



华南理工大学
South China University of Technology

课程设计报告书

题目：高级语言程序设计

学 院 电子与信息学院

专 业 电子科学与技术

学生姓名 陈艺荣

学生学号 201530301043

指导教师 张见威

课程编号 135160

课程学分 1.0

起始日期 2016/02/29-2016/06/22

教师评语	<div>教师签名：</div> <div>日期：</div>
成绩评定	
备注	

卓越学生信息管理系统

一、选题背景

当今社会，中小學生参加课外辅导的人数日益增加，补习班五花八门地涌现出来，但是很多课外辅导机构都不能很好地对学生的信息进行管理，同时在管理学生成绩方面，也不能向普通中小学那样，有自己的教务系统，这些都给课外辅导机构的老师带来不少麻烦。本课题主要针对补习班的特点，解决学生的基本信息管理、成绩管理等相关问题，为众多的辅导机构提供一个管理学生、帮助统计成绩的程序，使辅导机构更好地管理教学，进行教学评价，以及用最少的投入获得最高的回报。本课题主要应用 C++ 语言、简单的 MFC 控件，通过面向对象的编程思想，来实现各个功能。

二、方案论证(设计理念)

本程序的设计理念是：服务老师，提高课外辅导的教学管理效果。
基于此理念，我设计了以下实现方案：

- 1、通过 MFC 实现 Win32 窗体应用程序，从而提供友好的界面；
- 2、使用冒泡排序思想实现对成绩的排序；
- 3、使用文本文件对学生信息进行保存^[1]，从而使数据能完整地保存；
- 4、设置登录界面使得教师能够较为安全地保管自己的学生信息；

基于上述方案，对该程序的功能设计如下：

- 1、显示功能：输入的信息会在列表视图框即时显示出来，各种操作也会即时显示出来；
- 2、登录功能：使用帐号和密码进行登录，才能进行其他操作，其中密码输入提供掩码保护密码；
- 3、输入功能：可以输入学生信息，添加到信息库中；
- 4、查询功能：可以进行多种查询，包括输入单一信息、多个信息进行查询；
- 5、删除功能：输入学生的姓名或学号或电话号码进行定位删除；
- 6、排序功能：提供多种排序机制，可以选择 4 不同项目进行排序，可以选择升序排序或降序排序；
- 7、帮助功能：主菜单提供帮助版块，包括使用说明、问题反馈、程序介绍；

- 8、将信息导出到文件功能^[1]：可以通过主菜单的“导出文件”实现将显示
- 9、将信息从文件导出功能^[1]：可以通过主菜单的“导入文件”实现将文件中的信息导入到列表视图框；
- 10、使用教程功能；在登录前会通过“帮助”按钮提供程序的使用教程；
- 11、错误提示功能；对各种操作错误均提供弹窗进行温馨提醒；
- 12、统计功能；可以统计学生人数，男女个数，各科成绩的平均分，及格率，优秀率，方便老师进行成绩分析；
- 13、退出程序功能：提供多种方式退出程序；
- 14、最小化最大化功能。

以上方案，主要特点为：界面友好，功能齐全，容易使用；

三、过程论述

各个版块功能实现：

1、登录功能：

登录功能的设计如图 1 所示，

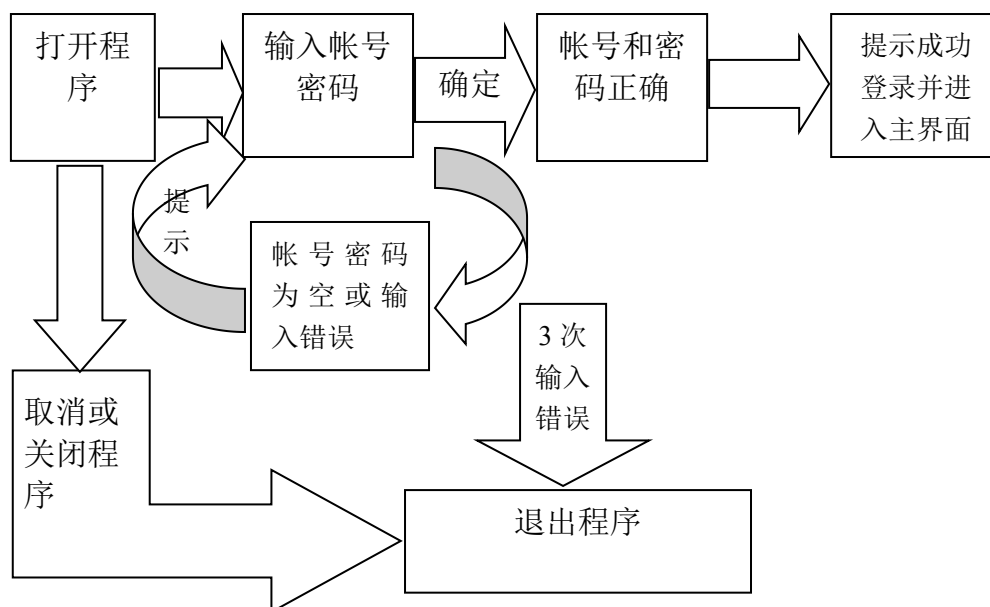


图 1 登录功能实现流程图

登录窗口通过确定按键函数 `void dengluDlg::OnConfirm()`、取消按键函数 `void dengluDlg::OnCancel()`、帮助按键函数 `void dengluDlg::OnLhelp()` 实现流程图中的

功能。

2、输入功能：

输入功能主要通过将 CEdit 类^[2] 编辑框与定义类成员绑定在一起，以及将 CButton 按钮^[2] 与类成员函数绑定在一起，流程见图 2，

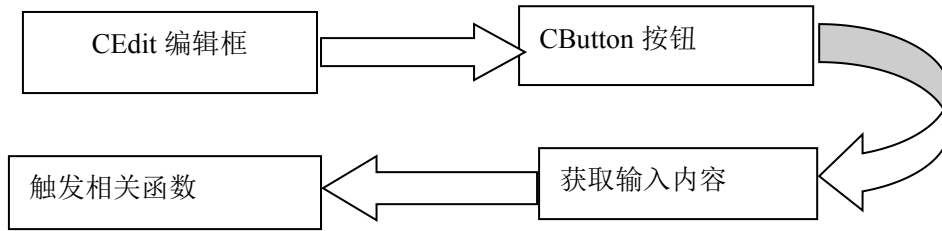


图 2 输入功能流程图

利用函数 `DDX_Text(pDX, IDC_EDIT_ZHANGHAO, m_strZhanghao)`^[2]、`DDX_Control(pDX, IDC_EDIT_ZHANGHAO, m_ZHANGHAO)`^[2] 实现 CEdit 编辑框与数据成员绑定，`ON_BN_CLICKED(ID_OK, OnConfirm)`^[2] 实现 CButton 按钮与相关函数绑定，利用 `m_ZHANGHAO.GetWindowText(m_strZhanghao)`^[2] 实现获取 CEdit 编辑框的内容。

3、显示功能：

显示功能包括：对学生信息进行显示和对操作结果进行显示。

其中，在本系统中，对学生信息进行显示是通过 CListCtrl 类列表视图控件^[2] 来实现的，通过以下函数设置列表的各列名称：

```
m_ListControl.InsertColumn(0, "学号", LVCFMT_LEFT, 106)[2];  
m_ListControl.InsertColumn(1, "姓名", LVCFMT_LEFT, 106);  
m_ListControl.InsertColumn(2, "性别", LVCFMT_LEFT, 106);  
m_ListControl.InsertColumn(3, "班别", LVCFMT_LEFT, 106);  
m_ListControl.InsertColumn(4, "电话", LVCFMT_LEFT, 106);  
m_ListControl.InsertColumn(5, "语文", LVCFMT_LEFT, 106);  
m_ListControl.InsertColumn(6, "数学", LVCFMT_LEFT, 106);  
m_ListControl.InsertColumn(7, "英语", LVCFMT_LEFT, 106);  
m_ListControl.InsertColumn(8, "总分", LVCFMT_LEFT, 106);
```

通过以下函数实现一行行地将信息在列表视图框中显示出来：

```
m_ListControl.InsertItem(0, m_id)[2]  
m_ListControl.SetItemText(0, 1, m_name)[2]
```

对操作结果进行显示则是利用 `MessageBox("卓越温馨提示：")` 函数实现。

4、查询功能：

查询功能包括：按学号查询、按姓名查询或者按电话号码查询，该部分功能的实现主要是利用列表视图框的相应函数进行实现，具体流程见图 3，

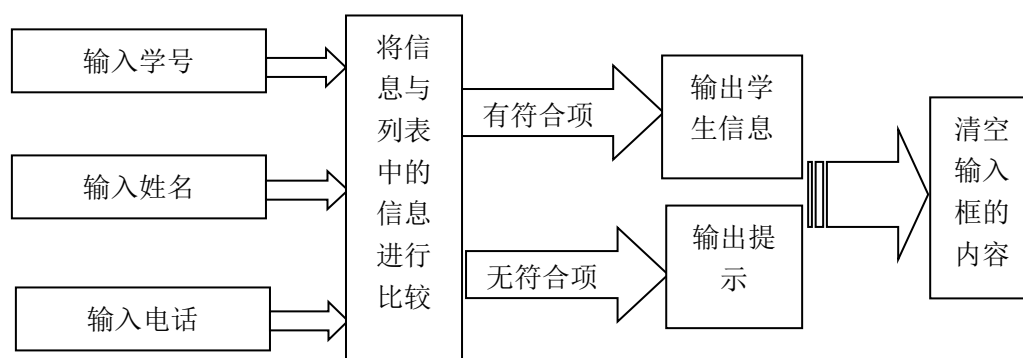


图 3 查询功能流程图

查询功能其实已经包含输入功能，其中尤为重要的是如何快速地将输入的内容与列表中的信息进行比较，在这里利用了 CListCtrl 类列表视图控件的函数：

```
m_ListControl.GetItemText(i, 0) [2]
```

实现快速比较。

5、删除功能：

该部分功能的实现与查询功能是相似的，只是在查询功能的基础上对信息进行删除，因此，其流程可以描述为图 4 所示的流程，

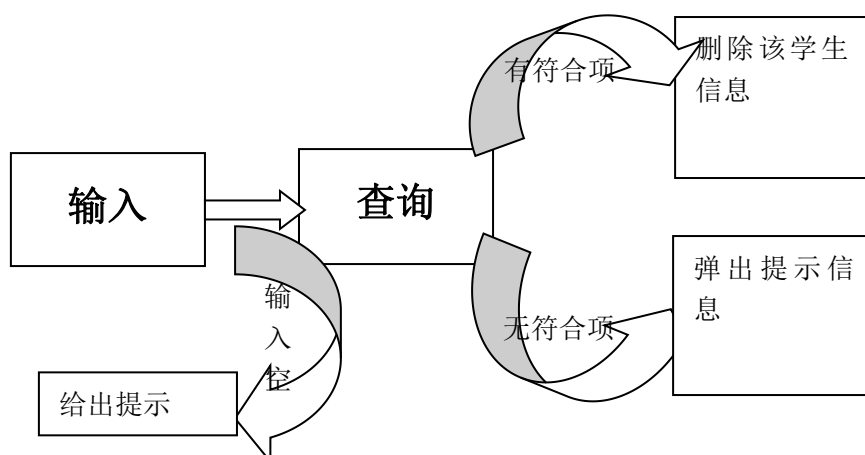


图 4 删除功能流程图

删除操作的快速实现也是借助于 CListCtrl 类列表视图控件的函数：

```
m_ListControl.DeleteItem(i)[2];
```

该函数实现对列表视图框的指定行进行删除操作。

6、排序功能：

本部分功能包括：对语文成绩升序或降序排序，对数学成绩升序或降序排序，对英语成绩升序或降序排序，对总分升序或降序排序。其中，考虑到辅导社的学生总数不是很大，同时，为了让使用本系统的老师方便观察各科成绩的情况，例如单科尖子的总分排名，本系统的排序均采用冒泡排序算法^[3]。

冒泡排序的示例代码^[3]如下：

```
//按语文成绩升序排序
```

```
void CStudentDlg::OnSort_chinese_shengxu()
```

```
{
```

```
    CString StrText1, StrText2;
```

```
    int num, i, flag, j;
```

```
    int score1, score2;
```

```
    num=0; //计数器[4]
```

```
    i=0;
```

while(m_ListControl.GetItemText(i, 5)!="") //当列表视图的语文列当前行非空时，执行代码^[2]

```
{
```

```
    num++; //计算总行数，其中 num=总行数+1，注意：最后一行为空！
```

```
    i++;
```

```
}
```

struct Item ListItem[MAXSTUDENT]; //最多保存 1000 个学生的信息，可以通过头文件 Resource.h 修改最大值

```
for(i=0; i<=num; i++)
```

```
{ //获取列视图框的第 i+1 行内容
```

```
    ListItem[i].StrID=m_ListControl.GetItemText(i, 0)[2];
```

```
    ListItem[i].StrName=m_ListControl.GetItemText(i, 1);
```

```
    ListItem[i].StrSex=m_ListControl.GetItemText(i, 2);
```

```
    ListItem[i].StrClass=m_ListControl.GetItemText(i, 3);
```

```
    ListItem[i].StrPhone=m_ListControl.GetItemText(i, 4);
```

```

        ListItem[i].StrChinese=m_ListControl.GetItemText(i, 5);
        ListItem[i].StrMaths=m_ListControl.GetItemText(i, 6);
        ListItem[i].StrEnglish=m_ListControl.GetItemText(i, 7);
        ListItem[i].StrScore=m_ListControl.GetItemText(i, 8);
    }
    for(i=num-1;i>=0;i--)
    {
        UpdateData(TRUE)[2];
        m_ListControl.DeleteItem(i);
        UpdateData(FALSE);
    }
    for(i=0;i<num-1;i++)
    {
        flag=1;
        for(j=0;j<num-i-1;j++)
        {
            StrText1=ListItem[j].StrChinese;
            score1=_tcstoul(StrText1, NULL, 10); //将 Cstring 转换为整数
            StrText2=ListItem[j+1].StrChinese;
            score2=_tcstoul(StrText2, NULL, 10);

            if(score1<score2) //当第 i 行的语文成绩小于第 i+1 行的语文成绩
时，交换两行的信息[3]
            {
                struct Item TempItem; //临时变量 TempItem 作为交换两行信息
的中间体[3]

                TempItem.StrID=ListItem[j].StrID;
                TempItem.StrName=ListItem[j].StrName;
                TempItem.StrSex=ListItem[j].StrSex;
                TempItem.StrClass=ListItem[j].StrClass;
                TempItem.StrPhone=ListItem[j].StrPhone;
                TempItem.StrChinese=ListItem[j].StrChinese;

```



```

TempItem.StrMaths=ListItem[j].StrMaths;
TempItem.StrEnglish=ListItem[j].StrEnglish;
TempItem.StrScore=ListItem[j].StrScore;

ListItem[j].StrID=ListItem[j+1].StrID;
ListItem[j].StrName=ListItem[j+1].StrName;
ListItem[j].StrSex=ListItem[j+1].StrSex;
ListItem[j].StrClass=ListItem[j+1].StrClass;
ListItem[j].StrPhone=ListItem[j+1].StrPhone;
ListItem[j].StrChinese=ListItem[j+1].StrChinese;
ListItem[j].StrMaths=ListItem[j+1].StrMaths;
ListItem[j].StrEnglish=ListItem[j+1].StrEnglish;
ListItem[j].StrScore=ListItem[j+1].StrScore;

ListItem[j+1].StrID=TempItem.StrID;
ListItem[j+1].StrName=TempItem.StrName;
ListItem[j+1].StrSex=TempItem.StrSex;
ListItem[j+1].StrClass=TempItem.StrClass;
ListItem[j+1].StrPhone=TempItem.StrPhone;
ListItem[j+1].StrChinese=TempItem.StrChinese;
ListItem[j+1].StrMaths=TempItem.StrMaths;
ListItem[j+1].StrEnglish=TempItem.StrEnglish;
ListItem[j+1].StrScore=TempItem.StrScore;
flag=0;
    }
}
if(flag==1)
    break;
}
for(i=0;i<num;i++)
{
    m_ListControl.InsertItem(0,ListItem[i].StrID); //插入节点函数
    m_ListControl.SetItemText(0,1,ListItem[i].StrName);

```

```

        m_ListControl.SetItemText(0, 2, ListItem[i].StrSex);
        m_ListControl.SetItemText(0, 3, ListItem[i].StrClass);
        m_ListControl.SetItemText(0, 4, ListItem[i].StrPhone);
        m_ListControl.SetItemText(0, 5, ListItem[i].StrChinese);
        m_ListControl.SetItemText(0, 6, ListItem[i].StrMaths);
        m_ListControl.SetItemText(0, 7, ListItem[i].StrEnglish);
        m_ListControl.SetItemText(0, 8, ListItem[i].StrScore);
    }

    MessageBox("卓越温馨提示：已经按语文成绩升序排序完毕！"); //输出排
序完毕提示[5]
}

```

运用上述代码，就可以实现图 5 所示的排序流程：

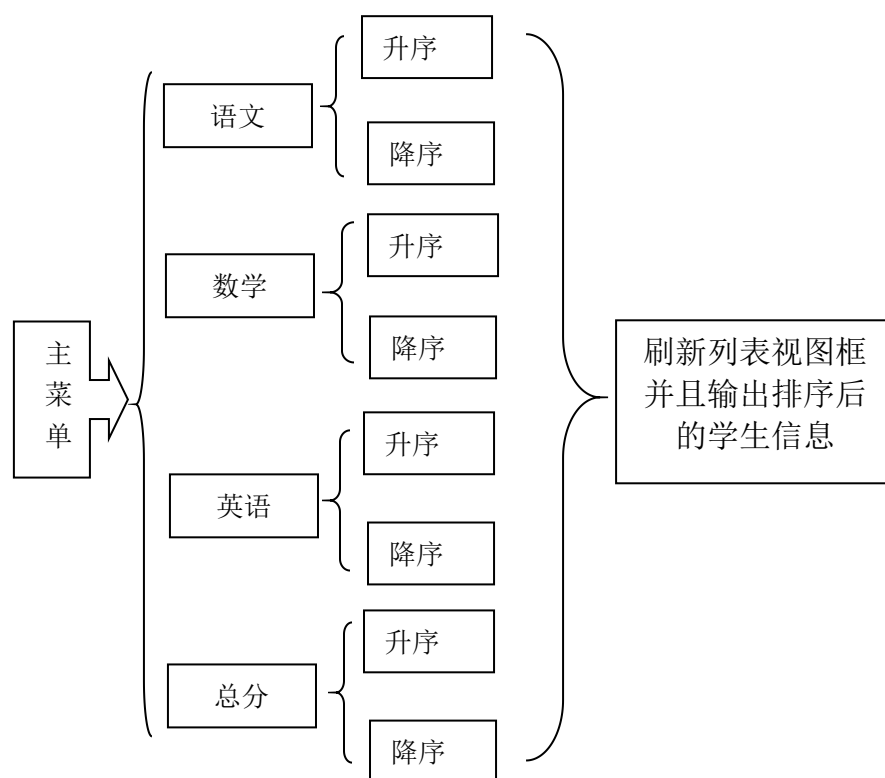


图 5 排序功能流程图

7、帮助功能

帮助功能包括 3 个版块：关于我们、使用说明和联系我们，对该部分功能的实现

均通过 MessageBox(“要提示的内容”)函数^[5]实现，当用户点击相应菜单按键时，就弹出相应的弹窗^[5]。对该部分的设计实现见图 6：



图 6 帮助功能设计图

8、将信息导出到文件功能

该功能是基于 C++ 的文件流^[4]实现的，实现本功能的函数如下：

//获取输入的信息并且将它保存在 C 盘中，用 ios::out 方式实现^[4]

```
void CStudentDlg::OnMenuItemDispaly()
{
    // TODO: Add your command handler code here
    //获取输入的信息并且将它保存在 C 盘中
    ofstream fp("C:\\studentxinxi.txt", ios::out); // 信息保存在 C 盘的
student.txt 文件中
    CString
StrID, StrName, StrSex, StrClass, StrPhone, StrChinese, StrMaths, StrEnglish, StrS
core;

    int i=0;
    for(i=0; m_ListControl.GetItemText(i, 0) != ""; i++) //当指针未有到达列表
视图的尾部时，就进行信息提取，将信息写进指定文本文件
    {
        StrID=m_ListControl.GetItemText(i, 0);
        StrName=m_ListControl.GetItemText(i, 1);
        StrSex=m_ListControl.GetItemText(i, 2);
        StrClass=m_ListControl.GetItemText(i, 3);
        StrPhone=m_ListControl.GetItemText(i, 4);
        StrChinese=m_ListControl.GetItemText(i, 5);
        StrMaths=m_ListControl.GetItemText(i, 6);
        StrEnglish=m_ListControl.GetItemText(i, 7);
        StrScore=m_ListControl.GetItemText(i, 8);
```

```

        fp<<StrID<<"    "<<StrName<<"    "<<StrSex<<"    "<<StrClass<<"
"<<StrPhone<<"    "<<StrChinese<<"    "<<StrMaths<<"    "<<StrEnglish<<"
"<<StrScore<<endl;
    }
    fp.close();
    MessageBox("卓越温馨提示：学生信息已保存至 C:\\studentxinxi.txt");
//提示学生信息保存完毕
}

```

值得注意的是，在 void CStudentDlg::OnMenuItemDispaly() 函数中，将学生信息输出到文件的方式是 ios::out 在打开文本文件时删除原来的数据而不是追加方式^[4]，这样的设计是为了和将信息导出到文件功能相照应，避免重复地导入学生信息。因此使用该功能前注意要先将原文本文件的学生信息导入到系统中。当然，使用 ios::out 也可以让用户在初次使用该软件的时候建立新的文本文件^[4]，而不会因此产生错误。

9、将信息从文件导出功能

因为整个系统在输出和输入时都是通过 CString 类对象传递数据，因此，本功能的实现与将信息导出到文件功能的实现是不同的。

实现函数代码：

```

void CStudentDlg::OnMenuItemDaoru()
{
    char
StrID1[30], StrName1[30], StrSex1[30], StrClass1[30], StrPhone1[30], StrChinese
1[30], StrMaths1[30], StrEnglish1[30], StrScore1[30];
    FILE* stream;
    stream=fopen("C:\\studentxinxi.txt", "r");
    int i=0;
    int num=0;
    struct Item ListItem[MAXSTUDENT];
        while(!feof(stream))
        {
            fscanf(stream, "%s", StrID1);
            fscanf(stream, "%s", StrName1);
            fscanf(stream, "%s", StrSex1);

```

```

fscanf(stream, "%s", StrClass1);
fscanf(stream, "%s", StrPhone1);
fscanf(stream, "%s", StrChinese1);
fscanf(stream, "%s", StrMaths1);
fscanf(stream, "%s", StrEnglish1);
fscanf(stream, "%s", StrScore1);
ListItem[i].StrID.Format("%s", StrID1);
ListItem[i].StrName.Format("%s", StrName1);
ListItem[i].StrSex.Format("%s", StrSex1);
ListItem[i].StrClass.Format("%s", StrClass1);
ListItem[i].StrPhone.Format("%s", StrPhone1);
ListItem[i].StrChinese.Format("%s", StrChinese1);
ListItem[i].StrMaths.Format("%s", StrMaths1);
ListItem[i].StrEnglish.Format("%s", StrEnglish1);
ListItem[i].StrScore.Format("%s", StrScore1);
i++;
num++; }
fclose(stream);
for(int i=0;i<num-1;i++)
{
    m_ListControl.InsertItem(0, ListItem[i].StrID);
    m_ListControl.SetItemText(0, 1, ListItem[i].StrName);
    m_ListControl.SetItemText(0, 2, ListItem[i].StrSex);
    m_ListControl.SetItemText(0, 3, ListItem[i].StrClass);
    m_ListControl.SetItemText(0, 4, ListItem[i].StrPhone);
    m_ListControl.SetItemText(0, 5, ListItem[i].StrChinese);
    m_ListControl.SetItemText(0, 6, ListItem[i].StrMaths);
    m_ListControl.SetItemText(0, 7, ListItem[i].StrEnglish);
    m_ListControl.SetItemText(0, 8, ListItem[i].StrScore);
}
MessageBox("卓越温馨提示： 已经成功从 C:\\studentxinxi.txt 导入学生信
息!");
}

```

在该函数中使用了两个较为关键的函数

`fscanf(stream, "%s", StrID1)`、`ListItem[i].StrID.Format("%s", StrID1);`

这两个函数用来实现信息从文件流输入到临时字符串变量，并将格式转为 Cstring 类保存到 `Item ListItem[MAXSTUDENT]` 结构数组中。

10、统计功能：

统计功能包括：对总人数统计，对男女人数统计，对各科的平均分、及格率、优秀率进行统计。流程图参考图 7。

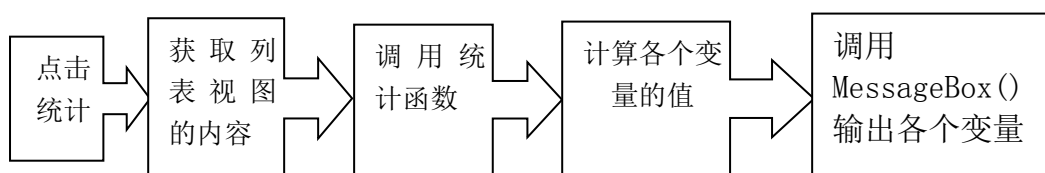


图 7 统计功能流程图

11、总功能模块：

对显示功能、登录功能、输入功能、查询功能、删除功能、排序功能、帮助功能、将信息导出到文件功能、将信息从文件导出功能、使用教程功能、错误提示功能、统计功能、退出程序功能、最小化最大化功能的联系指向流程图见图 8。

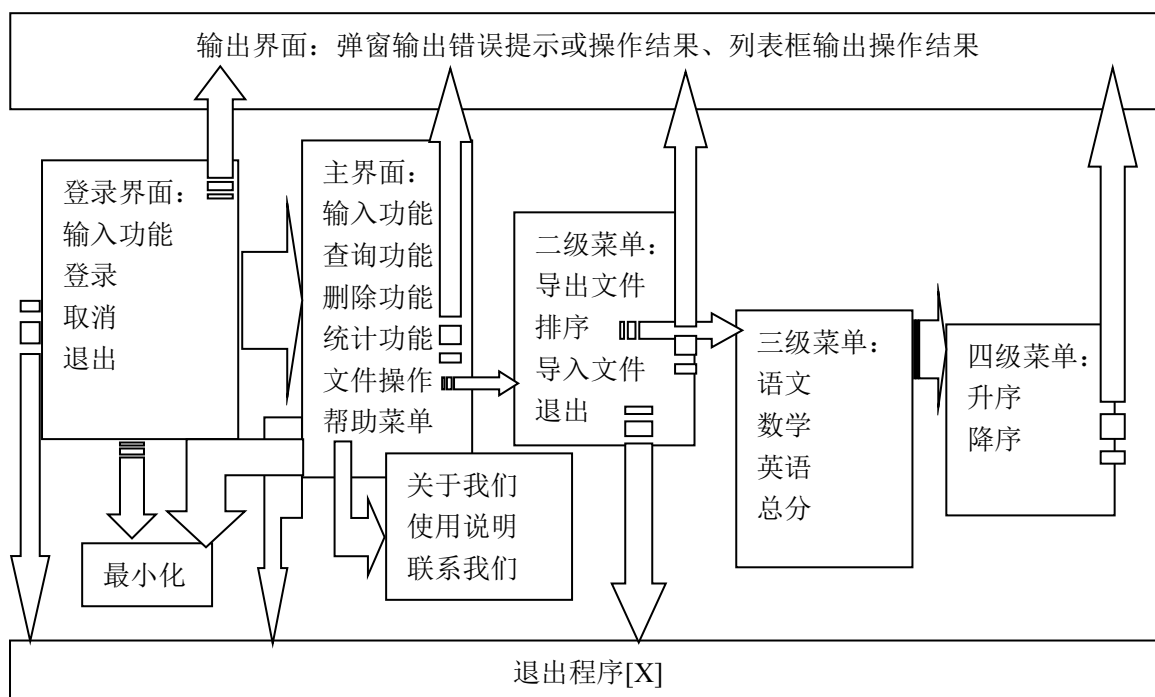


图 8 总功能流程图

通过图 8 可以看到，该系统包括两个窗口：登录窗口、主界面窗口，系统的界面设置比较友好，基本不会出现系统操作错误，功能之间相互联系。

四、结果分析

基于以上设计，卓越学生信息管理系统各个功能的效果图如下：

打开程序时，弹出登录欢迎界面，如图 9 所示



图 9 登录界面效果图



图 10 登录界面输入信息效果图

在登录界面单击“帮助”按键，弹出使用方法，见图 11

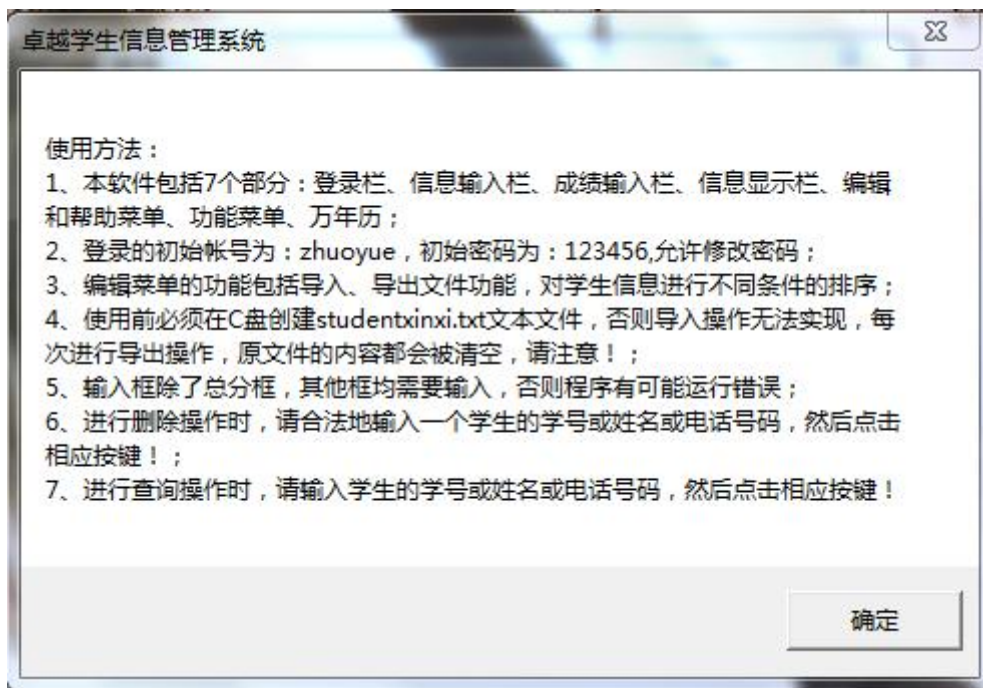


图 11 登录界面点击帮助键效果图

当输入帐号或密码错误时，弹出错误警告，见图 12



图 12 登录界面输入错误警告提示效果图

当连续 3 次登录失败时，系统会弹出警告，并且关闭程序，见图 13

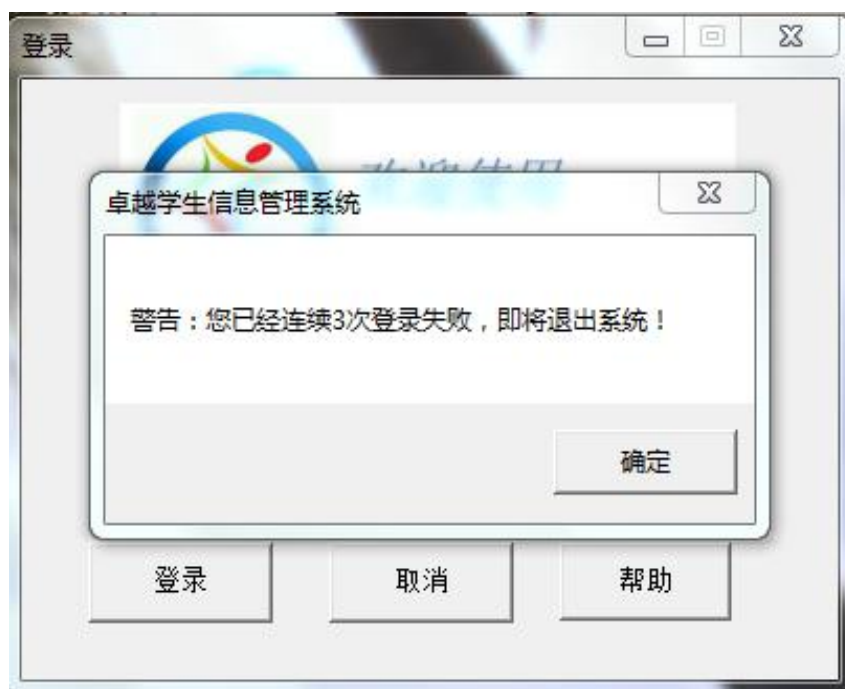


图 13 连续 3 次输入密码错误提示效果图

当用户输入的帐号和密码均正确时，系统会弹出登录成功说明窗口，并且在用户点击“确定”按键后，系统关闭登录窗口，显示主界面，见图 14 和图 15

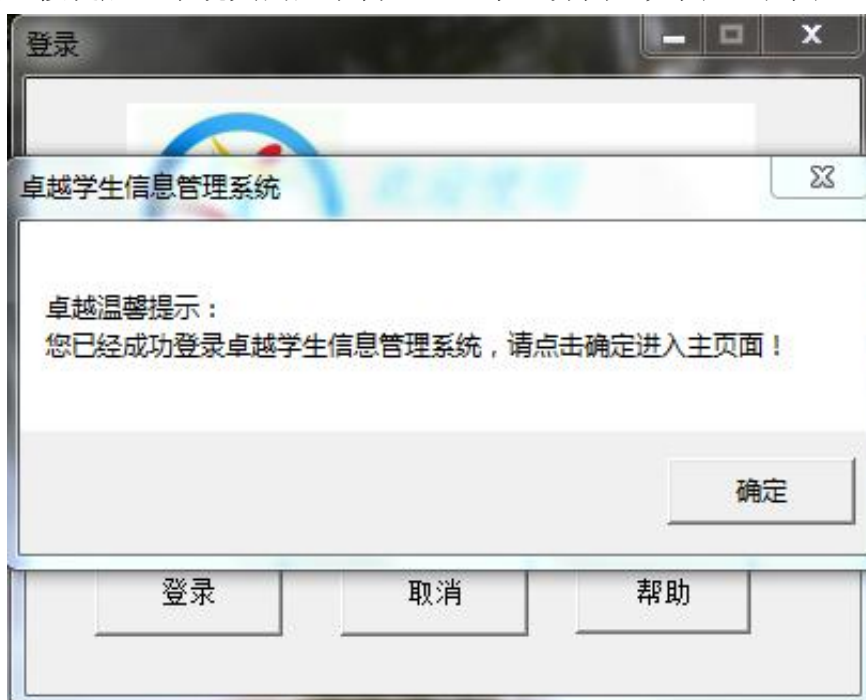


图 14 登录成功提示窗口效果图



图 15 系统的主界面效果图

点击“文件操作”，选择“导入文件”系统实现将学生信息从外部文件导入到列表视图框中，并且弹出成功导入文件的提示窗口，见图 16

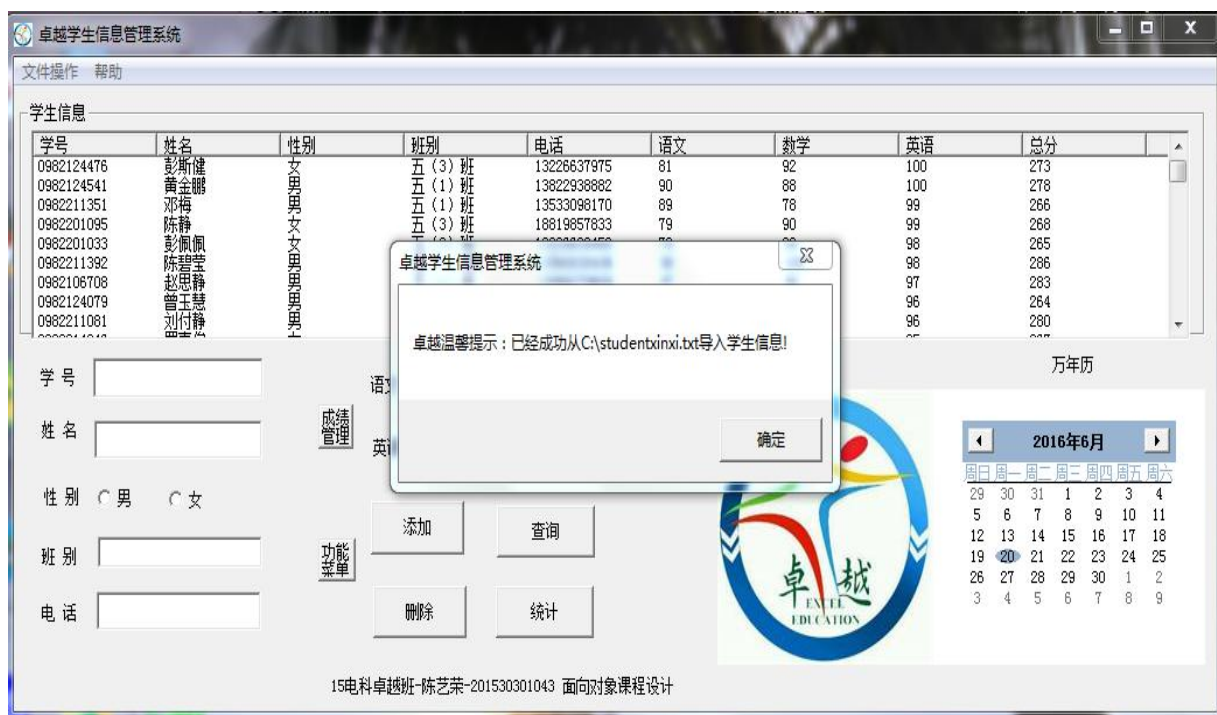


图 16 导入文件实现效果图

在界面的编辑框可以输入新生的相关信息，见图 17



图 17 信息输入效果图

当输入的信息有遗漏时，系统会弹出错误提示，见图 18

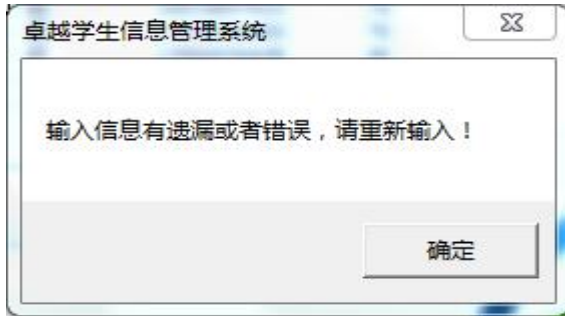


图 18 错误提示效果图

当合法地输入信息时，系统将该学生信息导入到列表视图框中，并弹出成功录入信息的提示，见图 19



图 19 成功学生添加信息提示

当用户在编辑框输入学生的学号或姓名或电话号码，点击删除按钮，系统会删除相关学生的所有信息并且刷新列表，弹出成功删除的提示框，见图 20

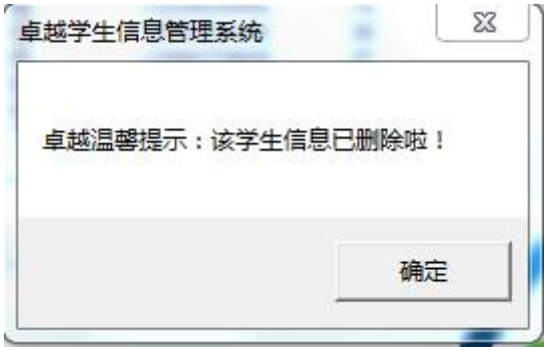


图 20 删除学生信息提示图

当用户右键单击系统主界面左上方的图标并在子菜单中选择关于卓越学生信息管理系统时，系统弹出相关说明，见图 21

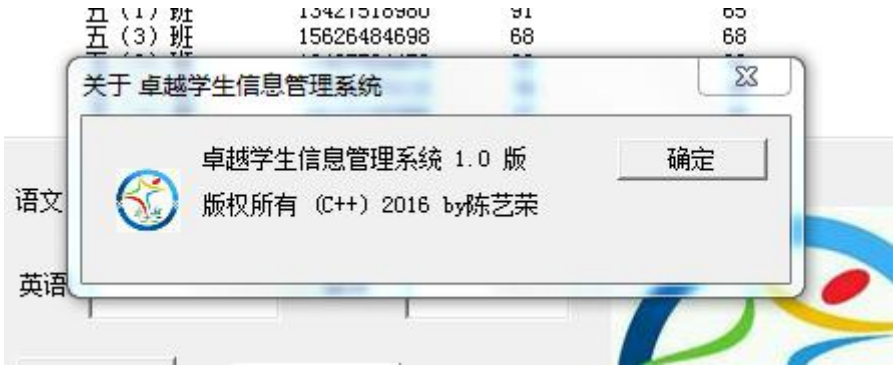


图 21 关于卓越学生信息管理系统效果图

当用户选择文件操作的排序时，会有语文、数学、英语、总分四个子菜单，而在这四个子菜单中，均有升序、降序两种选择，单击语文->降序，语文->升序，实现相关排序，效果图见图 22、图 23、图 24



图 22 主菜单设计图

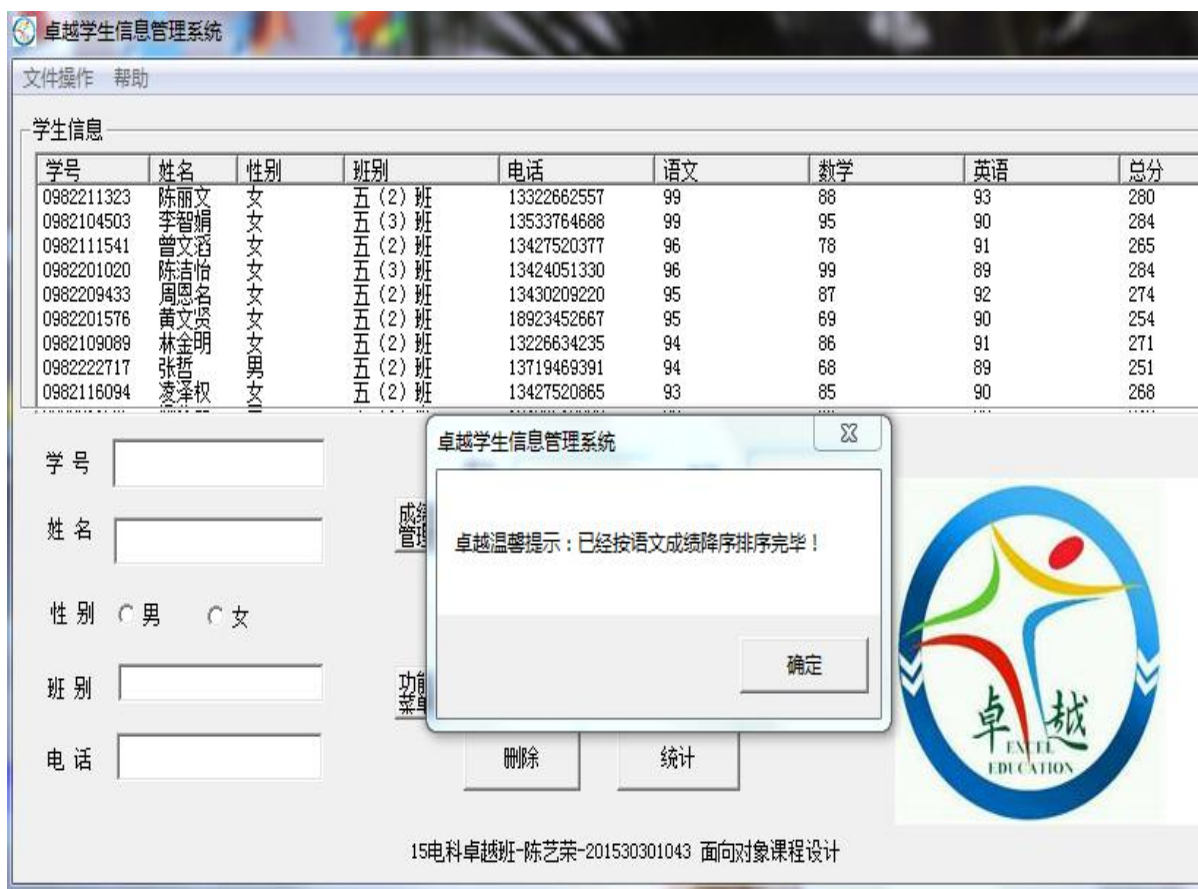


图 23 按语文成绩降序排序效果图

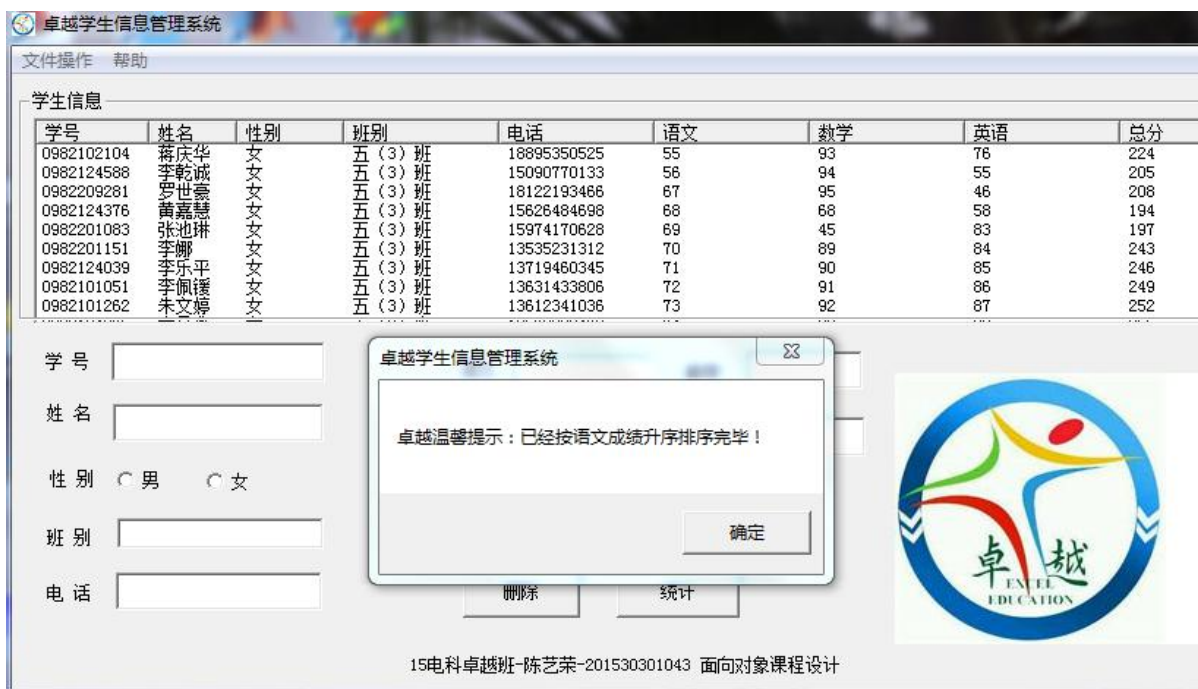


图 24 按语文成绩升序排序效果图

当用户选择主菜单右上方的最小化按钮[一]时，系统将最小化，见图 25



图 25 最小化效果图

当然，卓越学生信息管理系统还有不少个性化功能，例如万年历，个性化的卓越图标，参见图 26 和图 27



图 26 万年历效果图



图 27 个性化的.ico 程序图标

通过上述程序实现的结果，可以看出，本卓越学生信息管理系统已经将各个功能实现了，同时，由于本系统在设计时采用了面向对象方法来实现各个功能，因此，程序具有良好的扩展性，比如：可以添加科目、可以按学号、姓氏查找，可以进一步升级，增加教师管理，而这只需要增加一个教师类或结构以及子窗口即可，可以增加课程管理等等。

五、课程设计总结

我的课程设计历时一个月，其中，准备阶段将近 25 天，打代码、调试时间将近一周。因为在定位课程设计时我就打算将自己的系统实现界面化，而不是简单的黑框，因此，就要自学有关界面设计的实现方法。在选择方法时，有 QT（一个跨平台的 C++ 图形用户界面库）和 MFC(Microsoft Foundation Classes)，最终我选择了 MFC，一方面，MFC 可以直接通过 VS2010 或者 Visual C++6.0 实现，而不需要另外下载软件，这样就省去了适应一款新软件的使用的时间；另一方面，使用 MFC，本身就能很好地促进我对面向对象编程思想的理解，因为对微软基本类库的使用确实使我感受到面向对象编程的方便。

在学习 MFC 的基本控件的使用过程当中，我去图书馆借了两本书，分别为《Visual C++全能速查宝典》和《VC++.NET（2008）课程设计经典案例》。不得不说，这两本书对我在入门的过程中帮助很大。很多时候，通过百度虽然能找到不少资料，但是，不得不说，资料参差不齐，有时候甚至有可能导致自己误入歧途。而通过实体书本，则能很好地掌握相关知识，同时，也不会学错方向。当然，书本也有不足之处，我去图书馆找了半天，也没有找到基于 VS2010 的 MFC 学习资料，最后只好找到基于 Visual C++6.0 的资料书学习相应的控件类以及类成员函数的使用。同时，我在资料书上看不明白的函数使用等，通过在 MSDN 论坛查找，都能找到满意的答案，不得不说，MSDN 对学习编程是十分有帮助的，特别是想了解 MFC 里面某个类的使用，比如 CString 类、CEdit 类等等。

当然，在课程设计的过程中，让我感触最大的还是“实践”，即使花了 20 多天去了解 MFC，学习控件，然而，真正让我会基本的 MFC 的使用，还是开始打代码的那几天，从添加控件到设置控件个性化 ID 方便记忆，到修改控件的属性实现某些功能，以及使用类成员函数实现内容的获取与输出、将按键与相应函数关联等等。

其中，对我帮助特别大的函数有：

```
DDX_Text(pDX, IDC_EDIT_ID, m_id); //指定 IDC_EDIT_ID 控件的输入流向 m_id
```

```

DDX_Radio(pDX, IDC_RADIO_M, m_Sex);
DDX_Control(pDX, IDC_EDIT_ID, m_ID); //指定 IDC_EDIT_ID 控件的对象为 m_ID
ON_BN_CLICKED(IDC_BUTTON_ADD, OnAdd) //指定 IDC_BUTTON_ADD 控件触发 OnAdd
ON_COMMAND(ID_MENUITEM_DAORU, OnMenuitemDaoru) //指定 ID_MENUITEM_DAORU
控件触发 OnMenuitemDaoru
m_ListControl.InsertColumn(0, "学号", LVCFMT_LEFT, 106); //添加列
UpdateData(TRUE);
m_NAME.GetWindowText(m_name); //获取值
m_ListControl.InsertItem(0, m_id); //添加行
m_ListControl.SetItemText(0, 1, m_name); //对某行添加内容
m_score.Format("%d", mmzf); //格式化转换
MessageBox("卓越温馨提示：该学生信息已录入啦！"); //信息弹窗
m_ListControl.GetItemText(i, 0) //获取某行某列内容
m_ListControl.DeleteItem(i); //删除行
fscanf(stream, "%s", StrID1) //格式化获取

```

当然，在学习的过程中，我还接触到大量其他类成员函数，在学习这些函数的过程中，我深有体会的就是一定要熟悉参数的类型，同时也要知道控件获取的内容的类型，因为它们和控制台程序的标准输入输出流有很大的不同，不可以获取任意类型的参数，这一点如果没重视，在调试程序的时候会有很多错误。

同时，让我十分开心的是，自己可以坚持下去。在编写代码过程中，我一度想放弃，回到那种黑框界面就好，因为离答辩的时间不到几天，而且碰到的困难和错误很多，甚至连将文本编辑框的输入与变量绑定都未学会，身边也没有人能指导。我甚至已经开始打控制台应用程序的代码，可以说是已经放弃 MFC 实现界面这个想法了。然而我的舍友看到我设计的界面后，一直鼓励我，向我说当我在这个界面实现功能时，就比别人学多很多东西，同时这这也是一个很不错的经历。看着美丽却没有自己代码的界面，我决定继续坚持下去，周六周日都沉浸在打代码中，连续 5 天熬夜，终于突破瓶颈，实现了界面里的各个功能。因此，我想说，坚持在课程设计中尤为重要。我相信，坚持、不放弃这种精神，对以后参加各种竞赛、工作都是非常有帮助的。

参考文献

- [1] 周霭如, 林伟健. C++程序设计基础 (第4版) (上) [M]. 北京: 电子工业出版社, 2012.
- [2] 明日科技, 曹飞飞等. Visual C++全能速查宝典 [M]. 北京: 人民邮电出版社, 2012: 87-542.
- [3] 率辉. 2014版数据结构高分笔记 (第2版) [M]. 北京: 机械工业出版社, 2013: 219-258.
- [4] Stanley B. Lippman 等著, 李师贤等译. C++Primer 中文版 (第4版) [M]. 北京: 人民邮电出版社, 2010: 676-702.
- [5] 杜青等. VC++.NET (2008) 课程设计经典案例: 基于 C++/CLI [M]. 北京: 清华大学出版社, 2012: 147-210.