# 程序架构简单说明：

**各个class的功能介绍（此处只说一些看代码打死也想不出来的，一眼就能明白意思的就不在这里说了）：**

**类radix62 是为了存储密码，方便进行加密解密运算而建立起的数据结构，程序中所有密码均用radix62存储。radix62为62进制，0～9代表0～9，a～z代表10~25，A～Z代表26~61，故用户所能设置的密码中只能包含数字字符或英文字符。**

**目前radix62所支持的运算： 加法，减法及乘法（乘法有点BUG，当两乘数位数过大时，结果中会出现一些未规定的字符，BUG原因暂时不知。由于只有加减法，故加密操作只能使用移位码，安全性不高，写完系统之后有时间再改进）。**

**类radix62每一位上的成员由类radix62\_element存储，若不考虑算法的话，radix62可以近似看做存储radix62\_element的可动态变化的数组（使用了vector数据结构）。radix62\_element的数据成员中包括了一个叫做carry的int型变量，用来暂时存储加法减法以及乘法过程中产生的进位值，在用过之后会及时清零以避免下一次运算时产生污染。**

**类Info用来存储书籍信息**

**类record用来存储借书记录，其中bool型变量if\_return用来存储是否归还信息。记录在初始化之初默认该变量为false，只能通过成员函数return\_the\_book()来将改变量改为1，没有其他的更改该变量的接口。即，一条记录只能由未还更改为已还，不能由已还更改为未还，以此来避免续借一本书后可能出现原借书记录消失的现象的出现。**

**类user存储各用户的用户名，账号，密码以及权限，权限对应的操作如何展现暂时有点说不清，但等我写完主函数的部分就清晰了。**

**权限。该类中需要注意的是：**

**成员函数set\_keyword(const radix62 key);并没有提供对于keyword的再次核查（即再次输入密码与当前密码进行比对），所以密码检查部分需要在主函数中调用该函数时另外编写。**

**账号一经注册便不能更改，所以没有提供修改user\_number的任何接口。**

**为了防止破坏该类的封装性，没有提供对于操作符>>的重载。（重载后可能出现user u;cin>>u;然后账号密码等被更改的情况，不利于封装性的维护）所以在输入时代码要麻烦一些。**

**数据成员keyword中存储的是密码进行加密过的密文(为确保程序退出后在进入信息能够保存，每次程序退出前都将关键信息存入文本文件中，再次进入时再从该文件读取，所以不进行加密会有密码泄露的风险。)，加密的密钥为main.cpp中的secondary\_key（随便设，设啥都行，就是要长一点）。加密方法为最简单的移位法。**

**以上就是本程序用到的所有的类了，也就是数据结构了。本程序会根据以上的类建立三个表，分别用来存储用户信息，书籍信息，借书记录信息，暂且叫这三个表为User， Book和Record.目前只写完了所有的数据结构，程序的查询存储等等的逻辑还没有写，但是已经差不多了，没有用到SQL查询语句也不知道行不行，但是反正功能应该都是可以实现的。再写一下三个表的详细信息（没有遵照语法，毕竟我都忘了。。。）**

**User**

**账号 //为该表的键**

**姓名（昵称）**

**密码**

**权限**

**Book**

**书号（ISBN） //为该表的键**

**书名**

**作者名**

**Record**

**书号（ISBN） //外码**

**账号 //外码**

**借阅时间**

**是否归还**

**至于第几范式，ER图什么的自己分析一下吧，我已经忘了。。2017/8/8**