



# 实 验 报 告

实验题目：一个简单的选课系统

院（系）：数据科学与计算机学院

专业（班级）：计算机类

实验人：邱圣浩

学号：16337200

日期：2017 年 6 月 17 号

## 一. 实验目的

本实验面向 C++ 语言的初学者。

主要让实验者熟悉面向对象的编程思想以及类的使用。

## 二. 实验环境

本实验可基于 Visual Studio 或 Eclipse 或 Devc++ 平台开发，参考主流的编码规范，如 [Google C++ Style Guide \(中文版\)](#)

### 2.1 编程语言和开发工具

编程语言：ANSI C/C++

开发工具：Visual Studio 2017

### 2.2 编码规范

要求遵循良好的程序设计风格来设计和编写程序。基本编码规范：

1. 标识符的命名要到达顾名思义的程度；
2. 关键代码提供清晰、准确的注释；
3. 程序版面要求：
  - a) 不同功能块用空行分隔；
  - b) 一般一个语句一行；
  - c) 语句缩进整齐、层次分明。

## 三. 实验内容

实现一个简单的选课系统，可以提供给三种不同身份的用户使用（老师、学生、管理员）。

老师可用功能：查看所有课程、查看自己需要上的课程；

学生可用功能：查看所有课程、选课、退课、查看自己已有的课程

管理员功能：查看所有功能、增添课程、删除课程、删除学生、删除老师

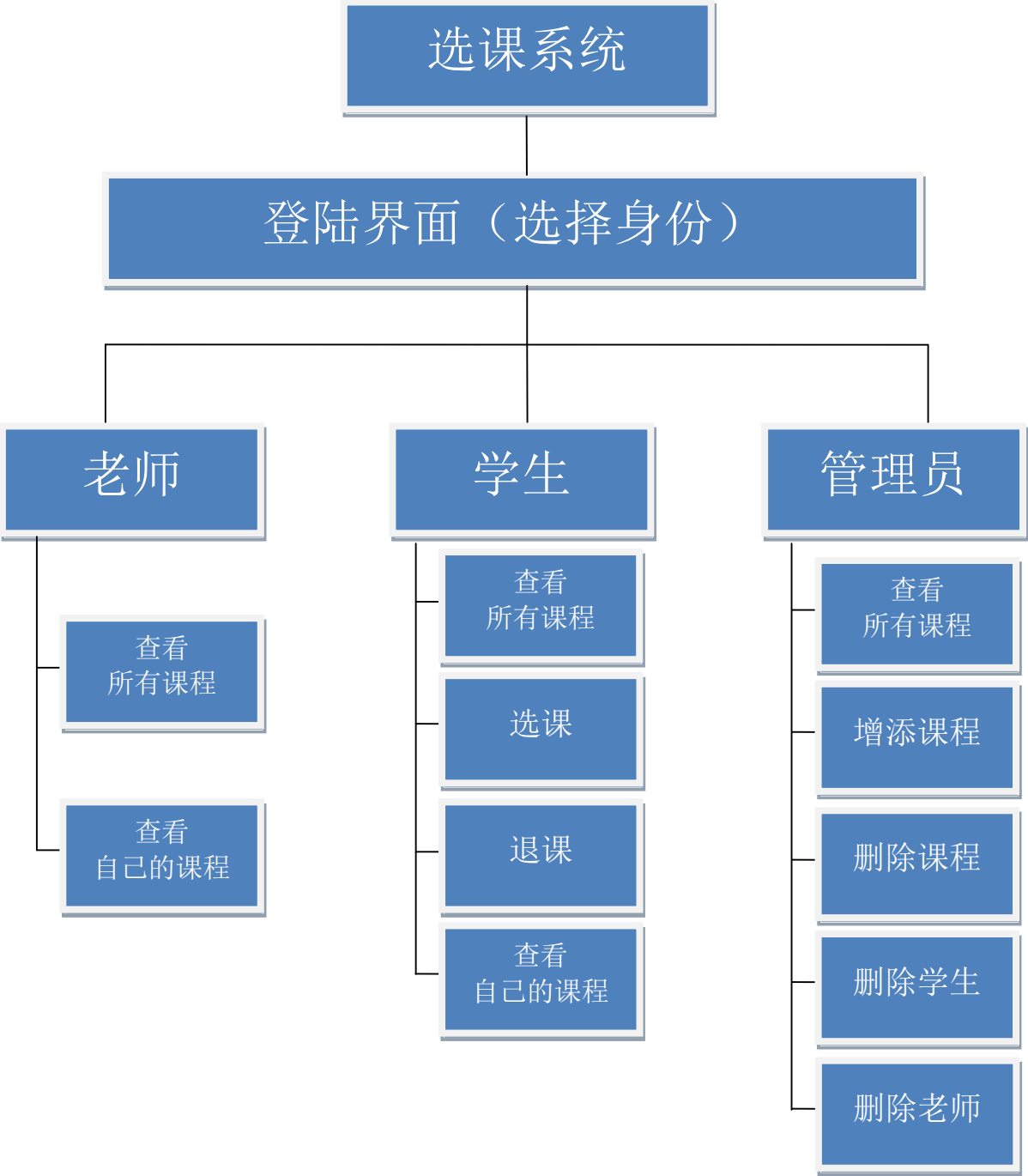
## 四. 分析与设计

### 4.1 、需求分析：

用户需要的是一个可以进行简单的选课退课查询等操作的程序，其需实现的功能如下：

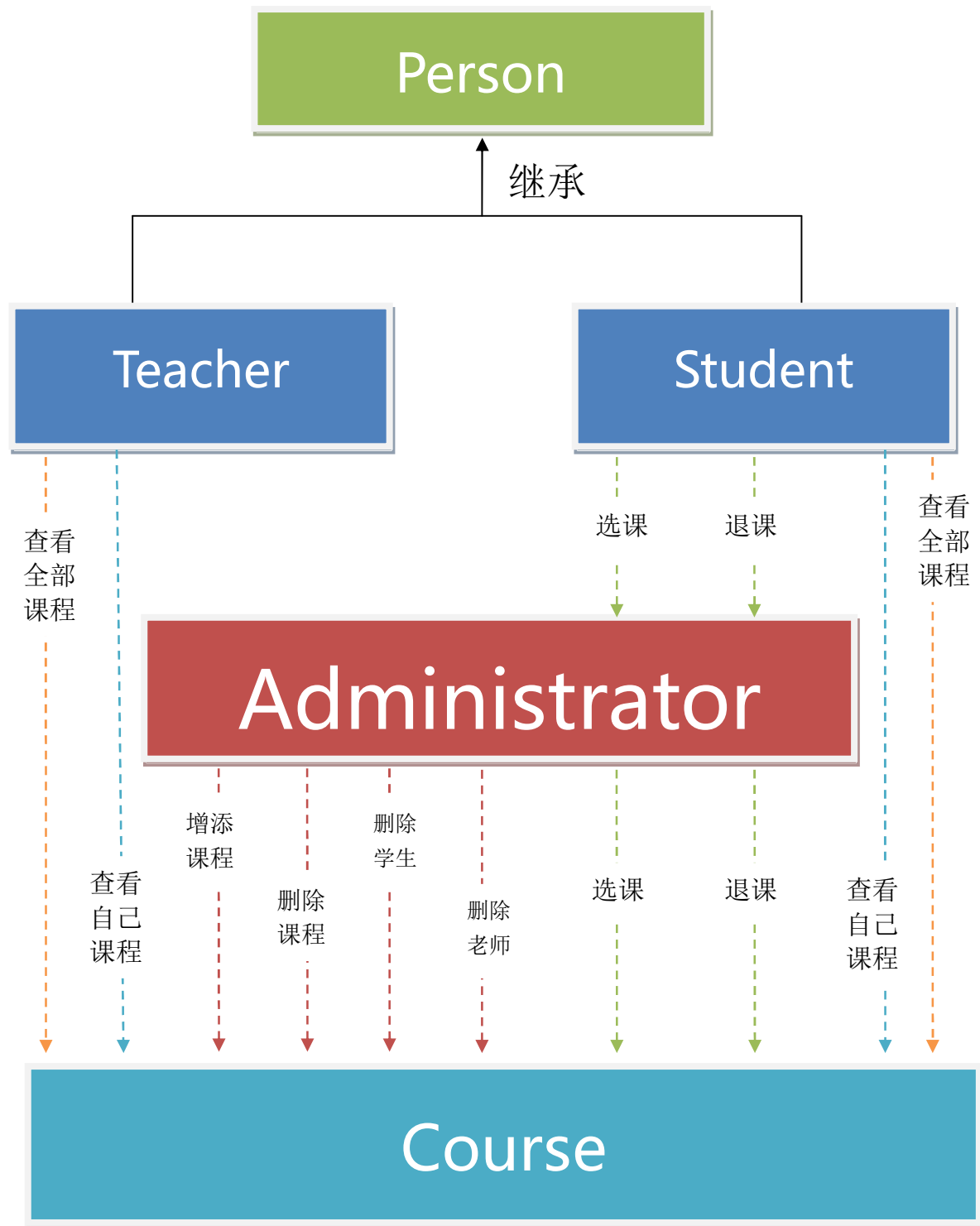
1. 简单的登陆界面（选择身份、输入用户密码）
2. 用户选择了相应的身份后，可以使用其身份对应的功能：
  - 1) 老师身份：
    - ①查看所有的课程
    - ②查询自己需要上的课程
  - 2) 学生身份：
    - ①查看所有课程
    - ②选课
    - ③退课
    - ④查看自己已有的课程
  - 3) 管理员身份：
    - ①查看所有课程
    - ②增添课程
    - ③删除课程
    - ④删除学生
    - ⑤删除老师

系统功能图



## 4.2、类结构设计

类关系图：



此程序一共用了五个类：

Person, Teacher, Student, Administrator 和 Course，其中 Teacher 和 Student 两个类都是 Person 类的派生类。

Teacher 类和 Student 类查看课程详细信息时直接访问 Course 中存放所有课程的一个静态的 public 的 map 成员。

Student 类进行选课和退课操作时通过 Administrator 类来间接对 Course 类进行操作。

Administrator 类还负责课程的增删、删除学生和老师的操作。

Person 类则主要存储了个人信息中的密码、名字、个人课程编号和 ID，Student 类（Teacher 类）中还有存放着所有学生（老师）的 map 成员，Course 类则主要负责记录课程的信息。

### 4.3、细节设计

接口设计：

**Administrator 类：**

```
public:
    Administrator();
    static Administrator* getAdministrator();
    bool checkPassword(const string&); //获取管理员密码
    bool chooseCourse(const int&, const int&); //通过老师的ID和课程的编号来选课
    bool dropCourse(const int&, const int&); //通过老师的ID和课程的编号来退课
    int addCourse(const int&, const string&); //通过老师的ID和课程的名字来增添课程
    bool deleteCourse(const int&); //通过课程的编号来删除课程
    bool deleteStudent(const int&); //通过学生的ID来删除学生
    bool deleteTeacher(const int&); //通过老师的ID来删除老师
```

## Course 类:

```
static int sum; //用于记录课程的总数目（包括被删的）并给课程编号
static map<int, Course*> allCourse;
Course(const string&); //以课程的名字为参数的构造函数
void setTime(const string&);
void setPosition(const string&);
void setTeacherName(const string&);
void setCourseName(const string&);
void setSerialNum(const int&);
const int getSerialNum();
const int getStudentSum();
const int getStudentId(const int&);
const string getTime();
const string getPosition();
const string getTeacherName();
const string getCourseName();
bool findStudent(const int&);
void addStudent(const int&); //课程被该编号的学生选中
void popStudent(const int&); //课程被该编号的学生退选
void display(); //显示该门课的详细信息
static void showAllCourse(); //显示全部课程的详细信息
void readStudent(ifstream&); //从文件中读取该门课程学生的ID信息并写入内存
void storeStudent(ostream&); //将该门课程学生的ID信息写到文件中
```

## Person 类:

```
public:
    void setName(const string&);
    void setId(const int&);
    void setPassword(const string&);
    const string getName() const;
    const string getPassword() const;
    const int getCourseSum() const;
    const int getId() const;
    void addCourse(const int&);
    void deleteCourse(const int&);
    void showOwnCourse() const;
    bool checkPassword(const string&);
    void readCourse(ifstream&); //从文件中读取该老师的课程编号信息并写入内存
    void storeCourse(ostream&); //将该老师自己的课程编号信息写到文件中
```

### Student 类:

```
public:
    static map<int, Student*> allStudent;//用于储存所有学生信息
    Student(const int&);//以学生的ID为参数的构造函数
    void chooseCourse(const int&);//选课
    void dropCourse(const int&);//退课
    void popAllCourse();//退选该学生的所有课程
```

### Teacher 类:

```
public:
    static map<int, Teacher*> allTeacher;//储存所有老师的信息
    Teacher(const int&);//以老师的ID为参数的构造函数
    void popAllCourse();//删除这老师任教的所有课程
```

### 数据成员设计:

### Administrator 类:

```
private:
    string password;
    static Administrator *administrator;
```

### Course 类:

```
private:
    int serialNum;
    string time, position, teacherName, courseName;
    vector<int> studentList;//存放选择此门课程的学生的ID
```

### Person 类:

```
private:
    int id;
    string name, password;
protected:
    vector<int> ownCourse;//记录自己所有课程的编号
```



## 五、实验结果

### 1、选择身份界面：

```
请选择你的权限：  
1. 管理员  
2. 老师  
3. 学生  
4. 退出系统  
>>> 3_
```

### 2、选择了学生身份：

```
请选择你要进行的操作：  
1. 登陆已有的账号  
2. 注册新的账号  
3. 返回  
>>> 1
```

### 登陆界面：

```
请输入你的学号：  
>>> 16337200  
请输入你的密码：  
>>> 16337200_
```

选择操作界面：

```
你好，zero同学，欢迎使用中山大学选课系统
```

```
请选择你要进行的操作：
```

1. 查看所有课程
  2. 选课
  3. 退课
  4. 查看自己已有的课程
  5. 返回选择权限菜单
- ```
>>> 1_
```

操作 1：查看所有课程：

```
以下是选课系统中可供选择的课程：
```

| Course_SerialNum | Course_Name | Teacher_Name | Time     | Position |
|------------------|-------------|--------------|----------|----------|
| 0                | C++         | yyh          | Mon:4-5  | C403     |
| 1                | C++         | yyh          | Fri:2-3  | B201     |
| 2                | 数学分析        | tzj          | Mon:2-3  | C501     |
| 3                | 数学分析        | tzj          | Wed:4-5  | D301     |
| 4                | 离散数学基础      | llz          | Tue:4-5  | C502     |
| 5                | 离散数学基础      | llz          | Thur:3-5 | C502     |

```
按任意键返回上一级菜单. . .
```

操作 4：查看自己已有的课程：

```
以下是你的所有课程：
```

| Course_SerialNum | Course_Name | Teacher_Name | Time     | Position |
|------------------|-------------|--------------|----------|----------|
| 0                | C++         | yyh          | Mon:4-5  | C403     |
| 1                | C++         | yyh          | Fri:2-3  | B201     |
| 3                | 数学分析        | tzj          | Wed:4-5  | D301     |
| 4                | 离散数学基础      | llz          | Tue:4-5  | C502     |
| 5                | 离散数学基础      | llz          | Thur:3-5 | C502     |

```
按任意键返回上一级菜单. . .
```

操作 2：选课：

```
请输入你想要选择的课程的课程编号：
```

```
>>> 2
```

```
选课成功！
```

```
输入“b”返回上一级菜单，按任意键继续进行选择. . .
```

```
b_
```

```

以下是你的所有课程:
Course_SerialNum Course_Name      Teacher_Name  Time          Position
0                C++                yyh           Mon:4-5       C403
1                C++                yyh           Fri:2-3       B201
3                数学分析          tzj           Wed:4-5       D301
4                离散数学基础      llz           Tue:4-5       C502
5                离散数学基础      llz           Thur:3-5      C502
2                数学分析          tzj           Mon:2-3       C501
-----
按任意键返回上一级菜单. . .

```

### 操作 3: 退课:

```

请输入你想要退选的课程编号:
>>> 2
退课成功!
输入“b”返回上一级菜单, 按任意键继续进行选择. . .

```

```

以下是你的所有课程:
Course_SerialNum Course_Name      Teacher_Name  Time          Position
0                C++                yyh           Mon:4-5       C403
1                C++                yyh           Fri:2-3       B201
3                数学分析          tzj           Wed:4-5       D301
4                离散数学基础      llz           Tue:4-5       C502
5                离散数学基础      llz           Thur:3-5      C502
-----
按任意键返回上一级菜单. . .

```

### 操作 5: 返回选择权限菜单:

```

请选择你的权限:
1. 管理员
2. 老师
3. 学生
4. 退出系统
>>> 2

```

### 3、选择了老师身份:

```

请选择你要进行的操作:
1. 登陆已有的账号
2. 注册新的账号
3. 返回
>>> 1

```

登陆：

```
请输入你的工号：
>>> 1
请输入你的密码：
>>> 1
```

```
你好，yyh老师，欢迎使用中山大学选课系统
-----
请选择你要进行的操作：
1. 查看所有课程
2. 查看自己已有的课程
3. 返回选择权限菜单
>>> _
```

操作 1：查看所有课程：

```
以下是选课系统中可供选择的课程：
Course_SerialNum Course_Name      Teacher_Name  Time          Position
0                C++          yyh           Mon:4-5       C403
1                C++          yyh           Fri:2-3       B201
2                数学分析    tzj           Mon:2-3       C501
3                数学分析    tzj           Wed:4-5       D301
4                离散数学基础  llz           Tue:4-5       C502
5                离散数学基础  llz           Thur:3-5      C502
-----
按任意键返回上一级菜单. . .
```

操作 2：查看自己已有的课程：

```
以下是你的所有课程：
Course_SerialNum Course_Name      Teacher_Name  Time          Position
0                C++          yyh           Mon:4-5       C403
1                C++          yyh           Fri:2-3       B201
-----
按任意键返回上一级菜单. . .
```

操作 3：返回选择权限菜单：

```
请选择你的权限：
1. 管理员
2. 老师
3. 学生
4. 退出系统
>>>
```

4、选择了管理员身份：

登陆界面：

```
请输入密码：
>>> 666666_
```

```
你好，管理员，欢迎使用中山大学选课系统
-----
请选择你要进行的操作：
1. 查看所有课程
2. 增添课程
3. 取消课程
4. 删除学生
5. 删除老师
6. 返回选择权限菜单
>>>
```

操作 1：查看所有课程

```
以下是选课系统中可供选择的课程：
Course_SerialNum Course_Name      Teacher_Name  Time          Position
0                C++          yyh           Mon:4-5       C403
1                C++          yyh           Fri:2-3       B201
2                数学分析     tzj           Mon:2-3       C501
3                数学分析     tzj           Wed:4-5       D301
4                离散数学基础  llz           Tue:4-5       C502
5                离散数学基础  llz           Thur:3-5      C502
-----
按任意键返回上一级菜单. . .
```

操作 2：增添课程：

```
请输入你要创建的课程名称：
>>> test
请输入任课老师的工号：
>>> 1
请输入此课程的上课时间：
>>> time
请输入此课程的上课地点：
>>> position_
```

```
成功添加此课程到系统。
输入“b”返回上一级菜单，按任意键重新进行操作. . .
```

```
以下是选课系统中可供选择的课程：
Course_SerialNum Course_Name      Teacher_Name  Time          Position
0                C++          yyh           Mon:4-5       C403
1                C++          yyh           Fri:2-3       B201
2                数学分析     tzj           Mon:2-3       C501
3                数学分析     tzj           Wed:4-5       D301
4                离散数学基础  llz           Tue:4-5       C502
5                离散数学基础  llz           Thur:3-5      C502
6                test          yyh           time          position
-----
按任意键返回上一级菜单. . .
```

操作 3：删除课程：

```
请输入你要删除的课程的编号:
>>> 6
成功将此课程从系统中移除
输入“b”返回上一级菜单, 按任意键重新进行操作. . .
_
```

```
以下是选课系统中可供选择的课程:
Course_SerialNum Course_Name      Teacher_Name  Time          Position
0                C++          yyh           Mon:4-5       C403
1                C++          yyh           Fri:2-3       B201
2                数学分析      tzj           Mon:2-3       C501
3                数学分析      tzj           Wed:4-5       D301
4                离散数学基础  llz           Tue:4-5       C502
5                离散数学基础  llz           Thur:3-5      C502
-----
按任意键返回上一级菜单. . .
```

#### 操作 4: 删除学生:

```
请输入你要从系统中移除的学生的学号:
>>> 16337200_
```

```
成功把此学生从系统中删除。
>>> 输入“b”返回上一级菜单, 按任意键重新进行操作. . .
```

删除后登陆已有学生账号会显示该学号未注册:

```
请输入你的学号:
>>> 16337200
【错误】该学号尚未注册
输入“b”返回上一层界面, 按任意键重新输入. . .
>>>
```

#### 操作 5: 删除老师:

```
请输入你要从系统中移除的老师的工号:
>>> 1
成功把此老师从系统中删除。
>>> 输入“b”返回上一级菜单, 按任意键重新进行操作. . .
_
```

删除该老师后, 其任教的课程也会被删除:

```
以下是选课系统中可供选择的课程:
Course_SerialNum Course_Name      Teacher_Name  Time          Position
2                数学分析      tzj           Mon:2-3       C501
3                数学分析      tzj           Wed:4-5       D301
4                离散数学基础  llz           Tue:4-5       C502
5                离散数学基础  llz           Thur:3-5      C502
-----
按任意键返回上一级菜单. . .
```



## 操作 6：返回选择权限菜单：

```
请选择你的权限：
1. 管理员
2. 老师
3. 学生
4. 退出系统
>>> _
```

## 4、退出系统：

```
请选择你的权限：
1. 管理员
2. 老师
3. 学生
4. 退出系统
>>> 4
Bye-bye!
按任意键继续. . .
```

## 六、设计心得

- 1、对原来的项目进行了修改，发现自己之前写的代码的结构挺乱的，而且修改起来太麻烦了。。。
- 2、完善了上一次项目没能做出的文件输入输出
- 3、用 map 来储存所有的学生老师和课程信息，使代码的复杂程度减少了不少。。。。。
- 4、增强了系统的健壮性：在应该输入数字的地方输入字符也不会出现死循环错误了，也算是完成了这学期项目里面一直没能实现的功能。

具体写法如下：

```

int enterNum(string& str, int& choose, stringstream& ss) {
    try {
        getline(cin, str);
        for (int i = 0; i < str.length(); i++) {
            if (str[i] < '0' || str[i] > '9')
                throw inputError();
        }
        ss << str;
        ss >> choose;
        ss.clear();
        return 0;
    }
    catch (inputError& ex) {
        system("cls");
        cout << "【非法输入】此处请输入数字\n";
        cout << "输入“b”返回上一级菜单，按任意键重新进行操作. . . \n";
        getline(cin, str);
        system("cls");
        if (str == "b")
            return -1;
        return 1;
    }
}

```

此处使用了 try、catch 结构，个人觉得要实现这个小功能的关键是程序的所有输入都要改成用 getline（）而不用 cin，把所有的输入都存进 string 再对它进行处理。

- 5、 发现 Student 类和 Teacher 类的重用率太高了，所以加入了一个新的 Person 类，并用它来派生出 Student 类和 Teacher 类。
- 6、 由于时间原因，来不及把整个系统都改成使用中介者模式，只是把其中一部分改成那样子的模式而已（比如选课和退课的时候），个人觉得中介者模式让程序的思路清晰了不少，就是 debug 需要修改的时候麻烦了一点
- 7、 最后说一点再 debug 的时候出现的小问题：  
代码出现这个错误：





```
3156     template<class _InIt,  
3157               class _Ty> inline  
3158     _InIt _Find_unchecked1(_InIt _First, _InIt _Last, const _Ty& _Val, false_type)  
3159     { // find first matching _Val  
3160       for (; _First != _Last; ++_First)  
3161         if (*_First == _Val)  
3162           break;  
3163       return (_First);  
3164     }
```

只好把一个个函数模块注释掉,再一个个取消注释运行来找 bug 的位置。最后发现原来是这里的 find 用错了:

//此处的 allTeacher 是一个 map 容器

```
/*find(Teacher::allTeacher.begin(), Teacher::allTeacher.end(), teacher)
```

改成

```
if (Teacher::allTeacher.find(teacher)/*find(Teacher::al
```

这样就对了!

因为上面 find () 函数的哪个用法是对于 vector 容器的用法,  
在 map 中并没有此用法。