

普通高校本科计算机专业特色教材精选·算法与程序设计

MFC Windows 应用程序设计 (第2版)

任 哲 等编著



清华大学出版社

普通高校本科计算机专业特色教材精选·算法与程序设计

MFC Windows 应用程序设计(第2版)

任 哲 等编著

清华大学出版社
北京

内 容 简 介

本书为《MFC Windows 应用程序设计》的第2版。本书沿袭前一版的编写原则和风格,在读者学习了C/C++的基础上,从Windows SDK程序设计出发,用C++类对Windows应用程序进行了模拟封装,重点介绍MFC的应用程序框架。随后以精练的语言介绍Windows的图形图像处理、应用程序界面、资源、文件、动态链接库、进程与线程管理等知识。最后,介绍组件对象模型(COM)、ActiveX技术、数据库和异常处理的基础知识。

本书内容丰富、结构合理、重点突出、注重应用、兼顾说理性和实用性、文字通俗易懂,以适当的方法加强了对MFC重点技术核心内容的介绍,使读者可以在有限的文字篇幅情况下获得尽可能大的信息量,适合作为普通高等院校的教学及教学参考用书,也可以作为其他计算机开发应用人员的参考书。

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。

版权所有,侵权必究。侵权举报电话:010-62782989 13501256678 13801310933

图书在版编目(CIP)数据

MFC Windows 应用程序设计/任哲等编著. —2版. —北京:清华大学出版社,2007.9
(普通高校本科计算机专业特色教材精选·算法与程序设计)
ISBN 978-7-302-15549-2

I. M… II. 任… III. ①C语言—程序设计—高等学校—教材 ②窗口软件, Windows—程序设计—高等学校—教材 IV. ①TP312 ②TP316.7

中国版本图书馆CIP数据核字(2007)第096749号

责任编辑:汪汉友 徐跃进

责任校对:时翠兰

责任印制:王秀菊

出版发行:清华大学出版社

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

c-service@tup.tsinghua.edu.cn

社总机:010-62770175

投稿咨询:010-62772015

地 址:北京清华大学学研大厦A座

邮 编:100084

邮购热线:010-62786544

客户服务:010-62776969

印 刷 者:北京市世界知识印刷厂

装 订 者:三河市溧源装订厂

经 销:全国新华书店

开 本:185×260 印 张:25.5

版 次:2007年9月第2版

印 数:1~4000

定 价:33.00元

字 数:569千字

印 次:2007年9月第1次印刷

本书如存在文字不清、漏印、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:010-62770177 转 3103 产品编号:025174-01

出版说明

INTRODUCTION

在我国高等教育逐步实现大众化后,越来越多的高等学校将会面向国民经济发展的第一线,为行业、企业培养各级各类高级应用型专门人才。为此,教育部已经启动了“高等学校教学质量和教学改革工程”,强调要以信息技术为手段,深化教学改革和人才培养模式改革。如何根据社会的实际需要,根据各行各业的具体人才需求,培养具有特色显著的人才,是我们共同面临的重大问题。具体地说,培养具有一定专业特色的和特定能力强的计算机专业应用型人才则是计算机教育要解决的问题。

为了适应 21 世纪人才培养的需要,培养具有特色的计算机人才,急需一批适合各种人才培养特点的计算机专业教材。目前,一些高校在计算机专业教学和教材改革方面已经做了大量工作,许多教师在计算机专业教学和科研方面已经积累了许多宝贵经验。将他们的教研成果转化为教材的形式,向全国其他学校推广,对于深化我国高等学校的教学改革是一件十分有意义的事情。

清华大学出版社在经过大量调查研究的基础上,决定组织出版一套“普通高校本科计算机专业特色教材精选”。本套教材是针对当前高等教育改革的新形势,以社会对人才的需求为导向,主要以培养应用型计算机人才为目标,立足课程改革和教材创新,广泛吸纳全国各地的高等院校计算机优秀教师参与编写,从中精选出版确实反映计算机专业教学方向的特色教材,供普通高等院校计算机专业学生使用。

本套教材具有以下特点:

1. 编写目的明确

本套教材是在深入研究各地各学校办学特色的基础上,面向普通高校的计算机专业学生编写的。学生通过本套教材,主要学习计算机科学与技术专业的基本理论和基本知识,接受利用计算机解决实际问题的基本训练,培养研究和开发计算机系统,特别是应用系统的基本能力。

2. 理论知识与实践训练相结合

根据计算学科的三个学科形态及其关系,本套教材力求突出学科的理论与实践紧密结合的特征,结合实例讲解理论,使理论来源于实践,又进一步指导实践。学生通过实践深化对理论的理解,更重要的是使学生学会理论方法的实际运用。在编写教材时突出实用性,并做到通俗易懂,易教易学,使学生不仅知其然,知其所以然,还要会其如何然。

3. 注意培养学生的动手能力

每种教材都增加了能力训练部分的内容,学生通过学习和练习,能比较熟练地应用计算机知识解决实际问题。既注重培养学生分析问题的能力,也注重培养学生解决问题的能力,以适应新经济时代对人才的需要,满足就业要求。

4. 注重教材的立体化配套

大多数教材都将陆续配套教师用课件、习题及其解答提示,学生上机实验指导等辅助教学资源,有些教材还提供能用于网上下载的文件,以方便教学。

由于各地区各学校的培养目标、教学要求和办学特色均有所不同,所以对特色教学的理解也不尽一致,我们恳切希望大家在使用教材的过程中,及时地给我们提出批评和改进意见,以便我们做好教材的修订改版工作,使其日趋完善。

我们相信经过大家的共同努力,这套教材一定能成为特色鲜明、质量上乘的优秀教材。同时,我们也希望通过本套教材的编写出版,为“高等学校教学质量和教学改革工程”做出贡献。

清华大学出版社

第2版 前言

PREFACE

由于广大读者的抬爱及清华大学出版社的鼓励,本书有幸再版。在此首先对广大读者和出版社表示衷心的感谢。自本书第1版出版以来,作者陆续收到一些读者来信,针对这些来信,作者准备谈以下两个问题。

1. 为什么要学习 MFC

作者在第1版前言中曾说过,通过对MFC的学习,除了可以使學生掌握一种Windows应用程序设计的基本方法之外,还可以使他们进一步全面、深刻地理解面向对象程序设计思想,从而把握程序设计方法的发展方向。对此,曾经有读者提出过疑问:Visual Basic、Delphi、C#.NET、Java不都是面向对象的吗?为什么非要学习目前看起来似乎有些落后的MFC呢?

诚然,从时间上来看,MFC是微软较早开发的Windows应用程序框架,历史比较悠久。但是,请读者要注意,历史悠久与落后绝不是同一概念。当然也不意味着只要学会了MFC就可以打遍天下,解决所有问题,但就MFC所蕴含及所展露的程序设计思想、代码实现技巧,则是其他开发工具所不及的。

作为世界上第一个应用程序框架类库的MFC,是在C++尚未完善,尚不支持某些功能的时候产生的,MFC设计者们就是使用这种当时还比较低级的C++实现了这个庞大程序框架的各种复杂功能,因此MFC的各个角落都充满了这些设计者的智慧和技巧,这些智慧和技巧都体现了当今软件工程技术的重要思想和发展方向,以至于对C++的发展和随后出现的Java、C#技术等,都产生了极为深刻的影响,甚至有些就是MFC技术的直接应用。作者认为,迄今为止还没有一个应用程序框架类库能具有如此之大的信息量,所以它是计算机专业教育和教学的绝好素材。

作者认为,通过对MFC的学习至少可以达到如下几个目的:一是可以学习Windows应用程序的特点(图形界面、多任务、事件驱动)、数据和函数的组织方法(资源、动态链接库)和结构(接收、分派消息的主程序模块和处理消息的消息相应程序模块)及其处理思想;二是通过MFC

这个大实例,可以学习面向对象程序设计的实际方法(例如文档/视图结构的构思);三是可以学习 MFC 设计者们的一些巧妙的程序设计方法和技巧(例如消息映射、对象动态创建、序列化);四是可以学习 Visual C++ 这个开发环境的使用,真可谓是一举多得。所以,作为每一个立志从事计算机工作的人来讲,不在一定程度上了解和掌握 MFC 将会是他的终生遗憾。

当然,其他开发工具也有它们各自的特点和优势,也值得学习。例如,Visual Basic 作为微软开发的第一套可视化开发环境,是它把 DOS 程序员从汪洋大海般的 Windows API 函数中解救出来,为 Windows 应用程序的设计和发展做出了不可磨灭的贡献。但是,它使用的并不是非学不可的 Basic 这种非主流语言,所以对于学时极为有限和宝贵的学生来说,是否非得在校学习它就值得考虑。而使用 PASCAL 语言的 Delphi 和使用 C/C++ 语言的 C++ Builder 以其提供的丰富组件,在快速开发应用程序方面确实具有一定的优势,但它们与 Visual Basic 一样,把 Windows 底层封装得太严了,程序员只剩下填空的份儿,以至于使它们成了一种“傻瓜”式的开发工具,这当然不适合以教授和学习程序设计思想和设计方法为目的的学校来使用。

当然,作者也认可业界“聪明人用 Delphi,高手用 VC”之类的说法,因为在实际工程中,为了满足工程期限的需要,而采用 Delphi 之类的快速开发工具是合理并应该的。但并不意味着学生在校期间一定要拿出大量的课时去学习这些快速开发工具(工具无所谓会不会的问题,而是熟不熟的问题),因为高等学校毕竟是教育机构而不是培训机构,不但要让学生知其然也更应该让其知其所以然,从而在此基础上形成较强的软件开发(甚至是研发)能力,而不是只让学生会几门流行的技术而使学生一辈子在技术发展背后疲于奔命。

那么学习 Windows SDK 编程又如何呢?作者认为,用函数编程这是 C 程序设计课程解决的问题,学生学习了 C 语言之后只要肯下功夫,多实践,自学 Windows SDK 程序设计是完全可以的,没有必要再浪费宝贵的课时了。

也正因为上述种种原因,所以本书既不以剖析 MFC 源代码为主,也不以介绍 Visual C++ 开发工具的使用为主,而是以重点介绍 MFC 框架的设计思想、理念和方法为基础,兼顾应用,从而使读者通过对 MFC 的学习,除了掌握 MFC 之外,还能进一步深化理解 C/C++、数据结构、操作系统乃至计算机硬件的相关知识,尽快提高程序设计能力并学习其他应用程序设计工具打下良好的基础。

2. 第2版的变化

自本书的第1版出版以来,笔者陆续接到许多读者的来信,其中既有教师也有学生和程序设计人员,在信中都在对本书褒奖之余对书的内容及编写提出了很好的意见与建议,并希望作者在有机会的条件下进行修改。在此,作者首先对这些关心本书的读者表示由衷的谢意。

综合了读者的意见之后,《MFC Windows 应用程序设计(第2版)》特做如下变化:

因为本书旨在说明 MFC 的思想,所以只要能以模拟方式说明其思想的,本书就采用了模拟方法。显然本书第1版在消息映射的模拟和叙述上过于潦草,给一部分读者学

习这部分时造成了一些不必要的困难。故此，作者在第2版中进行了必要的修改，使得这部分的介绍更为合理、清楚。

文档/视图结构是MFC应用程序框架的核心。为此，作者在第2版的第3章中适当增加了文档/视图结构程序框架各对象的创建顺序、对象之间的联系、消息的传递等内容，以期使读者对文档/视图结构及其各对象之间的关系有更清楚的认识和了解，为读者精研MFC打下一个更加良好的基础。另外，在这一章中还加强了对对象动态创建的介绍。

参考读者建议，作者把第9章的内容在顺序上做了改变：先介绍相对比较容易的CFile类，后介绍CArchive类，最后介绍文档序列化，并且加强了序列化部分的说理性。

异常及异常处理是一个健壮的应用程序必备特性，这方面的内容的缺失不能不说是第1版的一个缺憾，这次在第2版中把它补齐了。

当然，为追求内容的完整性，本书至少还应该补上MFC在网络上的应用（这也是读者的建议），但作者思虑再三后认为，作为教材本书还是不包含这部分内容为好，一是因为MFC在网络程序方面并不占优势，二是因为各个学校都是在C#.NET或Java等课程中处理该问题，如果本书包括这部分内容的话，在教学上会造成内容的重叠。

另外，正如一些读者所指出的，第14章“组件对象模型COM基础”出现在本书中显得比较生硬：第一，组件对象模型的内容对于本科生来讲似乎是难了一些；第二，这本书讲的是MFC，而这一章讲的是ATL，确实与本书不配套。关于第一点，作者认为不是问题，学生觉得难，一是因为没有好的教材，二是学生原来接触这方面的技术比较少，但不管是什么原因，在组件应用如此广泛的今天，计算机专业特别是以软件设计为培养目标的专业不向学生讲授这方面的知识是无论如何不行的，但是，现在大多数本科学校又没有相应的课程来介绍这方面的内容，再者组件技术把抽象类的应用发挥到了淋漓尽致的程度，不学它是不会深切领会抽象类的精神实质以及它的应用的，所以本书就勉为其难地编写了这部分的入门知识。为了降低学习这部分内容的台阶，作者重写了这一章。至于第二点，生硬就生硬了吧，因为ATL在组件设计方面确实要比MFC强许多。

正如前面所讲，MFC信息量巨大，涉及到程序设计的方方面面，因此需要读者在阅读本书时不但要不断地复习C/C++的内容，还要不时地阅读其他书籍，尤其是数据结构、操作系统等。

另外，还要多找一些实例上机做实验（提供这样例题的书籍及网上资料很多）。

本书在编写过程中参考了大量的相关文献和网上的资料，并引用了其中的一些例题、文字和插图，在此对这些文献和资料的作者表示诚挚的谢意。同时也对关心本书的读者表示万分的感谢。

参加本书编写工作的有任哲、房红征、李益民、车进辉。另外，刘泰辉博士对于本书的章节及内容的安排提出的良好意见和建议，也使本书增色不少。

将信息量如此巨大的MFC在一本50万字的教学用书中进行介绍，对水平有限的作者来说实在是一件难事，所以在此恳请读者对书中的缺点和错误提出批评和指正。同

时,也希望阅读本书的读者或使用本书作为教材的教师能与作者共同探讨,以期对本书作进一步的改进。欢迎读者及时与作者沟通。

通信地址: 北华大学计算机科学技术学院

邮编: 132021

电子信箱: renzhe71@sina.com.cn

祝各位教学有成!

作者

2007 年 7 月

第1版 前言

PREFACE

这些年来, Windows 一直是一个主流操作系统, 市场上对开发 Windows 应用程序的需求与日俱增。随着 Windows 操作系统的发展, Windows 应用程序变得越来越复杂, 使得单纯使用 Windows API 来开发 Windows 应用程序变得越来越困难。于是, 一些可视化的 Windows 开发工具(如 Visual Basic、Delphi、C++ Builder 等)如雨后春笋般地涌现出来, 这无疑给 Windows 应用的开发带来了极大的方便, 同时也大大地提高了开发效率。但是, 由于这些开发工具对应用程序框架的代码封装得过于严密, 因而使得对 Windows 低层应用的开发几乎成为不可能。而 Microsoft 基础类库(Microsoft foundation classes, MFC)只是使用 C++ 类对 Windows API 进行了封装, 它一方面较好地屏蔽了 Windows 编程的复杂性, 使得 Windows 应用程序的设计变得简单起来, 另一方面仍然允许有经验的程序员使用 Windows 的底层功能开发高效的 Windows 应用程序, 再加上与 MFC 配套的开发工具 Visual C++ 提供的功能强大的各种 Wizard(向导), 使得 MFC 成为了一个强有力的 Windows 应用程序的开发工具。

目前, 各高等学校已经把 C/C++ 列为理工科专业学生的必修课程, 这门课程使学生初步了解和掌握了面向对象程序设计的思想和方法, 这无疑为学生学习和掌握 MFC 打下了良好的基础, 同时对 MFC 的学习, 除了可以使学生掌握 Windows 应用程序设计的基本方法之外, 又可以使学生更进一步深刻、全面地理解面向对象程序设计的思想, 从而把握程序设计方法的发展方向。

基于上述这些理由, 各个高校都在 MFC 的教学方面进行了积极探索。本书是作者积数年教学经验, 为高等学校理工科学生编写的一部应用 MFC 进行 Windows 应用程序设计的入门教材。

全书共有 16 章, 参考教学时数 50 学时。

本书的特点是遵循循序渐进的教学原则, 以较大的篇幅从 Windows SDK 程序入手, 在介绍 Windows 应用程序一般特点的基础上, 重点介绍 Windows 的消息机制和 Windows 程序的基本结构, 以 C++ 类对其进行模拟

封装使学生迅速建立起 MFC 应用程序框架的基本概念; 然后以精练的语言介绍 Windows 的图形图像处理、应用程序界面、资源、动态链接库、进程与线程管理等知识。最后, 为使学生对软件工程的发展有一个初步的了解又介绍了组件对象模型 (COM)、ActiveX 技术和数据库的基本常识。

为配合教学和学习, 本书为每一个知识点都配以必要的实例, 力求通过实例让读者掌握 MFC 程序设计的特点及方法, 所有实例都在 Windows 98 和 Visual C++ 6.0 环境中测试通过。并且, 为配合教师教学, 本书为使用本教材的教师免费提供源代码和电子教案。

本书在编写过程中得到了吉林大学张长海教授及其他教师的悉心指导, 在本书的内容、章节编排等方面提出了宝贵意见和建议, 在此对他们的帮助表示衷心的感谢。本书在编写中还参考了大量的相关文献, 并引用了其中的一些例题和内容, 在此也对这些文献的作者表示诚挚的谢意。

参加本书编写工作的有任哲、李益民、车进辉。由于作者水平有限, 恳请读者对书中的缺点和错误提出批评和指正, 欢迎读者与作者联系。

通信地址: 北华大学计算机科学技术学院

邮编: 132021

电子信箱: renzhe71@sina.com.cn

作 者

2004 年 4 月

目 录

CONTENTS

第 1 章 Windows 应用程序基础知识	1
1.1 Windows 应用程序的基本概念	1
1.1.1 Windows 应用程序的界面	1
1.1.2 Windows 应用程序与 API 函数	1
1.1.3 Windows 的一个特殊数据类型——句柄	2
1.1.4 Windows 应用程序的数据类型	3
1.1.5 消息与消息循环	4
1.2 Windows 应用程序的结构	6
1.2.1 主函数	6
1.2.2 消息的处理部分——窗口函数	8
1.2.3 Windows 系统、主函数、窗口函数之间的关系	9
1.3 用函数封装 Windows 程序	15
1.4 窗口函数的另一种结构	19
小结	22
练习题	22
思考题	23
第 2 章 Windows 应用程序的类封装	25
2.1 应用程序主函数的 C++ 类封装	25
2.1.1 窗口类的声明	26
2.1.2 应用程序类的声明	27
2.1.3 主函数封装后的程序	28
2.2 派生类的应用	32
2.2.1 应用程序类的派生类	33
2.2.2 窗口类的派生类	35
2.3 窗口函数的封装——消息映射	39
2.3.1 消息处理函数的简单封装	40

2.3.2 消息映射	42
2.3.3 消息映射表的声明和实现	45
小结	48
练习题	48
思考题	49
第3章 MFC 应用程序框架	51
3.1 早期的应用程序框架及其 MFC 类	51
3.1.1 早期的应用程序框架	51
3.1.2 MFC 的窗口类	52
3.1.3 CWinApp 类	53
3.2 最简单的 MFC 程序实例	54
3.2.1 程序的编写	54
3.2.2 程序主函数的代码	56
3.3 应用程序的文档/视图结构	57
3.3.1 文档/视图结构的基本概念	57
3.3.2 单文档界面和多文档界面结构	59
3.4 文档类 CDocument 的派生类	60
3.5 视图类 CView 的派生类	61
3.6 窗口框架类 CFrameWnd 的派生类	62
3.7 文档模板类 CDocTemplate	64
3.8 应用程序类的派生类	65
3.8.1 应用程序类派生类的代码	65
3.8.2 程序员的主要工作	66
3.9 MFC 文档/视图应用程序框架中各个对象的关系	67
3.9.1 应用程序各对象创建的顺序	67
3.9.2 应用程序各对象之间的联系	68
3.9.3 文档/视图应用程序消息的传递	71
3.10 对象的动态创建	71
3.10.1 问题的提出与解决	71
3.10.2 类信息表及其声明和实现	76
3.10.3 对象类信息表的提取	77
小结	78
练习题	78
思考题	78
第4章 图形	79
4.1 DC 和 GDI	79

4.1.1 设备描述环境	79
4.1.2 Windows 的 GDI	80
4.2 CDC 类	80
4.3 CPen 类	82
4.4 CBrush 类	85
4.5 绘图模式	87
4.6 文本和 CFont 类	88
4.6.1 显示文本	88
4.6.2 字体和 CFont 类	91
4.7 CDC 的其他派生类	93
4.7.1 窗口用户区设备描述环境 CClientDC 类	93
4.7.2 图元文件设备描述环境 CMetaFileDC 类	94
小结	96
练习题	96
思考题	96
第 5 章 MFC 的通用类	97
5.1 简单数据类	97
5.1.1 点类 CPoint	97
5.1.2 矩形类 CRect	98
5.1.3 尺寸类 CSize	98
5.1.4 字符串类 CString	99
5.2 群体数据类	100
5.3 数据在文档/视图结构程序中的位置	102
小结	104
练习题	104
思考题	104
第 6 章 Windows 应用程序界面的设计	105
6.1 SDI 和 MDI 界面	105
6.2 SDI 界面窗口的样式	106
6.2.1 在 MFC AppWizard 中确定窗口样式	106
6.2.2 在函数 PreCreateWindow 中修改窗口的样式	107
6.2.3 可拆分为小窗口的 SDI 界面	111
6.3 拆分窗口的同步更新及更新效率	112
6.3.1 拆分窗口的同步更新	112
6.3.2 提高拆分窗口更新效率的方法	112
6.4 创建带有滚动条的窗口	117

小结	121
练习题	121
思考题	122
第7章 鼠标和键盘	123
7.1 处理鼠标消息	123
7.1.1 用户区鼠标消息	124
7.1.2 非用户区鼠标消息	128
7.1.3 鼠标消息的捕获	129
7.2 处理键盘消息	130
7.2.1 按键的虚拟码	130
7.2.2 一般按键消息	133
7.2.3 系统按键消息	134
7.3 应用程序窗口的焦点	135
小结	136
练习题	137
思考题	138
第8章 资源	139
8.1 资源和资源文件	139
8.1.1 什么是资源	139
8.1.2 资源头文件	140
8.1.3 资源描述文件	141
8.2 菜单	141
8.2.1 资源描述文件的菜单部分	141
8.2.2 编辑现有的菜单	143
8.2.3 自定义菜单	146
8.2.4 用菜单编辑器编辑菜单	148
8.2.5 菜单命令选项的动态修改	148
8.3 快捷键表	151
8.3.1 资源描述文件的快捷键表部分	151
8.3.2 使用快捷键编辑器编辑快捷键表	152
8.4 图标	152
8.4.1 制作图标	152
8.4.2 资源描述文件的图标部分	153
8.5 位图	154
8.5.1 资源描述文件的位图部分	154
8.5.2 位图资源的使用	154

小结	157
练习题	158
思考题	158
第 9 章 MFC 的文件处理机制	159
9.1 CFile 类	159
9.1.1 CFile 类常用成员函数	159
9.1.2 文件的创建与打开	159
9.1.3 文件的读写	161
9.1.4 CFile 的派生类 CMemFile	165
9.1.5 CFileFind 类	166
9.2 CArchive 类	167
9.2.1 插入符和提取符的重载	167
9.2.2 CArchive 类对象与 CFile 类对象的关联	168
9.3 对象的序列化	170
9.3.1 序列化的基本概念	170
9.3.2 MFC 的 Serialize 函数	171
9.3.3 MFC 应用程序的序列化过程	173
9.3.4 序列化的顺序	174
9.4 宏 DECLARE_SERIAL 和 IMPLEMENT_SERIAL	177
9.5 MFC 文件命令的默认处理流程	182
9.5.1 应用程序启动后文档的初始化流程	182
9.5.2 应用程序菜单命令的默认处理	182
小结	183
练习题	183
思考题	183
第 10 章 控件	185
10.1 标准控件及其使用	185
10.1.1 标准控件	186
10.1.2 静态文本控件	186
10.1.3 按钮控件	188
10.1.4 编辑控件	192
10.2 通用控件	194
10.2.1 进度条控件	194
10.2.2 微调器控件	195
10.2.3 图像列表控件	199
10.2.4 列表视图控件	200

10.3 控件的背景颜色	208
小结	210
练习题	211
第 11 章 对话框	213
11.1 对话框模板资源描述文件	213
11.2 自定义对话框类	215
11.2.1 MFC 的对话框类及用户自定义对话框类	216
11.2.2 模态对话框	217
11.3 对话框的数据交换和数据检验	221
11.3.1 数据交换	221
11.3.2 使用 Class Wizard 为对话框添加成员变量	223
11.3.3 数据检验	225
11.4 以对话框为主界面的应用程序	225
11.4.1 应用程序框架的设计	226
11.4.2 应用程序代码的设计	226
11.5 MFC 预置的通用对话框	228
11.5.1 文件对话框	228
11.5.2 字体选择对话框	230
11.6 非模态对话框	233
11.7 属性页	235
小结	238
练习题	239
思考题	239
第 12 章 进程与线程的管理	241
12.1 进程、线程及其优先权	241
12.1.1 进程、线程的基本概念	241
12.1.2 进程和线程的优先级	241
12.2 工作线程和用户界面线程	243
12.2.1 工作线程的创建	243
12.2.2 用户界面线程的创建	244
12.3 线程同步	246
12.3.1 线程同步的基本概念	246
12.3.2 事件	247
12.3.3 临界段	251
12.3.4 互斥体	253
12.3.5 信号量	254