刘 强 副教授
冯建华 副教授

北京大学

杨冬青 教授
陈 钟 教授
陈立军 副教授

北京航空航天大学

马殿富 教授
吴超英 副教授
姚淑珍 教授

中国人民大学

王 珊 教授
孟小峰 教授
陈 红 教授

北京交通大学

阮秋琦 教授

北京信息工程学院

孟庆昌 教授

北京科技大学

杨炳儒 教授

石油大学

陈 明 教授

天津大学

艾德才 教授

复旦大学

吴立德 教授
吴百锋 教授
杨卫东 副教授

华东理工大学

邵志清 教授

华东师范大学

杨宗源 教授

应吉康 教授

东华大学

乐嘉锦 教授

上海第二工业大学

蒋川群 教授

浙江大学

吴朝晖 教授

李善平 教授

南京大学

骆 斌 教授

南京航空航天大学

南京理工大学

南京邮电学院

苏州大学

江苏大学

武汉大学

华中科技大学

中南财经政法大学

华中师范大学

武汉理工大学

国防科技大学

中南大学

湖南大学

西安交通大学

西北大学

长安大学

西安石油学院

西安邮电学院

哈尔滨工业大学

吉林大学

长春工程学院

山东大学

山东科技大学

中山大学

厦门大学

福州大学

云南大学

重庆邮电学院

西南交通大学

秦小麟 教授

张功萱 教授

朱秀昌 教授

龚声蓉 教授

宋余庆 教授

何炎祥 教授

刘乐善 教授

刘腾红 教授

王林平 副教授

魏开平 教授

李中年 教授

赵克佳 教授

肖 依 副教授

陈松乔 教授

林亚平 教授

邹北骥 教授

沈钧毅 教授

齐 勇 教授

周明全 教授

巨永峰 教授

方 明 教授

陈莉君 副教授

郭茂祖 教授

徐一平 教授

毕 强 教授

沙胜贤 教授

孟祥旭 教授

郝兴伟 教授

郑永果 教授

潘小轰 教授

冯少荣 教授

林世平 副教授

刘惟一 教授

王国胤 教授

杨 燕 副教授

出版说明

改革开放以来,特别是党的十五大以来,我国教育事业取得了举世瞩目的辉煌成就,高等教育实现了历史性的跨越,已由精英教育阶段进入国际公认的大众化教育阶段。在质量不断提高的基础上,高等教育规模取得如此快速的发展,创造了世界教育发展史上的奇迹。当前,教育工作既面临着千载难逢的良好机遇,同时也面临着前所未有的严峻挑战。社会不断增长的高等教育需求同教育供给特别是优质教育供给不足的矛盾,是现阶段教育发展面临的基本矛盾。

教育部一直十分重视高等教育质量工作。2001年8月,教育部下发了《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》,提出了十二条加强本科教学工作提高教学质量的措施和意见。2003年6月和2004年2月,教育部分别下发了《关于启动高等学校教学质量与教学改革工程精品课程建设工作的通知》和《教育部实施精品课程建设提高高校教学质量和人才培养质量》文件,指出“高等学校教学质量和教学改革工程”,是教育部正在制订的《2003—2007年教育振兴行动计划》的重要组成部分,精品课程建设是“质量工程”的重要内容之一,教育部计划用五年时间(2003—2007年)建设1500门国家级精品课程,利用现代化的教育信息技术手段将精品课程的相关内容上网并免费开放,以实现优质教学资源共享,提高高等学校教学质量和人才培养质量。

为了深入贯彻落实教育部《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》精神,紧密配合教育部已经启动的“高等学校教学质量与教学改革工程精品课程建设工作”,在有关专家、教授的倡议和有关部门的大力支持下,我们组织并成立了“清华大学出版社教材编审委员会”(以下简称“编委会”),旨在配合教育部制定精品课程教材的出版规划,讨论并实施精品课程教材的编写与出版工作。“编委会”成员皆来自全国各类高等学校教学与科研第一线的骨干教师,其中许多教师为各校相关院、系主管教学的院长或系主任。

按照教育部的要求,“编委会”一致认为,精品课程的建设工作从开始就要坚持高标准、严要求,处于一个比较高的起点上;精品课程教材应该能够反映各高校教学改革与课程建设的需要,要有特色风格、有创新性(新体系、新内容、新手段、新思路,教材的内容体系有较高的科学创新、技术创新和理念创新的含量)、先进性(对原有的学科体系有实质性的改革和发展、顺应并符合新世纪教学发展的规律、代表并引领课程发展的趋势和方向)、示范性(教材所体现的课程体系具有较广泛的辐射性和示范性)和一定的前瞻性。教材由个人申报或各校推荐(通过所在高校的“编委会”成员推荐),经“编委会”认真评审,最后由清华大学出版社审定出版。

目前,针对计算机类和电子信息类相关专业成立了两个“编委会”,即“清华大学出版社计算机教材编审委员会”和“清华大学出版社电子信息教材编审委员会”。首批推出的特色精品教材包括:

(1) 高等学校教材·计算机应用——高等学校各类专业，特别是非计算机专业的计算机应用类教材。

(2) 高等学校教材·计算机科学与技术——高等学校计算机相关专业的教材。

(3) 高等学校教材·电子信息——高等学校电子信息相关专业的教材。

(4) 高等学校教材·软件工程——高等学校软件工程相关专业的教材。

(5) 高等学校教材·信息管理与信息系统

清华大学出版社经过近二十年的努力，在教材尤其是计算机和电子信息类专业教材出版方面树立了权威品牌，为我国的高等教育事业做出了重要贡献。清华版教材经过二十多年的精雕细刻，形成了技术准确、内容严谨的独特风格，这种风格将延续并反映在特色精品教材的建设中。

清华大学出版社教材编审委员会

E-mail: dingl@tup.tsinghua.edu.cn

前 言

近年来计算机的软件开发技术有了变革性的飞速发展，一大批面向对象程序设计语言及其开发工具相继出现，并已日益显示出其强大的生命力。

C++是一种高效、实用的程序设计语言，它既可进行过程化程序设计，也可进行面向对象程序设计，是目前编程人员使用最广泛的语言工具。

C++ Builder 是一种极有代表性的面向对象开发工具，它以 C++语言为基础，将面向对象的程序设计方法与数据库技术、网络技术以及可视化、事件驱动、代码自动生成等先进技术完美地结合在一起，使用它可以直观地、快速地开发出高质量的 Windows 应用程序。

程序设计语言与可视化的开发工具这两者是互相依存的。对于语言基础知识的学习，要求提供完善、便捷的运行环境，且应以应用程序的开发作为学习的最终目的；而对开发工具的理解与掌握，则要求以语言知识为基础，且在应用程序的开发中也要用到语言基础知识。本书将这两者较好地结合起来，力求揭示面向对象程序设计语言及面向组件的开发工具的内在联系。

本书分为两部分，第一部分(第1章~第9章)全面、系统地介绍了面向对象程序设计语言 C++的基本概念、基本语法和编程方法，第二部分(第10章~第15章)结合开发工具 C++ Builder 介绍了可视化的开发方法。在章节内容的设置上，既体现了可视化开发工具的强大功能，又强调了语言基础知识在应用程序开发中的重要作用。

本书包含了一系列面向对象的综合开发实例，在实例中运用了可视化的开发方法，体现了语言的基本概念与面向对象程序设计思想，使读者在掌握面向对象程序设计思想的基础上，运用开发工具所提供的强大功能，快速地开发出实用的 Windows 应用程序。

本书的特色是：

- (1) 将 C++的语言基础知识(语法、语义及相关的概念)与可视化开发方法融为一体。
- (2) 以 Windows 应用程序的开发作为学习的最终目的。

本教材可作为《数据结构(面向对象语言描述)》(清华大学出版社出版)的先修教材，在学习前应具备 C 语言的知识。本书所介绍的 C++以 ANSI C++为标准，所使用的 C++ Builder 是 C++ Builder 6.0 的版本。

由于作者水平有限、时间仓促，书中难免存在缺点与疏漏，敬请读者及同行们予以批评指正。

朱振元

目 录

第 1 章 面向对象程序设计概述.....	1
1.1 什么是面向对象程序设计	1
1.2 面向对象程序设计中的基本概念	3
1.2.1 类和对象.....	3
1.2.2 数据封装(信息隐蔽).....	4
1.2.3 继承性.....	4
1.2.4 多态性.....	5
1.3 C++与 C++ Builder 概述	6
1.3.1 C++语言	6
1.3.2 C++ Builder 开发工具	6
1.3.3 编程环境.....	7
第 2 章 C++中的一些新特性.....	9
2.1 输入/输出的新风格.....	9
2.2 const 修饰符	10
2.3 new 和 delete	12
2.4 引用.....	13
2.5 函数原型及参数默认值.....	16
2.6 内联函数.....	17
2.7 灵活的表达方式.....	18
2.7.1 注释行.....	18
2.7.2 强制类型转换.....	18
2.7.3 局部变量说明.....	19
2.7.4 结构名的使用.....	19
2.7.5 作用域运算符.....	19
2.8 应用实例: 链栈的过程化实现.....	20
2.8.1 问题描述.....	20
2.8.2 链栈相关的数据结构与函数.....	20
2.8.3 程序的处理过程.....	21
习题.....	22
第 3 章 类与对象	25
3.1 类的定义.....	25
3.2 接口部分与实现部分.....	26

3.2.1 类的接口部分.....	26
3.2.2 类的实现部分.....	27
3.2.3 内联函数.....	28
3.3 类的封装.....	29
3.4 对象的生成与访问.....	30
3.4.1 对象分配的三种区域.....	30
3.4.2 直接定义对象.....	31
3.4.3 定义对象指针以创建对象.....	32
3.5 this 指针.....	33
3.6 类的作用域.....	34
3.7 应用实例：数字式时钟模拟程序.....	36
3.7.1 问题描述.....	36
3.7.2 类的定义及实现.....	36
3.7.3 处理过程及输出结果.....	37
3.7.4 操作步骤.....	38
习题.....	38
第4章 构造函数与析构函数.....	40
4.1 构造函数的功能及特点.....	40
4.1.1 设置构造函数的必要性.....	40
4.1.2 构造函数的特点.....	41
4.1.3 构造函数的执行.....	42
4.2 构造函数的参数及其默认值.....	42
4.2.1 参数设置.....	42
4.2.2 设置参数的默认值.....	44
4.3 重载构造函数.....	46
4.4 构造函数的初始化表.....	47
4.5 析构函数.....	50
4.6 拷贝构造函数.....	51
4.6.1 拷贝构造函数的形式及功能.....	51
4.6.2 浅拷贝与深拷贝.....	52
4.6.3 拷贝构造函数的执行.....	53
4.7 无名对象与类型转换.....	55
4.7.1 无名对象的使用.....	55
4.7.2 类型转换.....	56
4.8 应用实例：整数集合运算.....	58
4.8.1 问题描述.....	58
4.8.2 Tintset 类的定义.....	58
4.8.3 Tintset 类的实现.....	59

4.8.4 程序的处理过程.....	61
习题.....	61
第5章 静态成员与友元	64
5.1 静态成员.....	64
5.1.1 静态数据成员.....	64
5.1.2 静态函数成员.....	65
5.2 友元.....	66
5.2.1 友元的概念.....	66
5.2.2 友元函数.....	66
5.2.3 友元类.....	68
5.3 const 修饰的对象及类成员	69
5.4 应用实例:链栈处理程序.....	70
5.4.1 问题说明.....	71
5.4.2 链栈类的定义及实现.....	71
5.4.3 程序的处理过程.....	72
习题.....	73
第6章 重载	75
6.1 函数的重载.....	75
6.2 运算符重载概述.....	76
6.3 类运算符重载的两种形式.....	77
6.3.1 友元运算符重载.....	77
6.3.2 成员运算符重载.....	79
6.4 几种特殊运算符的重载.....	81
6.4.1 下标运算符的重载.....	81
6.4.2 转换运算符的重载.....	82
6.4.3 赋值运算符的重载.....	84
6.5 应用实例:复数运算.....	86
6.5.1 复数类的定义.....	86
6.5.2 复数类的实现.....	86
6.5.3 复数运算演示程序.....	87
习题.....	88
第7章 类的继承	91
7.1 继承的概念与派生类定义.....	91
7.1.1 继承的概念.....	91
7.1.2 派生类的定义.....	92
7.2 派生类的访问控制.....	94

7.2.1 保护成员.....	94
7.2.2 访问控制.....	95
7.3 构造函数与析构函数的调用顺序.....	96
7.4 二义性及作用域操作符.....	98
7.4.1 由多个基类的同名成员产生的二义性.....	98
7.4.2 由多个父类的共同基类产生的二义性.....	99
7.4.3 支配规则的运用.....	101
7.5 虚拟继承.....	102
7.6 运行时的多态性及虚函数.....	103
7.7 纯虚函数与抽象类.....	106
7.8 应用实例.....	108
7.8.1 实例一：汽车与赛车信息管理.....	108
7.8.2 实例二：学生与教师评选程序.....	111
习题.....	113
第8章 模板	116
8.1 模板的概念.....	116
8.2 函数模板.....	117
8.3 类模板.....	119
8.4 应用实例：顺序栈处理程序.....	122
8.4.1 顺序栈类模板.....	122
8.4.2 程序的处理过程.....	123
习题.....	124
第9章 I/O 流类	126
9.1 C++中的 I/O 系统.....	126
9.2 标准 I/O 流类.....	127
9.3 格式控制.....	129
9.3.1 使用流类的成员函数.....	129
9.3.2 使用格式控制符.....	132
9.4 重载插入/提取运算符.....	134
9.5 文件流类.....	136
9.5.1 文件流类概述.....	136
9.5.2 文件的打开与关闭.....	137
9.5.3 文件的读写.....	138
9.6 应用实例：文件信息读写程序.....	141
9.6.1 程序中的类定义及数据结构.....	141
9.6.2 程序的处理过程.....	142
习题.....	143

第 10 章 C++ Builder 集成开发环境	147
10.1 面向对象开发工具中的基本概念	147
10.1.1 消息与事件驱动	147
10.1.2 可视化	147
10.1.3 事件处理	148
10.1.4 组件	148
10.1.5 属性	149
10.1.6 方法	149
10.2 VCL 类库	150
10.2.1 VCL 类库概述	150
10.2.2 组件的分类	150
10.2.3 组件的设置与引用	151
10.3 C++ Builder 的集成开发环境	152
10.3.1 主菜单及快捷按钮栏	152
10.3.2 组件板	154
10.3.3 对象监视器	154
10.3.4 窗体与代码编辑器	155
10.3.5 对象树形浏览器	157
10.3.6 工程管理	158
10.3.7 开发界面的调整	160
10.4 创建一个简单的 Windows 应用程序	160
10.4.1 C++ Builder 对 C++ 的扩展	160
10.4.2 创建应用程序的基本步骤	162
10.4.3 应用程序的基本组成	165
第 11 章 输入/输出处理	167
11.1 窗体设计	167
11.1.1 窗体类	167
11.1.2 窗体的主要属性	167
11.1.3 窗体的主要事件	169
11.1.4 窗体设计实例	169
11.2 基本输入/输出组件	170
11.2.1 标签	170
11.2.2 编辑框	171
11.2.3 数字增减器	172
11.2.4 字符串表格	173
11.3 选择输入组件	174
11.3.1 列表选择组件	174

11.3.2	组合框	175
11.3.3	复选框	176
11.3.4	无线按钮	176
11.3.5	分组框	177
11.3.6	无线按钮组	177
11.3.7	选择输入组件的应用实例	178
11.4	按钮与信息显示	179
11.4.1	基本按钮	179
11.4.2	图形按钮	180
11.4.3	信息显示对话框	181
11.5	应用实例：员工信息表维护程序	181
11.5.1	功能要求及组件设置	181
11.5.2	顺序表的类定义	182
11.5.3	顺序表 Tsxb 类的实现	182
11.5.4	程序功能的实现	184
习题		185
第 12 章	日期、时间及字符串处理	187
12.1	用户自定义字符串类	187
12.1.1	Tstring 类的定义	187
12.1.2	Tstring 类的实现	188
12.1.3	字符串类功能演示程序	191
12.2	系统提供的 AnsiString 类	194
12.2.1	AnsiString 类提供的方法	194
12.2.2	字符串处理的相关函数	197
12.3	用户自定义 Tdate 类	199
12.3.1	Tdate 类的定义	199
12.3.2	Tdate 类的实现	199
12.3.3	日期类功能演示程序	201
12.4	系统提供的 TDateTime 类	202
12.4.1	TDateTime 类的方法	202
12.4.2	日期和时间处理的相关函数	204
12.5	应用实例：将播出日程表作成程序	206
12.5.1	可视化组件 CCalendar	206
12.5.2	Timer 组件	206
12.5.3	将播出日程表作成程序	207
习题		209

第 13 章 图形图像处理	211
13.1 图形图像有关的类	211
13.1.1 TCanvas 类的基本属性	211
13.1.2 使用 Canvas 的绘图方法	213
13.1.3 TGraphics 类	216
13.1.4 TPicture 类	216
13.1.5 TBitmap 类	216
13.2 图形图像有关的组件	217
13.2.1 绘图板组件(PaintBox)	217
13.2.2 Shape 组件	218
13.2.3 图像显示组件(Image)	220
13.3 应用实例: 时钟模拟程序	220
13.3.1 功能要求及组件设置	221
13.3.2 时钟类的定义	221
13.3.3 时钟类的实现	222
13.3.4 程序功能的实现	223
13.3.5 操作步骤	225
习题	225
第 14 章 定制组件与异常处理	227
14.1 组件的继承	227
14.2 创建自定义组件的操作步骤	227
14.3 自定义组件实例	229
14.3.1 闪烁标签	229
14.3.2 自定义绘图板	232
14.4 异常处理	233
14.4.1 异常概述	233
14.4.2 异常类及异常处理机制	234
习题	237
第 15 章 多线程编程	239
15.1 线程的概念	239
15.2 C++ Builder 中的线程功能	239
15.2.1 线程的定义	239
15.2.2 线程的优先级	241
15.2.3 线程运行控制	242
15.2.4 线程的互斥与同步	242
15.3 线程应用实例: 八皇后演示程序	243

15.3.1	问题说明	243
15.3.2	功能要求及组件设置	243
15.3.3	实现要点	244
15.3.4	线程类定义	245
15.3.5	程序功能的实现	246
15.3.6	程序清单	247
习题		249
附录	习题参考答案	252

第 1 章 面向对象程序设计概述

C++是一种面向对象的程序设计语言，在学习 C++前，先了解面向对象程序设计的方法及特点是有必要的。在本章中将介绍面向对象程序设计方法的特点及其所涉及的基本概念，以及面向对象程序设计所使用的语言和开发工具。

1.1 什么是面向对象程序设计

在介绍面向对象程序设计方法之前，先来介绍一下传统的面向过程程序设计方法的一些特点。

例如要编制一个程序以演示栈操作的执行过程。在程序中用一个字符表示栈中的一个元素，由输入字符而引起栈中当前元素的变化。若输入字符为 P，则表示执行出栈操作，若为 E，则表示退出执行；否则将该字符推入栈中。栈的初始状态为空，程序从开始起顺序地执行，直至输入字符 E。

使用面向过程的方法可编制以下一段程序：

```
struct Tnode                //定义结点结构
{char data;
  Tnode* next;
};
typedef Tnode* stack;       //定义栈类型
void push(stack& s,char el); //入栈函数
char pop(stack& s);         //出栈函数
void prnt(stack s);         //该函数显示栈中元素
int main(int argc, char* argv[])
{stack s; char el;
  s=NULL; el='a';
  while (el != 'E' )
  { //输入一个字符并存入 el
    //对 el 进行判别并进行相应的处理
    //输出栈中的当前元素
  };
}
```

在上述程序中，定义了一个表示栈的变量 s 及入栈函数 push 和出栈函数 pop，它们都与栈操作有关。但在程序中，表示栈的数据与表示栈操作的函数之间并没有建立必然联系，在程序中的任何地方都可以对 s 进行访问。由此可见，在面向过程的程序设计方法中，数据和操作是分离的，而且程序是从开始至结束顺序地执行的（参见第 2 章中的应用实例）。

这种方法着眼于系统要实现的功能，从系统的输入和输出出发，分析系统要做哪些事情，进而考虑如何做这些事情，自顶向下地对系统的功能进行分解，来建立系统的功能结