

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 习题解答及上机实验(第2版)

任 哲 李益民 车进辉 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 习题解答及上机实验(第2版)

任 哲 李益民 车进辉 编著

清华大学出版社
北京

内 容 简 介

本书是《MFC Windows 应用程序设计(第2版)》的配套教材。书中对《MFC Windows 应用程序设计(第2版)》的习题做了详细的解答并提供了一些上机实验题目,力图通过这些练习和训练使读者能较好地理解和掌握 MFC Windows 应用程序框架及使用 MFC 类库编程的基本方法。在书后的附录 Visual C++ 6.0 集成开发工具介绍部分,简明扼要地介绍了创建应用程序框架的方法及使用调试开发工具的方法,为读者快速使用该开发工具提供了帮助。

本书适合作为高等学校和高等职业学校计算机专业的教学用书,同时也适合其他有一定 C++ 基础的读者,它可以在较短的时间内使读者了解和掌握开发 MFC Windows 应用程序的特点和设计方法。

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 应用程序设计习题解答及上机实验/任哲,李益民,车进辉编著. —2版.

—北京:清华大学出版社,2007.10

(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 978-7-302-15737-3

I. M… II. ①任… ②李… ③车… III. ①C语言—程序设计—高等学校—教学参考资料 ②窗口软件, Windows—程序设计—高等学校—教学参考资料 IV. TP312
TP316.7

中国版本图书馆 CIP 数据核字(2007)第 108187 号

责任编辑:汪汉友 徐跃进

责任校对:李建庄

责任印制:孟凡玉

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

印装者:北京市清华园胶印厂

经 销:全国新华书店

开 本:185×260 印 张:7.5

版 次:2007年10月第2版

印 数:1~4000

定 价:13.00元

字 数:169千字

印 次:2007年10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:025173-01

编审委员会

主任：蒋宗礼

副主任：李仲麟 何炎祥

委员：（排名不分先后）

王向东	宁 洪	朱庆生	吴功宜	吴 跃
张 虹	张 钢	张为群	余雪丽	陈志国
武 波	孟祥旭	孟小峰	胡金初	姚放吾
原福永	黄刘生	廖明宏	薛永生	

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的经验。并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，而这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

前言

PREFACE

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。如果在此基础上开设 MFC 课程,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,把握住计算机程序设计方法的发展方向,从而为今后进一步提高计算机程序设计能力打下坚实基础。

本书作为《MFC Windows 应用程序设计(第2版)》一书的配套教材,除了提供了《MFC Windows 应用程序设计(第2版)》教材中各章习题的解答外还设计了一些课后上机实验,学生通过这些必要的练习可以较快地了解 MFC 的框架并掌握使用 MFC 类库进行 Windows 应用程序设计的一般方法。

为使学生可以较快地熟悉 Visual C++ 开发环境中的众多开发工具,书后还有选择地介绍了 Visual C++ 部分开发工具的使用方法。

本书在编写中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

2007年7月

目 录

CONTENTS

第 1 章 Windows 应用程序基础知识习题解答及上机实验	1
1.1 习题解答	1
1.2 上机实验	2
第 2 章 Windows 应用程序的类封装习题解答及上机实验	7
2.1 习题解答	7
2.2 上机实验	7
第 3 章 MFC 的程序框架习题解答及上机实验	11
3.1 习题解答	11
3.2 上机实验	13
第 4 章 图形习题解答及上机实验	17
4.1 习题解答	17
4.2 上机实验	19
第 5 章 MFC 的通用类习题解答及上机实验	21
5.1 习题解答	21
5.2 上机实验	24
5.2.1 简单通用类的应用	24
5.2.2 群体类的应用	26
第 6 章 Windows 应用程序界面的设计习题解答及上机实验	29
6.1 习题解答	29
6.2 上机实验	30
6.2.1 实验 1	30
6.2.2 实验 2	31

第7章 鼠标和键盘习题解答及上机实验	33
7.1 习题解答	33
7.2 上机实验	43
第8章 资源习题解答及上机实验	47
8.1 习题解答	47
8.2 上机实验	56
第9章 MFC 的文件处理机制习题解答	59
第10章 控件习题解答	61
第11章 对话框习题解答及上机实验	63
11.1 习题解答	63
11.2 上机实验	68
第12章 进程与线程的管理习题解答	73
第13章 动态链接库习题解答	75
第14章 组件对象模型(COM)基础习题解答	77
第15章 ActiveX 应用基础习题解答	79
第16章 用 MFC 设计数据库应用程序习题解答	81
第17章 异常和异常处理习题解答	83
附录 A Visual C++ 开发环境简介	85

Windows 应用程序基础知识

习题解答及上机实验

1.1 习题解答

1-1 什么是 Windows SDK 程序?

答: 使用软件开发工具包 (software development kit) 开发出来的 Windows 应用程序叫做 Windows SDK 程序。

1-2 什么是 Windows API 函数?

答: 用来开发 Windows SDK 应用程序的软件开发工具包是用 C 语言编写的一个大型函数库, 这个库中的函数叫做 API 函数。

1-3 打开 windows.h 文件, 看一下 Windows 系统的句柄是什么数据类型的?

答: 整型。

1-4 试说明以下是什么句柄?

(1) HWND;

(2) HINSTANCE;

(3) HDC。

答: HWND 窗口句柄。

HINSTANCE 应用程序实例句柄。

HDC 图形设备环境句柄。

1-5 什么是事件? 试举例说明。

答: 能触发程序作出相应反应的因素或动作叫做“事件”。例如, 在键盘上按下了一个键、鼠标的单击或双击、应用程序窗口的显示和销毁等。

1-6 如何显示和更新窗口?

答: 调用函数 ShowWindow 显示窗口, 调用函数 UpdateWindow 更新窗口。

1-7 什么是消息循环?

答: 在创建了窗口的应用程序中, 应用程序将不断地从消息队列中获取消息, 并将消息指派给指定的窗口处理函数来处理, 然后再回来从消息队列获取消息, 这个不断重复的工作过程叫做消息循环。

1-8 Windows 应用程序的主函数有哪三个主要任务?

答: 注册窗口类、创建应用程序的窗口和建立消息循环。

1-9 说明 Windows 应用程序的主函数、窗口函数与 Windows 系统之间的关系。

答: Windows 应用程序的主函数和窗口函数都是系统调用的函数, 主函数是在应用程序启动时由系统首先调用的函数, 而窗口函数是主函数在消息循环中获得消息并把消息派送给系统之后, 由系统调用的用来处理消息的函数。

1.2 上机实验

实验内容:

在 Visual C++ 6.0 中创建 Win32 Application 工程。

实验目的:

1. 熟悉在 Visual C++ 6.0 中创建 Win32 Application 的过程。
2. 熟悉 Visual C++ 6.0 可以创建的三种 Win32 Application。
3. 在 Visual C++ 6.0 中查看帮助文件。

实验步骤:

(1) 选择【开始】|【程序】|Microsoft Visual Studio 6.0|Microsoft Visual C++ 6.0 命令启动 Visual C++ 6.0, 见图 1-1。

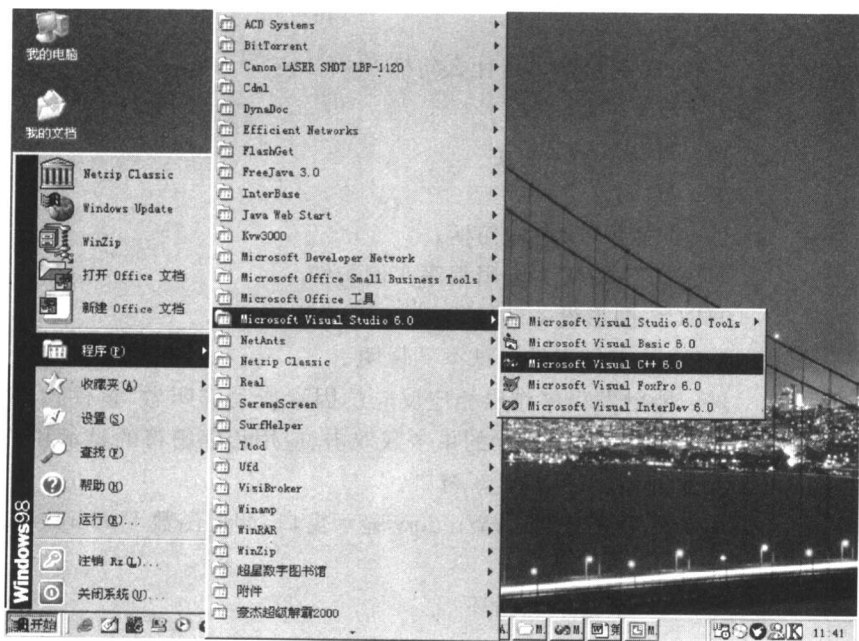


图 1-1 启动 Visual C++ 6.0

(2) 选择 Visual C++ 6.0 的菜单 File|New 命令打开 New 对话框,在这个对话框中选择 Projects 选项卡,见图 1-2。

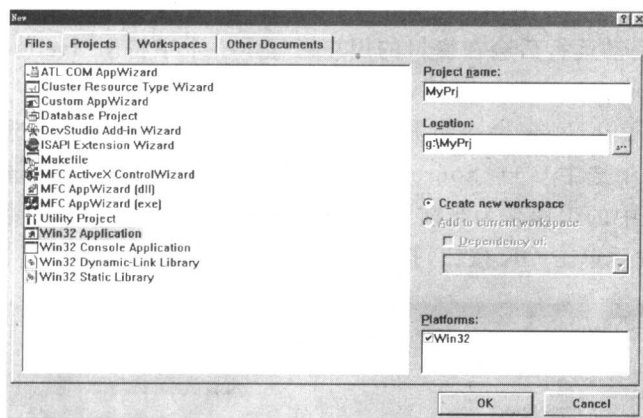


图 1-2 New 对话框的 Projects 选项卡

(3) 在左侧的列表框中选中 Win32 Application 选项,在 Project name 编辑框中填写工程名称(例如,MyPrj),在 Location 编辑框中选择存放工程文件的路径,其余使用默认选项,见图 1-2。最后,单击 OK 按钮,打开 Win32 Application-Step 1 of 1 对话框,见图 1-3。

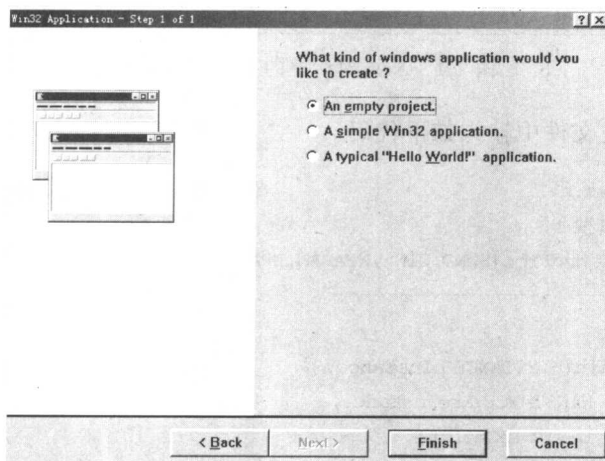


图 1-3 Win32 Application-Step 1 of 1 对话框

(4) 在图 1-3 所示的对话框中,有三种工程类型可以供选择:

- An empty project;
- A simple Win32 application;
- A typical “Hello World!” application.

选择 An empty project 选项,可以创建一个没有任何文件的空工程,用户需要自己向工程中添加所需要的文件。

选择 A simple Win32 application 选项,可以创建一个由系统自动生成必要代码的工程。

选择 A typical “Hello World!” application 选项,可以创建一个完整的 Win32 示例程序,这个程序在启动运行后会在窗口的用户区显示一个字符串“Hello World!”。

(5) 选择 An empty project 选项,创建一个空 Win32 工程。

(6) 选择菜单 File|New 命令打开 New 对话框,在这个对话框中选中 Files 选项卡,在卡中左侧的窗口中选中 C++ Source File,在确认 Add to project 复选项被选中且在它下面的下拉列表框中的工程名为当前工程名后,在 File 文本框中填写要创建文件的名称(例如,MyPrj)。最后单击 OK 按钮,打开文件,见图 1-4。

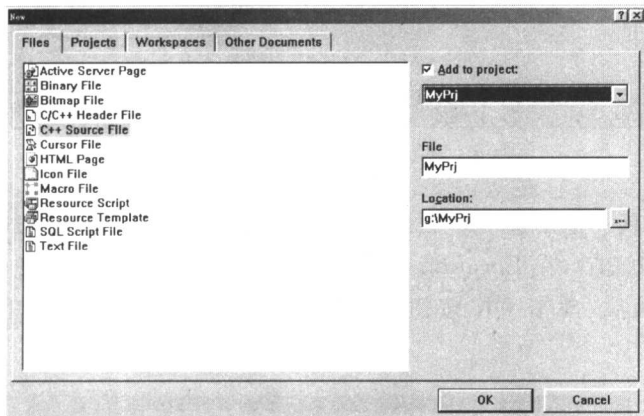


图 1-4 New 对话框的 Files 选项卡

(7) 在打开的空文件中输入如下代码。

```
#include<windows.h>                                //编写 Windows 程序必须要包含的头文件
//声明窗口函数原型
LRESULT CALLBACK WndProc (HWND,UINT,WPARAM,LPARAM);
//-----
//主函数
int WINAPI WinMain(HINSTANCE hInstance,
                  HINSTANCE PreInstance,
                  LPSTR lpCmdLine,
                  int nCmdShow)
{
    HWND hwnd;                                     //定义窗口句柄
    MSG msg;                                       //定义一个用来存储消息的变量
    char lpszClassName[]="窗口";
    WNDCLASS wc;                                  //定义一个窗口类变量
    wc.style=0;
    wc.lpfnWndProc=WndProc;
    wc.cbClsExtra=0;
    wc.cbWndExtra=0;
```

```
wc.hInstance=hInstance;
wc.hIcon=LoadIcon(NULL,IDI_APPLICATION);
wc.hCursor=LoadCursor(NULL,IDC_ARROW);
wc.hbrBackground=(HBRUSH)GetStockObject(WHITE_BRUSH);
wc.lpszMenuName=NULL;
wc.lpszClassName=lpszClassName;

RegisterClass(&wc); //注册窗口类

hwnd=CreateWindow(lpszClassName, //创建窗口
                  "Windows",
                  WS_OVERLAPPEDWINDOW,
                  120,50,800,600,
                  NULL,
                  NULL,
                  hInstance,
                  NULL);

ShowWindow(hwnd,nCmdShow); //显示窗口
UpdateWindow(hwnd);

while(GetMessage(&msg,NULL,0,0)) //消息循环
{
    TranslateMessage(&msg);
    DispatchMessage(&msg);
}
return msg.wParam;
}
//-----
//处理消息的窗口函数
LRESULT CALLBACK WndProc(HWND hwnd,
                          UINT message,
                          WPARAM wParam,
                          LPARAM lParam)
{
    switch(message)
    {
        case WM_LBUTTONDOWN: //鼠标左键按下消息
        {
            MessageBeep(0); //可以发出声音的 API 函数
        }
        break;
        case WM_DESTROY:
            PostQuitMessage(0);
    }
}
```

```

        break;
default:
    return DefWindowProc(hwnd,message,wParam,lParam);
}
return 0;
}
//-----

```

(8) 按下 Ctrl+F7 键编译这个程序。

(9) 如果程序编译通过,则按下 Ctrl+F5 功能键运行这个应用程序。

(10) 在教师的指导下排除程序中的错误。

(11) 按下 F11 功能键单步运行这个程序。

(12) 参照上述步骤创建一个 A simple Win32 application 类型的 Win32 工程,在分析了自动生成的代码后,对程序进行修改使它能实现前面工程一样的功能。

(13) 参照上述步骤创建一个 A typical “Hello World!” application 类型的 Win32 工程,在分析了自动生成的代码后,对程序进行修改使它能实现前面工程一样的功能。

(14) 在代码编辑窗口,把光标放在代码行

```

WNDCLASS wc;                                //定义一个窗口类变量

```

的 WNDCLASS 位置上按下 F1 功能键看看会出现什么情况。

第 2 章

CHAPTER

Windows 应用程序的类封装

习题解答及上机实验

2.1 习题解答

2-1 在窗体类 CFrameWnd 中需要封装哪些成员？

答：在窗体类 CFrameWnd 中要封装窗口句柄、窗口类的定义、注册窗口类、创建窗口、显示更新窗口。

2-2 应用程序类 CWinApp 应该具备哪些主要功能？

答：创建、显示应用程序的窗口和建立消息循环。

2-3 在 MFC 程序设计中，如果要建立拥有自己风格的主窗口，应该重写什么函数。

答：继承 CWinApp 类并需要重写该类的成员函数 InitInstance。

2-4 什么是消息映射表？

答：在 Windows SDK 应用程序的窗口函数中，是采用 switch-case 分支结构实现消息处理的，这种方式不适合面向对象设计的要求。因此 MFC 建立了一套自己的消息映射机制——消息映射表。从外观来看，这种表有些类似学校中使用的学生名册，学号相当于消息号，学生姓名就相当于消息处理函数名，学号和学生名一一对应（映射）。而 MFC 把实现表的代码用宏封装起来了。

2.2 上机实验

实验内容：

Windows 应用程序的类封装。

实验目的：

1. 理解 Windows 应用程序的类封装过程，体会类封装给应用程序设计带来的好处。
2. 在创建 Win32 工程中使用 MFC 类库。

实验步骤：

- (1) 选择菜单 File|New 命令打开 New 对话框。
- (2) 在 New 对话框的 Projects 选项卡中选择 Win32 Application 选项创建一个 Win32 Application 工程。
- (3) 在 Win32 Application-Step 1 of 1 对话框中选择创建空工程选项 An empty project。
- (4) 选择菜单 File|New 打开 New 对话框并选择 Files 选项卡,在选项卡中选择 C++ Source File 选项,创建源文件。
- (5) 选择菜单 Project|Settings 打开如图 2-1 所示的 Project Settings 对话框。

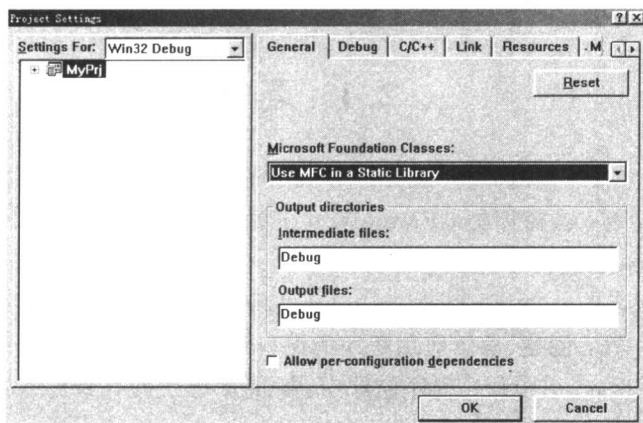


图 2-1 Project Settings 对话框

- (6) 在 Project Settings 对话框的 General 选项卡中的 Microsoft Foundation Classes 下拉列表框中选择 Use MFC in a Static Library 选项。其余使用默认选项,然后单击 OK 按钮。这样在创建的应用程序中就可以使用 MFC 类库了。
- (7) 在源文件中输入教材第 2 章的例 2-5 的源代码。

```
//需要包含的头文件-----
#include<afxwin.h>
//由 CFrameWnd 派生的 CMyWnd 类-----
class CMyWnd:public CFrameWnd
{
private:
    char * ShowText;                                //声明一个字符串为数据成员
public:
    afx_msg void OnPaint();                          //声明 WM_PAINT 消息处理函数
    afx_msg void OnLButtonDown();                    //鼠标左键按下消息处理函数
    DECLARE_MESSAGE_MAP()                            //声明消息映射
};
//消息映射的实现-----
BEGIN_MESSAGE_MAP(CMyWnd,CFrameWnd)
```