

目录

第一章 需求分析.....	2
1.1 问题描述.....	2
1.2 功能需求.....	2
1.3 开发环境.....	3
1.4 开发过程.....	3
第二章 概要设计.....	3
2.1 总体设计.....	3
2.2 类的定义.....	4
2.3 运行界面设计.....	5
第三章 详细设计.....	6
第四章 测试分析.....	8
4.1 测试程序执行情况.....	8
第五章 课程设计总结.....	27
附录：程序代码.....	28

第一章 需求分析

1.1 问题描述

大学的学生信息管理系统以数据为核心，为大学生提供了从入学到毕业全流程管理服务。

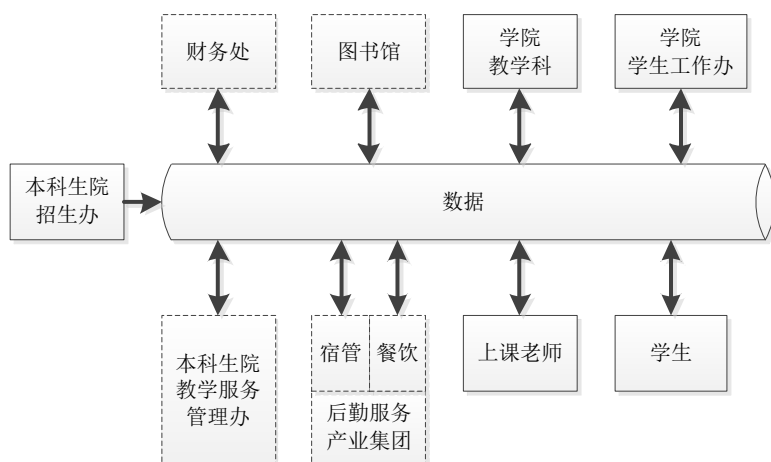


图 1 学生信息管理系统

如图 1 所示，学校中与学生相关的单位和老师包括：

- 本科生院招生办，负责高考招生录取、特殊类型招生选拔和第二学士学位招生录取工作；创建已录取学生的基本信息，包括：学生姓名、性别、身份证号、籍贯、高考成绩、录取学院与专业、学号等；
- 教学服务管理办公室：负责学生学籍管理、转专业与大类专业分流、学业警示等日常教学运转工作；负责辅修学士学位、第二学士学位、双学士学位教学运行管理；
- 财务处，学杂费收缴和奖贷金发放，日常提供饭费充值服务；
- 后勤服务产业集团，负责宿舍管理（分配/收回宿舍、日常进出管理）和餐饮服务（刷卡消费）等；
- 图书馆，为学生提供图书借阅服务；
- 学院教学科，学生选课、成绩管理、专业排名等服务；
- 教师，上课老师成绩登记；
- 学院学生工作办公室，奖学金评定、毕业生管理、推荐免试研究生；

1.2 功能需求

1. 人员管理：系统支持的人员可分为两种，学生与职工，每人可以维护自己的基本信息，但无法修改其操作权限；超级用户可以创建职工并为职工分配操作权限；
2. 安全管理：所有执行操作的人员，均需进行身份认证（用户名+密码）；
3. 权限管理：每个用户均有指定的操作权限，比如教学科老师可以查看学院所有学生的成绩，对同年级、

同专业的学生进行成绩排名，但不具有修改学生成绩的权限；教师可以录入所承担课程的学生成绩，但不可以查看学生其他课程的成绩；

4. 成绩管理：管理学生 4 年、8 个学期的成绩，每个学期最多 6 门课；

a) 教师输入成绩；

b) 学生查自己的成绩；

5. 奖学金评定：可以按学年对同一个专业学生成绩进行排序；

6. 推荐免试研究生

7. 毕业生管理

1.3 开发环境

Dev C++。

1.4 开发过程

第一天 选择课程设计题目，分析课题的要求

第二天 编程

第三天 编程及调试

第四天 写课程设计报告

第五天 提交课程设计报告

第二章 概要设计

2.1 总体设计

其主要功能模块有：

(1) 键盘输入

(2) 屏幕输出

(3) 文件读取

(4) 文件写入

(5) 统计

(6) 排序

(7) 编辑

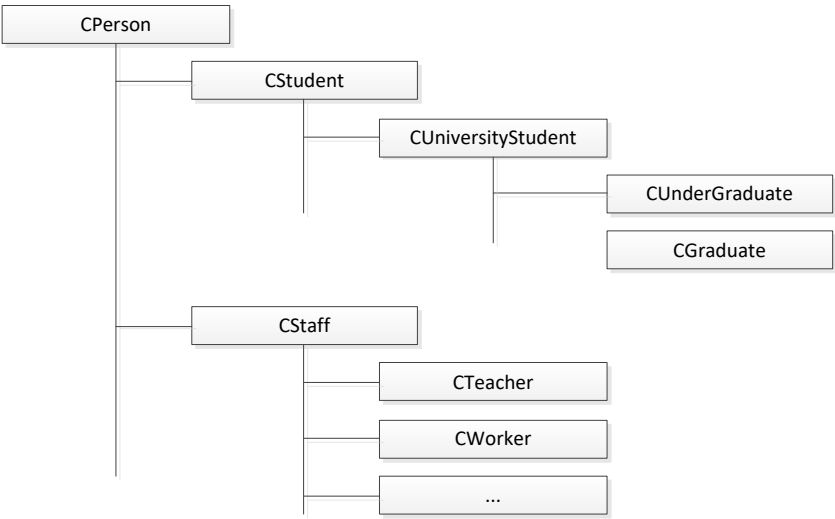
(8) 删除

各功能模块的调用应通过菜单进行选取。

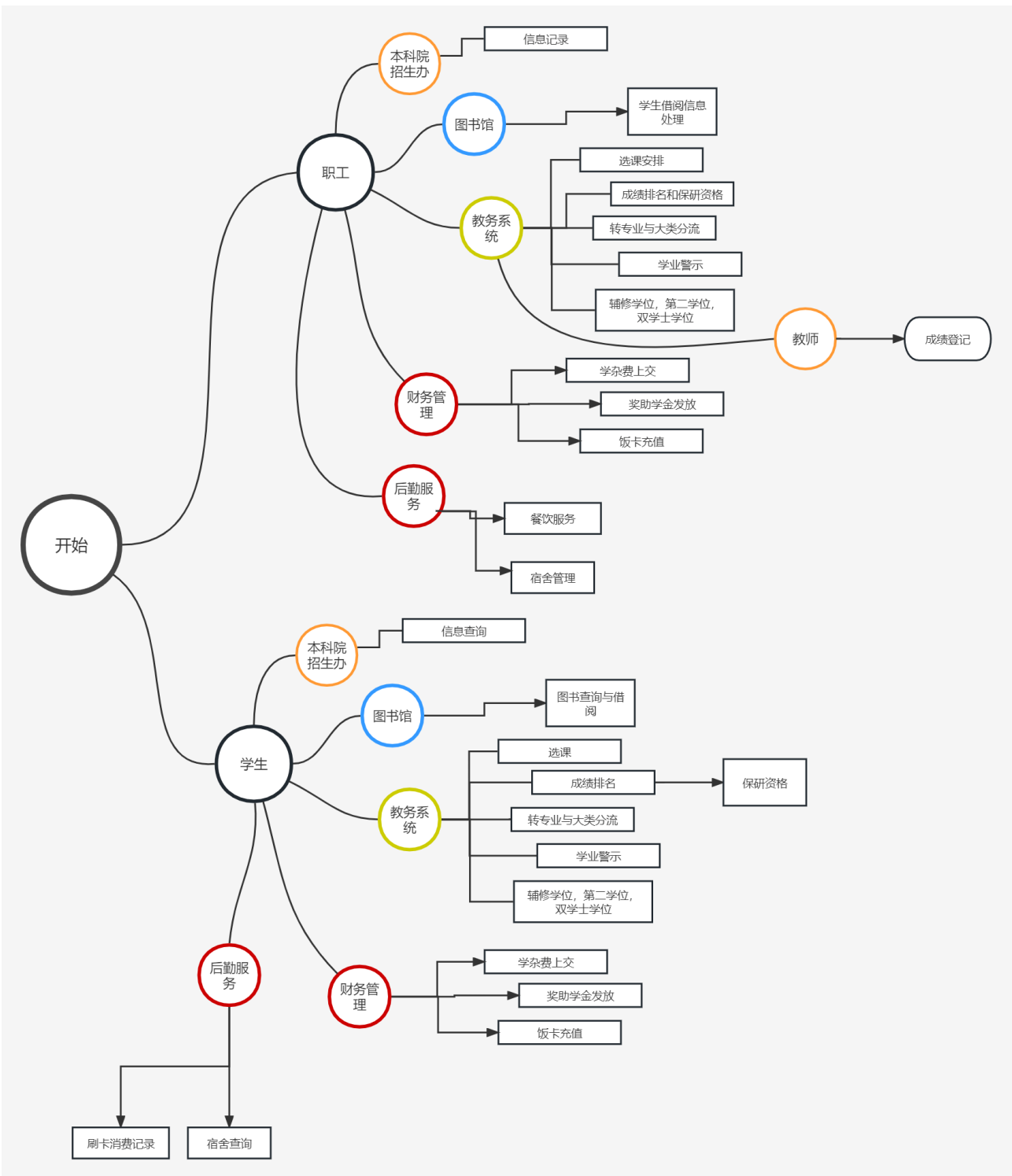
2.2 类的定义

按照图中所示来进行人员类编写

职工按照工作属性来分类



2.3 运行界面设计



第三章 详细设计

以人物类为例

```
//身份抽象类
class Identity
{
public:

    //操作菜单
    virtual void operMenu() = 0;

    string m_Name; //用户名
    string m_Pwd;  //密码
};
```

进行子类的操作

```
class Teacher :public Identity
{
public:

    //默认构造
    Teacher();

    //有参构造 (职工编号, 姓名, 密码)
    Teacher(int empId, string name, string pwd);

    //菜单界面
    virtual void operMenu();

    //本科院招生办信息记录
    void Zhaosheng();
    //查看学生信息
    void showComputer();

    //图书馆书籍管理
    void Shuji();

    //教务系统
    void Jiaowu();
    void chengjisystem();
    void Chengji() ;
    void xuankesystem();
    void xuanke();
```

```

//财务管理修改
void Caiwu();

//后勤服务保障
void Houqin();

int m_EmpId; //教师编号

void initVector();

vector<ComputerRoom> vCom;

void initVector2();

vector<Jiangxuejin> vJia;

};

```

对于各个成员需要登录的模块，采用大部分为函数少部分为新的类的方式进行构造

对于拥有较多函数的模块，进行分离，先声明，在实现

```

#pragma once
#include<iostream>
#include <string>
#include <fstream>

using namespace std;
class primary {
private:
    int number;
    string name;
    int age;
    string sex;
    int class_info;

public:
    void addstu(int nums, primary stu[101],int &index);
    void search_by_name(string name, primary stu[101],int index);
    void search_by_num(int num, primary stu[101],int index);
    void delete_stu(string name, primary stu[101],int &index);
    float CHI_average(primary stu[101],int index);
    float MATH_average(primary stu[101], int index);
    float ENG_average(primary stu[101],int index);

```

```

void total_sorting(primary stu[101],int index);
void CHI_sorting(primary stu[101],int index);
void MATH_sorting(primary stu[101],int index);
void ENG_sorting(primary stu[101],int index);
void output_list(primary stu[101],int amount);
void get_name();
void show_all(primary stu[101],int index);
void change_stu1(string name, primary stu[101], int index);
void change_stu2(int target, primary Stu[101], int index);
};

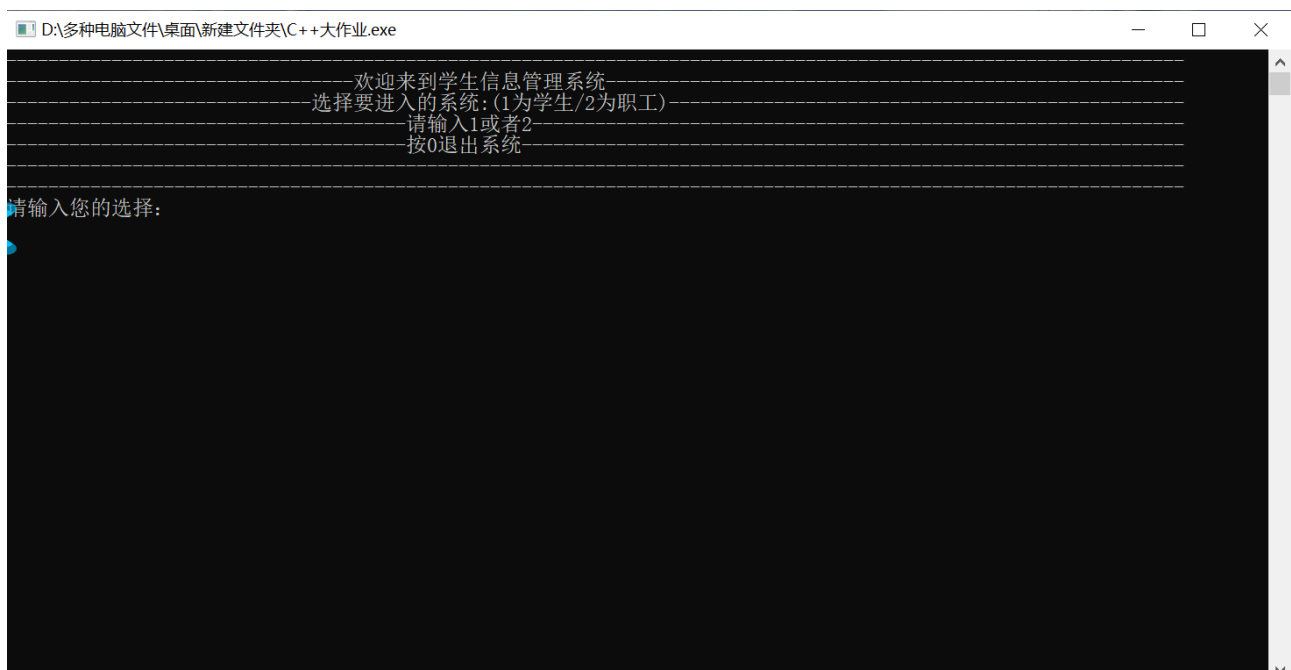
```

总体而言，以类为对象进行分析

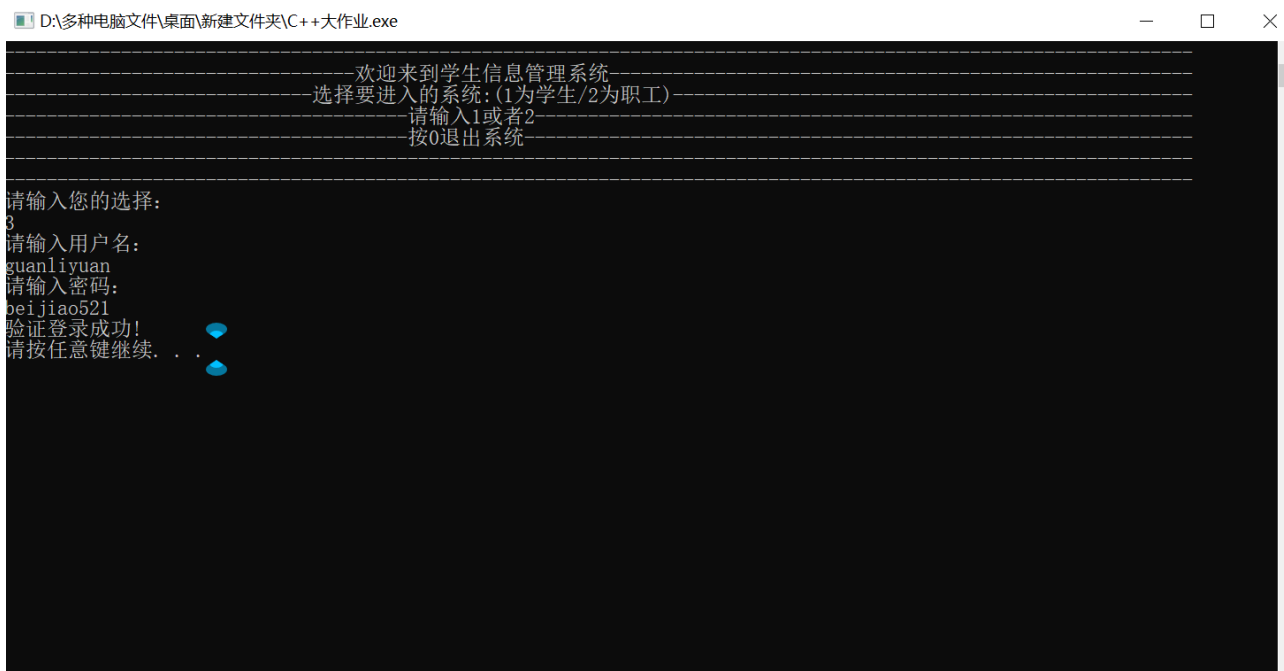
第四章 测试分析

4.1 测试程序执行情况

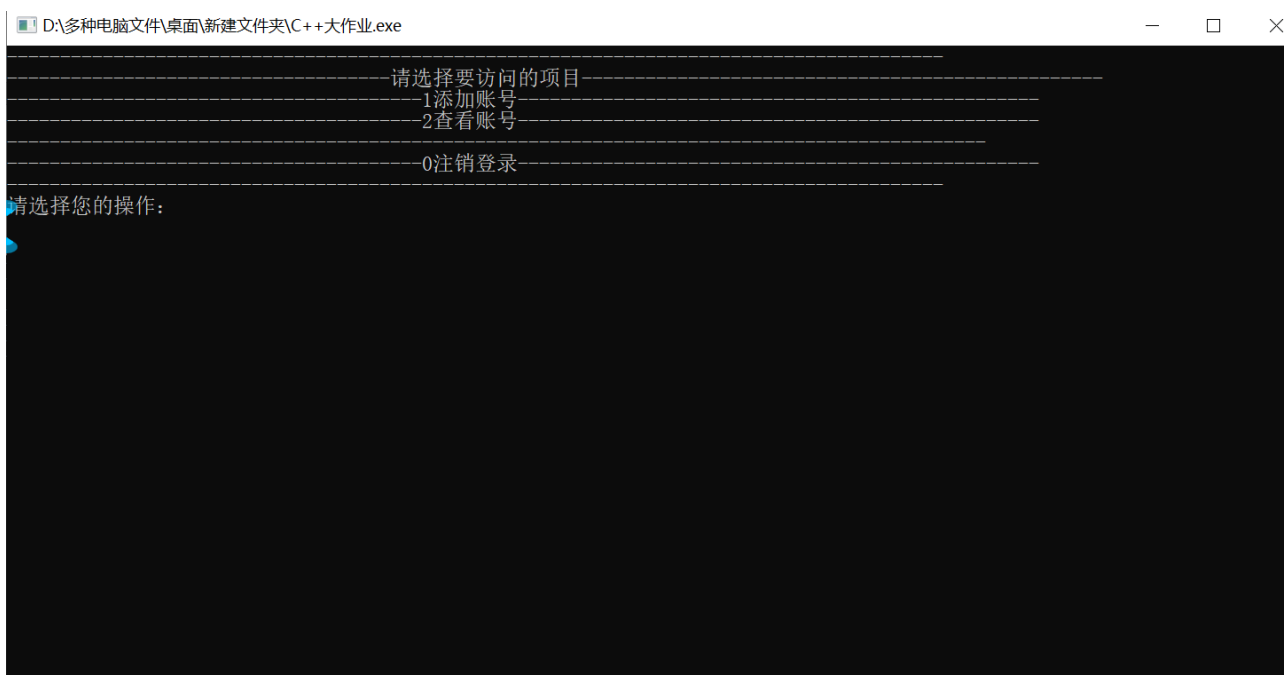
精简的首页



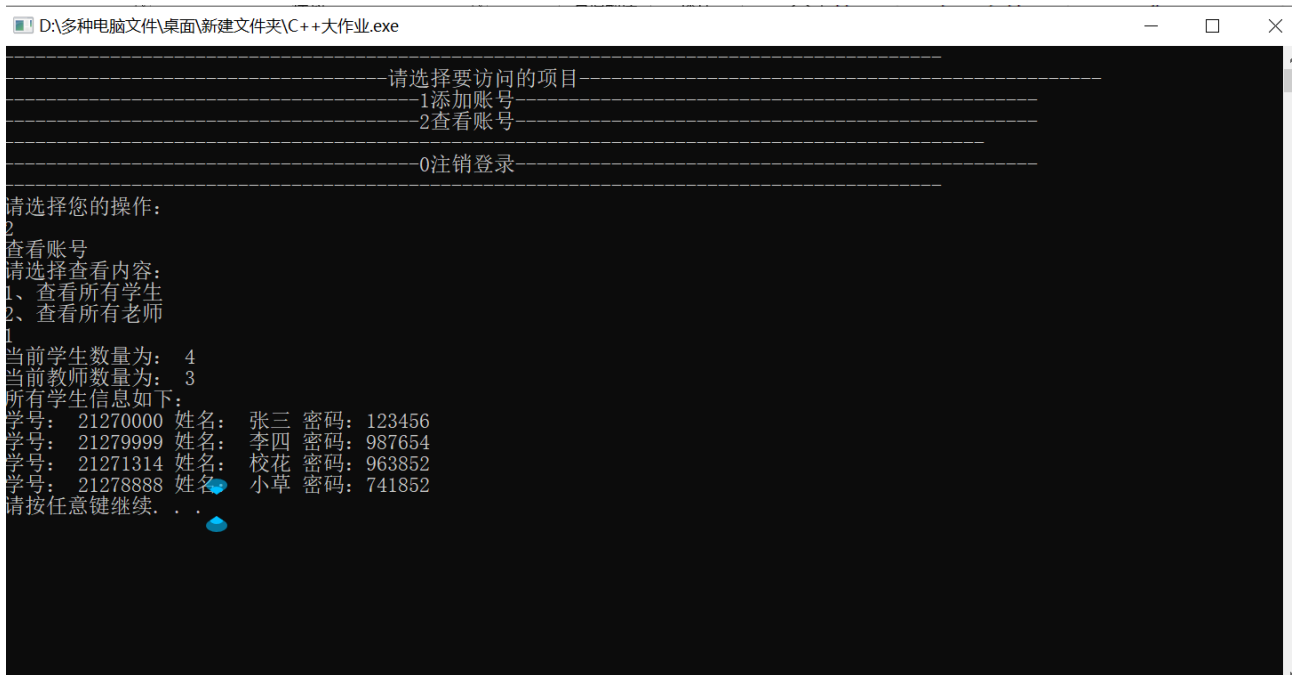
隐秘的管理员登录



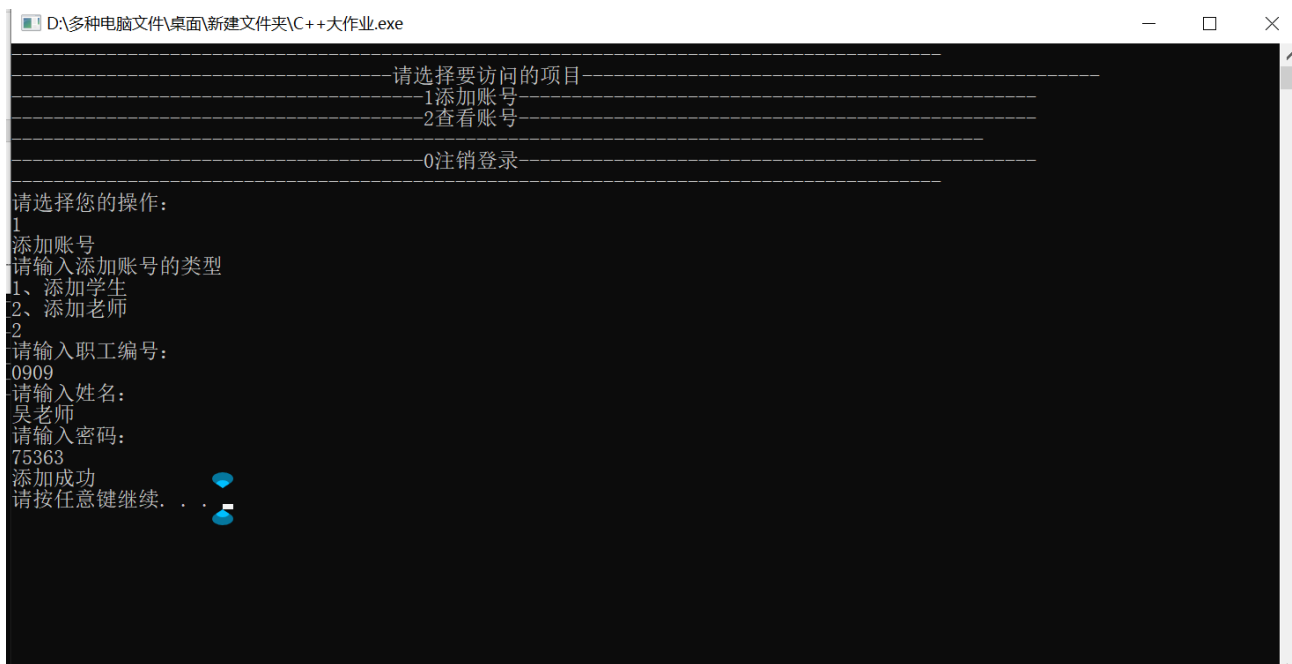
管理员可以进行学生账号老师账号的增加与修改



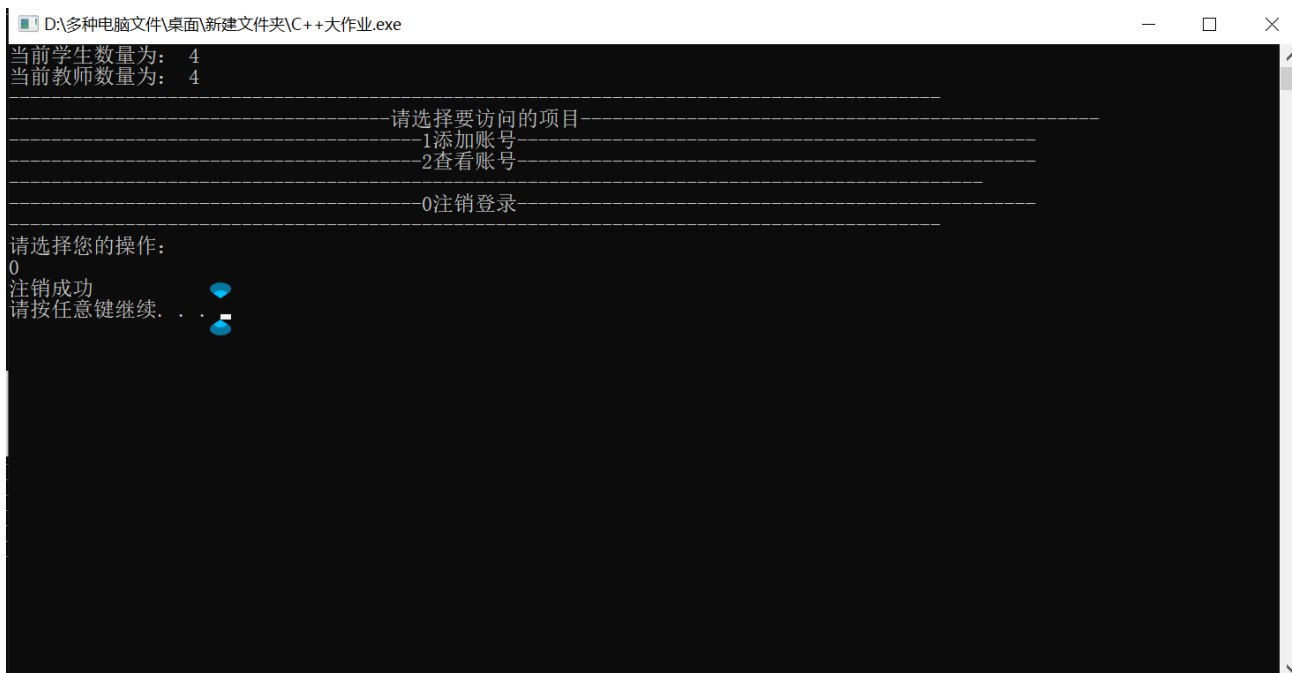
查看账号



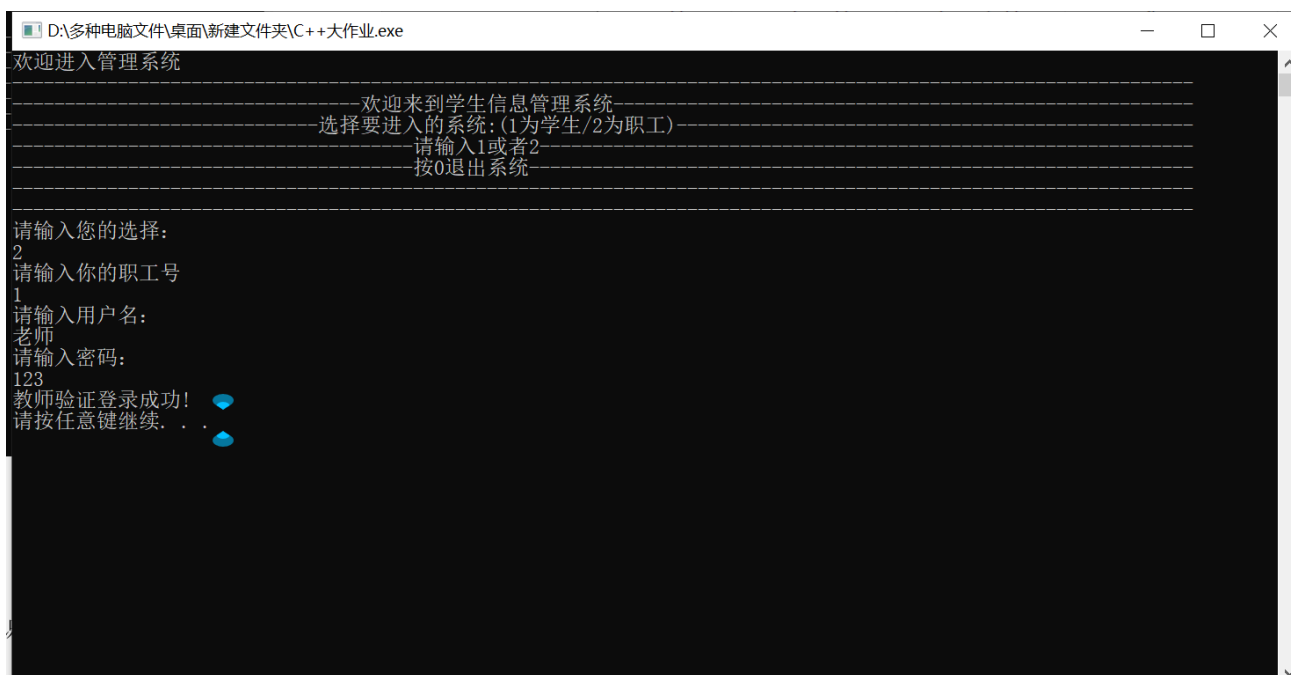
增添老师或学生的账号



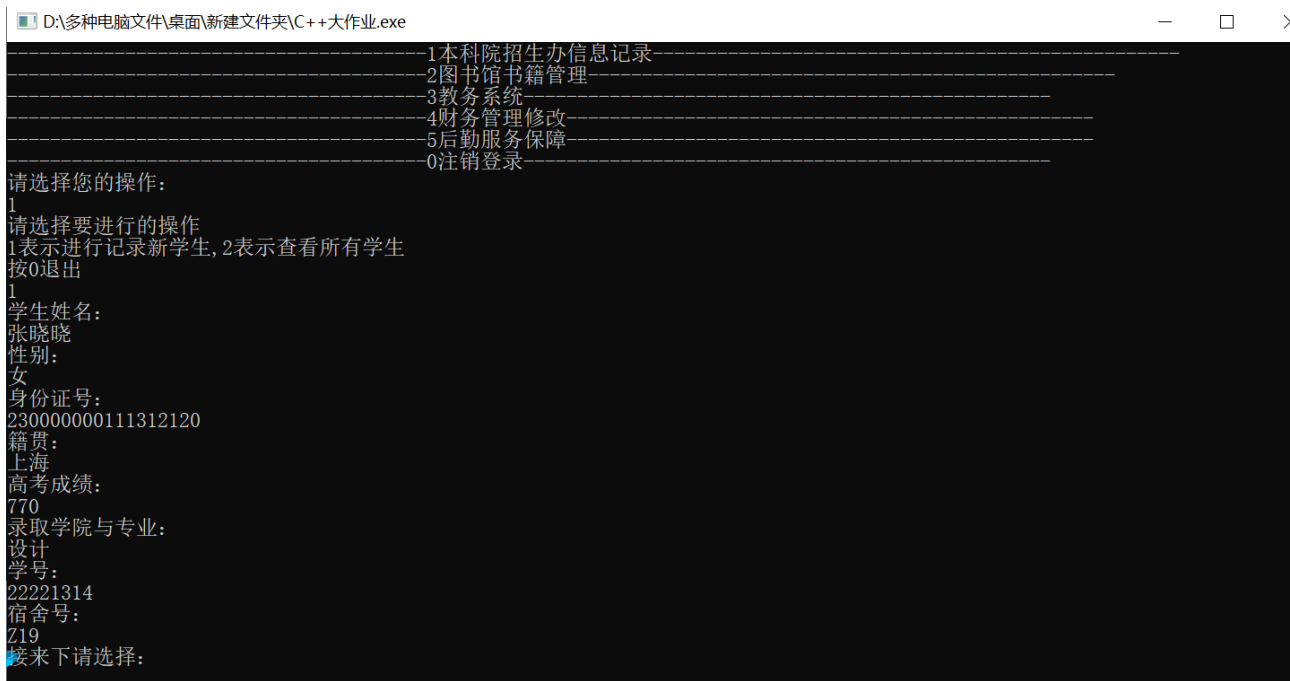
左上角有当前老师与学生账号数量



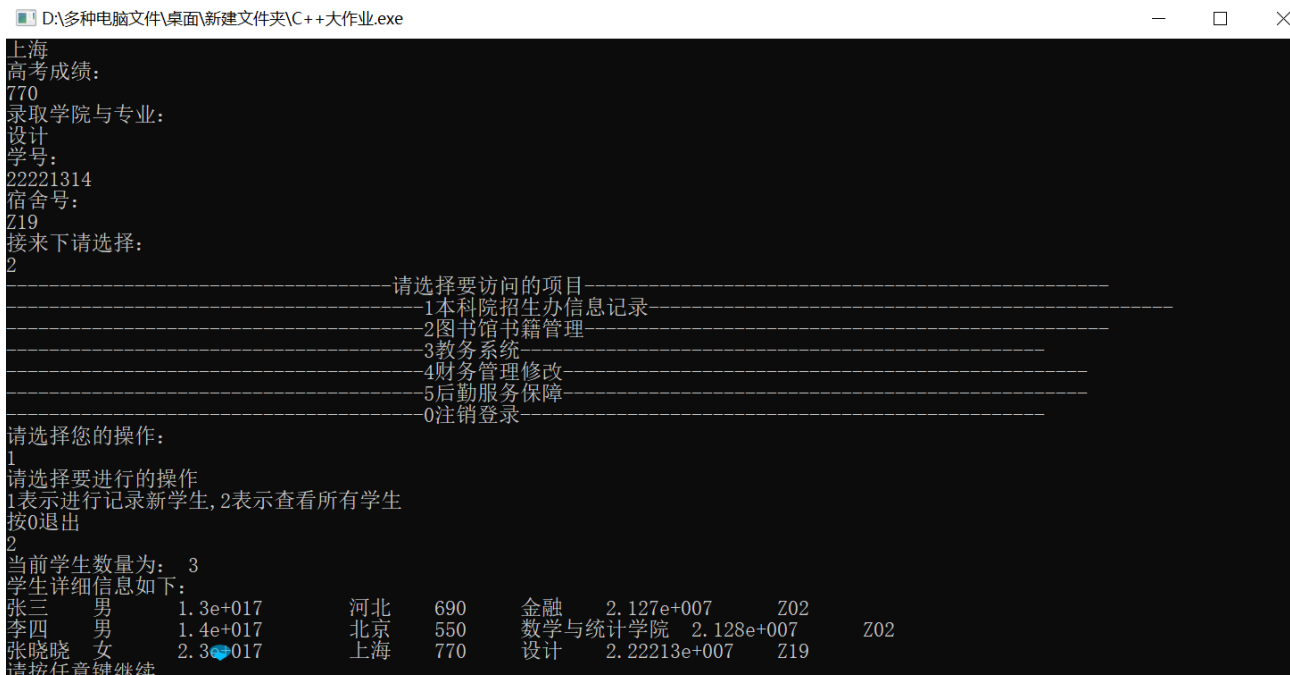
教师登录



增添学生信息



查看所有学生信息



图书馆借阅信息记录

```
D:\多种电脑文件\桌面\新建文件夹\C++大作业.exe

请选择要访问的项目
1本科院招生办信息记录
2图书馆书籍管理
3教务系统
4财务管理修改
5后勤服务保障
0注销登录

请选择您的操作：
2
***** 书库管理 *****
请键入操作选择
1 -在借书信息中添加记录
2 -按学号删除记录
3 -按学号查询图书信息
4 -建立和浏览图书文件
0 -退出
1
***** 入库登记 *****
请键入操作选择
1: 书籍信息录入 2: 修改借阅书籍 0: 退出
1
学号: 22221314
学生姓名: 张晓晓
借出(主/东): 东
剩余借阅天数: 50
已借阅数量: 8
请键入操作选择
1: 书籍信息录入 2: 修改借阅书籍 0: 退出
0
***** 书库管理 *****
请键入操作选择
1 -在借书信息中添加记录
2 -按学号删除记录
3 -按学号查询图书信息
4 -建立和浏览图书文件
0 -退出
```

按学号查询借书记录

```
D:\多种电脑文件\桌面\新建文件夹\C++大作业.exe

请选择要访问的项目
1本科院招生办信息记录
2图书馆书籍管理
3教务系统
4财务管理修改
5后勤服务保障
0注销登录

请选择您的操作：
2
***** 书库管理 *****
请键入操作选择
1 -在借书信息中添加记录
2 -按学号删除记录
3 -按学号查询图书信息
4 -建立和浏览图书文件
0 -退出
3
***** 查询 *****
请键入操作选择
1: 按学号查询 2: 浏览 0: 退出
1
学号(ID):
? 21270000
张三 主 10
请键入操作选择
1: 按学号查询 2: 浏览 0: 退出
```

浏览借书人员信息

```

-----1本科院招生办信息记录-----
-----2图书馆书籍管理-----
-----3教务系统-----
-----4财务管理修改-----
-----5后勤服务保障-----
-----0注销登录-----

请选择您的操作：
2
***** 书库管理 *****
请键入操作选择
    1 -在借书信息中添加记录
    2 -按学号删除记录
    3 -按学号查询图书信息
    4 -建立和浏览图书文件
    0 -退出
3
***** 查询 *****
请键入操作选择
1: 按学号查询    2: 浏览    0: 退出
1
学号(ID):
? 21270000
张三    主    10
请键入操作选择
1: 按学号查询    2: 浏览    0: 退出
2
学生姓名: 张三    借出校区: 主    借阅数量:    10
请键入操作选择
1: 按学号查询    2: 浏览    0: 退出

```

```

***** 书库管理 *****
请键入操作选择
    1 -在借书信息中添加记录
    2 -按学号删除记录
    3 -按学号查询图书信息
    4 -建立和浏览图书文件
    0 -退出
2
***** 销售登记 *****
请键入操作选择
1: 借阅登记    0: 退出
0
***** 书库管理 *****
请键入操作选择
    1 -在借书信息中添加记录
    2 -按学号删除记录
    3 -按学号查询图书信息
    4 -建立和浏览图书文件
    0 -退出
3
***** 查询 *****
请键入操作选择
1: 按学号查询    2: 浏览    0: 退出
2
学生姓名: 张三    借出校区: 主    借阅数量:    10
学生姓名: 李四    借出校区: 东    借阅数量:    2
学生姓名: 张晓晓    借出校区: 东    借阅数量:    8
请键入操作选择
1: 按学号查询    2: 浏览    0: 退出

```

```

***** 书库管理 *****
请键入操作选择
1 -在借书信息中添加记录
2 -按学号删除记录
3 -按学号查询图书信息
4 -建立和浏览图书文件
0 -退出
2
***** 销售登记 *****
请键入操作选择
1: 借阅登记      0: 退出
1
学号 (TP):
? 21270000
学号:21270000   学生姓名: 张三   借阅数量: 10
借阅数量:
? 5
现已借阅量o           5
请键入操作选择
1: 借阅登记      0: 退出

```

打印信息

```

***** 书库管理 *****
请键入操作选择
1 -在借书信息中添加记录
2 -按学号删除记录
3 -按学号查询图书信息
4 -建立和浏览图书文件
0 -退出
4
***** 建立文本文件 *****
文本文件已建立，要浏览文件吗？（ Y/N ）
Y
学号:21270000   学生姓名:张三   借出校区:主   剩余借阅天数:90  借阅数量:10
学号:21279999   学生姓名:李四   借出校区:东   剩余借阅天数:50  借阅数量:2
学号:22221314   学生姓名:张晓晓  借出校区:东   剩余借阅天数:50  借阅数量:8
学号:21270000   学生姓名:张三   借出校区:主   剩余借阅天数:90  借阅数量:5

***** 书库管理 *****
请键入操作选择
1 -在借书信息中添加记录
2 -按学号删除记录
3 -按学号查询图书信息
4 -建立和浏览图书文件
0 -退出

```

增添选课课程信息

```
-----请选择要访问的项目-----
1本科院招生办信息记录
2图书馆书籍管理
3教务系统
4财务管理修改
5后勤服务保障
0注销登录
-----

请选择您的操作:
3
请选择模块
1课程添加
2成绩录入
请输入:
1

-----欢迎来到选课系统-----

**      < 1 >添加课程信息
**      < 2 >显示所有课程信息
**      < 3 >通过课程名查询课程信息
**      < 4 >通过课程号查询课程信息
**      < 5 >将记录保存为txt文件
**      < 6 >删除课程信息记录
**      < 0 >退出此系统
-----

1
请输入需要添加的课程数
1
请输入第1个课程名:统计
请输入第1个课程的老师:梁老师
请输入第1个课程的学分:6
请输入第1个课程的学时:64
请输入第1个课程的课程号:2322445

-----欢迎来到选课系统-----

**      < 1 >添加课程信息
**      < 2 >显示所有课程信息
**      < 3 >通过课程名查询课程信息
**      < 4 >通过课程号查询课程信息
**      < 5 >将记录保存为txt文件
**      < 6 >删除课程信息记录
**      < 0 >退出此系统
-----

2
统计 64 梁老师 6

-----欢迎来到选课系统-----

**      < 1 >添加课程信息
**      < 2 >显示所有课程信息
**      < 3 >通过课程名查询课程信息
**      < 4 >通过课程号查询课程信息
**      < 5 >将记录保存为txt文件
**      < 6 >删除课程信息记录
**      < 0 >退出此系统
-----
```



```
1
请输入需要添加的课程数
2
请输入第1个课程名:
概率
请输入第1个课程的老师:张老师
请输入第1个课程的学分:5
请输入第1个课程的学时:34
请输入第1个课程的课程号:3949595
请输入第2个课程名:运筹学
请输入第2个课程的老师:赵老师
请输入第2个课程的学分:7
请输入第2个课程的学时:54
请输入第2个课程的课程号:34554
```

-----欢迎来到选课系统-----

```
**      < 1 >添加课程信息
**      < 2 >显示所有课程信息
**      < 3 >通过课程名查询课程信息
**      < 4 >通过课程号查询课程信息
**      < 5 >将记录保存为txt文件
**      < 6 >删除课程信息记录
**      < 0 >退出此系统
```

```
2
概率 34 张老师 5
运筹学 54 赵老师 7
0 0
```

-----欢迎来到选课系统-----

```
**      < 1 >添加课程信息
**      < 2 >显示所有课程信息
**      < 3 >通过课程名查询课程信息
**      < 4 >通过课程号查询课程信息
**      < 5 >将记录保存为txt文件
**      < 6 >删除课程信息记录
**      < 0 >退出此系统
```

```
5
```

-----欢迎来到选课系统-----

```
**      < 1 >添加课程信息
**      < 2 >显示所有课程信息
**      < 3 >通过课程名查询课程信息
**      < 4 >通过课程号查询课程信息
**      < 5 >将记录保存为txt文件
**      < 6 >删除课程信息记录
**      < 0 >退出此系统
```

```
-----欢迎来到选课系统-----
**      < 1 >添加课程信息
**      < 2 >显示所有课程信息
**      < 3 >通过课程名查询课程信息
**      < 4 >通过课程号查询课程信息
**      < 5 >将记录保存为txt文件
**      < 6 >删除课程信息记录
**      < 0 >退出此系统
-----
2
概率 34 张老师 5
运筹学 54 赵老师 7
0 0
-----欢迎来到选课系统-----
**      < 1 >添加课程信息
**      < 2 >显示所有课程信息
**      < 3 >通过课程名查询课程信息
**      < 4 >通过课程号查询课程信息
**      < 5 >将记录保存为txt文件
**      < 6 >删除课程信息记录
**      < 0 >退出此系统
-----
5
-----欢迎来到选课系统-----
**      < 1 >添加课程信息
**      < 2 >显示所有课程信息
**      < 3 >通过课程名查询课程信息
**      < 4 >通过课程号查询课程信息
**      < 5 >将记录保存为txt文件
**      < 6 >删除课程信息记录
**      < 0 >退出此系统
-----
4
请选择查询(1)或者修改(0)课程信息
1
请输入查询的课程学号:
34554
运筹学 54 赵老师 7
-----欢迎来到选课系统-----
**      < 1 >添加课程信息
**      < 2 >显示所有课程信息
**      < 3 >通过课程名查询课程信息
**      < 4 >通过课程号查询课程信息
**      < 5 >将记录保存为txt文件
**      < 6 >删除课程信息记录
**      < 0 >退出此系统
-----
```

增添学生成绩信息

```

感谢使用-----请选择要访问的项目-----
1本科院招生办信息记录
2图书馆书籍管理
3教务系统
4财务管理修改
5后勤服务保障
0注销登录

请选择您的操作：
3
请选择模块
1课程添加
2成绩录入
请输入：
2

-----成绩管理-----

**      < 1 >添加学生成绩信息记录
**      < 2 >显示所有学生成绩信息
**      < 3 >通过姓名查询修改学生信息
**      < 4 >通过学号查询修改学生信息
**      < 5 >将记录内学生排序
**      < 6 >将记录保存为txt文件
**      < 7 >删除学生信息记录
**      < 0 >退出此系统

-----

1
请输入需要添加的学生人数
2
请输入第1个学生的姓名:张晓晓
请输入第1个学生的性别:女
请输入第1个学生的专业:设计
请输入第1个学生的年级:22级
请输入第1个学生的学号:22221314
请输入第1个学生的专业课成绩:99
请输入第1个学生的微积分成绩:86
请输入第1个学生的英语综合成绩:90
请输入第2个学生的姓名:张三三
请输入第2个学生的性别:男
请输入第2个学生的专业:统计
请输入第2个学生的年级:21级
请输入第2个学生的学号:21271164
请输入第2个学生的专业课成绩:99
请输入第2个学生的微积分成绩:99
请输入第2个学生的英语综合成绩:99

-----成绩管理-----

**      < 1 >添加学生成绩信息记录
**      < 2 >显示所有学生成绩信息
**      < 3 >通过姓名查询修改学生信息
**      < 4 >通过学号查询修改学生信息
**      < 5 >将记录内学生排序
**      < 6 >将记录保存为txt文件
**      < 7 >删除学生信息记录
**      < 0 >退出此系统

```

成绩排名

```
-----成绩管理-----
**      < 1 >添加学生成绩信息记录
**      < 2 >显示所有学生成绩信息
**      < 3 >通过姓名查询修改学生信息
**      < 4 >通过学号查询修改学生信息
**      < 5 >将记录内学生排序
**      < 6 >将记录保存为txt文件
**      < 7 >删除学生信息记录
**      < 0 >退出此系统

5
请选择需要的排序方式:
<1>根据总分排序
<2>根据专业成绩排序
<3>根据微积分排序
<4>根据英语综合排序
2

-----成绩管理-----
**      < 1 >添加学生成绩信息记录
**      < 2 >显示所有学生成绩信息
**      < 3 >通过姓名查询修改学生信息
**      < 4 >通过学号查询修改学生信息
**      < 5 >将记录内学生排序
**      < 6 >将记录保存为txt文件
**      < 7 >删除学生信息记录
**      < 0 >退出此系统

2
姓名: 张晓晓  年级: 22级  性别: 女  专业: 设计  专业课成绩: 99  微积分成绩: 86  英语: 90  总成绩: 275
姓名: 张三三  年级: 21级  性别: 男  专业: 统计  专业课成绩: 99  微积分成绩: 99  英语: 99  总成绩: 297

-----成绩管理-----
**      < 1 >添加学生成绩信息记录
**      < 2 >显示所有学生成绩信息
**      < 3 >通过姓名查询修改学生信息
**      < 4 >通过学号查询修改学生信息
**      < 5 >将记录内学生排序
**      < 6 >将记录保存为txt文件
**      < 7 >删除学生信息记录
**      < 0 >退出此系统
```

5

请选择需要的排序方式:

- <1>根据总分排序
- <2>根据专业成绩排序
- <3>根据微积分排序
- <4>根据英语综合排序

1

成绩管理

- ** < 1 >添加学生成绩信息记录
- ** < 2 >显示所有学生成绩信息
- ** < 3 >通过姓名查询修改学生信息
- ** < 4 >通过学号查询修改学生信息
- ** < 5 >将记录内学生排序
- ** < 6 >将记录保存为txt文件
- ** < 7 >删除学生信息记录
- ** < 0 >退出此系统

5

请选择需要的排序方式:

- <1>根据总分排序
- <2>根据专业成绩排序
- <3>根据微积分排序
- <4>根据英语综合排序

1

成绩管理

- ** < 1 >添加学生成绩信息记录
- ** < 2 >显示所有学生成绩信息
- ** < 3 >通过姓名查询修改学生信息
- ** < 4 >通过学号查询修改学生信息
- ** < 5 >将记录内学生排序
- ** < 6 >将记录保存为txt文件
- ** < 7 >删除学生信息记录
- ** < 0 >退出此系统

2

姓名: 张三三 年级: 21级 性别: 男 专业: 统计 专业课成绩: 99 微积分成绩: 99 英语: 99 总成绩: 297
姓名: 张晓晓 年级: 22级 性别: 女 专业: 设计 专业课成绩: 99 微积分成绩: 86 英语: 90 总成绩: 275

成绩管理

- ** < 1 >添加学生成绩信息记录
- ** < 2 >显示所有学生成绩信息
- ** < 3 >通过姓名查询修改学生信息
- ** < 4 >通过学号查询修改学生信息
- ** < 5 >将记录内学生排序
- ** < 6 >将记录保存为txt文件
- ** < 7 >删除学生信息记录
- ** < 0 >退出此系统

删除某个学生信息

```
-----成绩管理-----
**      < 1 >添加学生成绩信息记录
**      < 2 >显示所有学生成绩信息
**      < 3 >通过姓名查询修改学生信息
**      < 4 >通过学号查询修改学生信息
**      < 5 >将记录内学生排序
**      < 6 >将记录保存为txt文件
**      < 7 >删除学生信息记录
**      < 0 >退出此系统
-----
6
-----成绩管理-----
**      < 1 >添加学生成绩信息记录
**      < 2 >显示所有学生成绩信息
**      < 3 >通过姓名查询修改学生信息
**      < 4 >通过学号查询修改学生信息
**      < 5 >将记录内学生排序
**      < 6 >将记录保存为txt文件
**      < 7 >删除学生信息记录
**      < 0 >退出此系统
-----
7
请输入需要删除的学生的名字
张三
已删除该学生信息
-----成绩管理-----
**      < 1 >添加学生成绩信息记录
**      < 2 >显示所有学生成绩信息
**      < 3 >通过姓名查询修改学生信息
**      < 4 >通过学号查询修改学生信息
**      < 5 >将记录内学生排序
**      < 6 >将记录保存为txt文件
**      < 7 >删除学生信息记录
**      < 0 >退出此系统
-----
2
姓名：张晓晓  年级：22级  性别：女  专业：设计  专业课成绩：99  微积分成绩：86  英语：90  总成绩：275
-----成绩管理-----
**      < 1 >添加学生成绩信息记录
**      < 2 >显示所有学生成绩信息
**      < 3 >通过姓名查询修改学生信息
**      < 4 >通过学号查询修改学生信息
**      < 5 >将记录内学生排序
**      < 6 >将记录保存为txt文件
**      < 7 >删除学生信息记录
**      < 0 >退出此系统
-----
```

奖学金查看

```
-----成绩管理-----
**      < 1 >添加学生成绩信息记录
**      < 2 >显示所有学生成绩信息
**      < 3 >通过姓名查询修改学生信息
**      < 4 >通过学号查询修改学生信息
**      < 5 >将记录内学生排序
**      < 6 >将记录保存为txt文件
**      < 7 >删除学生信息记录
**      < 0 >退出此系统
-----

2
姓名：张晓晓  年级：22级  性别：女  专业：设计  专业课成绩：99  微积分成绩：86  英语：90  总成绩：275
-----成绩管理-----
**      < 1 >添加学生成绩信息记录
**      < 2 >显示所有学生成绩信息
**      < 3 >通过姓名查询修改学生信息
**      < 4 >通过学号查询修改学生信息
**      < 5 >将记录内学生排序
**      < 6 >将记录保存为txt文件
**      < 7 >删除学生信息记录
**      < 0 >退出此系统
-----

0
感谢使用-----请选择要访问的项目-----
1本科院招生办信息记录-----
2图书馆书籍管理-----
3教务系统-----
4财务管理修改-----
5后勤服务保障-----
0注销登录-----

请选择您的操作：
4
请选择要进行的操作
1表示进行发放学生奖学金,2表示查看所有学生奖学金
按0退出
2
当前获奖学金的学生数量为： 5
获奖学金的学生：
张三      2.127e+007      奖学金9000
李四      2.128e+007      奖学金18000
校花      2.12713e+007   奖学金520
小草      2.12789e+007     奖学金8888
小国      2.1202e+007     奖学金5000
请按任意键继续. . .
```

给同学发奖金

```
-----请选择要访问的项目-----
1本科院招生办信息记录
2图书馆书籍管理
3教务系统
4财务管理修改
5后勤服务保障
0注销登录

请选择您的操作:
4
请选择要进行的操作
1表示进行发放学生奖学金, 2表示查看所有学生奖学金
按0退出
1
学生姓名:
张三
学号:
21271164
奖学金:
8000
接下来请选择:
2
-----请选择要访问的项目-----
1本科院招生办信息记录
2图书馆书籍管理
3教务系统
4财务管理修改
5后勤服务保障
0注销登录

请选择您的操作:
4
请选择要进行的操作
1表示进行发放学生奖学金, 2表示查看所有学生奖学金
按0退出
2
当前获奖学金的学生数量为: 6
获奖学金的学生:
张三      2. 127e+007      奖学金9000
李四      2. 128e+007      奖学金18000
校花      2. 12713e+007   奖学金520
小草      2. 12789e+007   奖学金8888
小国      2. 1202e+007   奖学金5000
张三三    2. 12712e+007   奖学金8000
请按任意键继续. . .
```

后勤保障

```
-----请选择要访问的项目-----
1本科院招生办信息记录
2图书馆书籍管理
3教务系统
4财务管理修改
5后勤服务保障
0注销登录

请选择您的操作:
5
当前学生数量为: 3
学生详细信息如下:
张三      男      学号      2. 127e+007      宿舍号Z02
李四      男      学号      2. 128e+007      宿舍号Z02
张晓晓    女      学号      2. 22213e+007    宿舍号Z19
请按任意键继续. . .
```



```

-----请选择要访问的项目-----
1本科院招生办信息记录
2图书馆书籍管理
3教务系统
4财务管理修改
5后勤服务保障
0注销登录

请选择您的操作:
0
注销成功
请按任意键继续. . .

```

```

-----欢迎来到学生信息管理系统-----
-----选择要进入的系统:(1为学生/2为职工)-----
-----请输入1或者2-----
-----按0退出系统-----

请输入您的选择:
1
请输入你的学号
21270000
请输入用户名:
张三
请输入密码:
123456
学生验证登录成功!
请按任意键继续. . .

```

学生登录

D:\多种电脑文件\桌面\新建文件夹\C++大作业.exe

```

-----请选择要访问的项目-----
1本科院招生办
2图书馆
3教务系统
4财务管理
5后勤服务
0注销登录

请选择您的操作:
1
恭喜您被我校录取
有关招生详细资料请拨打招生办电话或去南门招生办询问
-----请选择要访问的项目-----
1本科院招生办
2图书馆
3教务系统
4财务管理
5后勤服务
0注销登录

请选择您的操作:

```

查询借书记录

```
-----请选择要访问的项目-----
1本科院招生办
2图书馆
3教务系统
4财务管理
5后勤服务
0注销登录
-----

请选择您的操作：
2
***** 查询 *****
请键入操作选择
1: 按学号查询    2: 浏览  0: 退出
1
学号(ID):
? 21271164
学号输入错误
请键入操作选择
1: 按学号查询    2: 浏览  0: 退出
1
学号(ID):
? 21270000
张三      主          10
请键入操作选择
1: 按学号查询    2: 浏览  0: 退出
1
```

学生登录教务系统

```
请选择您的操作：
2
***** 查询 *****
请键入操作选择
1: 按学号查询    2: 浏览  0: 退出
1
学号(ID):
? 21271164
学号输入错误
请键入操作选择
1: 按学号查询    2: 浏览  0: 退出
1
学号(ID):
? 21270000
张三      主          10
请键入操作选择
1: 按学号查询    2: 浏览  0: 退出
2
学生姓名: 张三    借出校区: 主    借阅数量:    10
请键入操作选择
1: 按学号查询    2: 浏览  0: 退出
0
-----请选择要访问的项目-----
1本科院招生办
2图书馆
3教务系统
4财务管理
5后勤服务
0注销登录
-----

请选择您的操作：
3
请选择模块
1课程查询
2成绩查询
请输入:
```

第五章 课程设计总结

这学期通过学习 C++，对 C 语言和 C++ 语言都有了更深刻的认识，也学习到了面向对象程序设计的思想与方法。

C++ 的突出点在与有很多标准模版库，我们只需要对模版库进行掌握与了解就能完成我们想要完成的任务。

在编写类时，有些链接不容易处理，容易出现 bug，有时自己思路结构不清晰的话写出来的东西报错就很严重。

Dev c++ 编译环境不是很好，现在一般通用的编码方式是 UTF-8，而 Dev-C++ 使用的是 ANSI。容易出现中文乱码。改完代码后有时 win 文件出现错误，只能将其删掉重新编译。

对于学生端还有一部分未加完善，有待提高，没有将接口接入。

做学生端的时候不剩多长时间了，原本想着加快步伐，在 `student.cpp` 中添加接口链接已经成型的教师端的部分函数，将部分权限关闭，可无奈时间太短，未来得及实现，只得以实现部分，按照这个思路将所有项目完成应该是可以的。

希望自己能继续成长，不断向前。

最后感谢赵老师的陪伴与教诲。

附录：程序代码

```
#include<iostream>
#include"xinxiManager.h"
#include "globalFile.h"
#include "identity.h"
#include <fstream>
#include <string>
#include "student.h"
#include "teacher.h"
#include "manager.h"
using namespace std;

//管理员菜单
void managerMenu(Identity * &manager)
{
    while (true)
    {
        //管理员菜单
        manager->operMenu();

        Manager* man = (Manager*)manager;
        int select = 0;

        cin >> select;

        if (select == 1) //添加账号
        {
            cout << "添加账号" << endl;
            man->addPerson();
        }
        else if (select == 2) //查看账号
        {
            cout << "查看账号" << endl;
            man->showPerson();
        }

        else
        {
            delete manager;
            cout << "注销成功" << endl;
            system("pause");
            system("cls");
            return;
        }
    }
}
```

```

    }
}
}

```

//学生菜单

```
void studentMenu(Identity * &student)
```

```

{
    while (true)
    {
        //学生菜单
        student->operMenu();

        Student* stu = (Student*)student;
        int select = 0;

        cin >> select;

        if (select == 1)
        {
            stu->Zhaosheng();
            cout<<"恭喜您被我校录取"<<endl;
            cout<<"有关招生详细资料请拨打招生办电话或去南门招生办询问"<<endl;
        }
        else if (select == 2)
        {
            stu->Shuji();
        }
        else if (select ==3)
        {
            stu->Jiaowu();
        }
        else if (select ==4)
        {
            stu->Caiwu();
        }
        else if (select ==5)
        {
            stu->Houqin();
        }
        else
        {
            delete student;
            cout << "注销成功" << endl;
            system("pause");
            system("cls");
            return;
        }
    }
}

```

```

    }

}

//教师菜单
void TeacherMenu(Identity * &teacher)
{
    while (true)
    {
        //教师菜单
        teacher->operMenu();

        Teacher* tea = (Teacher*)teacher;
        int select = 0;

        cin >> select;

        if (select == 1)
        {

            tea->Zhaosheng();
        }


        else if (select == 2)
        {
            tea->Shuji();
        }
        else if (select ==3)
        {
            tea->Jiaowu();
        }
        else if (select ==4)
        {
            tea->Caiwu();
        }
        else if (select ==5)
        {
            tea->Houqin();
        }
        else
        {
            delete teacher;
            cout << "注销成功" << endl;

```

```

        system("pause");
        system("cls");
        return;
    }

}

//登录功能
int LoginIn(string fileName, int type)
{
    Identity * person = NULL;

    ifstream ifs;
    ifs.open(fileName.c_str(),ios::in);

    //文件不存在情况
    if (!ifs.is_open())
    {
        cout << "文件不存在" << endl;
        ifs.close();
        return 0;
    }

    int xuehao = 0;
    string name;
    string pwd;

    if (type == 1) //学生登录
    {
        cout << "请输入你的学号" << endl;
        cin >> xuehao;
    }
    else if (type == 2) //教师登录
    {
        cout << "请输入你的职工号" << endl;
        cin >> xuehao;
    }

    cout << "请输入用户名: " << endl;
    cin >> name;

    cout << "请输入密码: " << endl;
    cin >> pwd;
}

```

```

if (type == 1)
{
    //学生登录验证
    int fId;
    string fName;
    string fPwd;
    while (ifs >> fId && ifs >> fName && ifs >> fPwd)
    {
        if (xuehao == fId && name == fName && pwd == fPwd)
        {
            cout << "学生验证登录成功!" << endl;
            system("pause");
            system("cls");
            person = new Student(xuehao, name, pwd);
            //进入学生菜单
            studentMenu(person);
            return 1;
        }
    }
}
else if (type == 2)
{
    //教师登录验证
    int fId;
    string fName;
    string fPwd;
    while (ifs >> fId && ifs >> fName && ifs >> fPwd)
    {
        if (xuehao == fId && name == fName && pwd == fPwd)
        {
            cout << "教师验证登录成功!" << endl;
            system("pause");
            system("cls");
            person = new Teacher(xuehao, name, pwd);
            //进入教师菜单
            TeacherMenu(person);
            return 1;
        }
    }
}
else if (type == 3)
{
    //管理员登录验证
    string fName;
    string fPwd;

```



```

while (ifs >> fName && ifs >> fPwd)
{
    if (name == fName && pwd == fPwd)
    {
        cout << "验证登录成功!" << endl;
        //登录成功后，按任意键进入管理员界面
        system("pause");
        system("cls");
        //创建管理员对象
        person = new Manager(name,pwd);
        managerMenu(person);
        return 1;
    }
}

cout << "验证登录失败!" << endl;

system("pause");
system("cls");
return 0 ;
}

```

```

int main()
{
    xinxManager xm;
    int choice=0;
    int s1,s2;
    while(1)
    {
        xm.show_Menu();
        cout<<"请输入您的选择: "<<endl;
        cin>>choice;

        if (choice == 0) {
            xm.exitSystem();
            break;}
        else if (choice == 1) {

            if (LoginIn(STUDENT_FILE,1)==1){
                cout <<"欢迎进入学生系统"<<endl;

```

```

    }

    }

    else if (choice == 2) {

        if (LoginIn(TEACHER_FILE,2)==1){
            cout <<"欢迎进入职工系统"<<endl;

        }

    }

    else if (choice == 3) {

        if (LoginIn(ADMIN_FILE, 3)==1)
        {
            cout <<"欢迎进入管理系统"<<endl;
        }

    }

    else
    {
        system("cls");
        break;
    }

    }
    system("pause");
    return 0;

}

```

// Append.cpp

```

#include "ex10_15.h"
void Append( fstream &f )
{ int choice ;
  bookData book ;
  int key ;
  long num ;
  f.seekp( 0, ios::end ) ;    // 读指针移到文件末尾
  long posEnd = f.tellp() ;    // 记录文件尾位置
  cout << "***** 入库登记 *****\n" ;
  while (1)

```

```

{ cout << "请键入操作选择\n"
  << "1: 书籍信息录入\t"
  << "2: 修改借阅书籍\t"
  << "0: 退出\n";
cin >> choice;
switch ( choice )
{ case 1: // 追加新记录
  { cout << "学号: ";
    cin >> book.TP;
    cout << "学生姓名: ";
    cin >> book.bookName;
    cout << "借出 (主/东): ";
    cin >> book.bookChu;
    cout << "剩余借阅天数: ";
    cin >> book.money;
    cout << "已借阅数量: ";
    cin >> book.balance;
    f.write( (char *) & book, sizeof( bookData ) ); // 写入文件
    break;
  };
case 2: // 修改记录
  { f.seekp( 0, ios::beg ); // 写指针移到文件头
    cout << "学号(TP) : \n? ";
    cin >> key; // 输入书号
    do // 按书号查找, 读数据赋给结构变量 book
      { f.read((char *) & book, sizeof(bookData));
        } while ( book.TP != key && f.tellp() != posEnd );
    if ( book.TP == key ) // 找到记录
      { cout << "学号: " << book.TP << "\t" << "学生姓名: " << book.bookName << "\t" << "剩余借阅天
数: " << book.money << "\t" << "借阅数量: " << book.balance << endl;
        cout << "入库数量: \n? ";
        cin >> num;
        if ( num>0 ) book.balance += num; // 修改库存量
        else
          { cout << "数量输入错误\n";
            continue;
          }
        f.seekp( -long( sizeof( bookData ) ), ios::cur ); // 指针复位
        f.write( ( char * ) & book, sizeof( bookData ) ); // 写入文件
        cout << "现借阅量: \t\t" << book.balance << endl;
      }
    else
      cout << "学号输入错误\n";
    break;
  }
case 0: return;

```

```

    }
}
}
// CreateTxt.cpp
// 读出已有二进制文件数据， 写入文本文件。建立格式化的文本文件，便于浏览、打印
#include "ex10_15.h"
void CreateTxt(fstream& f)
{ fstream ftxt("d:\\booksFile.txt", ios::out); // 写方式打开文本文件
  bookData book;
  f.seekg(0, ios::end);
  long posEnd = f.tellg(); // 记录二进制文件末尾位置
  f.seekg(0, ios::beg); // 移动读指针到文件头
  cout << "***** 建立文本文件 *****\n";
  do
  { // 从二进制文件读记录
    f.read((char *) & book, sizeof(bookData));
    // 把记录写入文本文件
    ftxt << "学号:" << book.TP << "\t" << "学生姓名:" << book.bookName << "\t" << "借出校
    区:" << book.bookChu << "\t" << "剩余借阅天数:" << book.money << "\t" << "借阅数量:" << book.balance << endl;
    } while ( f.tellg() != posEnd );
  ftxt.close();
  cout << "文本文件已建立，要浏览文件吗？(Y/N)\n";
  char answer, s[80];
  cin >> answer;
  if ( answer=='Y' || answer=='y' )
  { ftxt.open( "d:\\booksFile.txt", ios::in ); // 重用流打开文件
    while( !ftxt.eof() ) // 按行显示文本文件
    { ftxt.getline( s, 80 );
      cout << s << endl;
    }
  }
  ftxt.close(); // 关闭文本文件
}
// Inquire.cpp
// 查询
#include "ex10_15.h"
void Inquire(fstream& f)
{ int choice;
  bookData book;
  int key;
  f.seekg(0, ios::end); // 读指针移到文件末尾
  long posEnd = f.tellg(); // 记录文件尾位置
  cout << "***** 查询 *****\n";
  while (1)
  { cout << "请键入操作选择\n"
    << "1: 按学号查询\t"

```

```

        << "2: 浏览\t"
        << "0: 退出\n";
    cin >> choice ;
    switch ( choice )
    { case 1:      // 按书号检索
        { f.seekg( 0, ios::beg );
          cout << "学号(ID): \n? ";
          cin >> key ;
          do
              { f.read((char *) & book , sizeof(bookData)) ;
                } while ( book.TP != key && f.tellg() != posEnd ) ;
          if ( book.TP == key )
              cout << book.bookName << '\t' << book.bookChu << '\t' << book.bookZuo<<'\t'<<
book.balance << endl;
          else
              { cout << "学号输入错误\n";
                continue;
              }
          break ;
        }
        case 2:      // 浏览文件
        { f.seekg( 0, ios::beg );
          do          // 输出所有记录
              { f.read((char *) & book , sizeof(bookData)) ;
                cout <<"学生姓名: "<< book.bookName << '\t'<<"借出校区: " << book.bookChu <<
'\t'<<"借阅数量: "<<'\t'<< book.balance << endl ;
                } while ( book.TP != key && f.tellp() != posEnd ) ;
          break ;
        }
        case 0:  return ;
    }
}
}

```

// Sale.cpp

```

#include "ex10_15.h"
void Sale(fstream& f)
{ int choice ;
  bookData book ;
  int key ;
  long num ;
  f.seekp( 0, ios::end ) ;
  long posEnd = f.tellp() ;      // 记录文件末尾位置
  cout << "***** 销售登记 *****\n" ;
  while (1)
  { cout << "请键入操作选择\n"

```

```

        << "1: 借阅登记\t"
        << "0: 退出\n";
    cin >> choice ;
    switch ( choice )
    { case 1:
        { f.seekp( 0, ios::beg );      // 从文件头开始检索
          cout << "学号(TP): \n? ";
          cin >> key ;
          do          // 按书号查找
              { f.read((char *) & book , sizeof(bookData)) ;
                } while ( book.TP != key && f.tellp() != posEnd ) ;
          if ( book.TP == key )      // 找到
              { cout<<"学号:"<<book.TP<<"\t"<<"学生姓名: "<<book.bookName<<"\t"<<"借阅数量:
"<<book.balance<<endl;
                cout << "借阅数量: \n? ";
                cin >> num ;
                if ( num>0 && book.balance>=num )    // 修改库存量
                    book.balance -= num ;
                else
                    { cout << "数量输入错误\n";
                      continue;
                    }
                f.seekp( -long( sizeof( bookData ) ), ios::cur );    // 文件指针复位
                f.write( (char *) & book , sizeof( bookData ) );    // 修改文件记录
                cout << "现已借阅量 o\t\t" << book.balance << endl ;
            }
            else
                cout << "学号输入错误\n";
            break ;
        }
        case 0 : return ;
    }
}
}
}
#pragma once
#include<iostream>
#include<string>
using namespace std;
//机房类
class ComputerRoom
{
public:
    string name;
    string xingbie;
    double shenfenzheng;
    string jiguan;

```

```

        int gaokao;
        string luquzhuanye;
        double m_Id;
        string sushehao;

};

class Jiangxuejin
{
public:
    string name;
    double m_Id;

    double jiangxuejin;

};

//ex10_15.h

#ifndef EX10_15_H
#define EX10_15_H
/*
#include<iostream.h>
#include<fstream.h>
#include<stdlib.h>
#include<string.h>
*/
#include <iostream>
#include <fstream>
#include <cstdlib>
#include <string>
using namespace std;
struct bookData
{ int TP ;
    char bookName[40] ; //学生姓名
    char bookChu[40] ; //借出校区（主校区/东校区）
    char bookZuo[40] ;
    int money; //剩余借阅天数
    long balance; //已借阅数量
};

void Append(fstream &f);
void Sale(fstream &f);
void Inquire(fstream &f);
void CreateTxt(fstream &f);
#endif

```

```

#pragma once

//管理员文件
#define ADMIN_FILE      "admin.txt"
//学生文件
#define STUDENT_FILE    "student.txt"
//教师文件
#define TEACHER_FILE    "teacher.txt"
//课程信息文件
#define COURSE_FILE     "course.txt"
//成绩文件
#define SCORE_FILE      "score.txt"
//详细信息
#define COMPUTER_FILE   "student xiangxi.txt"
#define JIANGXUEJIN     "jiangxuejin.txt"

#pragma once
#include<iostream>
using namespace std;

//身份抽象类
class Identity
{
public:

    //操作菜单
    virtual void operMenu() = 0;

    string m_Name; //用户名
    string m_Pwd;  //密码
};

#include "manager.h"

//默认构造
Manager::Manager()
{
}

//有参构造
Manager::Manager(string name, string pwd)

```



```

{
    this->m_Name = name;
    this->m_Pwd = pwd;
}

//选择菜单
void Manager::operMenu()
{
    cout <<"-----" <<endl;
    cout <<"----- 请 选 择 要 访 问 的 项 目
-----" <<endl;
    cout <<"-----1 添加账号-----" <<endl;
    cout <<"-----2 查看账号-----" <<endl;
    cout <<"-----" <<endl;
    cout <<"-----0 注销登录-----" <<endl;
    cout <<"-----" <<endl;
    cout <<"请选择您的操作： " <<endl;
}

//添加账号
void Manager::addPerson()
{
    cout << "请输入添加账号的类型" << endl;
    cout << "1、添加学生" << endl;
    cout << "2、添加老师" << endl;

    string fileName;
    string tip;
    ofstream ofs;

    int select = 0;
    cin >> select;

    string errorTip; //重复错误提示

    if (select == 1)
    {
        fileName = STUDENT_FILE;
        tip = "请输入学号： ";
        errorTip = "学号重复，请重新输入";
    }
}

```

```

    }
else
{
    fileName = TEACHER_FILE;
    tip = "请输入职工编号： ";
    errorTip = "职工号重复， 请重新输入";
}
ofs.open(fileName.c_str(), ios::out | ios::app);
int id;
string name;
string pwd;
cout << tip << endl;

while (true)
{
    cin >> id;

    bool ret = this->checkRepeat(id, 1);

    if (ret) //有重复
    {
        cout << errorTip << endl;
    }
    else
    {
        break;
    }
}

cout << "请输入姓名： " << endl;
cin >> name;

cout << "请输入密码： " << endl;
cin >> pwd;

ofs << id << " " << name << " " << pwd << " " << endl;
cout << "添加成功" << endl;

system("pause");
system("cls");

ofs.close();
this->initVector();

```

```

}
void printStudent(Student & s)
{
    cout << "学号:  " << s.m_Id << " 姓名:  " << s.m_Name << " 密码:  " << s.m_Pwd << endl;
}
void printTeacher(Teacher & t)
{
    cout << "职工号:  " << t.m_EmpId << " 姓名:  " << t.m_Name << " 密码:  " << t.m_Pwd << endl;
}

void Manager::showPerson()
{
    cout << "请选择查看内容:  " << endl;
    cout << "1、查看所有学生" << endl;
    cout << "2、查看所有老师" << endl;

    int select = 0;

    cin >> select;
    Manager::initVector();
    if (select == 1)
    {
        cout << "所有学生信息如下:  " << endl;
        for_each(vStu.begin(), vStu.end(), printStudent);
    }
    else
    {
        cout << "所有老师信息如下:  " << endl;
        for_each(vTea.begin(), vTea.end(), printTeacher);
    }
    system("pause");
    system("cls");
}

void Manager::initVector()
{
    vStu.clear();
    vTea.clear();
    //读取学生文件中信息
    ifstream ifs;
    ifs.open(STUDENT_FILE, ios::in);
    if (!ifs.is_open())
    {

```

```

        cout << "文件读取失败" << endl;
        return;
    }

```

```

Student s;
while (ifs >> s.m_Id && ifs >> s.m_Name && ifs >> s.m_Pwd)
{
    vStu.push_back(s);
}
cout << "当前学生数量为:  " << vStu.size() << endl;
ifs.close(); //学生初始化

```

```

//读取老师文件信息
ifs.open(TEACHER_FILE, ios::in);

```

```

Teacher t;
while (ifs >> t.m_EmpId && ifs >> t.m_Name && ifs >> t.m_Pwd)
{
    vTea.push_back(t);
}
cout << "当前教师数量为:  " << vTea.size() << endl;

```

```

ifs.close();

```

```

}

```

```

bool Manager::checkRepeat(int id, int type)
{
    if (type == 1)
    {
        for (vector<Student>::iterator it = vStu.begin(); it != vStu.end(); it++)
        {
            if (id == it->m_Id)
            {
                return true;
            }
        }
    }
    else
    {
        for (vector<Teacher>::iterator it = vTea.begin(); it != vTea.end(); it++)
        {
            if (id == it->m_EmpId)
            {
                return true;
            }
        }
    }
}

```

```

        }
    }
}
return false;
}

#pragma once
#include <iostream>
using namespace std;
#include "identity.h"
#include <string>
#include <fstream>
#include "globalFile.h"
#include <vector>
#include "student.h"
#include "teacher.h"
#include <algorithm>
#include "computerRoom.h"
class Manager :public Identity
{
public:

    //默认构造
    Manager();

    //有参构造  管理员姓名，密码
    Manager(string name, string pwd);

    //选择菜单
    virtual void operMenu();

    //添加账号
    void addPerson();

    //查看账号
    void showPerson();


    //初始化容器
    void initVector();

    //学生容器
    vector<Student> vStu;

```

```

//教师容器
vector<Teacher> vTea;

bool checkRepeat(int id, int type);

};

#include "primary.h"
using namespace std;
void primary::addstu(int nums, primary Stu[101], int& index) {

    for (int i = 0; i < nums; i++)
    {
        cout << "请输入第" << i + 1 << "个课程名:";
        cin >> Stu[i].name;

        cout << "请输入第" << i + 1 << "个课程的老师:";
        cin >> Stu[i].sex;

        cout << "请输入第" << i + 1 << "个课程的学分:";
        cin >> Stu[i].class_info;

        cout << "请输入第" << i + 1 << "个课程的学时:";
        cin >> Stu[i].age;

        cout << "请输入第" << i + 1 << "个课程的课程号:";
        cin >> Stu[i].number;

    }
    index += nums;
}

void primary::search_by_name(string Name, primary stu[101],int index) {
    for (int i = 0; i < index; i++) {
        if (stu[i].name == Name) {
            cout << stu[i].name << ' ' << stu[i].age << ' ' << stu[i].sex << ' ' << stu[i].class_info << ' ';

        }
    }
}

void primary::search_by_num(int num, primary stu[101],int index) {
    for (int i = 0; i < index; i++) {

```

```

        if (stu[i].number == num) {
            cout << stu[i].name << ' ' << stu[i].age << ' ' << stu[i].sex << ' ' << stu[i].class_info << ' ';

        }
    }
}

void primary::output_list(primary stu[101], int amount) {
    ofstream dataFile;
    dataFile.open("dataFile.txt", ofstream::app);
    fstream file("dataFile.txt", ios::out);
    // 写入数据
    for (int i = 0; i < amount; i++) {
        dataFile << "名称" << ": " << stu[i].name << ' ' << "学时: " << stu[i].age << ' ' << "老师: " << stu[i].sex
        << ' ' << "学分: " << stu[i].class_info << ' ';

    }
    dataFile.close();
}

void primary::delete_stu(string target, primary stu[101], int& index) {
    for (int i = 0; i < index; i++) {
        if (stu[i].name == target) {
            for (int j = i; j < index; j++) {
                stu[j] = stu[j + 1];
            }
        }
    }
    index--;
}

void primary::get_name() {
    cout << this->name;
}

void primary::show_all(primary stu[101], int index) {
    for (int i = 0; i < index; i++) {
        cout << stu[i].name << ' ' << stu[i].age << ' ' << stu[i].sex << ' ' << stu[i].class_info << ' ';

        cout << endl;
    }
}

void primary::change_stu1(string name1, primary Stu[101], int index) {

```

```

int temp;
for (int i = 0; i < index; i++) {
    if (Stu[i].name == name1) {
        temp = i;
        break;
    }
}
cout << "请输入修改后课程名:";
cin >> Stu[temp].name;

cout << "请输入修改后课程的老师:";
cin >> Stu[temp].sex;

cout << "请输入修改后课程的学分:";
cin >> Stu[temp].class_info;

cout << "请输入修改后课程的学时:";
cin >> Stu[temp].age;

cout << "请输入修改后课程的课程号:";
cin >> Stu[temp].number;

}

void primary::change_stu2(int target, primary Stu[101], int index) {
    int temp;
    for (int i = 0; i < index; i++) {
        if (Stu[i].number == target) {
            temp = i;
            break;
        }
    }
    cout << "请输入修改后课程名:";
    cin >> Stu[temp].name;

    cout << "请输入修改后课程的老师:";
    cin >> Stu[temp].sex;

    cout << "请输入修改后课程的学分:";
    cin >> Stu[temp].class_info;

    cout << "请输入修改后课程的学时:";
    cin >> Stu[temp].age;

    cout << "请输入修改后课程的课程号:";
    cin >> Stu[temp].number;
}

```



```

}
#pragma once
#include<iostream>
#include <string>
#include <fstream>

using namespace std;
class primary {
private:
    int number;
    string name;
    int age;
    string sex;
    int class_info;

public:
    void addstu(int nums, primary stu[101],int &index);
    void search_by_name(string name, primary stu[101],int index);
    void search_by_num(int num, primary stu[101],int index);
    void delete_stu(string name, primary stu[101],int &index);
    float CHI_average(primary stu[101],int index);
    float MATH_average(primary stu[101], int index);
    float ENG_average(primary stu[101],int index);
    void total_sorting(primary stu[101],int index);
    void CHI_sorting(primary stu[101],int index);
    void MATH_sorting(primary stu[101],int index);
    void ENG_sorting(primary stu[101],int index);
    void output_list(primary stu[101],int amount);
    void get_name();
    void show_all(primary stu[101],int index);
    void change_stu1(string name, primary stu[101], int index);
    void change_stu2(int target, primary Stu[101], int index);
};

#include "primarychengji.h"
using namespace std;
void primarychengji::addstu(int nums, primarychengji Stu[101], int& index) {

    for (int i = 0; i < nums; i++)
    {
        cout << "请输入第" << i + 1 << "个学生的姓名:";
        cin >> Stu[i].name;

        cout << "请输入第" << i + 1 << "个学生的性别:";
    }
}

```

```

        cin >> Stu[i].sex;

        cout << "请输入第" << i + 1 << "个学生的专业:";
        cin >> Stu[i].class_info;

        cout << "请输入第" << i + 1 << "个学生的年级:";
        cin >> Stu[i].age;

        cout << "请输入第" << i + 1 << "个学生的学号:";
        cin >> Stu[i].number;

        cout << "请输入第" << i + 1 << "个学生的专业课成绩:";
        cin >> Stu[i].Chinese;

        cout << "请输入第" << i + 1 << "个学生的微积分成绩:";
        cin >> Stu[i].Math;

        cout << "请输入第" << i + 1 << "个学生的英语综合成绩:";
        cin >> Stu[i].English;

        Stu[i].Total = Stu[i].Chinese + Stu[i].Math + Stu[i].English;
    }
    index += nums;
}

void primarychengji::search_by_name(string Name, primarychengji stu[101],int index) {
    for (int i = 0; i < index; i++) {
        if (stu[i].name == Name) {
            cout << stu[i].name << ' ' << stu[i].age << ' ' << stu[i].sex << ' ' << stu[i].class_info << ' ';
            cout << stu[i].Chinese << ' ' << stu[i].Math << ' ' << stu[i].English << ' ' << stu[i].Total;
        }
    }
}

void primarychengji::search_by_num(int num, primarychengji stu[101],int index) {
    for (int i = 0; i < index; i++) {
        if (stu[i].number == num) {
            cout << stu[i].name << ' ' << stu[i].age << ' ' << stu[i].sex << ' ' << stu[i].class_info << ' ';
            cout << stu[i].Chinese << ' ' << stu[i].Math << ' ' << stu[i].English << ' ' << stu[i].Total;
        }
    }
}

void primarychengji::output_list(primarychengji stu[101], int amount) {
    ofstream chengjidataFile;
    chengjidataFile.open("chengjidataFile.txt", ofstream::app);

```

```

    fstream file("chengjidataFile.txt", ios::out);
    // 写入数据
    for (int i = 0; i < amount; i++) {
        chengjidataFile << "姓名: " << stu[i].name << ' ' << "年级: " << stu[i].age << ' ' << "性别: " << stu[i].sex
        << ' ' << "专业: " << stu[i].class_info << ' ';
        chengjidataFile << "专业课分数: " << stu[i].Chinese << ' ' << "微积分分数: " << stu[i].Math << ' ' << "
        英语综合分数: " << stu[i].English << ' ' << "总分: " << stu[i].Total;

    }
    chengjidataFile.close();
}

void primarychengji::delete_stu(string target, primarychengji stu[101], int& index) {
    for (int i = 0; i < index; i++) {
        if (stu[i].name == target) {
            for (int j = i; j < index; j++) {
                stu[j] = stu[j + 1];
            }
        }
    }
    index--;
}

float primarychengji::CHI_average(primarychengji stu[101], int index) {
    float result = 0;
    for (int i = 0; i < index; i++) {
        result += stu[i].Chinese;
    }
    return (result / index);
}

float primarychengji::MATH_average(primarychengji stu[101], int index) {
    float result = 0;
    for (int i = 0; i < index; i++) {
        result += stu[i].Math;
    }
    return (result / index);
}

float primarychengji::ENG_average(primarychengji stu[101], int index) {
    float result = 0;
    for (int i = 0; i < index; i++) {
        result += stu[i].English;
    }
    return (result / index);
}

```

```

void primarychengji::total_sorting(primarychengji stu[101], int index) {
    for (int i = 1; i < index; i++)
    {
        int j = i - 1;

        //插入数组中的数据
        primarychengji temp = stu[i];

        while (temp.Total > stu[j].Total && j >= 0)
        {
            stu[j + 1] = stu[j];

            j--;
        }

        stu[++j] = temp;
    }
}

```

```

void primarychengji::CHI_sorting(primarychengji stu[101], int index) {
    for (int i = 1; i < index; i++)
    {
        int j = i - 1;

        //插入数组中的数据
        primarychengji temp = stu[i];

        while (temp.Chinese > stu[j].Chinese && j >= 0)
        {
            stu[j + 1] = stu[j];

            j--;
        }

        stu[++j] = temp;
    }
}

```

```

void primarychengji::ENG_sorting(primarychengji stu[101], int index) {
    for (int i = 1; i < index; i++)
    {
        int j = i - 1;

        //插入数组中的数据
        primarychengji temp = stu[i];

```

```

        while (temp.English > stu[j].English && j >= 0)
        {
            stu[j + 1] = stu[j];

            j--;
        }

        stu[++j] = temp;
    }
}

void primarychengji::MATH_sorting(primarychengji stu[101], int index) {
    for (int i = 1; i < index; i++)
    {
        int j = i - 1;

        //插入数组中的数据
        primarychengji temp = stu[i];

        while (temp.Math > stu[j].Math && j >= 0)
        {
            stu[j + 1] = stu[j];

            j--;
        }

        stu[++j] = temp;
    }
}

void primarychengji::get_name() {
    cout << this->name;
}

void primarychengji::show_all(primarychengji stu[101], int index) {
    for (int i = 0; i < index; i++) {
        cout << "姓名: " << stu[i].name << " " << "年级: " << stu[i].age << " " << "性别: " << stu[i].sex << " " << "
专业: " << stu[i].class_info << " ";
        cout << "专业课成绩: " << stu[i].Chinese << " " << "微积分成绩: " << stu[i].Math << " " << "英语:
" << stu[i].English << " " << "总成绩: " << stu[i].Total;
        cout << endl;
    }
}

void primarychengji::change_stu1(string name1, primarychengji Stu[101], int index) {

```

```

int temp;
for (int i = 0; i < index; i++) {
    if (Stu[i].name == name1) {
        temp = i;
        break;
    }
}
cout << "请输入修改后学生的姓名:";
cin >> Stu[temp].name;

cout << "请输入修改后学生的性别:";
cin >> Stu[temp].sex;

cout << "请输入修改后学生的专业:";
cin >> Stu[temp].class_info;

cout << "请输入修改后学生的年级:";
cin >> Stu[temp].age;

cout << "请输入修改后学生的学号:";
cin >> Stu[temp].number;

cout << "请输入修改后学生的专业课成绩:";
cin >> Stu[temp].Chinese;

cout << "请输入修改后学生的微积分成绩:";
cin >> Stu[temp].Math;

cout << "请输入修改后学生的英语综合成绩:";
cin >> Stu[temp].English;

Stu[temp].Total = Stu[temp].Chinese + Stu[temp].Math + Stu[temp].English;
}

void primarychengji::change_stu2(int target, primarychengji Stu[101], int index) {
    int temp;
    for (int i = 0; i < index; i++) {
        if (Stu[i].number == target) {
            temp = i;
            break;
        }
    }
    cout << "请输入修改后学生的姓名:";
    cin >> Stu[temp].name;

    cout << "请输入修改后学生的性别:";

```

```

    cin >> Stu[temp].sex;

    cout << "请输入修改后学生的专业:";
    cin >> Stu[temp].class_info;

    cout << "请输入修改后学生的年级:";
    cin >> Stu[temp].age;

    cout << "请输入修改后学生的学号:";
    cin >> Stu[temp].number;

    cout << "请输入修改后学生的专业课成绩:";
    cin >> Stu[temp].Chinese;

    cout << "请输入修改后学生的微积分成绩:";
    cin >> Stu[temp].Math;

    cout << "请输入修改后学生的英语综合成绩:";
    cin >> Stu[temp].English;

    Stu[temp].Total = Stu[temp].Chinese + Stu[temp].Math + Stu[temp].English;
}

#pragma once
#include<iostream>
#include <string>
#include <fstream>

using namespace std;
class primarychengji {
private:
    int number;
    string name;
    string age;
    string sex;
    string class_info;
    float English;
    float Math;
    float Chinese;
    float Total;
public:
    void addstu(int nums, primarychengji stu[101],int &index);
    void search_by_name(string name, primarychengji stu[101],int index);
    void search_by_num(int num, primarychengji stu[101],int index);
    void delete_stu(string name, primarychengji stu[101],int &index);
    float CHI_average(primarychengji stu[101],int index);

```

```

float MATH_average(primarychengji stu[101], int index);
float ENG_average(primarychengji stu[101],int index);
void total_sorting(primarychengji stu[101],int index);
void CHI_sorting(primarychengji stu[101],int index);
void MATH_sorting(primarychengji stu[101],int index);
void ENG_sorting(primarychengji stu[101],int index);
void output_list(primarychengji stu[101],int amount);
void get_name();
void show_all(primarychengji stu[101],int index);
void change_stu1(string name, primarychengji stu[101], int index);
void change_stu2(int target, primarychengji Stu[101], int index);
};

```

```

#include "student.h"
#include <fstream>
#include "teacher.h"
#include "globalFile.h"
#include "computerRoom.h"

```

//默认构造

```
Teacher::Teacher()
```

```
{
}
```

//有参构造 (职工编号, 姓名, 密码)

```
Teacher::Teacher(int empId, string name, string pwd)
```

```
{
    this->m_EmpId = empId;
    this->m_Name = name;
    this->m_Pwd = pwd;

```

```
}
```

//本科院招生办信息记录

```
void Teacher::Zhaosheng()
```

```
{
    cout<<"请选择要进行的操作"<<endl;
    cout<<"1 表示进行记录新学生,2 表示查看所有学生"<<endl;
    cout<<"按 0 退出"<<endl;
    //假如写错了, 要怎么办将其修正呢?
    int mmm=0;

```



```

cin>>mmm;
if(mmm==1)
{
    string name,xingbie,jiguan,luquzhuanye,sushehao;
    double shenfenzheng;
    int gaokao,xuehao;
    fstream f;
    while(mmm==1)
    {
        f.open("student xiangxi.txt",ios::out|ios::app);
        cout<<"学生姓名: "<<endl;
        cin>>name;
        cout<<"性别: "<<endl;
        cin>>xingbie;
        cout<<"身份证号: "<<endl;
        cin>>shenfenzheng;
        cout<<"籍贯: "<<endl;
        cin>>jiguan;
        cout<<"高考成绩: "<<endl;
        cin>>gaokao;
        cout<<"录取学院与专业: "<<endl;
        cin>>luquzhuanye;
        cout<<"学号: "<<endl;
        cin>>xuehao;
        cout<<"宿舍号: "<<endl;
        cin>>sushehao;

        f<<name<<"\t"<<xingbie<<"\t"<<shenfenzheng<<"\t"<<jiguan<<"\t"<<gaokao<<"\t"<<luquzhuanye<<"\t"<
<xuehao<<"\t"<<sushehao<<"\t"<<endl;
        f.close();

        cout<<"接下来请选择: " <<endl;
        cin>>mmm;

    }}
else if(mmm=2)
{
    Teacher::showComputer();
    return;
}

else
{
    return;
}

```

```

}
//查看学生信息
void Teacher::showComputer()
{
    Teacher::initVector();
    cout << "学生详细信息如下： " << endl;
    for (vector<ComputerRoom>::iterator it = vCom.begin(); it != vCom.end(); it++)
    {
        cout << it->name << "\t" << it->xingbie << "\t" << it->shenfenzheng << "\t" << it->jiguan << "\t" <<
it->gaokao << "\t" << it->luquzhuanye << "\t" << it->m_Id << "\t" << it->sushehao << "\t" << endl;
    }
    system("pause");
    system("cls");
}
void Teacher::initVector(){

    vCom.clear();
    //获取学生信息
    ifstream ifs;

    ifs.open(COMPUTER_FILE, ios::in);
    if (!ifs.is_open())
    {
        cout << "文件读取失败" << endl;
        return;
    }
    ComputerRoom c;
    while (ifs >> c.name && ifs >> c.xingbie && ifs >> c.shenfenzheng && ifs >> c.jiguan && ifs >>
c.gaokao && ifs >> c.luquzhuanye && ifs >> c.m_Id && ifs >> c.sushehao)
    {
        vCom.push_back(c);
    }
    cout << "当前学生数量为： " << vCom.size() << endl;

    ifs.close();

}
void Teacher::initVector2(){

    vJia.clear();

    ifstream ifs;

    ifs.open(JIANGXUEJIN, ios::in);

```

```

if (!ifs.is_open())
{
    cout << "文件读取失败" << endl;
    return;
}
Jiangxuejin j;
while (ifs >> j.name && ifs >> j.m_Id && ifs >> j.jiangxuejin)
{
    vJia.push_back(j);
}
cout << "当前获奖学金的学生数量为:  " << vJia.size() << endl;

ifs.close();

}

```

```

//图书馆书籍管理
void Teacher::Shuji()
{
// 查看所有学生借阅的书
// 输入学号查看      一次性打开文件
int choice ;
fstream iof( "booksFile.dat" , ios::app|ios::binary|ios::in|ios::out );
if (!iof)
{ cerr << "文件不存在，创建一个新文件" << endl;
}
while (1)
{ cout << "***** 书库管理 *****\n 请键入操作选择\n";
cout << "\t 1 -在借书信息中添加记录" << endl;
cout << "\t 2 -按学号删除记录" << endl;
cout << "\t 3 -按学号查询图书信息" << endl;
cout << "\t 4 -建立和浏览图书文件" << endl;
// cout << "\t 5 -" << endl;
cout << "\t 0 -退出" << endl;
cin  >> choice;
switch ( choice )
{

    case 1 :Append(iof) ; break ;
    case 2 :Sale(iof) ; break ;
    case 3 :Inquire(iof) ; break;
    case 4 :CreateTxt(iof) ; break ;

```

```

        // case 5 :del(iof); break ;
        case 0 :cout<<"\t-退出程序的运行！ \n"<< endl; return ;
        default : cout<<"\t-输入错误， 请重新输入！ \n" <<endl;
    }
}
iof.close();      // 关闭文件
return ;

}

void Teacher::xuankesystem() {
    cout
    "-----" << endl;
    cout << "----- 欢 迎 来 到 选 课 系 统
    -----" << endl;
    cout
    "-----" << endl;
    cout << "***      < 1 >添加课程信息" << endl;
    cout << "***      < 2 >显示所有课程信息" << endl;
    cout << "***      < 3 >通过课程名查询课程信息" << endl;
    cout << "***      < 4 >通过课程号查询课程信息" << endl;
    cout << "***      < 5 >将记录保存为 txt 文件" << endl;
    cout << "***      < 6 >删除课程信息记录" << endl;
    cout << "***      < 0 >退出此系统" << endl;
    cout
    "-----" << endl;
    cout
    "-----" << endl;
}

void Teacher::xuanke() {
    primary xxs[101];
    int index = 0;
    int in;
    Teacher::xuankesystem();
    while (1) {
        bool flag1;
        cin >> in;
        if (in == 0) {
            cout << "感谢使用";
            break;
        }
        else if (in == 1) {
            int nums;
            cout << "请输入需要添加的课程数" << endl;
            cin >> nums;

```

```

        xxs->addstu(nums,xxs,index);
        Teacher::xuankesystem();
    }
    else if (in == 2) {
        xxs->show_all(xx, index);
        Teacher::xuankesystem();
    }
    else if (in == 3) {
        int d;
        cout << "请选择查询(1)或者修改(0)课程信息" << endl;
        cin >> d;
        if (d == 1) {
            cout << "请输入查询的课程名:"<<endl;
            string name;
            cin >> name;
            xxs->search_by_name(name,xxs,index);
            Teacher::xuankesystem();
        }
        else if (d == 0) {
            cout << "请输入需要修改的课程名： " << endl;
            string name;
            cin >> name;
            xxs->change_stu1(name, xxs, index);
            Teacher::xuankesystem();
        }
    }
    else if (in == 4) {
        int d;
        cout << "请选择查询(1)或者修改(0)课程信息" << endl;
        cin >> d;
        if (d == 1) {
            cout << "请输入查询的课程学号:" << endl;
            int target;
            cin >> target;
            xxs->search_by_num(target, xxs, index);
            Teacher::xuankesystem();
        }
        else if (d == 0) {
            cout << "请输入需要修改的课程学号： " << endl;
            int target;
            cin >> target;
            xxs->change_stu2(target, xxs, index);
            Teacher::xuankesystem();
        }
    }
}

```

```

        else if (in == 5) {
            xxs->output_list(xxs, index);
            Teacher::xuankesystem();
        }
        else if (in == 6) {
            cout << "请输入需要删除的课程的名字" << endl;
            string name;
            cin >> name;
            xxs->delete_stu(name, xxs, index);
            cout << "已删除该课程信息" << endl;
            Teacher::xuankesystem();
        }
    }

}

void Teacher::chengjisystem() {

    cout << "-----" << endl;
    cout << "----- 成 绩 管 理 -----" << endl;
    cout << "-----" << endl;

    cout << "-----" << endl;
    cout << "**      < 1 >添加学生成绩信息记录" << endl;
    cout << "**      < 2 >显示所有学生成绩信息" << endl;
    cout << "**      < 3 >通过姓名查询修改学生信息" << endl;
    cout << "**      < 4 >通过学号查询修改学生信息" << endl;
    cout << "**      < 5 >将记录内学生排序" << endl;
    cout << "**      < 6 >将记录保存为 txt 文件" << endl;
    cout << "**      < 7 >删除学生信息记录" << endl;
    cout << "**      < 0 >退出此系统" << endl;
    cout << "-----" << endl;
    cout << "-----" << endl;
}

void Teacher::Chengji() {
    primarychengji xxs[101];
    int index = 0;
    int in;
    Teacher::chengjisystem();
    while (1) {
        bool flag1;
        cin >> in;
        if (in == 0) {
            cout << "感谢使用";

```

```

        break;
    }
    else if (in == 1) {
        int nums;
        cout << "请输入需要添加的学生人数" << endl;
        cin >> nums;
        xxs->addstu(nums,xxs,index);
        Teacher::chengjisystem();
    }
    else if (in == 2) {
        xxs->show_all(xxs, index);
        Teacher::chengjisystem();
    }
    else if (in == 3) {
        int d;
        cout << "请选择查询(1)或者修改(0)学生信息" << endl;
        cin >> d;
        if (d == 1) {
            cout << "请输入查询的学生姓名:"<<endl;
            string name;
            cin >> name;
            xxs->search_by_name(name,xxs,index);
            Teacher::chengjisystem();
        }
        else if (d == 0) {
            cout << "请输入需要修改的学生姓名: " << endl;
            string name;
            cin >> name;
            xxs->change_stu1(name, xxs, index);
            Teacher::chengjisystem();
        }
    }
    else if (in == 4) {
        int d;
        cout << "请选择查询(1)或者修改(0)学生信息" << endl;
        cin >> d;
        if (d == 1) {
            cout << "请输入查询的学生学号:" << endl;
            int target;
            cin >> target;
            xxs->search_by_num(target, xxs, index);
            Teacher::chengjisystem();
        }
        else if (d == 0) {
            cout << "请输入需要修改的学生学号: " << endl;
            int target;

```

```

        cin >> target;
        xxs->change_stu2(target, xxs, index);
        Teacher::chengjisystem();
    }
}
else if (in == 5) {
    cout << "请选择需要的排序方式:"<<endl;
    int d;
    cout << "<1>根据总分排序" << endl;
    cout << "<2>根据专业成绩排序" << endl;
    cout << "<3>根据微积分排序" << endl;
    cout << "<4>根据英语综合排序" << endl;
    cin >> d;
    if (d == 1) {
        xxs->total_sorting(xxs, index);
    }
    if (d == 2) {
        xxs->CHI_sorting(xxs, index);
    }
    if (d == 3) {
        xxs->MATH_sorting(xxs, index);
    }
    if (d == 4) {
        xxs->ENG_sorting(xxs, index);
    }
    Teacher::chengjisystem();
}
else if (in == 6) {
    xxs->output_list(xxs, index);
    Teacher::chengjisystem();
}
else if (in == 7) {
    cout << "请输入需要删除的学生的名字" << endl;
    string name;
    cin >> name;
    xxs->delete_stu(name, xxs, index);
    cout << "已删除该学生信息" << endl;
    Teacher::chengjisystem();
}
}
}
}

```

//教务系统


```

void Teacher::Jiaowu()
{
//    1 课程添加
//    2 成绩录入，加排名
//
    cout<<"请选择模块"<<endl;
    cout<<"1 课程添加"<<endl;
    cout<<"2 成绩录入"<<endl;
    cout<<"请输入: "<<endl;
    int jiaowuint;
    cin>>jiaowuint;

    if (jiaowuint ==1)
    {

        Teacher::xuanke();
    }
    else if(jiaowuint==2)
    {
        Teacher::Chengji();
    }

}

//财务管理修改
void Teacher::Caiwu()
{
//    1 发奖学金
//    2 查看余额
    cout<<"请选择要进行的操作"<<endl;
    cout<<"1 表示进行发放学生奖学金,2 表示查看所有学生奖学金"<<endl;
    cout<<"按 0 退出"<<endl;
    int mmm=0;
    cin>>mmm;
    if(mmm==1)
    {
        string name;

        double jiangxuejin,xuehao;
        fstream f;
        while(mmm==1)
        {
            f.open(    JIANGXUEJIN,ios::out|ios::app);
            cout<<"学生姓名: "<<endl;
            cin>>name;

```

```

        cout<<"学号: "<<endl;
        cin>>xuehao;
        cout<<"奖学金: "<<endl;
        cin>>jiangxuejin;
        f<<name<<"\t"<<xuehao<<"\t"<<jiangxuejin<<"\t"<<endl;
        f.close();

        cout<<"接下来请选择: " <<endl;
        cin>>mmm;

    }}
    else if(mmm==2)
    {
        Teacher::initVector2();
        cout << "获奖学金的学生: " << endl;
        for (vector<Jiangxuejin>::iterator it = vJia.begin(); it != vJia.end(); it++)
        {
            cout<<it->name<<"\t"<<it->m_Id<<"\t 奖学金"<<it->jiangxuejin<<"\t"<<endl;

        }
        system("pause");
        system("cls");
        return;
    }
    else
    {
        return;
    }
}

//后勤服务保障
void Teacher::Houqin()
{
    //查找 学生和老师
    //宿舍和前面联系起来
    Teacher::initVector();
    cout << "学生详细信息如下: " << endl;
    for (vector<ComputerRoom>::iterator it = vCom.begin(); it != vCom.end(); it++)
    {
        cout << it->name <<"\t"<< it->xingbie<<"\t"<< "学号 "<<"\t"<< it->m_Id<<"\t 宿舍号 "<<
it->sushehao<<"\t"<< endl;
    }
    system("pause");
    system("cls");
}
}

```

```

//默认构造
Student::Student()
{
}

//有参构造(学号、姓名、密码)
Student::Student(int id, string name, string pwd)
{
    this->m_Id = id;
    this->m_Name = name;
    this->m_Pwd = pwd;
}

//菜单界面
void Student::operMenu()
{
    cout <<"----- 请 选 择 要 访 问 的 项 目
-----" <<endl;
    cout <<"-----1 本科院招生办-----" <<endl;
    cout <<"-----2 图书馆-----" <<endl;
    cout <<"-----3 教务系统-----" <<endl;
    cout <<"-----4 财务管理-----" <<endl;
    cout <<"-----5 后勤服务-----" <<endl;
    cout <<"-----0 注销登录-----" <<endl;
    cout <<"请选择您的操作: " <<endl;
}

//本科院招生办
void Student::Zhaosheng()
{
    //查找自己的信息 (身份证.....
    ifstream f("student xiangxi.txt", ios::in);
    if (!f)
    {
        cout <<"error" <<endl;
        return ;
    }

    string line;
    string a;
}

```

```

//图书馆
void Student::Shuji()
{
    //查找的时候再设一个函数 没有为零
    // 借几本书
    // 借书 ++
    // 还书 --
    //
    fstream iof( "booksFile.dat" , ios::app|ios::binary|ios::in|ios::out );
    Inquire(iof);
}

```

```

//教务系统
void Student::Jiaowu()
{
    // 1 课程号 课程名称 学时 老师 （查找，
    // 2 查看成绩与排名

```

```

    cout<<"请选择模块"<<endl;
    cout<<"1 课程查询"<<endl;
    cout<<"2 成绩查询"<<endl;
    cout<<"请输入： "<<endl;
    int jiaowuint;
    cin>>jiaowuint;

```

```

    if (jiaowuint ==1)
    {
//        Teacher::xuanke();

//        Student::xuanke();
    }
    else if(jiaowuint==2)
    {
        //Teacher::Chengji();
    }

```

```

}

```

```

//财务管理
void Student::Caiwu()
{

```

```

// 1 查看自己的余额
// 2 充值 （直接文件修改
// 3 查看奖助学金
}

//后勤服务
void Student::Houqin()
{
// 查自己宿舍，查第一个文件
}

#pragma once
#include<iostream>
using namespace std;
#include "identity.h"

#define _CRT_SECURE_NO_WARNINGS

#include<vector>

#include "ex10_15.h"
#include "primary.h"
#include "primarychengji.h"

//学生类
class Student :public Identity
{
public:
    //默认构造
    Student();

    //有参构造(学号、姓名、密码)
    Student(int xuehao, string name, string pwd);

    //菜单界面
    virtual void operMenu();

    //本科院招生办信息记录
    void Zhaosheng();

    //图书馆书籍管理
    void Shuji();

    //教务系统

```

```

        void Jiaowu();
//    void xuanke();

//财务管理修改
void Caiwu();

//后勤服务保障
void Houqin();

//学生学号
int m_Id;

private :
    int xuehao;

};
//chengji 类
class Score{

public:

float cLauage;//c 语言成绩

float math;//高数成绩

float english;//英语成绩

};

#include"teacher.h"
#include <fstream>
#include"globalFile.h"
#include"computerRoom.h"

//菜单界面
void Teacher::operMenu()
{
    cout    <<"-----    请    选    择    要    访    问    的    项    目
-----"<<endl;
    cout    <<"-----1    本    科    院    招    生    办    信    息    记    录
-----"<<endl;
    cout    <<"-----2    图    书    馆    书    籍    管    理
-----"<<endl;
    cout <<"-----3 教务系统-----"<<endl;

```

```

    cout <<"-----4 财务管理修改-----" <<endl;
    cout <<"-----5 后勤服务保障-----" <<endl;
    cout <<"-----0 注销登录-----" <<endl;
    cout <<"请选择您的操作： " <<endl;
}

```

```

#pragma once
#define _CRT_SECURE_NO_WARNINGS
#include<iostream>
using namespace std;
#include "identity.h"
#include<vector>
#include"computerRoom.h"
#include "ex10_15.h"
#include "primary.h"
#include "primarychengji.h"

class Teacher :public Identity
{
public:

    //默认构造
    Teacher();

    //有参构造 (职工编号, 姓名, 密码)
    Teacher(int empId, string name, string pwd);

    //菜单界面
    virtual void operMenu();

    //本科院招生办信息记录
    void Zhaosheng();
    //查看学生信息
    void showComputer();

    //图书馆书籍管理
    void Shuji();

    //教务系统
    void Jiaowu();
    void chengjisystem();
    void Chengji() ;
    void xuankesystem();
    void xuanke();

```

```

//财务管理修改
void Caiwu();

//后勤服务保障
void Houqin();

int m_EmpId; //教师编号

void initVector();

vector<ComputerRoom> vCom;

void initVector2();

vector<Jiangxuejin> vJia;

};

#include "xinxiManager.h"

xinxiManager::xinxiManager()
{

}

void xinxiManager::show_Menu()
{
    cout
<<"-----"<<endl;
    cout <<"----- 欢 迎 来 到 学 生 信 息 管 理 系 统
-----"<<endl;
    cout <<"----- 选 择 要 进 入 的 系 统 :(1 为 学 生 /2 为 职
工)-----"<<endl;
    cout <<"----- 请 输 入 1 或 者
2-----"<<endl;
    cout <<"----- 按 0 退 出 系 统
-----"<<endl;
    cout
<<"-----"<<endl;
    cout
<<"-----"<<endl;

}

```



```

void xinxiManager::exitSystem()
{
    cout << "欢迎下次使用" << endl;
    system("pause");
    exit(0);
}

int xinxiManager::system1()
{
    cout << "----- 请 选 择 要 访 问 的 项 目
-----" << endl;

    cout << "-----1 本科院招生办-----" << endl;
    cout << "-----2 图书馆-----" << endl;
    cout << "-----3 教务系统-----" << endl;
    cout << "-----4 财务管理-----" << endl;
    cout << "-----5 后勤服务-----" << endl;

    int s1;
    cin >> s1;
    if (s1==1||s1==2||s1==3||s1==4||s1==5)
    {
        return s1;
    }
    else
    {
        return 0;
    }
}

int xinxiManager::system2()
{
    cout << "----- 请 选 择 要 访 问 的 项 目
-----" << endl;

    cout << "-----1 本 科 院 招 生 办 信 息 记 录
-----" << endl;

    cout << "-----2 图 书 馆 书 籍 管 理
-----" << endl;

    cout << "-----3 教务系统-----" << endl;
    cout << "-----4 财务管理修改-----" << endl;
    cout << "-----5 后勤服务保障-----" << endl;

    int s2;
    cin >> s2;
    if (s2==1||s2==2||s2==3||s2==4||s2==5)
    {
        return s2;
    }
    else
    {
        return 0;
    }
}

```

```

}
}

void xinxiManager::createStudent()
{
    string name,xingbie,jiguan,luquzhuanye;
    int shenfenzheng,gaokao,xuehao;
    cout<<"学生姓名: "<<endl;
    cin>>name;
    cout<<"性别: "<<endl;
    cin>>xingbie;
    cout<<"身份证号: "<<endl;
    cin>>shenfenzheng;
    cout<<"籍贯: "<<endl;
    cin>>jiguan;
    cout<<"高考成绩: "<<endl;
    cin>>gaokao;
    cout<<"录取学院与专业: "<<endl;
    cin>>luquzhuanye;
    cout<<"学号: "<<endl;
    cin>>xuehao;

}

//xinxiManager::saveStudent()
//{
//    ofstream ofs;
//    ofs.open("student.csv", ios::out | ios::app); // 用输出的方式打开文件 -- 写文件
//    //将每个人数据写入到文件中
//    for (vector<int>::iterator it = vVictory.begin(); it != vVictory.end(); it++)
//    {
//        ofs << *it << ","
//        << m_Speaker[*it].m_Score[1] << ",";
//    }
//    ofs << endl;
//    //关闭文件
//    ofs.close();
//    cout << "记录已经保存" << endl;
//}

xinxiManager::~xinxiManager()
{
}

```

```
#pragma once
#include<iostream>
using namespace std;

//信息管理类
class xinxiManager
{
public:

    //构造函数
    xinxiManager();

    //展示菜单
    void show_Menu();

    void exitSystem();

    void initStudent();
    int  system1();
    int  system2();
    void createStudent();
    void saveStudent();

    //析构函数
    ~xinxiManager();
};
```