这个学习档案记录了我对题目的部分思路，更详细的内容请参考代码

**Level 0**

第一题

只需要写一些if条件句，判断输入的字符串是不是指定的关键词，再根据关键词输出对应的内容即可

第二题

我用了stl里的映射（map<string,string> m）来存取key和value。

第一步，读取文件里的数据，用ifstream函数打开文件，getline读取整行字符串并在遇到空行时直接跳过。随后用for循环遍历读取到的字符串，找到冒号的位置并记录下来，没有冒号则判定为错行并跳过。最后根据找到的冒号位置索引，使用substr函数将最开始的字符串拆分成两个字串，分别对应key和value，并存到m里

第二步，通过输入的key查询value，用map的find函数和end函数来判断对应的value是否存在，若存在就直接输出m[key]，不存在则输出error

**Level 1**

这题的图书馆预约系统涉及了多个参数，包括楼层、星期、座位的行列、使用者名字和座位预约状态。对此，我构造了一个结构体类型的四维数组a，四个维度分别代表楼层、星期和座位的行列，结构体edge包含使用者名字和座位预约状态两个成员。

数据存储方面，除了要使数据不会在第二次打开程序后被清理，还要让数据按照其在内存中的存储形式原样输出到磁盘上存放。因此，这里用到了二进制文件，同时满足了上述两种要求。

这题我还构建了四个函数。get\_date函数可以判断输入的字符串是星期几，并返回对应的数字，不存在则返回-1；read函数使用ifstream函数从二进制文件中读取数据，并存放到数组a中；save函数使用ofstream将数组a中的数据保存至二进制文件中；clear函数通过四个for循环重置数组a中的所有数据，并在最后调用save函数进行保存。

主程序包含两个while循环。第一个用于登录界面，输入命令为Quit时退出；第二个用于登录后的预约系统，输入命令为Exit时退出。

主程序通过输入的第一个单词以识别命令，判断关键词如Reserve、Reservation等，以及调用get\_date函数检测单词是否属于星期。识别完成后，即执行对应的操作。

**Level 2**

Level 2是基于Level 1的改版，主要进行了优化和拓展了功能。

优化方面，涵盖了命令集和用户提示。新的命令集的每条命令都有对应的关键词，输入方式也改为了逐层输入，新增了更详细的用户提示。例如输入Query后，程序会输出“请输入日期”，输入日期后，程序会输出“请输入楼层”。每条命令都有对应的用户提示，加上新增的能够显示所有可执行命令的help命令，让用户无需记住命令关键词或格式也可正常使用。

功能方面，Level 2在管理员命令方面进行了大量拓展。新增了拓展图书馆座位的Extend、减少图书馆座位的Reduce、重置指定楼层或日期的用户预约信息的Modify。除此之外，原有的Reserve和Query也加上了管理员独有的功能。在管理员模式下，Reserve命令可用于修改任意预约信息，Query命令则可以查询预约者的名字。

Level 2定义了新的结构体Sysdata，包含fx和fy两个成员，用于存储图书馆座位的行列边界；同时在原有的结构体edge里新增了布尔类型的成员ava，用于记录座位是否可用。两者结合，实现了Extend和Reduce的功能。