

**课 程 设 计 报 告 书**

**实验课程名称 《程序设计课程设计》**

**开 课 学 院 网络空间安全学院**

**指导老师姓名 姚晔/申延召**

**小 组 成 员： F001 （19061440）**

**Se4ky（19201209）**

**GCY （19081113）**

**2020 年 12 月 30 日**

选题名称：学生信息管理系统

**要求一级标题选择列表中的标题1，二级标题先择列表中的标题2，三级标题选择列表中的标题3，不建议使用四级及以上标题。正文均为5号宋体，首行缩进2字符，行间距固定20磅，两端对齐。**

# 1. 目的和要求

本组选题为学生信息管理系统，我们对所选择系统进行分析后，发现目前已经存在的相关系统存在着诸如系统界面过于苍白，系统在人机交互的过程中人员操作感差等问题。我们希望能够在传统的信息管理系统的基础上，优化相关数据结构，改良操作界面的人员交互性。在优化相关界面的过程中，我们遇到的问题是若基于原有的dos框架，则操作界面不论如何优化，仍然有一定的困难；若调用其他的文件，则代码量及相关课外内容较多。这是二者需要的权衡的利弊。最终我们采用了后者来进行相关的创新。我们还预计设置两个菜单界面，在欢迎界面设置两个部分，一个部分由管理员进入，管理员部分进入后需要进行密码的验证，可以通过，在用户界面上将会显示为教师系统；另一部分在创新点之外，我们还应会保证基础功能的正常实现，例如增删查改的基础存在性。其中，学生信息将会以学号为主键进行保存。

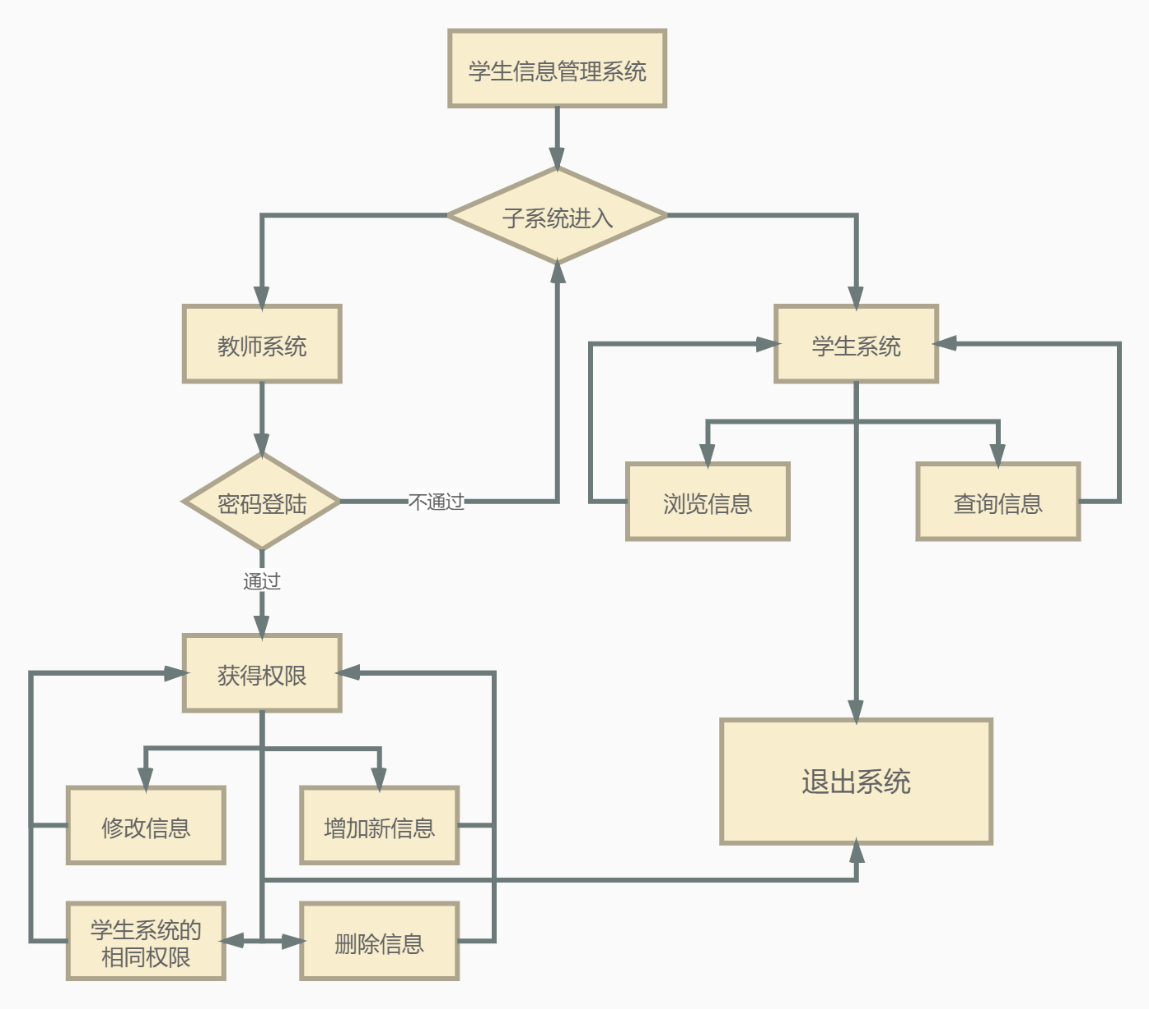
学生信息包括：学号，姓名，年龄，性别，出生年月，电话，E-mail，高等数学成绩、大学英语成绩、线性代数成绩等。

学生系统功能包括：信息查询

教师系统功能包括：信息浏览、信息查询、信息录入、信息修改

# 2. 总体设计

该思路基本如下：



基本分为两个子系统，教师系统和学生系统，其中学生系统用于浏览信息和查询信息，教师系统用于查询信息、新增信息、删除信息以及学生系统存在的相关功能。

设计思路为自顶向下依次实现，再自下而上逐个整合的操作方式。

# 3. 数据及数据结构设计描述

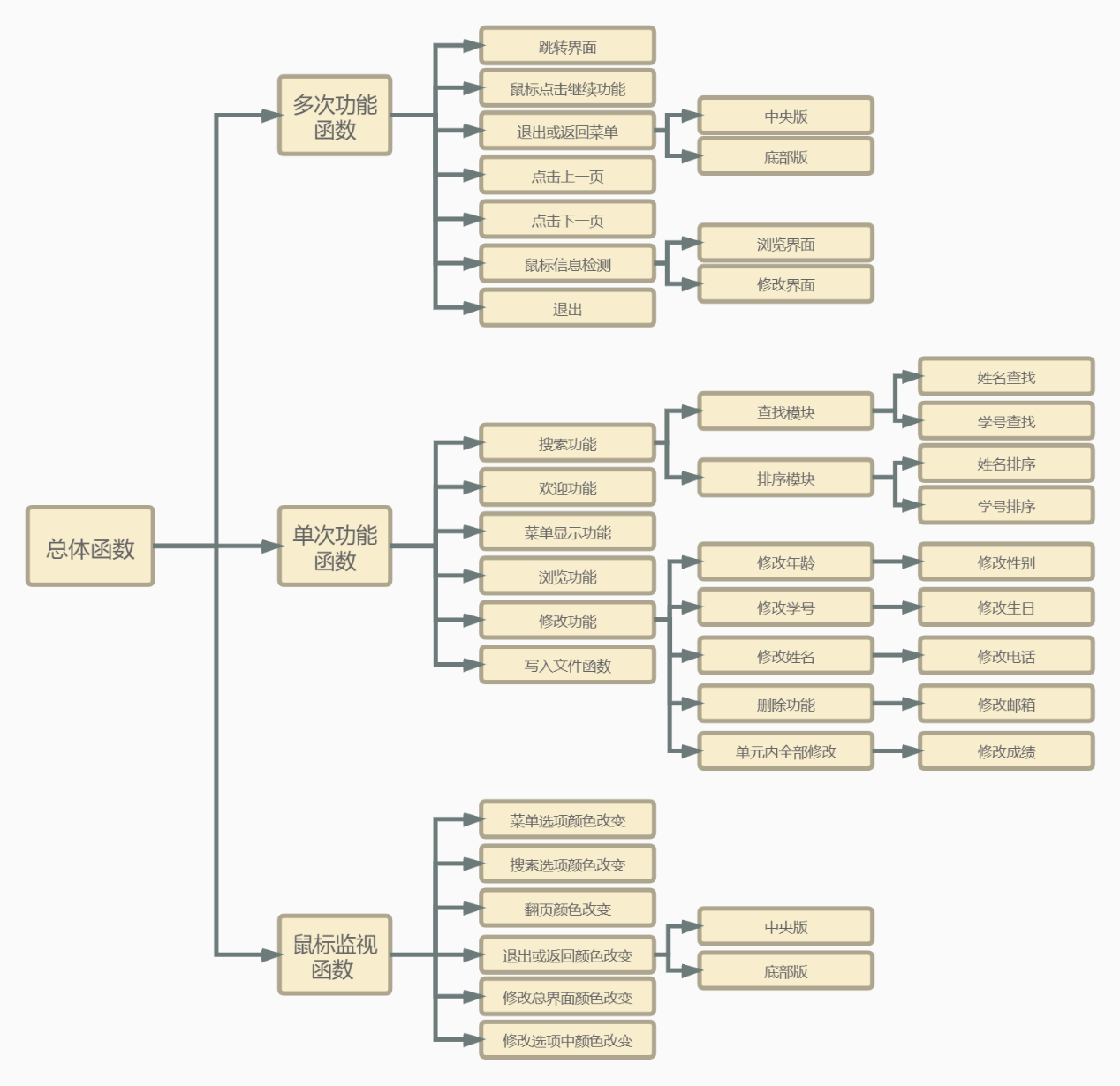
主要的数据结构为三个结构体，分别为学生信息结构体、学生成绩结构体、学生结点结构体。学生信息结构体体包括学号，姓名，年龄，性别，出生年月，电话，E-mail共计7个数据成员；学生成绩结构体包括高等数学成绩、大学英语成绩、线性代数成绩共计3个数据成员；学生结点结构体包括一个学生信息结构体、一个学生成绩结构体、一个头指针和一个尾指针。

两个全局变量：page1和page2分别表示总页码数量和当前页码，用于制作浏览时产生的目录。

所有的学生信息都通过链表的形式进行存储，通过双链表进行相应的删除、查找、排序等基本操作。选用链表的理由是其插入、删除等速度更快。

# 4. 详细设计

相关函数主要分布如下：



相关模块设计如下：

## 4.1学生模块

### 4.1.1 搜索功能

该功能是学生模块下的一个子模块。该功能可以进行按照学号的查找操作。由于链表的相关特性，进行排序完成后，无法提高查找的效率，我们考虑通过跳表的方式进行相关的查找，跳表是应用在链表下的数据结构，利用二分的方式进行。

在查找的过程中，我们针对输入的学号进行查找，由于仅有学号查找一种方式，因此我们只会显示一个学生的相关信息。

## 4.2教师模块

### 4.2.1登陆功能

该功能分为多个子模块，首先包括了教师的登陆情况。我们设计了一组账号和密码，要求输入账号与密码进行验证。这里的账号是我们在后台进行存储后的字符串，均为Sir，登录时是单纯是静态的匹配，无法增加新的账号。

### 4.2.2修改功能

我们将学生信息存储在Debug文件夹下的Info文件中。我们可以通过直接修改Info文件来修改诸多的学生信息，也可以通过调用该模块进行修改。该功能是教师模块下的一个子模块，并且可以修改学生的任意一个信息。信息包括年龄，学号，姓名，性别，生日，电话，邮箱，成绩，成绩修改将会包括三个课程的成绩一起进行修改。为了优化用户体验，我们还设置了全部修改，从而减少工作量。除此之外，增加学生人数，减少学生人数也同样可以进行操作。这样就减少了用户的工作量，增加了其使用效率。

### 4.2.3搜索功能

该功能是教师模块下的一个子模块，并且分为查找和排序两个子模块。在这个功能的两个子模块下，均包含“按姓名”、“按学号”两种方式进行。排序完成后可以进行查找。由于链表的相关特性，进行排序完成后，无法提高查找的效率，我们考虑通过跳表的方式进行相关的查找，跳表是应用在链表下的数据结构，利用二分的方式进行。

在排序的过程中，我们按照字符串顺序进行相关的比较，采用冒泡排序进行。

在查找的过程中，我们针对输入的关键字进行查找，若存在多个满足条件的结果，则依次输出相关的结果。

### 4.2.3校对功能

由于学号是唯一的，因此不论在新增学生信息，还是修改学生学号的过程中，都会进行相关的检查。若该学号已经存在，则提示修改失败，需要重新输入。

### 4.2.4浏览功能

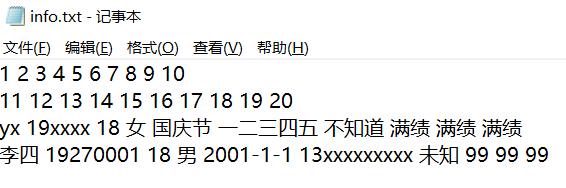
该功能同样是一个子模块。简单的说，浏览功能就是单纯浏览、观看系统中的相关信息。在此的情况下仅能够进行浏览，无法删除、增加、修改等对信息的操作。其浏览的顺序按照系统排序的顺序进行，也就是说，若进行排序的操作，则浏览的相关顺序将会发生对应的改变。

## 4.3导航模块

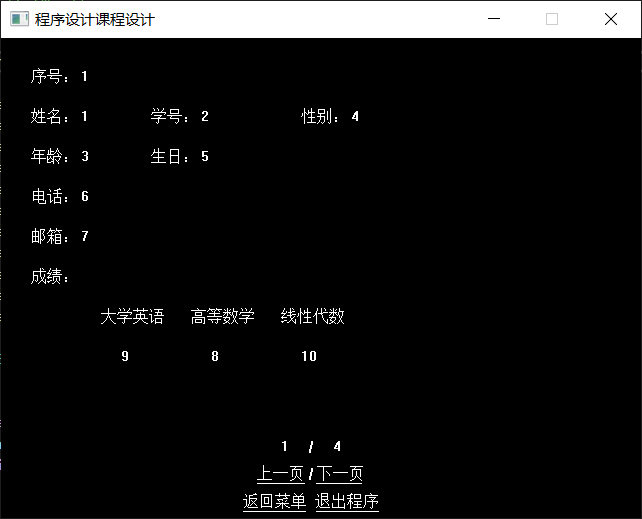
该模块是所有图形化界面的总称。其内部包括了欢迎功能、菜单显示功能、界面跳转功能、信息浏览按钮等均在此模块下。简单地说，该模块包括了全部的图形化界面。我们通过easyx库中的相关函数进行。通过检测鼠标点击的信号，判断此时鼠标的位置，执行相关函数，从而实现我们的功能。

# 5. 系统测试分析

我们将所有学生信息存储在了Info.txt文件中，每行信息分别为姓名、学号、年龄、性别、生日、手机号、邮箱、高等数学成绩、大学英语成绩、线性代数成绩在最开始的文件中初始化信息，如下图所示。



我们可以在学生界面中对相关数据进行浏览。



之后是添加的数据，我们添加 YX 19xxxxxx 18 女 不知道 199xxxxxx 未知 满绩 满绩 满绩 这一条消息记录，通过教师界面进行。



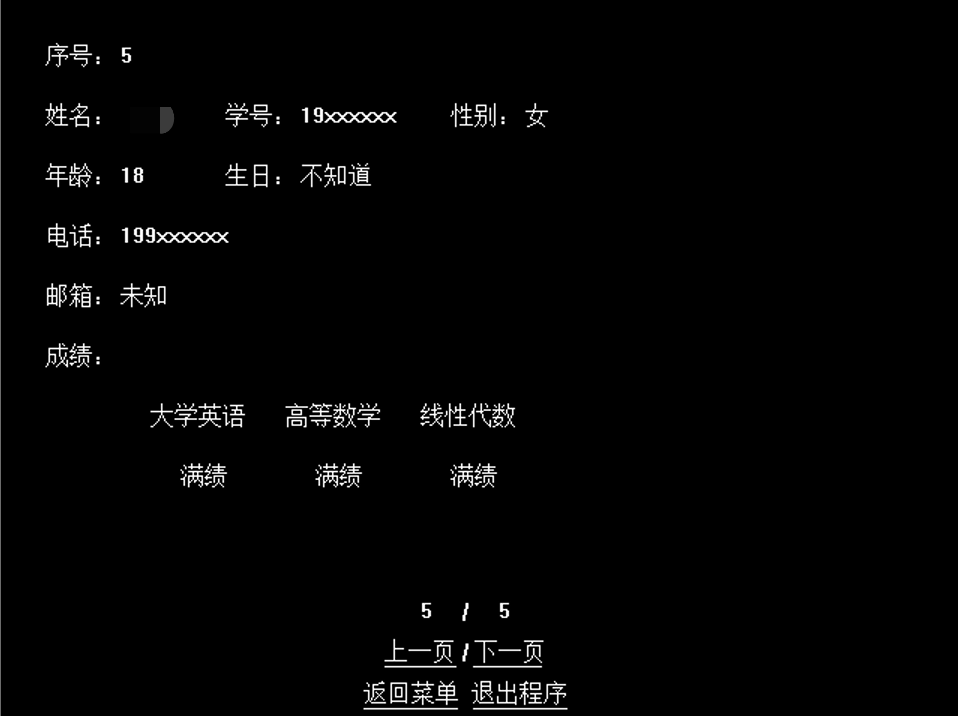
此时我们需要输入账户及密码。记录中目前仅有账号密码均为Sir的一个账户。若输入不存在的账户，就会返回如下信息。



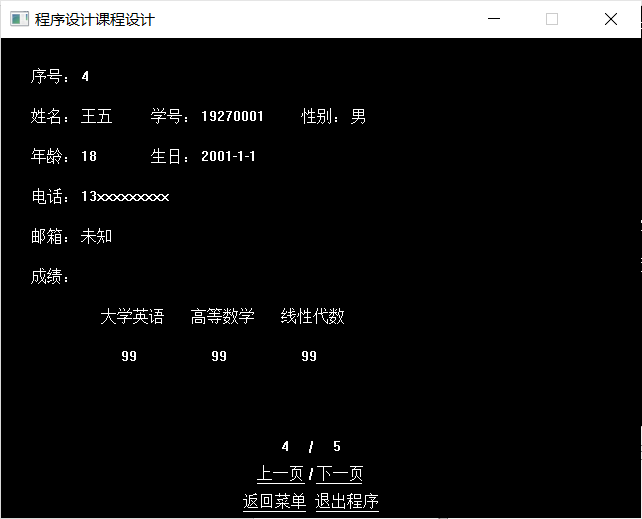
若多次输入均不正确，则会强制退出会主菜单。

正确输入账号和密码之后即可获得教师权限，且不用多次进行登陆。

进入教师界面之后，即可进行相关的信息修改以及删除操作。刚刚加入的杨菥即可顺利加入，信息如下：



例如将李四的姓名改为王五，相关如下：



以上为我们全部的相关测试结果。

# 6. 结论与心得

程序基本完成了目标的基本功能。并且利用easyx库实现了比dos界面的优化，提升了用户体验。实现了基本要求的增删查改的功能。在创新点上，我们完成了窗口框体设计等多种图形界面上的设计及优化，教师系统和学生系统的设置也初步完成。我们认为该系统的难点在于使用链表进行相关的排序，相对耗费时长更久。囿于技术原因，尚未完全掌握跳表及其使用方式，因此查找仅通过线性的O(n)时间复杂度进行了相关查找，若完成了跳表的操作，时间复杂度将会优化为O(logn)，但相应的空间开销将会增加。我们相信此处可以在未来进行相应的优化。

Se4ky：在开发的过程中作为第一个书写的人员，意识到了项目开始阶段有一个明确的目标及相关规划的重要性。很大程度上规划的好坏决定了后续开发的难度。链表的排序相对于原有的排序有许多的不同，存在一定的难度，花费时间相对久。但是其余部分的开发难度不打，但重复工作量的情况会使人感到疲惫。

GCY：在开发的过程中负责了UI界面的设计及其相关优化，感受到尽管算法是程序的核心，但是UI设计的好坏决定了用户的体验。UI的学问相当大，如何照顾好相关的用户体验是一个不可忽视的重要因素。在开发的过程中，发现了相关函数往往多次并且重复出现，为了简化编码过程，我选择了宏定义的形式简化代码结构，增强可读性及维护性。

F001：在开发过程中负责了Root账户的相关设置。由于是最后加入的Root，需要对之前的代码进行比较大的修改，也需要对全局有一个更加详细的了解。在修改的过程中，新增的一个子系统选择界面导致之后返回菜单的按钮将会重新回到子系统选择中。为了提升用户体验，防止多次输入密码的尴尬情况，我们设计了输入一次账号密码之后即可重复进入的方式，更加简单也更加合理。除此之外，更多的层数意味着更多的操作方式，因此在之前的界面中也加入了许多的返回键按钮，用于返回上一级菜单。

总的来说，我们实现了相关创新点，但是在算法层面还有可以提高的部分。还学习到了easyx库中的相关函数及其使用方法，也可以将easyx运用在之后别的系统中进行开发。由于是信息管理系统，我们没有考虑加入图片及声音等更多的元素，也是可以提升的点。此外，囿于时间原因，学生的密码信息以及老师的密码信息及其相关修改，是更多需要考虑的点。