南京航空航天大学金城学院

《C++语言程序设计》课程设计报告

计算器（Qt、MFC、Python、Java 版本）
打地鼠游戏（Qt 版本）
学生信息管理系统（Qt 版本）
图书馆信息管理系统

学号：2017024416　　姓名：**鲍禹辰（组长）**

学号：2017024234　　姓名：叶洵葳

学号：2017024412　　姓名：陈昱良

学号：2017024229　　姓名：徐天宇

日期：2018.9.4

注：本次课设选题完成数量较多，超额完成任务。故我组选取 MFC 版本计算器与 Qt 版本学生管理系统进行验收答辩。其他几份设计在文档尾部给出源码参考。

MFC:计算器

目　　录

一、需求分析

为方便商场对货品信息的管理的自动化，特设计此系统。由于 c++程序的可移植性和可维护性较强，且数据比较安全，所以采用 c++进行设计。

二、程序的主要功能

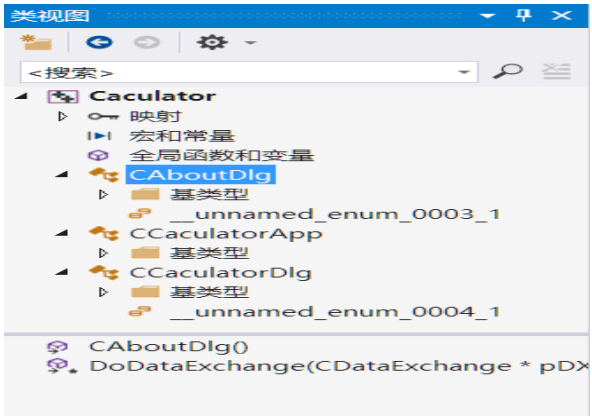
系统的主要功能有：
① 窗口界面的计算器；
② 通过按钮键输入数据；
③ 能够完成加、减、乘、除、求倒数等一般运算；
④ 输出结果具有一定精度。

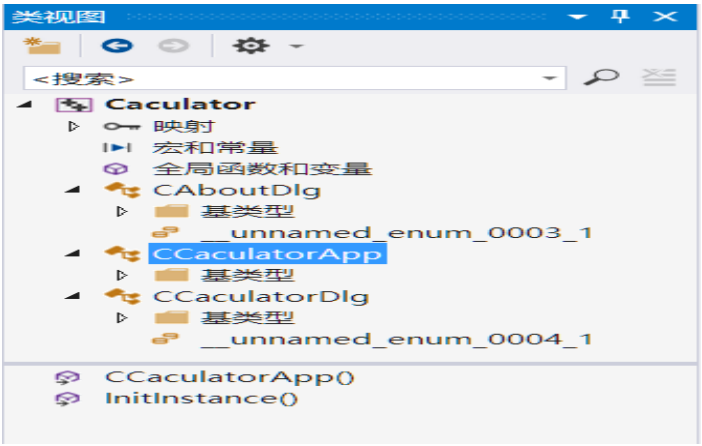
三、程序运行平台

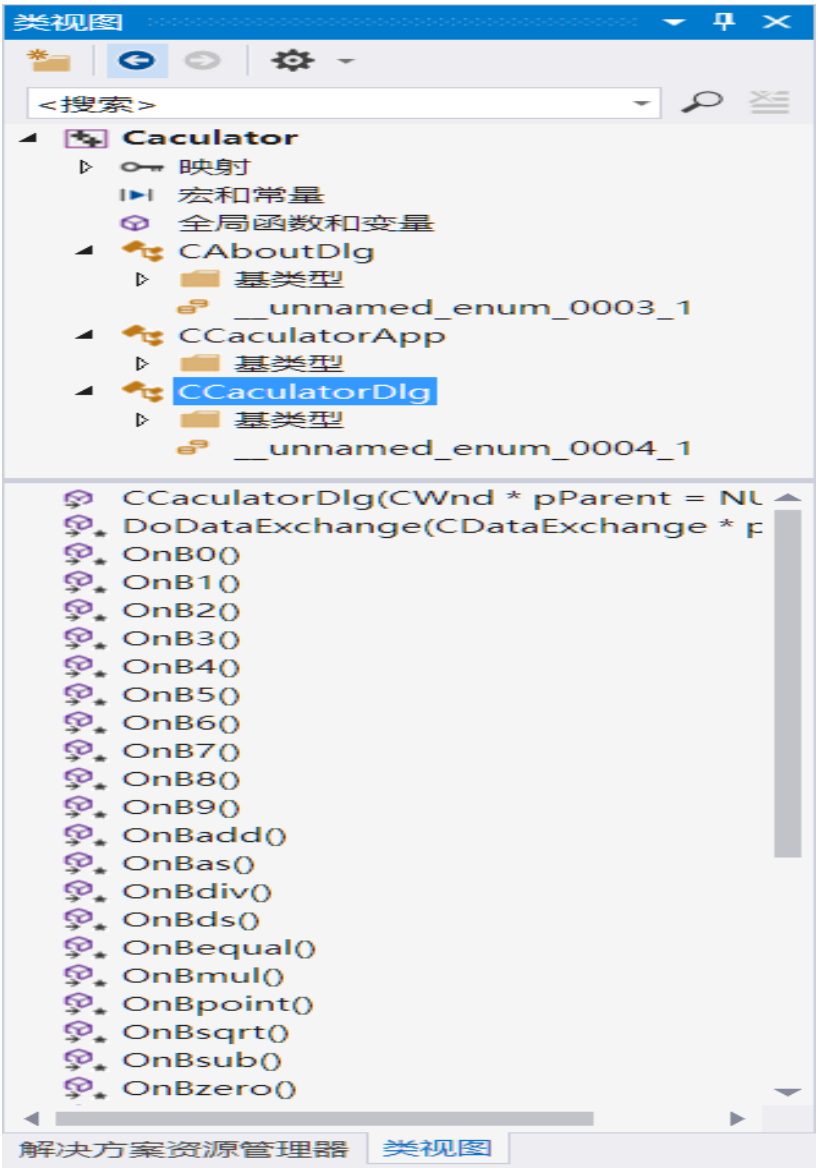
microsoft visual studio 2017
vc++6.0

四、程序类的说明

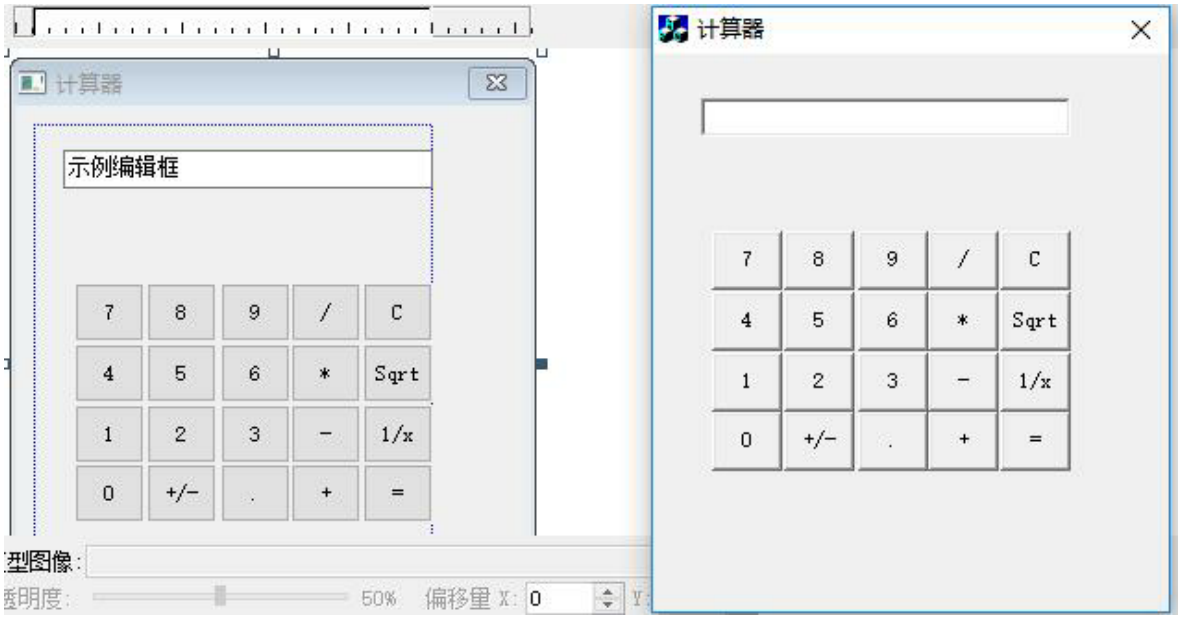
使用 MFC 进行开发，用类封装
下面是程序主要的类
因为 MFC 的开发习惯，使用了大量的类


类视图
<搜索>
Caculator
映射
宏和常量
全局函数和变量
CAboutDlg
基类型
__unnamed_enum_0003_1
CCaculatorApp
基类型
CCaculatorDlg
基类型
__unnamed_enum_0004_1
CAboutDlg()
DoDataExchange(CDataExchange * pDX


类视图
<搜索>
Caculator
映射
宏和常量
全局函数和变量
CAboutDlg
基类型
__unnamed_enum_0003_1
CCaculatorApp
基类型
CCaculatorDlg
基类型
__unnamed_enum_0004_1
CCaculatorApp()
InitInstance()


类视图
<搜索>
Caculator
映射
宏和常量
全局函数和变量
CAboutDlg
基类型
__unnamed_enum_0003_1
CCaculatorApp
基类型
CCaculatorDlg
基类型
__unnamed_enum_0004_1
CCaculatorDlg(CWnd * pParent = NU
DoDataExchange(CDataExchange * p
OnB0()
OnB1()
OnB2()
OnB3()
OnB4()
OnB5()
OnB6()
OnB7()
OnB8()
OnB9()
OnBadd()
OnBas()
OnBdiv()
OnBds()
OnBequal()
OnBmul()
OnBpoint()
OnBsqrt()
OnBsub()
OnBzero()
解决方案资源管理器
类视图

五、模块分析



程序有完整的加减乘除，支持开根号和倒数
有一定的高精度，在一定程度上模仿现实生活里的计算器，体现了拟物性

六、存在的不足与对策、编程体会

界面简单，代码重复性过多
改进，增加界面功能，加入函数模块，减少代码复用

七、程序主要控件源代码

```
// CaculatorDlg.cpp : implementation file
//

#include "stdafx.h"
#include "Caculator.h"
#include "CaculatorDlg.h"


#ifdef _DEBUG
#define new DEBUG_NEW
#undef THIS_FILE
static char THIS_FILE[] = __FILE__;
#endif

/////////////////////////////////////////////////////////////////////////////
// CAboutDlg dialog used for App About

class CAboutDlg : public CDialog
{
public:
	CAboutDlg();

// Dialog Data
	//{{AFX_DATA(CAboutDlg)
	enum { IDD = IDD_ABOUTBOX };
	//}}AFX_DATA

	// ClassWizard generated virtual function overrides
```

```
	//{{AFX_VIRTUAL(CAboutDlg)
	protected:
	virtual void DoDataExchange(CDataExchange* pDX);    // DDX/DDV support
	//}}AFX_VIRTUAL

// Implementation
protected:
	DECLARE_MESSAGE_MAP()
};

CAboutDlg::CAboutDlg() : CDialog(CAboutDlg::IDD)
{

}

void CAboutDlg::DoDataExchange(CDataExchange* pDX)
{
	CDialog::DoDataExchange(pDX);
}

BEGIN_MESSAGE_MAP(CAboutDlg, CDialog)
END_MESSAGE_MAP()


CCaculatorDlg::CCaculatorDlg(CWnd* pParent)
	: CDialog(CCaculatorDlg::IDD, pParent)
{
	m_result = _T("");
	m_hIcon = AfxGetApp()->LoadIcon(IDR_MAINFRAME);
}

void CCaculatorDlg::DoDataExchange(CDataExchange* pDX)
{
	CDialog::DoDataExchange(pDX);
	DDX_Text(pDX, IDC_RESULT, m_result);
}

BEGIN_MESSAGE_MAP(CCaculatorDlg, CDialog)
	ON_WM_SYSCOMMAND()
	ON_WM_PAINT()
	ON_WM_QUERYDRAGICON()
	ON_BN_CLICKED(IDC_B0, OnB0)
	ON_BN_CLICKED(IDC_B1, OnB1)
	ON_BN_CLICKED(IDC_B2, OnB2)
	ON_BN_CLICKED(IDC_B3, OnB3)
	ON_BN_CLICKED(IDC_B4, OnB4)
	ON_BN_CLICKED(IDC_B5, OnB5)
	ON_BN_CLICKED(IDC_B6, OnB6)
	ON_BN_CLICKED(IDC_B7, OnB7)
	ON_BN_CLICKED(IDC_B8, OnB8)
	ON_BN_CLICKED(IDC_B9, OnB9)
	ON_BN_CLICKED(IDC_BADD, OnBadd)
	ON_BN_CLICKED(IDC_BDIV, OnBdiv)
	ON_BN_CLICKED(IDC_BEQUAL, OnBequal)
	ON_BN_CLICKED(IDC_BMUL, OnBmul)
	ON_BN_CLICKED(IDC_BSUB, OnBsub)
	ON_BN_CLICKED(IDC_BZERO, OnBzero)
	ON_BN_CLICKED(IDC_BSQRT, OnBsqrt)
	ON_BN_CLICKED(IDC_BAS, OnBas)
	ON_BN_CLICKED(IDC_BDS, OnBds)
	ON_BN_CLICKED(IDC_BPOINT, OnBpoint)
END_MESSAGE_MAP()

BOOL CCaculatorDlg::OnInitDialog()
{
	CDialog::OnInitDialog();
	ASSERT((IDM_ABOUTBOX & 0xFFF0) == IDM_ABOUTBOX);
```

```
	ASSERT(IDM_ABOUTBOX < 0xF000);

	CMenu* pSysMenu = GetSystemMenu(FALSE);
	if (pSysMenu != NULL)
	{
		CString strAboutMenu;
		strAboutMenu.LoadString(IDS_ABOUTBOX);
		if (!strAboutMenu.IsEmpty())
		{
			pSysMenu->AppendMenu(MF_SEPARATOR);
			pSysMenu->AppendMenu(MF_STRING, IDM_ABOUTBOX, strAboutMenu);
		}
	}

	SetIcon(m_hIcon, TRUE);			// Set big icon
	SetIcon(m_hIcon, FALSE);		// Set small icon
	num1="";
	num2="";
	fh="";
	ischar=FALSE;//初始化时设置为 FALSE;
	sw=0;
	whichx=0;
	GetWindowRect(&rectsci);
	rectstan=rectsci;
	CRect r;
	GetDlgItem(IDC_RESULT)->GetWindowRect(&r);
	rectstan.right=r.right+10;
	return TRUE;  // return TRUE  unless you set the focus to a control

}

void CCaculatorDlg::OnSysCommand(UINT nID, LPARAM lParam)
{
	if ((nID & 0xFFF0) == IDM_ABOUTBOX)
	{
		CAboutDlg dlgAbout;
		dlgAbout.DoModal();
	}
	else
	{
		CDialog::OnSysCommand(nID, lParam);
	}
}

void CCaculatorDlg::OnPaint()
{
	if (IsIconic())
	{
		CPaintDC dc(this); // device context for painting

		SendMessage(WM_ICONERASEBKGND, (WPARAM) dc.GetSafeHdc(), 0);

		// Center icon in client rectangle
		int cxIcon = GetSystemMetrics(SM_CXICON);
		int cyIcon = GetSystemMetrics(SM_CYICON);
		CRect rect;
		GetClientRect(&rect);
		int x = (rect.Width() - cxIcon + 1) / 2;
		int y = (rect.Height() - cyIcon + 1) / 2;

		// Draw the icon
		dc.DrawIcon(x, y, m_hIcon);
	}
	else
	{
		CDialog::OnPaint();
```

```
	}
}
HCURSOR CCaculatorDlg::OnQueryDragIcon()
{
	return (HCURSOR) m_hIcon;
}

void CCaculatorDlg::OnB0()
{
	if(ischar==FALSE)
	{
		num1=num1+"0";
		m_result=num1;
		UpdateData(FALSE);
	}
	if(ischar==TRUE)
	{
		num2=num2+"0";
	   GetDlgItem(IDC_RESULT)->GetWindowText(m_result);
		m_result=m_result+"0";
		UpdateData(FALSE);
	}

}

void CCaculatorDlg::OnB1()
{
	if(ischar==FALSE)
	{
		num1=num1+"1";
		m_result=num1;
		UpdateData(FALSE);
	}
   if(ischar==TRUE)
	{
		num2=num2+"1";
	   GetDlgItem(IDC_RESULT)->GetWindowText(m_result);
		m_result=m_result+"1";
		UpdateData(FALSE);
	}
}

void CCaculatorDlg::OnB2()
{
	if(ischar==FALSE)
	{
		num1=num1+"2";
		m_result=num1;
		UpdateData(FALSE);
	}
	if(ischar==TRUE)
	{
		GetDlgItem(IDC_RESULT)->GetWindowText(m_result);
		num2=num2+"2";
		m_result=m_result+"2";
		UpdateData(FALSE);
	}
}

void CCaculatorDlg::OnB3()
{
	if(ischar==FALSE)
	{
		num1=num1+"3";
		m_result=num1;
		UpdateData(FALSE);
	}
```

```
	if(ischar==TRUE)
	{
		GetDlgItem(IDC_RESULT)->GetWindowText(m_result);
		num2=num2+"3";
		m_result=m_result+"3";
		UpdateData(FALSE);
	}
}

void CCaculatorDlg::OnB4()
{
	if(ischar==FALSE)
	{
		num1=num1+"4";
		m_result=num1;
		UpdateData(FALSE);
	}
	if(ischar==TRUE)
	{
		GetDlgItem(IDC_RESULT)->GetWindowText(m_result);
		num2=num2+"4";
		m_result=m_result+"4";
		UpdateData(FALSE);
	}
}

void CCaculatorDlg::OnB5()
{
	if(ischar==FALSE)
	{
		num1=num1+"5";
		m_result=num1;
		UpdateData(FALSE);
	}
	if(ischar==TRUE)
	{
		GetDlgItem(IDC_RESULT)->GetWindowText(m_result);
		num2=num2+"5";
		m_result=m_result+"5";
		UpdateData(FALSE);
	}
}

void CCaculatorDlg::OnB6()
{
	if(ischar==FALSE)
	{
		num1=num1+"6";
		m_result=num1;
		UpdateData(FALSE);
	}
	if(ischar==TRUE)
	{
		GetDlgItem(IDC_RESULT)->GetWindowText(m_result);
		num2=num2+"6";
		m_result=m_result+"6";
		UpdateData(FALSE);
	}
}

void CCaculatorDlg::OnB7()
{
	if(ischar==FALSE)
	{
		num1=num1+"7";
		m_result=num1;
		UpdateData(FALSE);
```

```
	}
	if(ischar==TRUE)
	{
		GetDlgItem(IDC_RESULT)->GetWindowText(m_result);
		num2=num2+"7";
		m_result=m_result+"7";
		UpdateData(FALSE);
	}
}

void CCaculatorDlg::OnB8()
{
	if(ischar==FALSE)
	{
		num1=num1+"8";
		m_result=num1;
		UpdateData(FALSE);
	}
	if(ischar==TRUE)
	{
		GetDlgItem(IDC_RESULT)->GetWindowText(m_result);
		num2=num2+"8";
		m_result=m_result+"8";
		UpdateData(FALSE);
	}
}

void CCaculatorDlg::OnB9()
{
	if(ischar==FALSE)
	{
		num1=num1+"9";
		m_result=num1;
		UpdateData(FALSE);
	}
	if(ischar==TRUE)
	{
		GetDlgItem(IDC_RESULT)->GetWindowText(m_result);
		num2=num2+"9";
		m_result=m_result+"9";
		UpdateData(FALSE);
	}
}

void CCaculatorDlg::OnBadd()
{
	GetDlgItem(IDC_RESULT)->GetWindowText(m_result);
	m_result=m_result+"+";
	UpdateData(FALSE);
	ischar=TRUE;
	sw=1;
}

void CCaculatorDlg::OnBdiv()
{
	GetDlgItem(IDC_RESULT)->GetWindowText(m_result);
	m_result=m_result+"/";
	UpdateData(FALSE);
	ischar=TRUE;
	sw=4;
}

void CCaculatorDlg::OnBequal()
{
	double number1=atof(num1);
	double number2=atof(num2);
	if(1)
```

```
	{
		double number1=atof(num1);
	   double number2=atof(num2);
		double result=0;
		switch(sw)
		{
		case 1:result=number1+number2;break;
	   case 2:result=number1-number2;break;
	   case 3:result=number1*number2;break;
	   case 4:result=number1/number2;break;
	   case 5:result=(double)sqrt(number1);break;
	   case 6:result=1/number1;break;
	   default:MessageBox("bfrheygh");break;
		}
	  m_result="";
	  num2="";
	  num1.Format("%lf",result);
	  ischar=TRUE;
	  m_result.Format("%g",result);
	  UpdateData(FALSE);
	}
}

void CCaculatorDlg::OnBmul()
{
	GetDlgItem(IDC_RESULT)->GetWindowText(m_result);
	m_result=m_result+"*";
	UpdateData(FALSE);
	ischar=TRUE;
	sw=3;
}

void CCaculatorDlg::OnBsub()
{
	GetDlgItem(IDC_RESULT)->GetWindowText(m_result);
	m_result=m_result+"-";
	UpdateData(FALSE);
	ischar=TRUE;
	sw=2;
}

void CCaculatorDlg::OnBzero()
{
	m_result="";
	ischar=FALSE;
	sw=0;
	num1="";
	num2="";
	UpdateData(FALSE);
}

void CCaculatorDlg::OnBsqrt()
{
	ischar=TRUE;
	sw=5;
}

void CCaculatorDlg::OnBas()
{
	CString as="-";

		m_result=as+num1;
		num1=as+num1;
		UpdateData(FALSE);
	if(ischar==TRUE)
	{
		GetDlgItem(IDC_RESULT)->GetWindowText(m_result);
```

```
            num2=num2+"9";
            m_result=m_result+"-";
            UpdateData(FALSE);
        }

}

void CCaculatorDlg::OnBds()
{
        GetDlgItem(IDC_RESULT)->GetWindowText(m_result);
        m_result=m_result+"1/x";//fh;
        UpdateData(FALSE);
        ischar=TRUE;
        sw=6;
}

void CCaculatorDlg::OnBpoint()
{
        if(ischar==FALSE)
        {
            num1=num1+".";
            m_result=num1;
            UpdateData(FALSE);
        }
        if(ischar==TRUE)
        {
            GetDlgItem(IDC_RESULT)->GetWindowText(m_result);
            num2=num2+".";
            m_result=m_result+".";
            UpdateData(FALSE);
        }
}


// Caculator.cpp : Defines the class behaviors for the application.
//

#include "stdafx.h"
#include "Caculator.h"
#include "CaculatorDlg.h"

#ifdef _DEBUG
#define new DEBUG_NEW
#undef THIS_FILE
static char THIS_FILE[] = __FILE__;
#endif

/////////////////////////////////////////////////////////////////////////////
// CCaculatorApp

BEGIN_MESSAGE_MAP(CCaculatorApp, CWinApp)
        //{{AFX_MSG_MAP(CCaculatorApp)
            // NOTE - the ClassWizard will add and remove mapping macros here.
            //    DO NOT EDIT what you see in these blocks of generated code!
        //}}AFX_MSG
        ON_COMMAND(ID_HELP, CWinApp::OnHelp)
END_MESSAGE_MAP()

/////////////////////////////////////////////////////////////////////////////
// CCaculatorApp construction

CCaculatorApp::CCaculatorApp()
{
        // TODO: add construction code here,
        // Place all significant initialization in InitInstance
}
```

```
/////////////////////////////////////////////////////////////////////////////
// The one and only CCaculatorApp object

CCaculatorApp theApp;

/////////////////////////////////////////////////////////////////////////////
// CCaculatorApp initialization

BOOL CCaculatorApp::InitInstance()
{
	AfxEnableControlContainer();

	// Standard initialization
	// If you are not using these features and wish to reduce the size
	//  of your final executable, you should remove from the following
	//  the specific initialization routines you do not need.

#ifdef _AFXDLL
	Enable3dControls();			// Call this when using MFC in a shared DLL
#else
	Enable3dControlsStatic();	// Call this when linking to MFC statically
#endif

	CCaculatorDlg dlg;
	m_pMainWnd = &dlg;
	int nResponse = dlg.DoModal();
	if (nResponse == IDOK)
	{
		// TODO: Place code here to handle when the dialog is
		//  dismissed with OK
	}
	else if (nResponse == IDCANCEL)
	{
		// TODO: Place code here to handle when the dialog is
		//  dismissed with Cancel
	}

	// Since the dialog has been closed, return FALSE so that we exit the
	//  application, rather than start the application's message pump.
	return FALSE;
}




// Caculator.cpp : Defines the class behaviors for the application.
//

#include "stdafx.h"
#include "Caculator.h"
#include "CaculatorDlg.h"

#ifdef _DEBUG
#define new DEBUG_NEW
#undef THIS_FILE
static char THIS_FILE[] = __FILE__;
#endif

/////////////////////////////////////////////////////////////////////////////
// CCaculatorApp

BEGIN_MESSAGE_MAP(CCaculatorApp, CWinApp)
	//{{AFX_MSG_MAP(CCaculatorApp)
		// NOTE - the ClassWizard will add and remove mapping macros here.
		//    DO NOT EDIT what you see in these blocks of generated code!
	//}}AFX_MSG
	ON_COMMAND(ID_HELP, CWinApp::OnHelp)
```

```
END_MESSAGE_MAP()

/////////////////////////////////////////////////////////////////////////////
// CCaculatorApp construction

CCaculatorApp::CCaculatorApp()
{
	// TODO: add construction code here,
	// Place all significant initialization in InitInstance
}

/////////////////////////////////////////////////////////////////////////////
// The one and only CCaculatorApp object

CCaculatorApp theApp;

/////////////////////////////////////////////////////////////////////////////
// CCaculatorApp initialization

BOOL CCaculatorApp::InitInstance()
{
	AfxEnableControlContainer();

	// Standard initialization
	// If you are not using these features and wish to reduce the size
	//  of your final executable, you should remove from the following
	//  the specific initialization routines you do not need.

#ifdef _AFXDLL
	Enable3dControls();			// Call this when using MFC in a shared DLL
#else
	Enable3dControlsStatic();	// Call this when linking to MFC statically
#endif

	CCaculatorDlg dlg;
	m_pMainWnd = &dlg;
	int nResponse = dlg.DoModal();
	if (nResponse == IDOK)
	{
		// TODO: Place code here to handle when the dialog is
		//  dismissed with OK
	}
	else if (nResponse == IDCANCEL)
	{
		// TODO: Place code here to handle when the dialog is
		//  dismissed with Cancel
	}

	// Since the dialog has been closed, return FALSE so that we exit the
	//  application, rather than start the application's message pump.
	return FALSE;
}
```

Qt 学生管理系统：

目　录

一：需求分析

学生信息管理系统是针对学校人事处的大量业务处理工作而开发的管理软件，主要用于学校学生信息管理，总体任务是实现学生信息关系的系统化、科学化、规范化和自动化，其主要任务是用计算机对学生各种信息进行日常管理，如查询、修改、增加、删除，另外还考虑到学生选课，针对这些要求设计了学生信息管理系统。学生信息档案的管理对于学校的管理者来说至关重要，学生信息是高等学校非常重要的一项数据资源，是一个教育单位不可缺少一部分。特别是近几年来，国家政策的调整，我国高等院校大规模的扩招，给高等院校的教学管理、学生管理、后勤管理等方面都带来不少的冲击。其包含的数据量大，涉及的人员面广，而且需要及时更新，故较为复杂，难以单纯地依靠人工管理，而且传统的人工管理方式既不易于规范化，管理效率也不高，目前我国各类高等院校中还有相当一部分学生档案管理还停留在纸介质的基础上，尤其是中、小学对学生档案的管理更是落后，这样的管理机制已经不能适应时代发展的要求，其管理方法将浪费许多人力和物力。随着科学技术的不断提高，计算机科学与技术日渐成熟，计算机应用的普及已进入人类社会生活的各个领域，并发挥着越来越重要的作用。这种传统的手工管理模式必然被以计算机为物质基础的信息管理方法所取代。

作为计算机应用的一部分，使用计算机对学生档案进行管理，有着手工管理所无法比拟的优点，如：检索迅速、查找方便、可靠性高、存储量大、保密性好、寿命长、成本低等。这些优点能够极大地提高学生档案管理的效率，也是学校向科学化、正规化管理发展的必要条件，更是各个高等院校与世界接轨的重要条件

二：程序的主要功能

2、添加功能

添加一个学生的信息，包括学生的姓名、学号、性别、院系、语文、数学、英语、总分

3、删除功能

删除之前再次确认。确认后只需提供该学生学号便可对该职员信息进行删除，如不确认则回到主界面供用户重新选择功能

4、排序显示功能

以学生的总分、语文、数学、英语成绩升序显示所有学生的详细信息

5、学生查询功能

只需提供该学号、姓名任一项均可跳过输入，便可显示该学生的详细信息

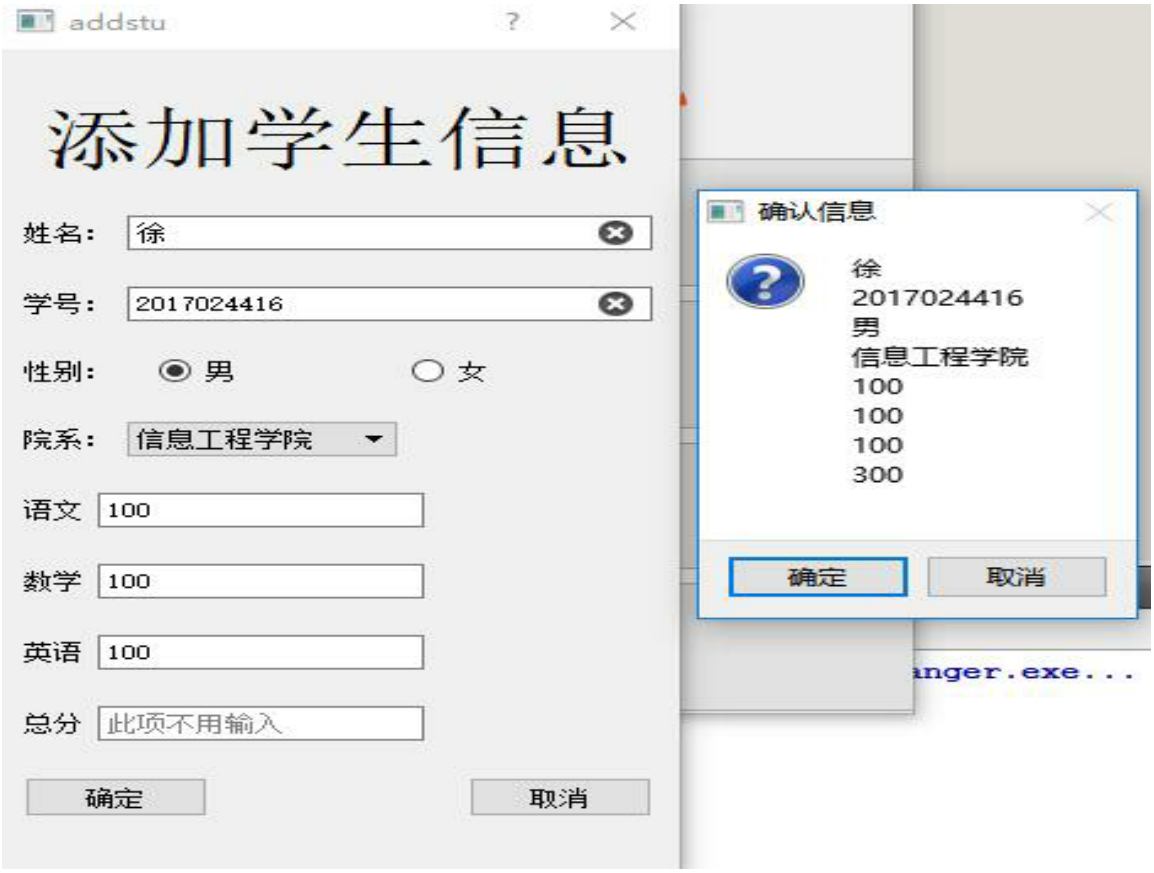
三：程序运行平台

Qt Creator

运行程序，进入登录界面如下：

1、点击添加学生信息模块

添加一个学生的信息，包括学生的姓名、学号、性别、院系、语文、数学、英语、总分

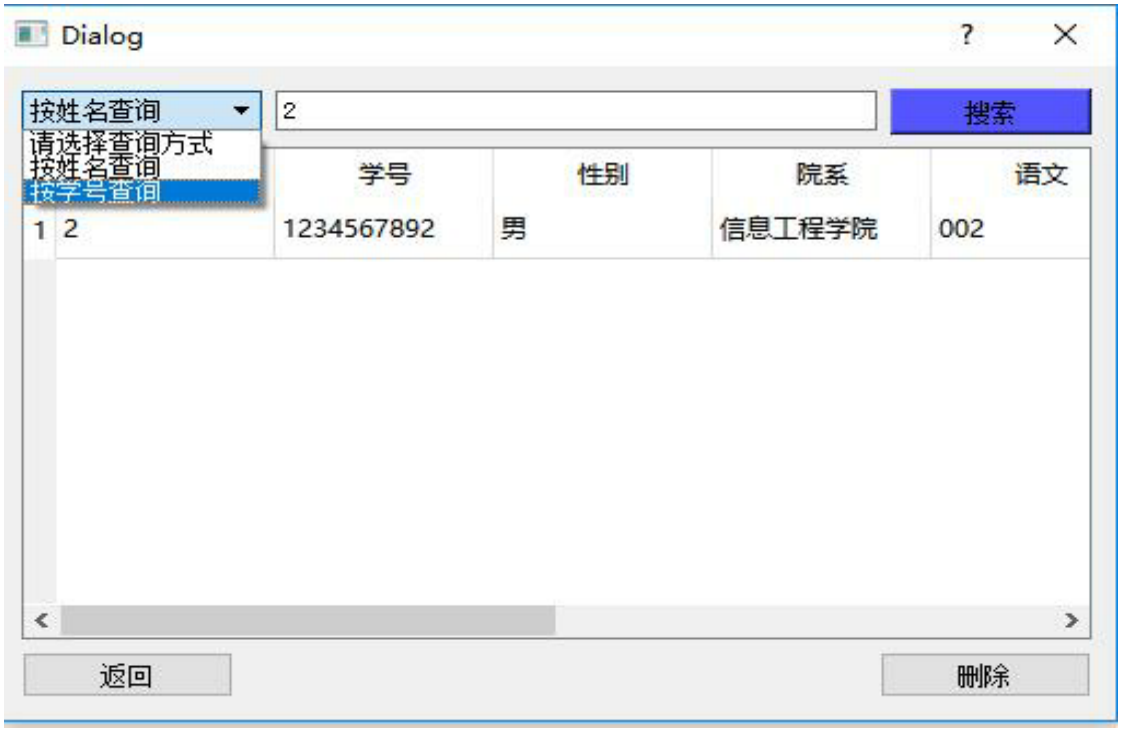


2、点击进入排序显示模块

以学生的总分、语文、数学、英语成绩升序显示所有学生的详细信息

3、点击进入查询信息模块

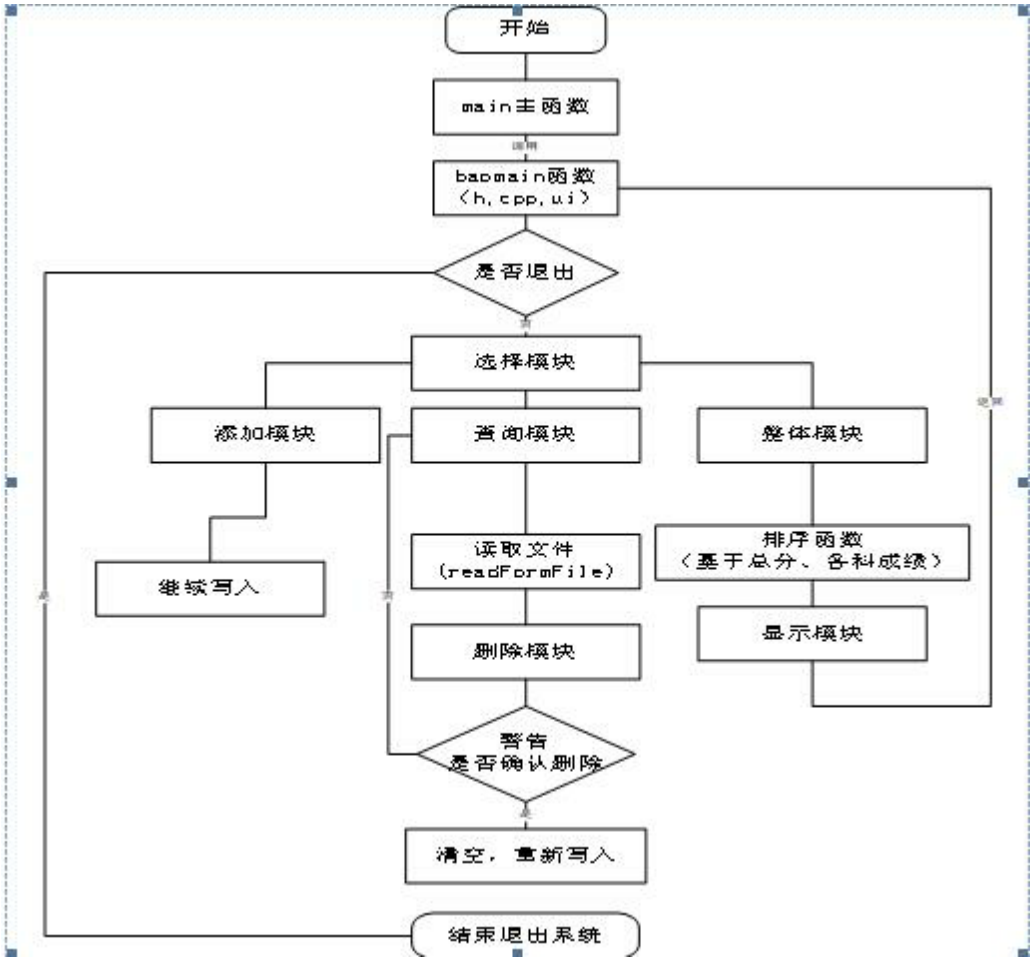
只需要提供：学号、姓名任一项均可跳过输入，便可显示该学生的详细信息



查询后可执行删除功能

删除之前再次确认，如不确认则回到主界面供用户重新选择模块

四、系统总框架图



程序类的说明

1) baomain 函数

baomain 函数被 main 主函数调用，然后进入主界面显示

2) readFormFile 函数

readFormFile 函数功能为读取磁盘文件中的学生信息

模块分析

1、添加学生信息模块

先判断磁盘是否已满。如果已满，提示已满并返回选择操作菜单。如未满，则请用户依次输入学生姓名、学号、性别、院系、语文、数学、英语、总分等各项信息（任一项可暂时空缺）。如添加成功，则提示成功，返回选择操作菜单；反之，直接返回选择操作菜单。

2、删除学生信息模块

删除前再请用户确认一次是否删除。如果不确认，返回选择操作菜单。如确认，请用户选择需要删除的学生，再执行删除操作。如删除成功，直接返回选择操作菜单。

3、查询学生信息模块

先判断信息表是否为空。如果为空，提示为空并返回选择操作菜单。如不为空，请用户输入学生信息（只需姓名、学号任一项）。如果未找到学生，提示未找到，返回选择操作菜单。如找到学生，在屏幕上显示该学生项的各项信息。

5、 排序显示职员信息模块先判断信息表是否为空。如果为空，提示为空并返回选择操作菜单。如果不为空，以学生的总分、语文、数学、英语各项（基于用户选择）作为依据进行排序，将排序后的结果显示在界面上。

比较有特色的函数：

1）baomain 函数

1、函数分析

baomain 函数被 main 主函数调用，然后进入主界面显示

2、函数源代码

2）readFormFile 函数

1、函数分析

readFormFile 函数功能为读取磁盘文件中的学生信息

2、函数源代码

存在的不足与编程体会：

1）存在的不足

还可以加入更高级的算法进行学生各科成绩信息的排序，同时也可以加入修改学生模块，增添更多的功能。

2）编程体会

1、多写通用公用的类、代码结构要易于维护，修改起来较容易

2、注意编码风格，变量、函数、常量的命名一目了然，方便日后维护

3、整个系统编码一定要有规范，调用函数的代码根据返回的状态做相应的处理，高度模块化，条理清晰，合理有序

4、要创建更友好的用户界面，让交互数据输入更方便，减少错误输入的影响。在编写交互式程序时，应该事先预料到用户可能会输入错误，然后设计程序处理用户的错误输入，在用户出错时提醒用户再次输入

5、要切身体会用户的使用环境和需求，增加功能实现，反复改进代码

6、debug 的过程很辛苦，要不断模拟用户可能的操作来调试测试程序。要有足够的耐心和心理调节能力，要耐住寂寞、忍受孤独

参考文献：

《C Primer Plus（Sixth Edition）》

——Stephen Prata

《Data Structure And Algorithm Analysis in C(Second Edition)》

——Mark Allen Weiss

《Parts1-4：Fundamental， Data Structure， Sorting， Searching（Third Edition）》

——Robert Sedgewick

《Qt Creator 快速入门》

——霍亚飞

程序源代码：

如下：项目源代码第一个：

项目源码部分：

学生管理系统：

添加学生：(addstu)

addstu.h

```
#ifndef ADDSTU_H
#define ADDSTU_H
#include <QDialog>
#include<QButtonGroup>
#include<QString>
```

```
#include"querystu.h"
namespace Ui {
class addstu;
}
class addstu : public QDialog
{
    Q_OBJECT

public:
    explicit addstu(QWidget *parent = 0);
    ~addstu();
    void clearUserInterface();
    void writeToFile(QString cnt);

private slots:
    void on_B_ok_clicked();

    void on_B_no_clicked();
private:
    Ui::addstu *ui;
    QButtonGroup *GenderGroup;
    QButtonGroup *StatusGroup;
};
#endif // ADDSTU_H
```

addstu.cpp

```
#include "addstu.h"
#include "ui_addstu.h"
#include<QMessageBox>       //提供一个小提示框
/*     eg QMessageBox msgBox;对象
       msgBox.setText("   ")
       msgBox.exce();
       显示文本
       msgBox.setText("大标文");
       msgBox.setInformativeText("小标文");
       msgBox.setStandardButton(QMessageBox::Save|QMessageBox::Discord|QMessageBox::Cannel);
       msgBox.setDefaultButton(QMessageBox::Save);//设置默认的按键
       int let=msgBox.exec();*/
#include<QAbstractButton>
#include<QPushButton>
#include<QString>
#include<QLabel>
#include<QButtonGroup>
#include<QFile>
#include<QTextStream>
#include<QIODevice>//这三个是文件处理的头文件；
#include"querystu.h"
#include<QStringList>
#include<QWidget>
#include<QtWidgets>
addstu::addstu(QWidget *parent) :
    QDialog(parent),
    ui(new Ui::addstu)
{
    ui->setupUi(this);
}

addstu::~addstu()
{
    delete ui;
}
```

```
void addstu::on_B_ok_clicked()
{
    QString name=this->ui->le_name->text();
    QString id=this->ui->le_id->text();
    QString sex;
    QString chinese=this->ui->le_chinese->text();
    if(chinese.length()==1)
        chinese="00"+chinese;
    else if(chinese.length()==2)
        chinese="0"+chinese;
    QString math=this->ui->le_math->text();
    if(math.length()==1)
        math="00"+math;
    else if(math.length()==2)
        math="0"+math;
    QString english=this->ui->le_english->text();
    if(english.length()==1)
        english="00"+english;
    else if(english.length()==2)
        english="0"+english;
    QString erro=this->ui->le_tot->text();
    GenderGroup = new QButtonGroup(this);
    GenderGroup->addButton(ui->rbtn_male,0);
    GenderGroup->addButton(ui->rbtn_fmale,1);
    double num1,num2,num3,num4;//将字符串转化成数值，double 型
    num1=chinese.toDouble();
    num2=math.toDouble();
    num3=english.toDouble();
    num4=num1+num2+num3;
    QString tot=QString::number(num4);//至此转化完毕,并
    if(ui->rbtn_male->isChecked())
    {
        sex="男";
    }
    else
    {
        sex="女";
    }
    QString dev=this->ui->cbb_yx->currentText();
    QString content=name+'\n'+id+'\n'+sex+'\n'+dev+'\n'+chinese+'\n'+math+'\n'+english+'\n'+tot+'\n';
    QString cnt=name+" "+id+" "+sex+" "+dev+" "+chinese+" "+math+" "+english+" "+tot+"\n";
    int ret;
    if(name.length()<1||id.length()!=10||erro!="")
    {
        QMessageBox::critical(this,"错误","信息填写不正确,请重新检查","确定");
    }
    else
    {
        ret = QMessageBox::question(this,"确认信息",content,"确定","取消");
    }
    if(ret==0)
    {
        clearUserInterface();
        writeToFile(cnt);
    }

}
void addstu::clearUserInterface()
{
    this->ui->le_name->clear();
    this->ui->le_id->clear();
    this->ui->rbtn_male->setChecked(true);
    this->ui->cbb_yx->setCurrentIndex(0);
```

```
    this->ui->le_chinese->clear();
    this->ui->le_math->clear();
    this->ui->le_english->clear();
    this->ui->le_tot->clear();
    this->ui->le_name->setFocus();
}
void addstu::writeToFile(QString cnt)
{
    QFile file("stu.txt");
    if(!file.open(QIODevice::Append|QIODevice::Text))
    {
        QMessageBox::critical(this,"错误","保存信息失败，请重新输入","确定");
        return;
    }
    QTextStream out(&file);//输入语句对应 file 的文件路径,stu.txt;
    out<<cnt;//这两句的作用是将文件的内容写入 stu.txt 中;
    file.close();

}
void addstu::on_B_no_clicked()
{
    this->close();
}
```

addstu.cpp 中 OK 组代码如下：

```
QString name=this->ui->name->text();
QString id=this->ui->le_id->text();
QMessageBox msgBox;
msgBox.setText("确认信息");
msgBox.setInformativeText(name+'\n'+id);
msgBox.setStandardButtons(QMessageBox::Save|QMessageBox::Cannel);
msgBox.setDefaultButton(QMessageBox::Save);


content=name+'\n'+id+'\n'+sex+'\n'+age+'\n'+ins;
if(name.length()<1||id.length!=10)
{
    QMessageBox::critical;
    int ret=msgBox.exec();
}
if(ret==0)
{
    clearUserInterface();
    writeToFile(cnt);
}
```

设置按键选项成为中文：

```
msgBox.addButton("确定",QMessageBox::AcceptRole);
msgBox.addButton("取消",QMessageBox::RejectRole);
//这两行最后的选项是属于类型选项
```

设置默认的自定义按钮：

```
QPushButton 在#include<QPushButton>中
QMessageBox::information(this,"确认",name+'\n'+id,QMessageBox::OK|QMessageBox::Cannel);
"确认"，"取消"可改为中文的按钮
QMessageBox::critical,warning,question 与上格式一样。
clearButton;//清除按钮可用
```

comboBOx 的使用：

```
QString sex=this->ui->sexGroup->checkedButton()->text();
//作用获取单选框的文本内容
```

```
QList<QAbstractButton*>ins_list;
this->ui->insGroup->button();
//获取下拉列表的内容

this->ui->le_name->clear();
this->ui->le_id->clear();
//实现学号与姓名的清空
this->ui->rbtn_male->setChecked(true);
//使性别选项恢复成男性
this->cbb_age->setCurrentText("17");
//将年龄默认为 17 岁，但这样做不太好。(语言不同问题);

this->ui->cbb_age->setCurrentIndex(0);
this->ui->cbb_yx->setCurrentIndex(0);
//恢复默认的选项，即第一个选项

将起点设置位置：
this->ui->name->setFocus();


void addstu::writeToFile(QString cnt)
{
    QFile file("stu.txt");
    if(!file.open(QIODevice::Append|QIODevice::Text))
        //如果打开不存在自动创建
        //以文本的格式写入
        {
            QMessageBox::critical(=====);
        }
    QTextStream out(&file);
    out<<cnt;
}


baomain.h
#ifndef BAOMAIN_H
#define BAOMAIN_H

#include <QMainWindow>
#include"addstu.h"
#include"querystu.h"
#include"stuall.h"
namespace Ui {
class baomain;
}

class baomain : public QMainWindow
{
  Q_OBJECT

public:
  explicit baomain(QWidget *parent = 0);
  ~baomain();

private slots:
  void on_rbt_add_clicked();

  void on_rbt_look_clicked();

  void on_rbt_tuichu_clicked();

  void on_rbt_all_clicked();
```

```
private:
    Ui::baomain *ui;
    addstu a;
    QueryStu b;
    stuall bao;

};

#endif // BAOMAIN_H
```

baomain.cpp

```
#include "baomain.h"
#include "ui_baomain.h"
#include"addstu.h"
#include"querystu.h"
baomain::baomain(QWidget *parent) :
    QMainWindow(parent),
    ui(new Ui::baomain)
{
    ui->setupUi(this);
}

baomain::~baomain()
{
    delete ui;
}

void baomain::on_rbt_add_clicked()
{
   //addstu w;
  // w.exec();//模态视图造成阻塞
    this->a.show();
}

void baomain::on_rbt_look_clicked()
{
    //this->b.readFormfile();
    this->b.show();

}

void baomain::on_rbt_tuichu_clicked()
{
    this->close();
}

void baomain::on_rbt_all_clicked()
{
    this->b.readFormfile();
    this->bao.show();
}
```

querystu.h

```
#ifndef QUERYSTU_H
#define QUERYSTU_H
#include <QDialog>
#include<QString>
#include<QFile>
#include<QList>
#include<QStandardItem>
#include<QStandardItemModel>
```

```
#include"addstu.h"
namespace Ui {
class QueryStu;
}

class QueryStu : public QDialog
{
    Q_OBJECT

public:
    explicit QueryStu(QWidget *parent = 0);
    ~QueryStu();
    int readFormfile();
    void doQuery(int index,QString cnt);
    void display(int row,QStringList subs);
    QList<QString>stu_lines;
private slots:
    void on_pb_bao_clicked();
    void on_rbt_back_clicked();
    void on_rbt_shanchu_clicked();
private:
    Ui::QueryStu *ui;
    QStandardItemModel *model;

};

#endif // QUERYSTU_H


querystu.cpp 中
#include "querystu.h"
#include "ui_querystu.h"
#include<QIODevice>
#include<QTextStream>
#include<QMessageBox>
#include<QDebug>
#include<QStringList>
#include<QFile>
int shu;
QString line1;
QueryStu::QueryStu(QWidget *parent) :
    QDialog(parent),
    ui(new Ui::QueryStu)
{
    ui->setupUi(this);
    if(readFormfile()==-1)
    {
        QMessageBox::critical(this,"严重错误","文件打开失败,请重试","确定");
        this->close();
    }

    this->model=new QStandardItemModel;
    this->model->setHorizontalHeaderItem(0, new QStandardItem(QObject::tr("姓名")));
    this->model->setHorizontalHeaderItem(1, new QStandardItem(QObject::tr("学号")));
    this->model->setHorizontalHeaderItem(2, new QStandardItem(QObject::tr("性别")));
    this->model->setHorizontalHeaderItem(3, new QStandardItem(QObject::tr("院系")));
    this->model->setHorizontalHeaderItem(4, new QStandardItem(QObject::tr("语文")));
    this->model->setHorizontalHeaderItem(5, new QStandardItem(QObject::tr("数学")));
    this->model->setHorizontalHeaderItem(6, new QStandardItem(QObject::tr("英语")));
    this->model->setHorizontalHeaderItem(7, new QStandardItem(QObject::tr("总分")));
    this->ui->tableView->setModel(model);
}
```

```
QueryStu::~QueryStu()
{
    delete ui;
}
int QueryStu::readFormfile()
{
    QFile file("stu.txt");
    if(!file.open(QIODevice::ReadOnly|QIODevice::Text))
    {
        return -1;
    }
    QTextStream in(&file);
    while(!in.atEnd())
    {
        QString line=in.readLine();
        stu_lines.append(line);
    }
    qDebug()<<stu_lines.length();
    file.close();
}

void QueryStu::on_pb_bao_clicked()
{
    int index=this->ui->cbb_method->currentIndex();
    QString cnt=this->ui->le_cnt->text();
    doQuery(index,cnt);
}
void QueryStu::doQuery(int index, QString cnt)
{
    int i=0;
    int row=0;
    for(i=0;i<stu_lines.length();i++)
    {
        QString line=stu_lines.at(i);
        //line=line.trimmed();//如果字符串结尾是空格,则自动去除;
        QStringList subs=line.split(" ");//以" "作为分格，将一行字符串进行拆分;
        switch(index)
        {
            case 1:
            if(cnt==subs.at(0))
            {
                shu=i;
                display(row,subs);

            }
                break;
            case 2:
            if(cnt==subs.at(1))
            {
                shu=i;

                display(row,subs);

            }
                break;

        }
    }
}
void QueryStu::display(int row, QStringList subs)
{
    int i=0;
```

```
    for(i=0;i<subs.length();i++)
    {
        this->model->setItem(row,i,new QStandardItem(subs.at(i)));
    }
}

void QueryStu::on_rbt_back_clicked()
{
    this->close();
}

void QueryStu::on_rbt_shanchu_clicked()
{

    QFile file("stu.txt");
    if(!file.open(QIODevice::Append|QIODevice::Truncate|QIODevice::Text))
    {
        QMessageBox::critical(this,"错误","保存信息失败，请重新输入","确定");
        return;
    }
    file.resize(0);
    QTextStream out(&file);//输入语句对应 file 的文件路径,stu.txt
    stu_lines.removeAt(shu);
    for(int i=0;i<stu_lines.length();i++)
    {
        out<<stu_lines.at(i)<<endl;
    }
    file.close();
    QMessageBox::question(this,"删除","删除成功","确定");
    file.close();
    stu_lines.clear();
    readFormfile();

}

stuall.h 中
#ifndef STUALL_H
#define STUALL_H
#include <QDialog>
#include"addstu.h"
#include"querystu.h"
#include<QMessageBox>
#include<QAbstractButton>
#include<QPushButton>
#include<QString>
#include<QLabel>
#include<QButtonGroup>
#include<QFile>
#include<QTextStream>
#include<QIODevice>
#include<QStringList>
namespace Ui {
class stuall;
}

class stuall : public QDialog
{
    Q_OBJECT

public:
    explicit stuall(QWidget *parent = 0);
    ~stuall();
    void stuallpaixu(int baoyuchen);
```

```
private slots:
    void on_rbt_tot_clicked();

    void on_rbt_chinese_clicked();

    void on_rbt_math_clicked();

    void on_rbt_english_clicked();

    void on_rbt_return_clicked();

    void on_lowstu_clicked();

private:
    Ui::stuall *ui;
    QStandardItemModel *model;
    QList<QString>stu_all;
    QList<QString>stu_allstu;
    addstu a;
    QueryStu b;
};

#endif // STUALL_H
```

stuall.cpp 中

```
#include "stuall.h"
#include "ui_stuall.h"
#include<QMessageBox>
#include<QAbstractButton>
#include<QPushButton>
#include<QString>
#include<QLabel>
#include<QButtonGroup>
#include<QFile>
#include<QTextStream>
#include<QIODevice>//这三个是文件处理的头文件；
#include"querystu.h"
#include<QStringList>
QString yexunwei;
stuall::stuall(QWidget *parent) :
    QDialog(parent),
    ui(new Ui::stuall)
{
    ui->setupUi(this);
    this->model=new QStandardItemModel;
    this->model->setHorizontalHeaderItem(0, new QStandardItem(QObject::tr("姓名")));
    this->model->setHorizontalHeaderItem(1, new QStandardItem(QObject::tr("学号")));
    this->model->setHorizontalHeaderItem(2, new QStandardItem(QObject::tr("性别")));
    this->model->setHorizontalHeaderItem(3, new QStandardItem(QObject::tr("院系")));
    this->model->setHorizontalHeaderItem(4, new QStandardItem(QObject::tr("语文")));
    this->model->setHorizontalHeaderItem(5, new QStandardItem(QObject::tr("数学")));
    this->model->setHorizontalHeaderItem(6, new QStandardItem(QObject::tr("英语")));
    this->model->setHorizontalHeaderItem(7, new QStandardItem(QObject::tr("总分")));
    this->ui->tableView->setModel(model);

}
stuall::~stuall()
{
    delete ui;
}
```

```
void stuall::on_rbt_tot_clicked()
{
    stuallpaixu(7);
}
void stuall::stuallpaixu(int baoyuchen)
{
    int digit=0;
    QFile file("stu.txt");
    if(!file.open(QIODevice::ReadOnly|QIODevice::Text))
    {
        return;
    }
    QTextStream in(&file);
    while(!in.atEnd())
    {
        QString line1=in.readLine();
        stu_all.append(line1);
        digit++;
    }
    file.close();
    int i=0;
    int row=0;
    for(i=0;i<digit;i++)
    {
        QString line=stu_all.at(i);
        QStringList subs=line.split(" ");
        int j=0;
        for(j=0;j<subs.length();j++)
        {
            this->model->setItem(row,j,new QStandardItem(subs.at(j)));
        }
        row++;
    }
    this->ui->tableView->sortByColumn(baoyuchen,Qt::DescendingOrder);
}

void stuall::on_rbt_chinese_clicked()
{
    stuallpaixu(4);
}

void stuall::on_rbt_math_clicked()
{
    stuallpaixu(5);
}

void stuall::on_rbt_english_clicked()
{
    stuallpaixu(6);
}

void stuall::on_rbt_return_clicked()
{
    this->close();
}

void stuall::on_lowstu_clicked()
{
    int digit=0;
    QFile file("stu.txt");
    if(!file.open(QIODevice::ReadOnly|QIODevice::Text))
    {
        return;
```

```
    }
    QTextStream in(&file);
    while(!in.atEnd())
    {
        QString line1=in.readLine();
       // yexunwei=line1
        QStringList subs=line1.split(" ");
        if(subs.at(4).toDouble()<60||subs.at(5).toDouble()<60||subs.at(6).toDouble()<60)
        {
            stu_allstu.append(line1);
            digit++;
        }
    }
    file.close();
    int i=0;
    int row=0;
    for(i=0;i<digit;i++)
    {
        QString line=stu_allstu.at(i);
        QStringList subs1=line.split(" ");
        int j=0;
        for(j=0;j<subs1.length();j++)
        {
            this->model->setItem(row,j,new QStandardItem(subs1.at(j)));
        }
        row++;
    }
    this->ui->tableView->sortByColumn(7,Qt::DescendingOrder);
}
```

main.cpp 中

```
#include "addstu.h"
#include <QApplication>
#include"baomain.h"
int main(int argc, char *argv[])
{
    QApplication a(argc, argv);
    //addstu w;
   // w.show();
    baomain w;
w.show();
   return a.exec();
}
```

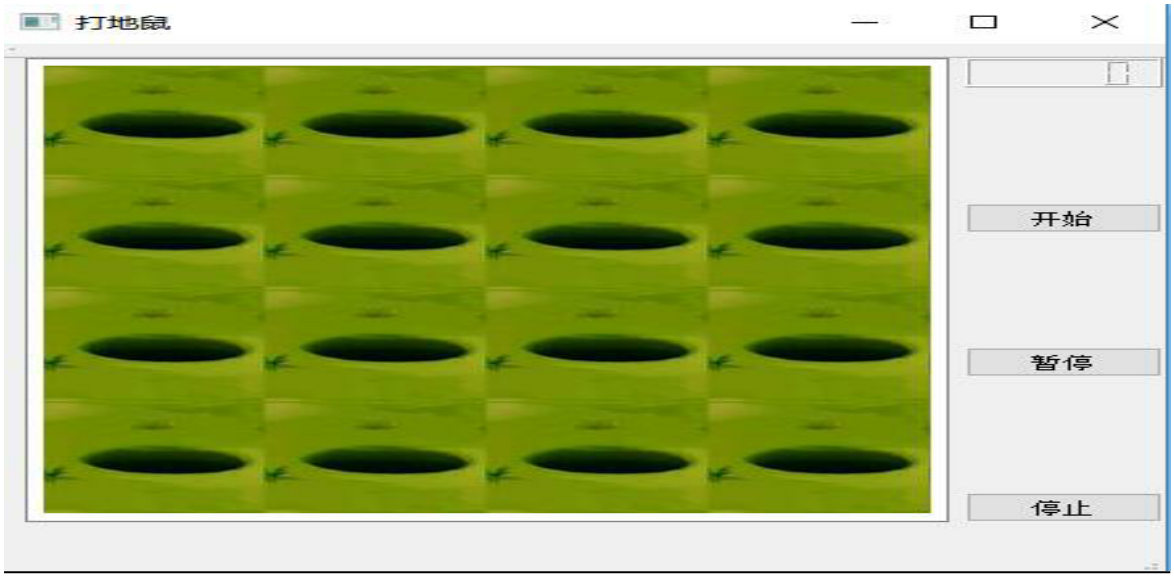
QT 中 QString 类与字符串之间的相互转化

```
QString str="1234";
float d=str.toFloat();
double d=str.toDouble();
int d=str.toInt();
float a=12.34;
```

数字转化为字符串：

```
QString str=QString::number(a);

#include<QTextStream>
while(!in.atEnd())//判断是否读取完文件。
{
        QString line=in.readLine();
        stu_lines.append(line);
}
file.close();
```

打地鼠：打地鼠：



handler.h

```
#ifndef HANDLER_H
#define HANDLER_H

#include <QObject>

class handler : public QObject
{
    Q_OBJECT
private:
    explicit handler(QObject *parent = 0);
    ~handler();

signals:
    void beatMouse();//定义了一个信号；
    //定义一个信号只需要写一个函数声明就可以
public slots:
private:
    static handler *hand;
    //返回给类外的程序使用
public:
```

```
    static handler *getInstance();
    //通过调用这个函数获取定义的静态变量。
    void addScore();//实现加分功能。
};

#endif // HANDLER_H
```

handler.cpp 中代码

```
#include "handler.h"

handler::handler(QObject *parent) : QObject(parent)
{

}

handler::~handler()
{

}
handler *handler::hand=new handler;
handler *handler::getInstance()
{
    return hand;
}
//单一的获取方式，单例问题，具有全局变量的特性。
void handler::addScore()
{
    //发送自定义的信号；
   emit beatMouse();
    //当函数被调用的时候通过 emit 发送信号
}
```

mainwindow.h

```
#ifndef MAINWINDOW_H
#define MAINWINDOW_H

#include <QMainWindow>
#include<myscene.h>
#include<QCloseEvent>
namespace Ui {
class MainWindow;
}

class MainWindow : public QMainWindow
{
    Q_OBJECT

public:
    void closeEvent(QCloseEvent *event);
    explicit MainWindow(QWidget *parent = 0);
    ~MainWindow();
    void scoregui0();

private:
    Ui::MainWindow *ui;
    myscene *sc;//场景类在 mainwindow.h 中的对象
    int score;
 private slots:
```

```
    void UpdateScore();
};

#endif // MAINWINDOW_H
```

mainwindow.cpp 中代码

```
#include "mainwindow.h"
#include "ui_mainwindow.h"
#include<QCloseEvent>
#include<QMessageBox>
#include "handler.h"
MainWindow::MainWindow(QWidget *parent) :
    QMainWindow(parent),
    ui(new Ui::MainWindow)
{
    ui->setupUi(this);
    this->sc=new myscene;
    this->ui->graphicsView->setScene(sc);
    connect(this->ui->btn_start,SIGNAL(clicked(bool)),this->sc,SLOT(startGame()));
    //实现了开始键的功能
    connect(this->ui->btn_pause,SIGNAL(clicked(bool)),this->sc,SLOT(pauseGame()));
    //实现暂停键
    connect(this->ui->btn_stop,SIGNAL(clicked(bool)),this->sc,SLOT(stopGame()));
    //实现停止键
    this->score=0;
    handler *hand=handler::getInstance();
    connect(hand,SIGNAL(beatMouse()),this,SLOT(UpdateScore()));
}
void MainWindow::closeEvent(QCloseEvent *event)
{
    int ret=QMessageBox::question(this,"确定","你确定要关闭吗？","是","否");
    if(ret==1)
    {
        event->ignore();//忽略默认事件(关闭事件)
    }
}

MainWindow::~MainWindow()
{
    delete ui;
}
void MainWindow::UpdateScore()
{
    this->score+=10;
    this->ui->baoyuchen->display(this->score);
}
void MainWindow::scoregui0()
{
    this->score=0;
    this->ui->baoyuchen->display(this->score);
}
```

myitem.h

```
#ifndef MYITEM_H
#define MYITEM_H
#include<QGraphicsPixmapItem>
#include<QString>
#include<QGraphicsSceneMouseEvent>
```

```
class myitem : public QGraphicsPixmapItem
{
public:
    void setMouse(bool mouse);
    void mousePressEvent(QGraphicsSceneMouseEvent *event);
    myitem();
    void setPic(QString path);
    void mouseReleaseEvent(QGraphicsSceneMouseEvent *event);
    ~myitem();
    bool isMouse();
    void setStart(bool start);
    bool isStart();
private:
    bool mouse;
    bool start;
    //当点击开始的时候 为真，停止的时候为假
};

#endif // MYITEM_H
//单例问题，在很多情况下可以当做全局变量去使用
```

myitem.cpp 中代码

```
#include "myitem.h"
#include<QPixmap>
#include<QDebug>
#include<handler.h>
#include<QCursor>
myitem::myitem()
{
    this->setPixmap(QPixmap(":/bg/pic/51.jpg"));
    this->start=false;
    this->mouse=false;//在图元开始之前进行的一个初始化设定
    this->setCursor(QCursor(QPixmap(":/bg/pic/64.jpg")));
}

myitem::~myitem()
{

}
void myitem::setPic(QString path)
{
    this->setPixmap(QPixmap(path));
}
void myitem::mousePressEvent(QGraphicsSceneMouseEvent *event)
{
    //qDebug()<<"鼠标按下";
    this->setCursor(QCursor(QPixmap(":/bg/pic/62.jpg")));
    if(this->isStart())
    {
        handler *hand=handler::getInstance();
        if(this->isMouse())
        {
            //qDebug()<<"打的是地鼠";
            hand->addScore();
            this->setPixmap(QPixmap(":/bg/pic/54.jpg"));
        }
        /*else
        {
            //qDebug()<<"打的不是地鼠";
        }*/
    }
```

```
}
void myitem::mouseReleaseEvent(QGraphicsSceneMouseEvent *event)
{
    this->setCursor(QCursor(QPixmap(":/bg/pic/64.jpg")));
}

void myitem::setMouse(bool mouse)
{
    this->mouse=mouse;
}
bool myitem::isMouse()
{
    return this->mouse;
}
void myitem::setStart(bool start)
{
    this->start=start;
}
bool myitem::isStart()
{
    return this->start;
}
```

myscene.h

```
#ifndef MYSCENE_H
#define MYSCENE_H

#include <QObject>
#include<QGraphicsScene>
#include<myitem.h>
#include<QTimer>
class myscene : public QGraphicsScene
{
    Q_OBJECT
public:
    explicit myscene(QObject *parent = 0);
    ~myscene();

signals:

public slots:
    void showMouse();
    void startGame();
    void pauseGame();
    void stopGame();

private:
    QTimer *ptimer;//定义一个指针
    myitem *item[16];//安置照片，即打地鼠的背景画面。
};

#endif // MYSCENE_H
```

myscene.cpp 中代码

```
#include "myscene.h"
#include<stdlib.h>//背景以及地鼠出现消失的功能实现
#include"handler.h"
myscene::myscene(QObject *parent) : QGraphicsScene(parent)
{
```

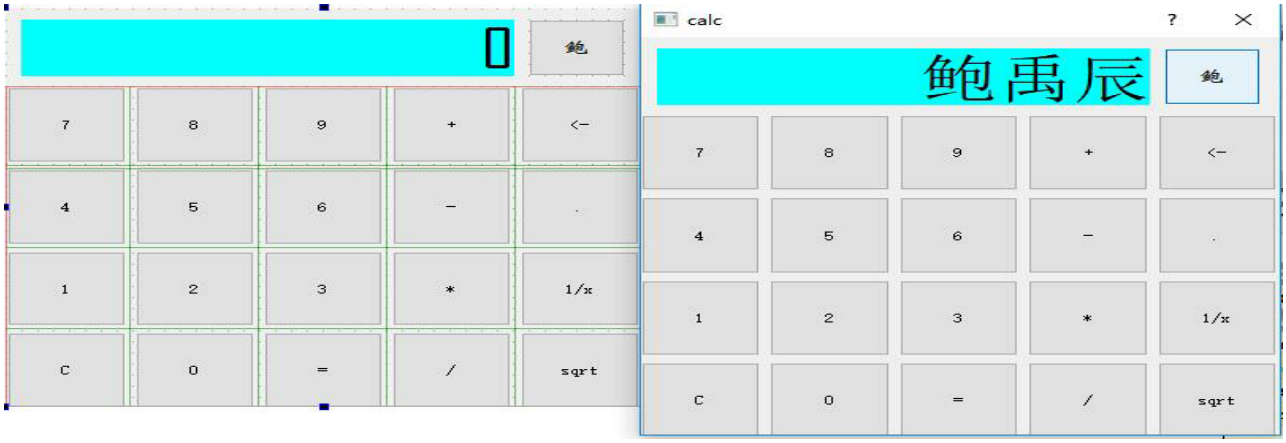
```
    int i=0;
    for(i=0;i<16;i++)
    {
        this->item[i]=new myitem;
        this->item[i]->setPos(i/4*this->item[i]->boundingRect().width(),i%4*this->item[i]
->boundingRect().height());
        this->addItem(this->item[i]);
    }
    this->ptimer=new QTimer;
    connect(this->ptimer,SIGNAL(timeout()),this,SLOT(showMouse()));
    //this->ptimer->start(1000);
}
void myscene::showMouse()
{
   //如何在老鼠出来之后再让他回去？
   //再换回原来的图片。。
   //随机一个数 0--15
   //先用 1--3 实验
   int count=rand()%3+1;//随机弹出 1-3 个老鼠
   int i=0;
   for(i=0;i<16;i++)
   {
       this->item[i]->setPic(":/bg/pic/51.jpg");
       this->item[i]->setMouse(false);
       //初始时时为背景图片此时不是老鼠为 false
   }
   for(i=0;i<count;i++)
   {
       int index=rand()%16;//老鼠的 16 个位置随机出现
       this->item[index]->setPic(":/bg/pic/52.jpg");
       this->item[index]->setMouse(true);
       //当用地鼠替换背景图片之后，此时是老鼠，设定为 true;

   }
}
/*事件驱动
 * 点击事件   发送 clicked 绑定信号
 * 右击事件
 * 编辑
 * 关闭事件  关闭程序*/
myscene::~myscene()
{

}
void myscene::startGame()
{
   int i=0;
   for(i=0;i<16;i++)
   {
       this->item[i]->setStart(true);
   }



   this->ptimer->start(1000);
   //为了使按下暂停或者停止的时候，鼠标点击不会产生反应
   //所以 设置了 bool start bool isStart()  void setStart()
   //当为 true 时正常运行，当点击暂停或者停止的时候设置成 false
   //鼠标动作对界面无反应。
}
void myscene::pauseGame()
{
```

```
    int i=0;
    for(i=0;i<16;i++)
    {
        this->item[i]->setStart(false);
    }
    this->ptimer->stop();
}
void myscene::stopGame()
{
    int i=0;
    for(i=0;i<16;i++)
    {
        this->item[i]->setStart(false);
    }
    this->ptimer->stop();
    for(i=0;i<16;i++)
    {
        this->item[i]->setPic(":/bg/pic/51.jpg");
        this->item[i]->setMouse(false);
        //初始时时为背景图片此时不是老鼠为 false
    }
    //停止键不同于暂停键的地方，清空游戏界面，数据。

}
```

main.cpp 中代码

```
#include "mainwindow.h"
#include <QApplication>
#include<stdio.h>
#include<time.h>
int main(int argc, char *argv[])
{
    QApplication a(argc, argv);
    srand((unsigned)time(NULL));//设置一个随机数的种子。
    MainWindow w;
    w.show();

    return a.exec();
}
```

Qt 计算器：



鲍禹辰牌商标： <---<

Calc.h

```
#ifndef CALC_H
```

```
#define CALC_H
#include"model.h"
#include <QDialog>
#include<QString>
namespace Ui {
class calc;
}
class calc : public QDialog
{
    Q_OBJECT
public:
    explicit calc(QWidget *parent = 0);
    ~calc();
private slots:
    void getBtn0();
    void getBtn1();
    void getBtn2();
    void getBtn3();
    void getBtn4();
    void getBtn5();
    void getBtn6();
    void getBtn7();
    void getBtn8();
    void getBtn9();
    void on_BC_clicked();
    void on_Badd_clicked();
    void on_dengyu_clicked();
    void on_jian_clicked();
    void on_cheng_clicked();
    void on_chushu_clicked();
    void on_dian_clicked();
    void on_daoshu_clicked();
    void on_pushButton_16_clicked();
    void on_Backspace_clicked();
    void on_bao_clicked();
private:
    Ui::calc *ui;
    QString tmp;
    model * mode;
};
#endif // CALC_H
```

Model.h

```
#ifndef MODEL_H
#define MODEL_H
#include<QString>
class model
{
public:
    model();
    ~model();
    void setNum1(double num1);
    void setNum2(double num2);
    void setFlag(QString flag);
    QString docalc();
private:
    double num1;
    double num2;
```

```
    QString flag;
};
#endif // MODEL_H
```

Calc.cpp

```
#include "calc.h"
#include "ui_calc.h"
#include<math.h>

calc::calc(QWidget *parent) :
    QDialog(parent),
    ui(new Ui::calc)
{
    ui->setupUi(this);
    this->tmp="";
    this->mode=new model;
    connect(this->ui->B0,SIGNAL(clicked(bool)),this,SLOT(getBtn0()));
    connect(this->ui->B1,SIGNAL(clicked(bool)),this,SLOT(getBtn1()));
    connect(this->ui->B2,SIGNAL(clicked(bool)),this,SLOT(getBtn2()));
    connect(this->ui->B3,SIGNAL(clicked(bool)),this,SLOT(getBtn3()));
    connect(this->ui->B4,SIGNAL(clicked(bool)),this,SLOT(getBtn4()));
    connect(this->ui->B5,SIGNAL(clicked(bool)),this,SLOT(getBtn5()));
    connect(this->ui->B6,SIGNAL(clicked(bool)),this,SLOT(getBtn6()));
    connect(this->ui->B7,SIGNAL(clicked(bool)),this,SLOT(getBtn7()));
    connect(this->ui->B8,SIGNAL(clicked(bool)),this,SLOT(getBtn8()));
    connect(this->ui->B9,SIGNAL(clicked(bool)),this,SLOT(getBtn9()));
}
calc::~calc()
{
    delete ui;
}
void calc::getBtn0()
{
    if(this->tmp!="")
    {
        this->tmp+=this->ui->B0->text();
        this->ui->label->setText(this->tmp);
    }
}
void calc::getBtn1()
{
    this->tmp+=this->ui->B1->text();
    this->ui->label->setText(this->tmp);
}
void calc::getBtn2()
{
    this->tmp+=this->ui->B2->text();
    this->ui->label->setText(this->tmp);
}
void calc::getBtn3()
{
    this->tmp+=this->ui->B3->text();
    this->ui->label->setText(this->tmp);
}
void calc::getBtn4()
{
    this->tmp+=this->ui->B4->text();
    this->ui->label->setText(this->tmp);
```

```
}
void calc::getBtn5()
{
    this->tmp+=this->ui->B5->text();
    this->ui->label->setText(this->tmp);
}
void calc::getBtn6()
{
    this->tmp+=this->ui->B6->text();
    this->ui->label->setText(this->tmp);
}
void calc::getBtn7()
{
    this->tmp+=this->ui->B7->text();
    this->ui->label->setText(this->tmp);
}
void calc::getBtn8()
{
    this->tmp+=this->ui->B8->text();
    this->ui->label->setText(this->tmp);
}
void calc::getBtn9()
{
    this->tmp+=this->ui->B9->text();
    this->ui->label->setText(this->tmp);
}
void calc::on_BC_clicked()
{
    this->tmp="";
    this->ui->label->setText("0");
}
void calc::on_Badd_clicked()
{
    double num1=this->tmp.toDouble();
    this->mode->setNum1(num1);
    this->tmp="";
    this->ui->label->setText("0");
    QString ex=this->ui->Badd->text();
    this->mode->setFlag(ex);
}
void calc::on_dengyu_clicked()
{
    double num=this->tmp.toDouble();
    this->mode->setNum2(num);
    QString res = this->mode->docalc();
    this->ui->label->setText(res);
    this->tmp=res;
}
void calc::on_jian_clicked()
{
      double num1=this->tmp.toDouble();
      this->mode->setNum1(num1);
    this->tmp="";
    this->ui->label->setText("0");
    QString ex=this->ui->jian->text();
    this->mode->setFlag(ex);
}
void calc::on_cheng_clicked()
{
```

```
    double num1=this->tmp.toDouble();
    this->mode->setNum1(num1);
    this->tmp="";
    this->ui->label->setText("0");
    QString ex=this->ui->cheng->text();
    this->mode->setFlag(ex);
}
void calc::on_chushu_clicked()
{
    double num1=this->tmp.toDouble();
    this->mode->setNum1(num1);
    this->tmp="";
    this->ui->label->setText("0");
    QString ex=this->ui->chushu->text();
    this->mode->setFlag(ex);
}
void calc::on_dian_clicked()
{
    this->tmp+=this->ui->dian->text();
    this->ui->label->setText(this->tmp);
}
void calc::on_daoshu_clicked()
{
    double num1=1/(this->tmp.toDouble());
    this->tmp=QString::number(num1);
    this->ui->label->setText(this->tmp);
}
void calc::on_pushButton_16_clicked()
{
    double num1=this->tmp.toDouble();
    num1=sqrt(num1);
    this->tmp=QString::number(num1);
    this->ui->label->setText(this->tmp);
}
void calc::on_Backspace_clicked()
{
   this->tmp=this->tmp.left(this->tmp.length()-1);
   this->ui->label->setText(this->tmp);
}
void calc::on_bao_clicked()
{
    this->tmp="";
    this->tmp="鲍禹辰";
    this->ui->label->setText(this->tmp);
}
```

Model.cpp

```
#include "model.h"
model::model()
{
    this->num1=0;
    this->num2=0;
}
model::~model()
{

}
void model::setNum1(double num1)
{
```

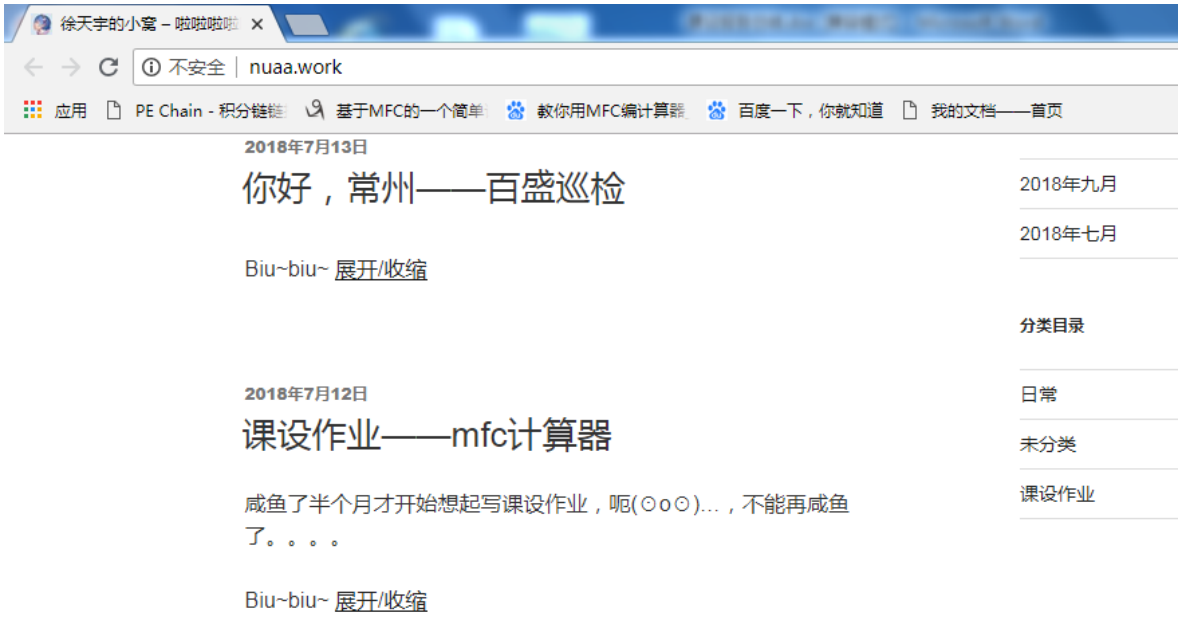
```
    this->num1=num1;
}
void model::setNum2(double num2)
{
    this->num2=num2;
}
void model::setFlag(QString flag)
{
    this->flag=flag;
}

QString model::docalc()
{
    double result=0;
    if(this->flag=="+")
    {
      result=this->num1+this->num2;
    }
    else if(this->flag=="-")
    {
      result=this->num1-this->num2;
    }
    else if(this->flag=="*")
    {
       result=this->num1*this->num2;
    }
    else if(this->flag=="/")
    {
          if(this->num2==0)
          {
              return "ERROR";
          }
        result=this->num1/this->num2;
     }
    else
    {
       return QString::number(this->num2);
    }
    return QString::number(result);
}
```

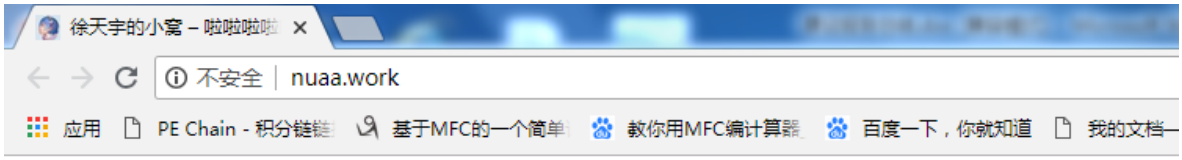
Main.cpp

```
#include "calc.h"
#include <QApplication>
int main(int argc, char *argv[])
{
    QApplication a(argc, argv);
    calc w;
    w.show();
    return a.exec();
}
```

MFC 计算器：


徐天宇的计算器
一
二
计算结果
46
35
1610.000000
+
-
×
÷
=
关于 xty
xty 1.1.8 版
徐天宇版权所有 (C) 2018
确定
1
2
6
7
8
9
0
关于此软件
nuaa.work


徐天宇的小窝 – 啦啦啦啦
不安全 | nuaa.work
应用
PE Chain - 积分链接
基于MFC的一个简单
教你用MFC编计算器
百度一下，你就知道
我的文档——首页
2018年7月13日
你好，常州——百盛巡检
Biu~biu~ 展开/收缩
2018年7月12日
课设作业——mfc计算器
咸鱼了半个月才开始想起写课设作业，呃(⊙o⊙)…，不能再咸鱼了。。。。
Biu~biu~ 展开/收缩
2018年九月
2018年七月
分类目录
日常
未分类
课设作业



2018年7月12日

课设作业——mfc计算器

咸鱼了半个月才开始想起写课设作业，呃(⊙o⊙)…，不能再咸鱼了。。。。

Biu~biu~ 展开/收缩

1.工具使用VC，创建MFC AppWizard[exe],设定好路径和工程名。

2.用按钮控件添加按钮（按你自己喜好来）。

2.添加完成后，双击控件添加对应的消息响应。

3.给控件消息响应函数添加具体代码部分。

加法：

```
void CMyDlg::OnButton11()
{
// TODO: Add your control notification handler code here
UpdataData() ;
double n_result;
n_result=m_num1+m_num2;
m_result=n_result;
char sz[20];
sprintf(sz,"%f",m_result);            /数组保存并格式化打印输出
```

```
// xty.cpp : Defines the class behaviors for the application.
//

#include "stdafx.h"
#include "xty.h"
#include "xtyDlg.h"

#ifdef _DEBUG
#define new DEBUG_NEW
#undef THIS_FILE
static char THIS_FILE[] = __FILE__;
#endif

/////////////////////////////////////////////////////////////////////////////
// CXtyApp

BEGIN_MESSAGE_MAP(CXtyApp, CWinApp)
        //{{AFX_MSG_MAP(CXtyApp)
                // NOTE - the ClassWizard will add and remove mapping macros here.
                //    DO NOT EDIT what you see in these blocks of generated code!
        //}}AFX_MSG
        ON_COMMAND(ID_HELP, CWinApp::OnHelp)
END_MESSAGE_MAP()
```

```
/////////////////////////////////////////////////////////////////////////////
// CXtyApp construction

CXtyApp::CXtyApp()
{
	// TODO: add construction code here,
	// Place all significant initialization in InitInstance
}

/////////////////////////////////////////////////////////////////////////////
// The one and only CXtyApp object

CXtyApp theApp;
// xtyDlg.cpp : implementation file
//

#include "stdafx.h"
#include "xty.h"
#include "xtyDlg.h"

#ifdef _DEBUG
#define new DEBUG_NEW
#undef THIS_FILE
static char THIS_FILE[] = __FILE__;
#endif

/////////////////////////////////////////////////////////////////////////////
// CAboutDlg dialog used for App About

class CAboutDlg : public CDialog
{
public:
	CAboutDlg();

// Dialog Data
	//{{AFX_DATA(CAboutDlg)
	enum { IDD = IDD_ABOUTBOX };
	//}}AFX_DATA

	// ClassWizard generated virtual function overrides
	//{{AFX_VIRTUAL(CAboutDlg)
	protected:
	virtual void DoDataExchange(CDataExchange* pDX);    // DDX/DDV support
	//}}AFX_VIRTUAL

// Implementation
protected:
	//{{AFX_MSG(CAboutDlg)
	//}}AFX_MSG
	DECLARE_MESSAGE_MAP()
};

CAboutDlg::CAboutDlg() : CDialog(CAboutDlg::IDD)
{
	//{{AFX_DATA_INIT(CAboutDlg)
	//}}AFX_DATA_INIT
}

void CAboutDlg::DoDataExchange(CDataExchange* pDX)
{
	CDialog::DoDataExchange(pDX);
	//{{AFX_DATA_MAP(CAboutDlg)
	//}}AFX_DATA_MAP
}

BEGIN_MESSAGE_MAP(CAboutDlg, CDialog)
	//{{AFX_MSG_MAP(CAboutDlg)
```

```
                        // No message handlers
        //}}AFX_MSG_MAP
END_MESSAGE_MAP()

/////////////////////////////////////////////////////////////////////////////
// CXtyDlg dialog

CXtyDlg::CXtyDlg(CWnd* pParent /*=NULL*/)
        : CDialog(CXtyDlg::IDD, pParent)
{
        //{{AFX_DATA_INIT(CXtyDlg)
        m_num1 = 0.0;
        m_num2 = 0.0;
        m_result = 0.0;
        //}}AFX_DATA_INIT
        // Note that LoadIcon does not require a subsequent DestroyIcon in Win32
        m_hIcon = AfxGetApp()->LoadIcon(IDR_MAINFRAME);
}

void CXtyDlg::DoDataExchange(CDataExchange* pDX)
{
        CDialog::DoDataExchange(pDX);
        //{{AFX_DATA_MAP(CXtyDlg)
        DDX_Control(pDX, IDC_EDIT3, m_CResult);
        DDX_Text(pDX, IDC_EDIT1, m_num1);
        DDX_Text(pDX, IDC_EDIT2, m_num2);
        DDX_Text(pDX, IDC_EDIT3, m_result);
        //}}AFX_DATA_MAP
}

BEGIN_MESSAGE_MAP(CXtyDlg, CDialog)
        //{{AFX_MSG_MAP(CXtyDlg)
        ON_WM_SYSCOMMAND()
        ON_WM_PAINT()
        ON_WM_QUERYDRAGICON()
        ON_BN_CLICKED(IDC_BUTTON11, OnButton11)
        ON_BN_CLICKED(IDC_BUTTON12, OnButton12)
        ON_BN_CLICKED(IDC_BUTTON13, OnButton13)
        ON_BN_CLICKED(IDC_BUTTON14, OnButton14)
        ON_EN_SETFOCUS(IDC_EDIT1, OnSetfocusEdit1)
        ON_EN_SETFOCUS(IDC_EDIT2, OnSetfocusEdit2)
        ON_WM_LBUTTONDOWN()
        ON_BN_CLICKED(IDC_BUTTON1, OnButton1)
        ON_BN_CLICKED(IDC_BUTTON2, OnButton2)
        ON_BN_CLICKED(IDC_BUTTON3, OnButton3)
        ON_BN_CLICKED(IDC_BUTTON4, OnButton4)
        ON_BN_CLICKED(IDC_BUTTON5, OnButton5)
        ON_BN_CLICKED(IDC_BUTTON6, OnButton6)
        ON_BN_CLICKED(IDC_BUTTON7, OnButton7)
        ON_BN_CLICKED(IDC_BUTTON8, OnButton8)
        ON_BN_CLICKED(IDC_BUTTON9, OnButton9)
        ON_BN_CLICKED(IDC_BUTTON10, OnButton10)
        ON_BN_CLICKED(IDC_BUTTON15, OnButton15)
        ON_WM_MOUSEMOVE()
        //}}AFX_MSG_MAP
END_MESSAGE_MAP()

/////////////////////////////////////////////////////////////////////////////
// CXtyDlg message handlers

BOOL CXtyDlg::OnInitDialog()
{
GetDlgItem(IDC_LINK)->GetWindowRect(&m_pRectLink);
ScreenToClient(&m_pRectLink);//将屏幕坐标转换为客户坐标
        CDialog::OnInitDialog();
        nEditFlag = 0;
```

```
		// Add "About..." menu item to system menu.

		// IDM_ABOUTBOX must be in the system command range.
		ASSERT((IDM_ABOUTBOX & 0xFFF0) == IDM_ABOUTBOX);
		ASSERT(IDM_ABOUTBOX < 0xF000);

		CMenu* pSysMenu = GetSystemMenu(FALSE);
		if (pSysMenu != NULL)
		{
			CString strAboutMenu;
			strAboutMenu.LoadString(IDS_ABOUTBOX);
			if (!strAboutMenu.IsEmpty())
			{
				pSysMenu->AppendMenu(MF_SEPARATOR);
				pSysMenu->AppendMenu(MF_STRING, IDM_ABOUTBOX,
strAboutMenu);
			}
		}

		// Set the icon for this dialog.  The framework does this automatically
		//  when the application's main window is not a dialog
		SetIcon(m_hIcon, TRUE);			// Set big icon
		SetIcon(m_hIcon, FALSE);		// Set small icon

		// TODO: Add extra initialization here

		return TRUE;  // return TRUE  unless you set the focus to a control
}

void CXtyDlg::OnSysCommand(UINT nID, LPARAM lParam)
{
		if ((nID & 0xFFF0) == IDM_ABOUTBOX)
		{
			CAboutDlg dlgAbout;
			dlgAbout.DoModal();
		}
		else
		{
			CDialog::OnSysCommand(nID, lParam);
		}
}

// If you add a minimize button to your dialog, you will need the code below
//  to draw the icon.  For MFC applications using the document/view model,
//  this is automatically done for you by the framework.

void CXtyDlg::OnPaint()
{
		if (IsIconic())
		{
			CPaintDC dc(this); // device context for painting

			SendMessage(WM_ICONERASEBKGND, (WPARAM) dc.GetSafeHdc(), 0);

			// Center icon in client rectangle
			int cxIcon = GetSystemMetrics(SM_CXICON);
			int cyIcon = GetSystemMetrics(SM_CYICON);
			CRect rect;
			GetClientRect(&rect);
			int x = (rect.Width() - cxIcon + 1) / 2;
			int y = (rect.Height() - cyIcon + 1) / 2;

			// Draw the icon
			dc.DrawIcon(x, y, m_hIcon);
		}
		else
		{
```

```
                CDialog::OnPaint();
        }
}

// The system calls this to obtain the cursor to display while the user drags
//  the minimized window.
HCURSOR CXtyDlg::OnQueryDragIcon()
{
        return (HCURSOR) m_hIcon;
}

void CXtyDlg::OnButton11()
{
        // TODO: Add your control notification handler code here
        UpdateData();
        double n_result;
        n_result = m_num1 + m_num2;
        m_result = n_result;
        char sz[20];
        sprintf(sz,"%f",m_result);
        m_CResult.SetWindowText(sz);

}

void CXtyDlg::OnButton12()
{
        // TODO: Add your control notification handler code here
        UpdateData();
        double n_result;
        n_result = m_num1 - m_num2;
        m_result = n_result;
        char sz[20];
        sprintf(sz,"%f",m_result);
        m_CResult.SetWindowText(sz);

}

void CXtyDlg::OnButton13()
{
        // TODO: Add your control notification handler code here
        UpdateData();
        double n_result;
        n_result = m_num1 * m_num2;
        m_result = n_result;
        char sz[20];
        sprintf(sz,"%f",m_result);
         m_CResult.SetWindowText(sz);

}

void CXtyDlg::OnButton14()
{
        // TODO: Add your control notification handler code here
        UpdateData();
        double n_result;
        n_result = m_num1 / m_num2;
        m_result = n_result;
        char sz[20];
        sprintf(sz,"%f",m_result);
        m_CResult.SetWindowText(sz);

}

void CXtyDlg::OnSetfocusEdit1()
{
        // TODO: Add your control notification handler code here
        nEditFlag = 1;
```

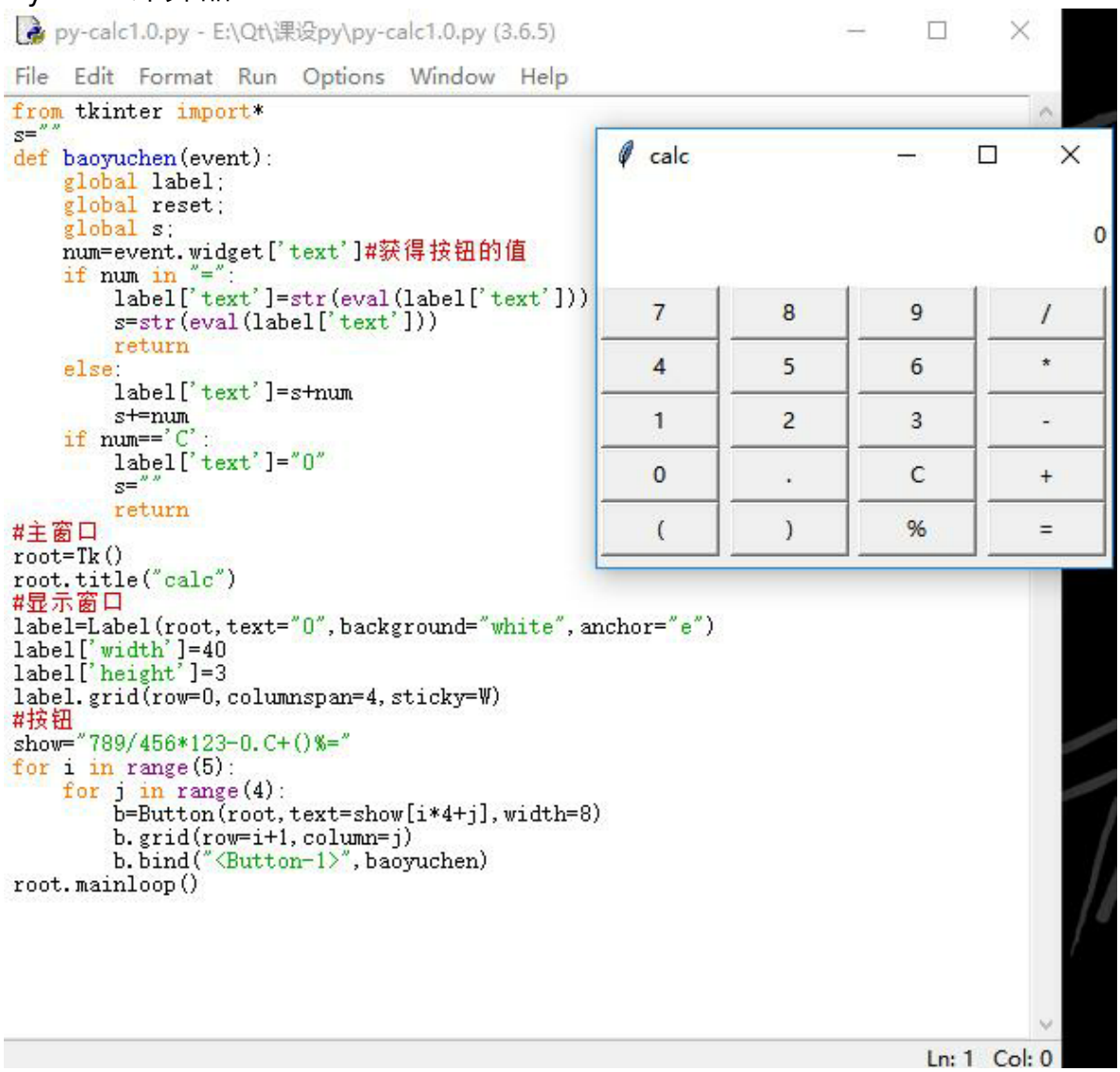
```
}

void CXtyDlg::OnSetfocusEdit2()
{
	// TODO: Add your control notification handler code here
	nEditFlag = 2;
}

void CXtyDlg::OnLButtonDown(UINT nFlags, CPoint point)
{
	// TODO: Add your message handler code here and/or call default
	this->SetFocus();
	nEditFlag=0;
	CDialog::OnLButtonDown(nFlags, point);
if
(point.x>m_pRectLink.left&&point.x<m_pRectLink.right&&point.y>m_pRectLink.top&&point.y<m_pRectLi
nk.bottom)

//此处添加判断坐标算法

{ if (nFlags==MK_LBUTTON)//鼠标左键按下

 { //为改善鼠标效果，此处加入以上变换鼠标形状的代码

ShellExecute(0, NULL, "http://nuaa.work", NULL,NULL, SW_NORMAL);

//也可以添加电子邮件的链接

 }
}
}

void CXtyDlg::OnButton1()
{
	// TODO: Add your control notification handler code here
	UpdateData(TRUE);
	if (nEditFlag==1)
	{
		m_num1=m_num1*10+1;
	}
	else if(nEditFlag==2)
	{
		m_num2=m_num2*10+1;
	}
	UpdateData(FALSE);
}

void CXtyDlg::OnButton2()
{
	// TODO: Add your control notification handler code here
	UpdateData(TRUE);
	if (nEditFlag==1)
	{
		m_num1=m_num1*10+2;
	}
	else if(nEditFlag==2)
	{
		m_num2=m_num2*10+2;
	}
	UpdateData(FALSE);
}

void CXtyDlg::OnButton3()
{
	// TODO: Add your control notification handler code here
```

```
	UpdateData(TRUE);
	if (nEditFlag==1)
	{
		m_num1=m_num1*10+3;
	}
	else if(nEditFlag==2)
	{
		m_num2=m_num2*10+3;
	}
	UpdateData(FALSE);
}

void CXtyDlg::OnButton4()
{
	// TODO: Add your control notification handler code here
	UpdateData(TRUE);
	if (nEditFlag==1)
	{
		m_num1=m_num1*10+4;
	}
	else if(nEditFlag==2)
	{
		m_num2=m_num2*10+4;
	}
	UpdateData(FALSE);
}

void CXtyDlg::OnButton5()
{
	// TODO: Add your control notification handler code here
	UpdateData(TRUE);
	if (nEditFlag==1)
	{
		m_num1=m_num1*10+5;
	}
	else if(nEditFlag==2)
	{
		m_num2=m_num2*10+5;
	}
	UpdateData(FALSE);
}

void CXtyDlg::OnButton6()
{
	// TODO: Add your control notification handler code here
	UpdateData(TRUE);
	if (nEditFlag==1)
	{
		m_num1=m_num1*10+6;
	}
	else if(nEditFlag==2)
	{
		m_num2=m_num2*10+6;
	}
	UpdateData(FALSE);
}

void CXtyDlg::OnButton7()
{
	// TODO: Add your control notification handler code here
	UpdateData(TRUE);
	if (nEditFlag==1)
	{
		m_num1=m_num1*10+7;
	}
	else if(nEditFlag==2)
	{
```

```
                m_num2=m_num2*10+7;
        }
        UpdateData(FALSE);
}

void CXtyDlg::OnButton8()
{
        // TODO: Add your control notification handler code here
        UpdateData(TRUE);
        if (nEditFlag==1)
        {
                m_num1=m_num1*10+8;
        }
        else if(nEditFlag==2)
        {
                m_num2=m_num2*10+8;
        }
        UpdateData(FALSE);
}

void CXtyDlg::OnButton9()
{
        // TODO: Add your control notification handler code here
        UpdateData(TRUE);
        if (nEditFlag==1)
        {
                m_num1=m_num1*10+9;
        }
        else if(nEditFlag==2)
        {
                m_num2=m_num2*10+9;
        }
        UpdateData(FALSE);
}

void CXtyDlg::OnButton10()
{
        // TODO: Add your control notification handler code here
        UpdateData(TRUE);
        if (nEditFlag==1)
        {
                m_num1=m_num1*10+0;
        }
        else if(nEditFlag==2)
        {
                m_num2=m_num2*10+0;
        }
        UpdateData(FALSE);
}

void CXtyDlg::OnButton15()
{
        // TODO: Add your control notification handler code here
        CAboutDlg dlgAbout;
        dlgAbout.DoModal();
}

void CXtyDlg::OnMouseMove(UINT nFlags, CPoint point)
{
        // TODO: Add your message handler code here and/or call default
        //下面设置鼠标在静态文本区时，将光标设成小手状

if
(point.x>m_pRectLink.left&&point.x<m_pRectLink.right&&point.y>m_pRectLink.top&&point.y<m_pRectLi
nk.bottom)
```

```
//此处添加判断坐标算法

{
HCURSOR hCursor;

hCursor=AfxGetApp()->LoadCursor(IDC_HAND);

//将鼠标设为小手状

SetCursor(hCursor);
}
	CDialog::OnMouseMove(nFlags, point);
}
/////////////////////////////////////////////////////////////////////////////
// CXtyApp initialization

BOOL CXtyApp::InitInstance()
{
	AfxEnableControlContainer();

	// Standard initialization
	// If you are not using these features and wish to reduce the size
	//  of your final executable, you should remove from the following
	//  the specific initialization routines you do not need.

#ifdef _AFXDLL
	Enable3dControls();			// Call this when using MFC in a shared DLL
#else
	Enable3dControlsStatic();	// Call this when linking to MFC statically
#endif

	CXtyDlg dlg;
	m_pMainWnd = &dlg;
	int nResponse = dlg.DoModal();
	if (nResponse == IDOK)
	{
		// TODO: Place code here to handle when the dialog is
		//  dismissed with OK
	}
	else if (nResponse == IDCANCEL)
	{
		// TODO: Place code here to handle when the dialog is
		//  dismissed with Cancel
	}

	// Since the dialog has been closed, return FALSE so that we exit the
	//  application, rather than start the application's message pump.
	return FALSE;
}
```

Python 计算器：



图书管理系统：

Mytype.h

```
#include<bits/stdc++.h>
#include<iostream>
#include<string>
#include<cstdlib>
using namespace std;
class CBook//图书类
{
        private:
                int code;
                char name[20];
                bool Is_Stock;     //true 为在书架，false 为 借出
                bool Is_Existing;  //true 为存在,false 为删除
```

```
        public:
        CBook();//默认构造函数
        CBook(int no,char na[]);
        void setcode(int n);
        int getcode();
        void setname(char na[]);
        char *getname();
        bool get_Stock();
        void set_Stock(bool);
        bool get_Existing();
        void set_Existing(bool);
};
//用户类
class CUser
{
        private:
                int code;
                char name[20];
        public:
                CUser();
                CUser(int id,char na[]);
                void setcode(int n);
                int getcode();
                void setname(char na[]);
                char *getname();
                void show();
 };
 //用户类的派生类-----读者类
 class CReader : public CUser
 {
        private:
                bool Is_Existing;
                bool Is_Borrowing;
                CBook brbook;
        public:
                CReader();
                CReader(int id,char na[]);
                bool get_Existing();
                void set_Existing(bool);
                bool get_Borrowing();
                void set_Borrowing(bool);
                void setbrbook(CBook bt);
                CBook getbrbook();
                void show();
  };
  //读者管理类
class CReaderManager
{
  public:
        CReader rdarray[50];
        int top;
  public:
        CReaderManager();
          ~CReaderManager();
        void addreaders();
        int findreaders(int suffix);
        int findreaders();
        void editreaders();
        void delreaders();
```

```
        void listreaders();
        //int findreaders();
};
//图书管理类:
class CBookManager
{
        public:
        CBook btarray[50];
        int top;
        public:
        CBookManager();
        ~CBookManager();
        void addbooks();
        int findbooks();
        int        findbooks(int suffix);
        void editbooks();
        void delbooks();
        void listbooks();
};
```

Classlmp.cpp

```
//#include<bits/stdc++.h>
#include<fstream>
#include"Mytype.h"
CUser::CUser(){}
CUser::CUser(int id,char na[])
{
        code=id;
        strcpy(name,na);
}
void CUser::setcode(int n)
{
        code=n;
}
int CUser::getcode()//返回图书编号
{
        return code;
}
void CUser::setname(char na[])
{
        strcpy(name,na);
}
char *CUser::getname()
{
        return name;
}
void CUser::show()
{
        cout<<"系统管理员编号:"<<code<<"系统管理员姓名:"<<name;
}
//读者类的实现
CReader::CReader(){}
 CReader::CReader(int id,char na[]) : CUser(id,na)
 {
        ls_Existing=1;
        ls_Borrowing=false;
 }
 bool CReader::get_Existing()
 {
```

```
	return ls_Existing;
}
void CReader::set_Existing(bool n)
{
	ls_Existing=n;
}
bool CReader::get_Borrowing()
{
	return ls_Borrowing;
}
void CReader::set_Borrowing(bool n)
{
	ls_Borrowing=n;
}
void CReader::setbrbook(CBook bt)
{
	brbook=bt;
}
CBook CReader::getbrbook()
{
	return brbook;
}
void CReader::show()
{
	char *js;
	char *sc;
	if(ls_Existing==true)
	{
		cout<<"读者编号："<<this->getcode()<<endl;
		cout<<"读者姓名："<<this->getname()<<endl;
		if(ls_Borrowing==true)
			cout<<"已借图书名称："<<brbook.getname();
		else
			cout<<"目前没有已借图书！"<<endl;
	}
	else
		cout<<"该读者已经从系统删除！"<<endl;
}
//图书类的实现
CBook::CBook(){}
CBook::CBook(int id,char na[])
{
	code=id;
	strcpy(name,na);
	ls_Stock=true;
	ls_Existing=true;
}
void CBook::setcode(int n)
{
	code=n;
}
int CBook::getcode()
{
	return code;
}
void CBook::setname(char na[])
{
	strcpy(name,na);
}
```

```
char *CBook::getname()
{
	return name;
}
bool CBook::get_Stock()
{
	return Is_Stock;
}
void CBook::set_Stock(bool n)
{
	Is_Stock=n;
}
bool CBook::get_Existing()
{
	return Is_Existing;
}
void CBook::set_Existing(bool n)
{
	Is_Existing=n;
}
//读者管理类的实现
CReaderManager::CReaderManager()
{
	CReader rd;
	top=-1;
	fstream file("reader.txt",ios::in);//打开文件
	while(1)
	{
		file.read((char*)&rd,sizeof(rd));
		if(!file) break;
		top++;
		rdarray[top]=rd;
	}
	file.close();//关闭文件
}
CReaderManager::~CReaderManager()
{
	fstream file("reader.txt",ios::out);
	for(int i=0;i<=top;i++)
	{
		if(rdarray[i].get_Existing()==true)
		file.write((char*)&rdarray[i],sizeof(rdarray[i]));
	}
	file.close();
}
void CReaderManager::addreaders()//添加读者
{
	CReader rd;
	int no;
	char rname[20];
	cout<<"请输入学号：";
	cin>>no;
	cout<<endl<<"请输入姓名：";
	cin>>rname;
	cout<<endl;
	rd.setcode(no);
	rd.setname(rname);
	rd.set_Borrowing(false);
	rd.set_Existing(true);
```

```
		top++;
		rdarray[top]=rd;
		return;
}
int CReaderManager::findreaders(int suffix)
{
		int no;
		cout<<"请输入学号：";
		cin>>no;
		for(int i=0;i<=top;i++)
		{
			if(rdarray[i].getcode()==no && rdarray[i].get_Existing()==true)
			{
				return i;
			}
		}
		return -1;
}
int CReaderManager::findreaders()
//查找读者，这个函数用于读者维护操作
{
		int no;
		char value[3];
		cout<<"请输入学号：";
		cin>>no;
		for(int i=0;i<=top;i++)
		{
			if(rdarray[i].getcode()==no&&rdarray[i].get_Existing()==true)
			{
				if(rdarray[i].get_Borrowing()==true)
					strcpy(value,"借");
				if(rdarray[i].get_Borrowing()==false)
					strcpy(value,"无");
				cout<<"学号："<<rdarray[i].getcode()<<"姓名："
				<<rdarray[i].getname()<<"是否借书:"<<value<<endl;
				char ch;
				cout<<"按任意键继续....."<<endl;
				cin>>ch;
				return i;
			}
		}
		return -1;
}
void CReaderManager::editreaders()
{
		int cur;
		cur=findreaders();
		if(cur==-1)
		{
			cout<<"没有该学生"<<endl;
			return;
		}
		cout<<"请修改数据："<<endl;
		cout<<endl<<"姓名:";
		cin>>rdarray[cur].getname();
		return;
}
void CReaderManager::delreaders()
{
```

```
	int cur;
	cur=findreaders();
	if(cur==-1)
	{
		return;
	}
	if(rdarray[cur].get_Borrowing()==true)
	{
		cout<<"该用户借了图书：请还书后再删除"<<endl;
		return;
	}
	rdarray[cur].set_Existing(false);
	return;
}
void CReaderManager::listreaders()
{
	char value[3];
	for(int i=0;i<=top;i++)
	{
		if(rdarray[i].get_Existing()==true)
		{
			if(rdarray[i].get_Borrowing()==true)
				strcpy(value,"借");
			if(rdarray[i].get_Borrowing()==false)
				strcpy(value,"无");
			cout<<"学号："<<rdarray[i].getcode()<<"姓名："
			<<rdarray[i].getname()<<"是否借书:"<<value<<endl;
		}

	}
	char ch;
	cout<<"按任意键继续.....\n";
	cin>>ch;
	return;
}
//图书 管理类的实现
CBookManager::CBookManager()
{
	CBook b;
	top=-1;
	fstream file("book.txt",ios::in);
	while(1)
	{
		file.read((char*)&b,sizeof(b));
		if(!file)
			break;
		top++;
		btarray[top]=b;
	}
	file.close();
}
CBookManager::~CBookManager()
{
	fstream file("book.txt",ios::out);
	for(int i=0;i<=top;i++)
	{
		if(btarray[i].get_Existing()==true)
			file.write((char*)&btarray[i],sizeof(btarray[i]));
	}
```

```
		file.close();
}
void CBookManager::addbooks()
{
		CBook bk;
		int no;
		char bname[20];
		cout<<"请输入书号：";
		cin>>no;
		cout<<endl<<"请输入书名:";
		cin>>bname;
		cout<<endl;
		bk.setcode(no);
		bk.setname(bname);
		bk.set_Existing(true);
		bk.set_Stock(true);
		top++;
		btarray[top]=bk;
		return;
}
int CBookManager::findbooks(int suffix)
{
		int no;
		cout<<"请输入书号：";
		cout<<top<<endl;
		cin>>no;
		for(int i=0;i<=top;i++)
		{
			if(btarray[i].getcode()==no&&btarray[i].get_Existing()==true)
			{
				return i;
			}
		}
		return -1;
}
int CBookManager::findbooks()
{
		int no;
		char value[3];
		cout<<"请输入书号：";
		cin>>no;
		for(int i=0;i<=top;i++)
		{
			if(btarray[i].getcode()==no&&btarray[i].get_Existing()==true)
			{
				if(btarray[i].get_Stock()==true)
					strcpy(value,"在架");
				if(btarray[i].get_Stock()==false)
					strcpy(value,"借出");
				cout<<"书号："<<btarray[i].getcode()<<"书的名称："
				<<btarray[i].getname()<<"图书状态："<<value<<endl;
				cout<<"按任意键继续："<<endl;
				char ch;
				cin>>ch;
				return i;
			}
		}
		return -1;
}
```

```
void CBookManager::editbooks()
{
	int cur;
	cur=findbooks();
	if(cur==-1)
	{
		cout<<"没有这本书"<<endl;
		return;
	}
	cout<<"书号：  "<<btarray[cur].getcode()<<"书的名称：  "
	<<btarray[cur].getname()<<endl;
	cout<<"请修改数据：  "<<endl;
	cout<<endl<<"书的名称：  ";
	cin>>btarray[cur].getname();
	return;
}
void CBookManager::delbooks()
{
	int cur;
	cur=findbooks();
	if(cur=-1)
	{
		return;
	}
	if(btarray[cur].get_Stock()==false)
	{
		cout<<"该图书已经借出，请归还后再删除。"<<endl;
		char ch;
		cout<<"按任意键继续.....\n";
		cin>>ch;
		return;
	}
	btarray[cur].set_Existing(false);
	return;
}
void CBookManager::listbooks()
{
	char value[6];
	for(int i=0;i<=top;i++)
	{
		if(btarray[i].get_Existing()==true)
		{
			if(btarray[i].get_Stock()==true)
				strcpy(value,"在架");
			if(btarray[i].get_Stock()==false)
				strcpy(value,"借出");
			cout<<"书号：  "<<btarray[i].getcode()<<"图书状态:"
			<<value<<endl;
		}
	}
	char ch;
	cout<<"按任意键继续.....\n";
	cin>>ch;
	return;
}
```

Duzheguali.cpp:

```
#include"Mytype.h"
void readerjiemian();
```

```
void duzhemanage()
{
	char in;
	CReaderManager rm;
	do
	{
		system("cls");
		readerjiemian();
		cin>>in;
		switch(in)
		{
			case '1':
			rm.addreaders();
			break;
			case '2':
			rm.editreaders();
			break;
			case '3':
			rm.delreaders();
			break;
			case '4':
			rm.findreaders();
			break;
			case '5':
			rm.listreaders();
			break;
			case '6':
			break;
			default:
			cout<<"输入错误，请重新输入"<<endl;
		}
	}while(in!='6');
}
```

Guanliyuan.cpp:

```
#include<iostream>
#include<fstream>
using namespace std;
#include"Mytype.h"
void staffjiemian();
void bookmanage();
void duzhemanage();
int superyz()
{
	CUser super;
	int id;
	cout<<"请输入您的编号:";
	cin>>id;
	fstream ff("init.txt",ios::in);
	if(!ff)
	{
```

```
                cout<<"open file error!\n";
                exit(1);
        }
        ff.read((char*)&super,sizeof(CUser));
        ff.close();
        if(id!=super.getcode())
                return 0;
        else
                return 1;
}
void sysmanage()
{
        int f;
        f=superyz();
        if(f)
        {
                char in1;
                do
                {
                        system("cls");
                        staffjiemian();
                        cin>>in1;
                        switch(in1)
                        {
                                case '1':
                                bookmanage();
                                break;
                                case '2':
                                duzhemanage();
                                break;
                                case '3':
                                break;
                                default:
                                cout<<"输入错误，请重新输入。"<<endl;
                        }
                }while(in1!='3');
        }
        else
                cout<<"您不是图书管理员";
}
```

Library.cpp

```
#include<iostream>
#include<cstdlib>
#include"Mytype.h"
#include<fstream>
using namespace std;
void sysmanage();
void jiehuan();
void mainjiemian();
void init()
```

```
{
	int id;
	char n[20];
	cout<<"请输入系统管理员编号：";
	cin>>id;
	cout<<"请输入系统管理员姓名：";
	cin>>n;
	CUser super(id,n);
	fstream ff("init.txt",ios::out);
	if(!ff)
	{
		cout<<"open file error!\n";
		exit(1);
	}
	ff.write((char*)&super,sizeof(CUser));
	ff.close();
	cout<<"系统初始化成功！按任意键继续.....";
	cin.get();
	cin.get();
}
//主函数
int main()
{
	char in;
	do
	{
		system("cls");
		mainjiemian();
		cin>>in;
		switch(in)
		{
			case '1':
			sysmanage();
			break;
			case '2':
			jiehuan();
			break;
			case '3':
			init();
			break;
			case '4':
			break;
			default:
			cout<<"输入错误，请重新输入。"<<endl;
		}
	}while(in!='4');
	return 0;
}
```

Menu.cpp:

```
#include<iostream>
```

```
#include<bits/stdc++.h>
using namespace std;
void mainjiemian()//登录主界面
{
	cout<<"*******欢迎使用图书管理系统*************"<<endl;
	cout<<endl;
	cout<<"*******请选择您的身份  ***********"<<endl;
	cout<<"********1  系统管理员  ***********"<<endl;
	cout<<"********2  图书管理员  ***********"<<endl;
	cout<<"********3  系统初始化  ***********"<<endl;
	cout<<"********4  退出系统    ***********"<<endl;
	cout<<"*********************************************"<<endl;
	cout<<endl;
	cout<<"请选择数字:";
}
void staffjiemian()//管理员界面
{
	cout<<"*********************************************"<<endl;
	cout<<"***********图书管理系统***************"<<endl;
	cout<<"******                   ***********"<<endl;
	cout<<"******                   ***********"<<endl;
	cout<<"******     (1)图书管理    ***********"<<endl;
	cout<<"******                   ***********"<<endl;
	cout<<"******     (2)用户管理    ***********"<<endl;
	cout<<"******                   ***********"<<endl;
	cout<<"******     (3)退出        ***********"<<endl;
	cout<<"******                   ***********"<<endl;
	cout<<"*************************************"<<endl;
}
void bookjiemian()//图书维护子菜单
{
	cout<<"|-----------------------------------|"<<endl;
	cout<<"|           图书维护              |"<<endl;
	cout<<"|          (1) 增加图书           |"<<endl;
	cout<<"|          (2) 更改图书           |"<<endl;
	cout<<"|          (3) 删除图书           |"<<endl;
	cout<<"|          (4) 查找图书           |"<<endl;
	cout<<"|          (5) 显示图书           |"<<endl;
	cout<<"|          (6) 返回上一层         |"<<endl;
	cout<<"|-----------------------------------|"<<endl;
}
void readerjiemian()//读者维护子菜单
{
	cout<<"|-------------------------------------|"<<endl;
	cout<<"|           读者维护              |"<<endl;
	cout<<"|           (1) 增加读者          |"<<endl;
	cout<<"|           (2) 更改读者          |"<<endl;
	cout<<"|           (3) 删除读者          |"<<endl;
	cout<<"|           (4) 查找读者          |"<<endl;
	cout<<"|           (5) 显示读者          |"<<endl;
	cout<<"|           (6) 返回上一层        |"<<endl;
```

```
	cout<<"|--------------------------------------|"<<endl;
}
void yewujiemian()//借还书
{
	cout<<"*******欢迎使用图书管理系统*************"<<endl;
	cout<<endl;
	cout<<"*******请选择您的操作  ***********"<<endl;
	cout<<"********1  借书管理    ***********"<<endl;
	cout<<"********2  还书管理    ***********"<<endl;
	cout<<"********3  返回上一层  ***********"<<endl;
	cout<<"*********************************************"<<endl;
	cout<<endl;
}

Tushuguanli.cpp:
#include<iostream>
//#include<bits/stdc++.h>
#include<fstream>
using namespace std;
#include"Mytype.h"
void bookjiemian();
void bookmanage()
{
	char in;
	CBookManager bm;
  do
   {
	system("cls");
		bookjiemian();
		cin>>in;
	switch(in)
	{
		case '1':
		bm.addbooks();
		break;
		case '2':
		bm.editbooks();
		break;
		case '3':
		bm.delbooks();
		break;
		case '4':
		bm.findbooks();
		break;
		case '5':
		bm.listbooks();
		break;
		case '6':
		break;
		default:
		cout<<"输入错误，请重新输入。"<<endl;
```

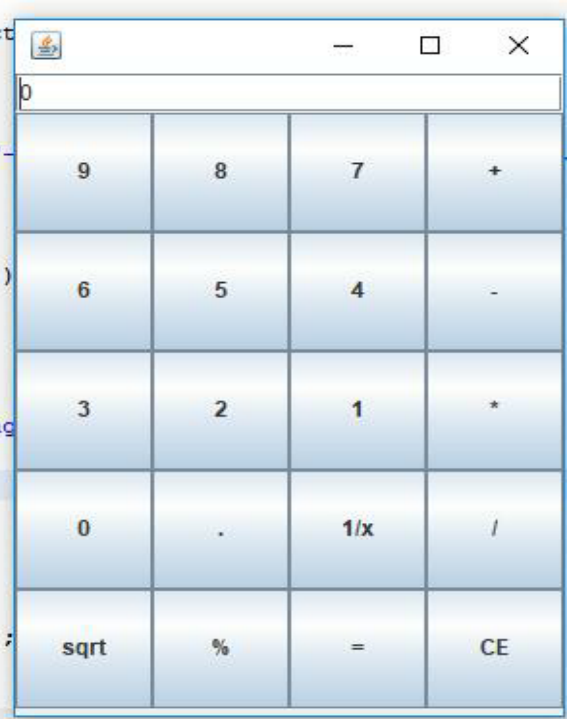
```
            }
  }while(in!='6');
}

Yewu.cpp:
#include<iostream>
#include<fstream>
using namespace std;
#include"Mytype.h"
void yewujiemian();
//借书处理
void borrowbook()
{
            int rsuffix,bsuffix;
            CBookManager t_bd;
            CReaderManager t_rd;
            rsuffix=t_rd.findreaders(0);
            bsuffix=t_bd.findbooks(0);
            char ch;
            if(bsuffix==-1)
            {
                        cout<<"书不存在！\n";
                        cout<<"按任意键继续.....\n";
                        cin>>ch;
                        return;
            }
            if(rsuffix==-1)
            {
                        cout<<"读者不存在";
                        cout<<"按任意键继续.....\n";
                        cin>>ch;
                        return;
            }
            if(t_bd.btarray[bsuffix].get_Stock()==false)
            {
                        cout<<"图书已经借出，请选择其他图书"<<endl;
                        cout<<"按任意键继续.....\n";
                        cin>>ch;
                        return;
            }
            if(t_rd.rdarray[rsuffix].get_Borrowing()==true)
            {
                        cout<<"你已经借过图书，请先归还再借其他图书"<<endl;
                        cout<<"按任意键继续.....\n";
                        cin>>ch;
                        return;
            }
            t_bd.btarray[bsuffix].set_Stock(false);
            t_rd.rdarray[rsuffix].set_Borrowing(true);
            t_rd.rdarray[rsuffix].setbrbook(t_bd.btarray[bsuffix]);
            return;
```

```
}
//还书处理:
void returnmanage()
{
	int rsuffix,bsuffix;
	CBookManager t_bd;
	CReaderManager t_rd;
	rsuffix=t_rd.findreaders();
	bsuffix=t_bd.findbooks();
	char ch;
	if(bsuffix==-1)
	{
		cout<<"书不存在！\n";
		cout<<"按任意键继续.....\n";
		cin>>ch;
		return;
	}
	if(rsuffix==-1)
	{
		cout<<"读者不存在！\n";
		cout<<"按任意键继续.....\n";
		cin>>ch;
		return;
	}
	if(t_rd.rdarray[rsuffix].getbrbook().getcode()==t_bd.btarray[bsuffix].getcode())
	{
		t_bd.btarray[bsuffix].set_Stock(true);
		t_rd.rdarray[rsuffix].set_Borrowing(false);
		CBook cb;
		t_rd.rdarray[rsuffix].setbrbook(cb);
	}
	else
	{
		cout<<"请重新输入，读者借的图书号不正确"<<endl;
		cout<<"按任意键继续.....\n"<<endl;
		cin>>ch;
		return;
	}
	return;
}
void jiehuan()
{
	char in1;
	do
	{
		system("cls");
		yewujiemian();
		cin>>in1;
		switch(in1)
		{
			case '1':
```

```
                    borrowbook();
                    break;
                    case '2':
                    returnmanage();
                    break;
                    case '3':
                    break;
                    default:
                    cout<<"输入错误，请重新输入。"<<endl;
                }
        }while(in1!='3');
}
```

Java 版计算器：

```
mport java.util.*;
ublic class Bao16 extends JFrame implements Act

    double number = 0.0;
    String operator = "=";
    String str[]={"9","8","7","+","6","5","4","-
            "sqrt","%","=","CE"};
    boolean isFirstDigit = true;
    JPanel panel=new JPanel();
    JPanel panel2=new JPanel(new BorderLayout())
    JTextField display = new JTextField("0");
    public Bao16()
    {
        int digit=0;
        JButton[] buttons = new JButton[str.leng
        this.setSize(300,400);
        this.setLocation(400, 200);
        panel2.add(display);
        panel.setLayout(new GridLayout(5,4));
         for (int i = 0; i < str.length; i++)
         {
               buttons[i] = new JButton(str[i]);
               panel.add(buttons[i]);
         }
```




```
import java.util.*;
import java.awt.*;
import java.awt.event.*;
import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JPanel;
import javax.swing.JTextField;
public class Bao16 extends JFrame implements ActionListener
{
        double number = 0.0;
        String operator = "=";
        String str[]={"9","8","7","+","6","5","4","-","3","2","1","*","0",".","1/x","/",
                      "sqrt","%","=","CE"};
        boolean isFirstDigit = true;
        JPanel panel=new JPanel();
        JPanel panel2=new JPanel(new BorderLayout());
        JTextField display = new JTextField("0");
        public Bao16()
        {
                int digit=0;
```

```
                JButton[] buttons = new JButton[str.length];
                this.setSize(300,400);
                this.setLocation(400, 200);
                panel2.add(display);
                panel.setLayout(new GridLayout(5,4));
                 for (int i = 0; i < str.length; i++)
                 {
          buttons[i] = new JButton(str[i]);
          panel.add(buttons[i]);
                 }
                 this.getContentPane().setLayout(new BorderLayout());
    this.getContentPane().add("North", panel2);
    this.getContentPane().add("Center", panel);
                 this.setVisible(true);

                 for (int i = 0; i < str.length; i++)
        buttons[i].addActionListener(this);
                  display.addActionListener(this);
}
public static void main(String[] args)
{
               Bao16 bao=new Bao16();
}
public void actionPerformed(ActionEvent e)
{
                  String target=e.getActionCommand();
      String label = e.getActionCommand();
      if (target.equals("CE"))
         handleReset();
      else if ("0123456789.".indexOf(label) > 0)
         handleNumber(label);
      else
         handleOperator(label);
}
private void handleOperator(String key) {

               if (operator.equals("+"))
          number += Double.valueOf(display.getText());
      else if (operator.equals("-"))
         number -= Double.valueOf(display.getText());
      else if (operator.equals("*"))
         number *= Double.valueOf(display.getText());
      else if (operator.equals("/"))
         number /= Double.valueOf(display.getText());
      else if (operator.equals("="))
         number = Double.valueOf(display.getText());
      else if(operator.equals("1/x"))
                  number=1/Double.valueOf(display.getText());
      else if(operator.equals("sqrt"))
                  number=Math.sqrt(Double.valueOf(display.getText()));
      else if(operator.equals("%"))
                  number%=Double.valueOf(display.getText());
      display.setText(String.valueOf(number));
      operator = key;
      isFirstDigit = true;
}
private void handleNumber(String key) {
                if (isFirstDigit)
          display.setText(key);
```

```
            else if ((key.equals(".")) && (display.getText().indexOf(".") < 0))
                display.setText(display.getText() + ".");
            else if (!key.equals("."))
                display.setText(display.getText() + key);
            isFirstDigit = false;
        }
        private void handleReset() {
                    display.setText("0");
            isFirstDigit = true;
            operator = "=";
        }
}
```