**20210805 day03项目测试优化与总结**

## day03目标

### 1、项目名称

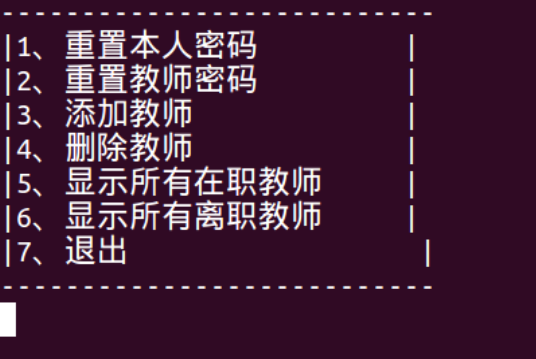
score\_manage/学生成绩管理系统

### 2、day03目标

* 学生、教师、校长首先进行单独调试
* 针对出现的bug进行讨论并提出解决方案
* 改进代码，增添必要注释
* 进行联调debug

## 主要业务的调试 v3.0

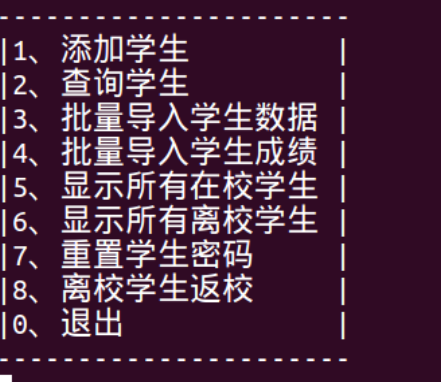
* principal校长部分：



第一轮debug及优化:

1. 重置本人密码没有二次确认（y\Y）
2. 重置密码，开始为直接打开校长密码文件，以文本文档形式，安全性不够
3. 重置教师密码函数，与队友写的重置函数有重复
4. 添加教师，学号有重复问题，没有考虑周全
5. 删除教师的方法太复杂，现在方法为网上顶
6. 文件打开关闭有问题，数据导入读取不出来

* teacher部分



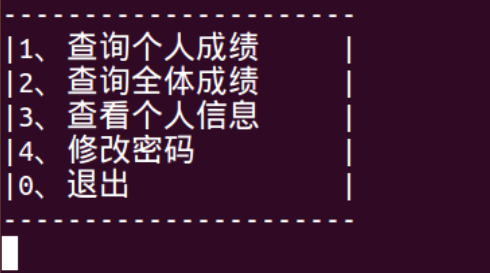
此为查询学生的二级菜单：



第一轮debug及优化：

1. 修改了初始化冻结名单的文件读取方式："w+"->"r"；
2. 删除了TCH\_MAX的重复定义；
3. 校长端补充了教师文件读写函数；
4. 在teacher.c添加了激活学生函数；
5. 在principal.c添加了激活教师函数；
6. 批量导入学生数据功能更改，从原先的全部数据导入缩减到"姓名 性别"格式
7. 学号自动生成，密码和成绩全部置零；
8. 添加教师功能加入了检索离职教师工号的代码；
9. 成绩批量导入，有脏数据现象；

* student学生部分



第一轮debug及优化：

1. 全体成绩显示没有对齐；
2. 修改密码是学生独有的功能，从工具函数分离重写比较合适
3. 查询完全体成绩，进行了排名，文件里面也将学生排名再存入，是个bug

## 项目的优化及不足v4.0

第二轮debug及联调后，我们进行了项目展示，在导师及同学的指正下，我们依旧出现了一些测试时没有考虑周全的问题。

* 项目没有考虑到的优化v5.0

1. 各种展示界面的整齐度及美观度进行了优化；
2. 登录时输入账号工号与密码如果越界会产生段错误
3. 关于教师批量导入成绩，没有明确结束的非法标识符
4. 重置或者修改密码加上判断，不能和初始密码和原密码一样
5. 工具函数有过多的冗余，需要进行必要的增删
6. 内存优化不到位，比如结构体内存对齐问题
7. 缺少密码的\*字符保护输入
8. 密码如果以文本形式储存，安全性太低

## 项目总结

* 个人总结：

张丁月：

我这次主要负责工具函数的编写和部分的登录界面编写，主要完成的功能有：强制修改密码，修改密码，确认密码，重置密码，校长登陆，学生登录，老师登录等。  
  有哪些错误：  
  结构体使用错误，混淆了结构体指针和结构体变量。文件操作不熟练，比如打开有没有及时关闭，指针没有及时置空等，导致文件读写会出现问题  
  走过的弯路：  
  首先，函数功能不够合理，错误的把一个要求当做一个功能写成了一个函数，实际上，一个要求里面可能包含多项功能，要求之间也可能会有功能重复的部分。我在编写的时候没有认识到这一点，导致写出来的函数会有好几个功能，代码阅读起来比较复杂不够好理解，若是要修改起来，也是牵一发而动全身，很难修改。  
  比如：修改密码的要求，里面应该有身份确认（即输入原密码），修改密码，密码确认（输入两遍新密码）等多个功能，应该分为三个函数来写。同时，重置密码部分也会用到密码确认部分，若是分为三个函数，就可以减少代码量，看起来更加简洁。  
  启示：  
  在下手编写之前，应该首先罗列好功能大纲，争取做到一个功能对应一个函数，完成某个函数的时候可以尝试思考一下这个函数能不能拆分，优化，而不是等到全部完成才开始考虑优化，这样会导致不好改动

程龙新：

编写teacher.h的头文件的过程中，依旧是由于构想不够明确，没有写明确的计划书的原因，导致了教师的函数较少，耦合性较高，这在之后的完善中也逐渐改正，但确实也是走了不少弯路。

在teacher结构体的编写过程中，虽然注意到了内存对齐的问题，但是对这方便的知识不够熟练，自己算出的字节数与调试中反馈的结构体的字节数不匹配，为了保持团队进度，放弃了回头重新学习和结构体合理排序。

在结构体的编写过程中，发现了之前学习中没有注意到的点，就是关于结构体的全局变量的定义，我一开始是定义在源文件中，在头文件中声明了结构体指针，但在make的过程中发现，这种定义方式并不能被接受，经过网上搜索资料和自己尝试，才知道结构体需要直接定义在头文件中，补充了之前有关这方面的知识的缺漏。

在最后补充注释的过程中，意识到了自己前面在编写函数的时候很少写注释，写代码的时候的想法也没有作注释写下，这说明自己在写函数的时候没有能够有意识的以注释的方式进行规划，这也是在之后的项目中需要注意的地方。

陶开泽：

在本次项目中，我丰富了建立大型工程相关的知识，在初次的生成文件过程中，意识到了结构的不合理，对文件结构进行了改进，由基础的main,student,teacher,tools,principalg这五个主要内容，增加了login文件，由main单独调用以保证main.c的干净整洁，使项目整体更加结构化，降低了耦合性。

Makefile过程由于自己之前进行单独的练习过，已经不再那么生疏，这一次的项目较为轻松的就能够搭建出框架，但是Makefile文件的编写还是不够熟练，出现了可执行文件串项目命名错误导致找不到的情况，还有一些文件的依赖建立的也不够完善，不过这是由于最初的构想和思路不够详细，不足以满足本项目落实之后的需求，这在之后写代码的过程中可以着重考虑。

接着就是编写程序的入口，设计了以携带参数的方式运行可执行文件就可以跳转到校长的登录界面，刚好用上了最近最新的知识，这也是自己比较满意的一点。

由于之前Makefile的编写，就可以再程序入口完善的情况下进行整个项目的运行并对单个功能进行调试，而不是单独外部写一个文件调用后另外调试，这样可以再后续的过程中减少很多错误。

最初的构想中，打算整个项目都面向文件，只对文件进行读写，减少堆内存的使用，后来发现这种思路太过复杂，由于我们的项目规模不大，所以能将文件直接读入堆内存中，统一操作，统一保存，这样子就不需要使用文件指针的相关函数，也降低了代码编写的难度。

在编写的过程中，也发现了自己对字符串相关的函数有些薄弱，比如getchar的运用，并不能很好的分清楚什么时候能够正确地吞下'\n'，只是凭感觉在使用。

由于初步构想的原因，导致整合的时候发现有功能过于复杂的函数存在，吸取了这次的经验，在之后若有团队项目，分工会更加明确，对各个文件的功能落实会更加清晰。

在本次项目中，我认为自己增加的经验主要在于制作大型工程的规划方面，明白了自己主要的薄弱点大概就是对与项目的构想和规划不够完善，在自己做的同时应该关注队友的进度和想法，适当地帮助队友，和队友沟通也会让自己的更轻松，项目进度更快。

* 团队总结（王怡宁）

关于这次团队项目合作，其实总体来说还是比较成功的。

首先感谢陶开泽同学，作为首席开发师，提供了很关键的思路以及思路指导，能够主动承担其中关键业务代码的编写，总的来说，他是一位很靠谱思维敏捷的开发工程师；也很感谢张丁月同学的刻苦努力，编写了其中耦合度最高的工具函数，有问题沟通的很及时，帮忙debug也很认真；同时也很感谢程龙新同学，虽然在项目进行中我们交流不够多，但是在最后的总结环节，他能够完成自己的任务不出错，非常感谢他靠谱的责任心！

感谢老师的批评指正，毕竟第一次完整的团队项目，我也总结一些团队未来的改进点。

* 1. 最初的构想和思路不够全面，不足以满足本项目落实之后的需求，注释也不够详细。
  2. 和每位成员项目进度跟进的不及时，功能有一些重复性
  3. 知识点应用不熟练，写的效率有点低
  4. 代码的通用性和耦合度没有考虑全面，往后应当善用使用宏函数
  5. 思维不够系统，有些边界问题没有考虑清楚
* 团队分工

**队长：王怡宁**

1. **思维框架的搭建**
2. **项目报告书的撰写**
3. **校长部分代码的实现**

**首席开发：陶开泽**

1. **编译脚本的撰写**
2. **主函数登录函数的撰写**
3. **教师部分代码的实现**

**工程师：张丁月**

**工具函数的实现**

**工程师：程龙新**

**学生部分代码的实现**