

课 程 设 计 报 告

课程名称 计算机程序设计基础（2）

班 级： 无 47 班
学 号： 2014011216
姓 名： 刘前

2015 年 7 月 26 日

课程设计任务书

1. 课程设计目的

面向对象程序设计课程设计是集中实践性环节之一，是学习完《计算机程序设计基础 2》课程后进行的一次全面的综合练习。要求学生达到熟练掌握 C++ 语言的基本知识和技能；基本掌握面向对象程序设计的思想和方法；能够利用所学的基本知识和技能，解决简单的面向对象程序设计问题，从而提高动手编程解决实际问题的能力。

2. 课题题目

- 1) 学生成绩管理系统
- 2) 图书管理系统

3. 设计要求

1) 设计课题题目：

每个同学都完成 2 道课题。 后面有范题，仅供同学们参考，不列入本次课程设计的课题。

2) 根据课题完成以下主要工作：

①完成系统需求分析，包括：

系统设计目的与意义；

系统功能需求（**系统流程图**）；

输入输出的要求。

②完成系统总体设计，包括：

系统功能分析；

系统功能模块划分与设计（**系统功能模块图**）。

③完成系统详细设计：

数据文件；

类层次图；

界面设计与各功能模块实现。

④系统调试：

调试出现的主要问题，编译语法错误及修改，重点是运行逻辑问题修

改和调整。

⑤使用说明书及编程体会：说明如何使用你编写的程序，详细列出每一步的操作步骤。

⑥关键源程序（带注释）。

3) 按规定格式完成课程设计报告，并[在网络学堂上按时提交](#)。

4) 不得抄袭他人程序、课程设计报告，每个人应体现自己的个性设计。

4. 程序设计的基本要求

(1) 要求利用面向对象的方法以及 C++ 的编程思想来完成系统的设计；

(2) 要求在设计的过程中，建立清晰的类层次；

(3) 每套系统中设计的类的数目不少于 4 个，每个类中要有各自的属性（多于 3 个）和方法（函数多于 3 个）；

(4) 需要定义一个抽象类，采用继承方式派生这些类。并设计一个多重继承的派生类。

(5) 在程序设计中，引入多函数的多态性、运算符重载等机制。

(6) 自己设计测试数据，将测试数据存在文件中，通过文件来进行数据读写来测试。

5. 创新要求：

在基本要求达到后，可进行创新设计，如根据查找结果进行修改的功能。

6. 进度安排：

小学期 第 1 周	7 月 20 日	学生成绩管理系统构思、设计 School_Member 类和 Student 类、编写 Student 和 Teacher 类
	7 月 21 日	编写 Manage 类和 Director 类，编写 学生成绩管理系统.cpp
	7 月 22 日	课程设计报告-学生成绩管理系统部分，图书 管理系统构思、Book 类 Manage_Book 类
	7 月 23 日	编写 Manage_Book 和 Reader 类，编写 Manage_Reader 类和 Manage 类
	7 月 24 日	编写图书管理系统.cpp，课程设计报告-图书 管理系统部分
小学期 第 2 周	7 月 27 日	
	7 月 28 日	
	7 月 29 日	
	7 月 30 日	
	7 月 31 日	

【提示】评分表在本课程设计报告的最后一页

一、学生成绩管理系统

目 录

1. 系统需求分析	5
2. 总体设计	8
3. 详细设计	10
4. 系统调试	15
5. 使用说明书	16
6. 总结	27
附录：源程序清单	28
测试数据文件	54

1. 系统需求分析

目前，学生的学习成绩在很大程度上体现了该生的在校表现，因而，对学生成绩的管理及进行的相关数据处理具有重要的意义。然而，现今学校的学生规模很大，课程也很丰富，学生成绩的数据也十分庞杂，导致学生成绩管理是一个非常繁琐与复杂的工作。为了减轻人的工作负担，运用计算机程序、设计出一个学生成绩管理系统应运而生。

（1）系统设计目的与意义

学生成绩管理系统的目的就在于方便学校对学生的成绩进行处理和应用，使教务人员从复杂的成绩管理中解脱出来。计算机具有运算速度快、处理能力强、可靠性高、存储量大等特点，这是传统的人工管理所不能企及的。

通过学生成绩管理系统，可以实现对学生成绩的各项处理，并且提高成绩管理的效率和水平；同时，对学生成绩的深入分析可以判断学生的学习方式，甚至能据此针对性的提出一些学习的建议，这对学校教务教学工作的发展具有重要的推动作用。

（2）系统功能需求：

功能需求

对不同的使用人员有不同的功能限制。

① 对于学生而言：

通过登录获得权限之后，即可进行成绩查询，即系统的输出功能。分为两个功能板块。一个是查询个人成绩，另一个是成绩排名。

学生输入个人密码，根据选项系统会输出学生的个人信息（姓名、学号、各科成绩），各课程名称、成绩及成绩排名，GPA 及 GPA 排名。

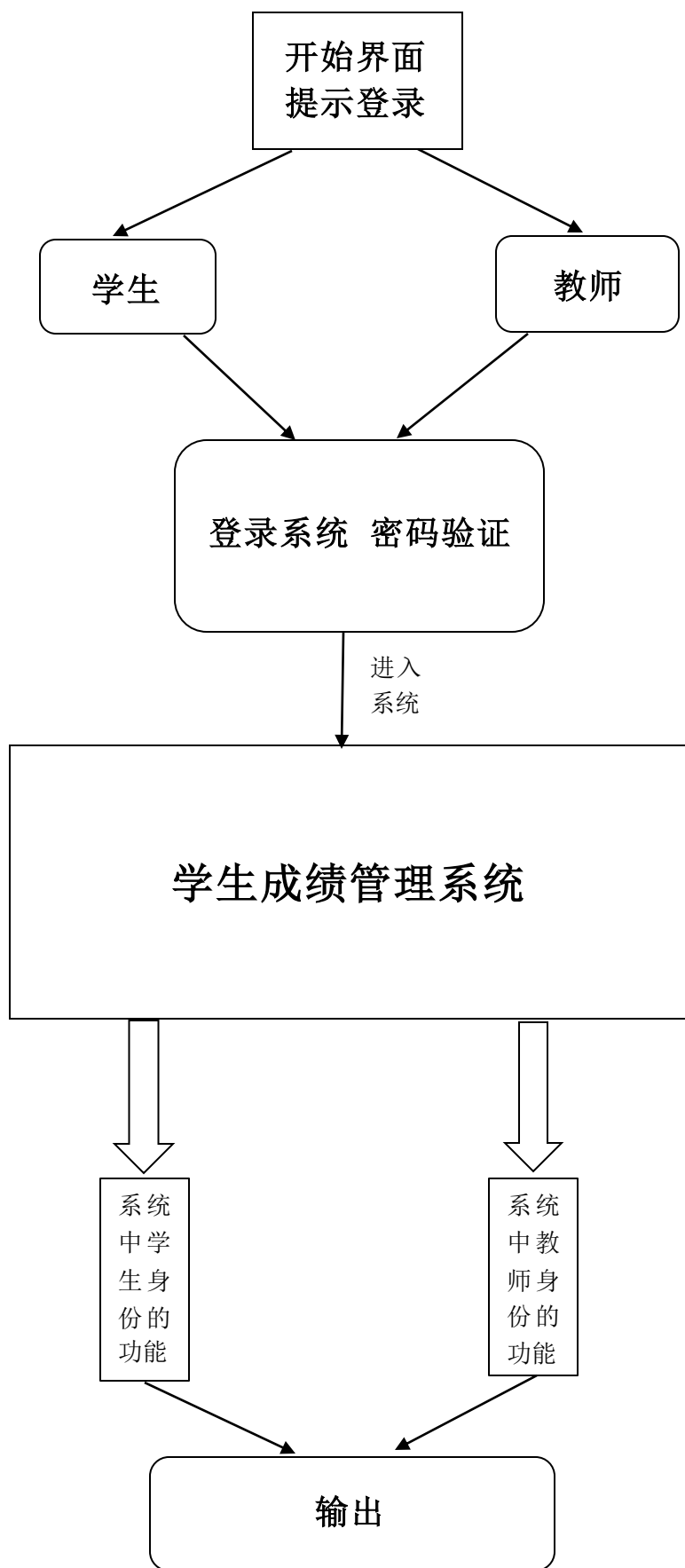
② 对于系统的管理员（老师），通过登录获得权限之后，可以进行学生个人信息和成绩的添加、录入、修改、查询和删除等功能。

根据不同的需求，系统需要实现以下功能：

- 1) 考虑到学生的流动，学生信息及成绩必须能够被添加或删除。添加学生包括学生个人信息以及各学科成绩；删除学生即意味着将学生从该系统中彻底删除；
- 2) 考虑到评分时的失误以及初次录入时的错误，学生成绩必须能够修改，而且只能由老师修改，以保证成绩的准确性；
- 3) 为方便老师了解全部学生的学习情况，系统需要提供查询功能及学生排名功能。教师输入学号即可查询学生成绩（单科或 GPA）
[注意]：教师查询成绩是可以显示学生的姓名，但在学生查询板块中不显示姓名（保护学生隐私）；
- 4) 系统能够从外部文件中导入学生的信息及成绩，极大地方便学生成绩录入系统；
- 5) 同时，针对不同教师不同学科的需要，成绩管理系统还应该能够实现分学科显示全部学生的成绩以及平均成绩；等等。

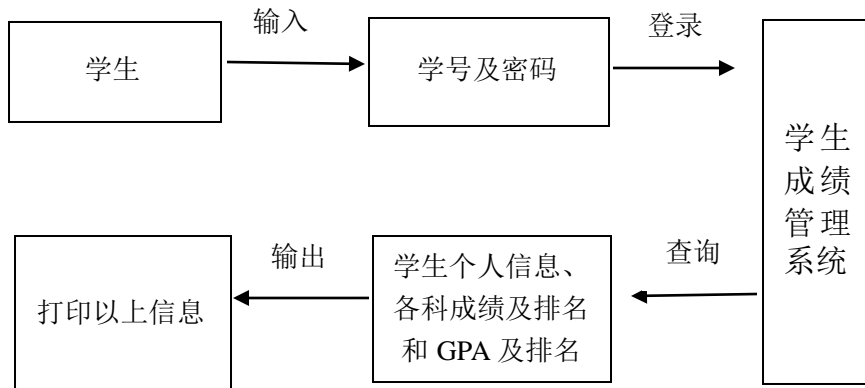
以上，均为学校成绩管理系统所应具备的功能。

系统流程图

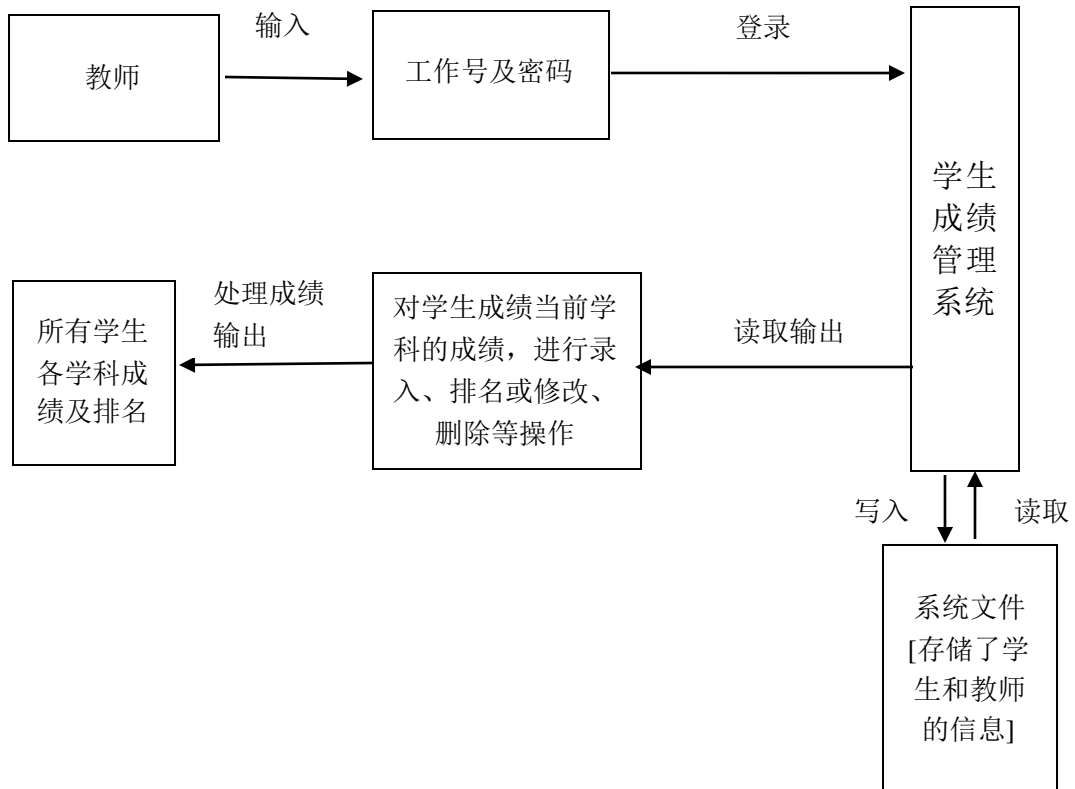


学生成绩管理系统根据不同使用者，实现的功能也不同，下面的流程图分别针对学生身份和教师身份。

① 学生（查询成绩）



② 教师



(3) 输入与输出要求:

进入系统，需要先输入学号（工作号）及密码，通过验证之后方可进入系统，以此保护系统的安全。教师和学生因身份不同，所以系统执行的操作及功能也不同，因而需要设计学生、教师两个不同的界面。

在两大不同的功能板块中，

①学生只能进行成绩以及排名的查询，即输出。输出内容包括学生的个人信息及成绩和排名的情况；（可以分为所有信息打印和个别信息进行打印）；

②输入则由教师进行，包括学生成绩的添加、修改和删除等，输入时注意是分学科进行输入和整体输入均可。主要是通过文件将学生的信息及成绩读入系统，以供查询；另外对学生成绩的处理也将存入文件中。

2. 系统总体设计

系统功能分析：

1) 系统进入界面：

人性化地提示用户输入个人的身份（学生 or 教师），然后进入不同功能板块的登录界面。

2) 登录界面：

学生输入学号和密码，系统在确认学号和密码匹配无误之后，显示“登录成功”，进入“学生”板块。教师输入工作号和密码，系统在确认工作号和密码匹配无误之后，显示“登录成功”，进入“教师”板块。

【注意】

系统针对学生和教师的功能不同，而且学生选用学号登录是因为：用户姓名可能相同，但学号是一一对应的。因此大规模情况下不宜用“姓名”登录。

3) 学生板块功能分析：

学生从系统中本身只需了解个人成绩及排名即可，因而功能为以上两个。

a) 个人成绩查询：

只能查询自己的各科成绩和 GPA，出于不同的需要，可以分别查询单科成绩，也可以一次性查询所有成绩（包括 GPA）；

b) 成绩排名查询：

成绩的排名不直接以数字的形式体现，而是将该科目下所有学生的成绩从高到低排出来（为了在一定程度上保护个人隐私，每个学生不显示姓名），同样能够直观的得到个人的成绩排名；

4) 教师板块功能分析：

a) 添加学生信息及成绩：

考虑到学生的流动性（迁入），根据系统界面的提示，按步骤录入学生的姓名、学号及各学科的成绩，即可添加学生到成绩管理系统，实现学生数目的增加。

b) 删除学生信息及成绩：

考虑到学生的流动性（迁出），根据系统界面的提示，输入所要删除的学生的学号，即可将学生从系统中删除。

c) 修改学生成绩：

考虑到教师批卷失误及向系统录入分数时的失误，因而需要实现对学生成绩的更改的功能。选择修改学生成绩后，根据提示完成对学生成绩的修改，并存入系统中。

- d) 查询学生成绩:
根据提示, 输入所要查询的学生的序号 (为了应对学生姓名相同的情况, 因此使用学号来查询), 即可分科目查询或者一次性显示该生的所有成绩;
- e) 成绩排名查询:
与学生的查询功能基本相同, 但可以显示出每位学生的名字, 从而帮助各学科老师更好地了解学生的学习状况。
- f) 导入外部文件:
大量的成绩数据时, 为了避免一一录入系统的繁杂过程, 特为系统设定了从外部文件导入数据的功能。可以从指定的文件内读入学生信息及对应成绩。
- g) 显示所有成绩:
一次性将所有学生的所有科目的成绩以及 GPA 输出, 更加直观、方便地进行比较和了解。
- h) 显示平均成绩:
显示各个学科的平均成绩, 一定程度上体现了学生的整体水平及考试难度, 对教师以后的教学及考试有一定的反馈效果。

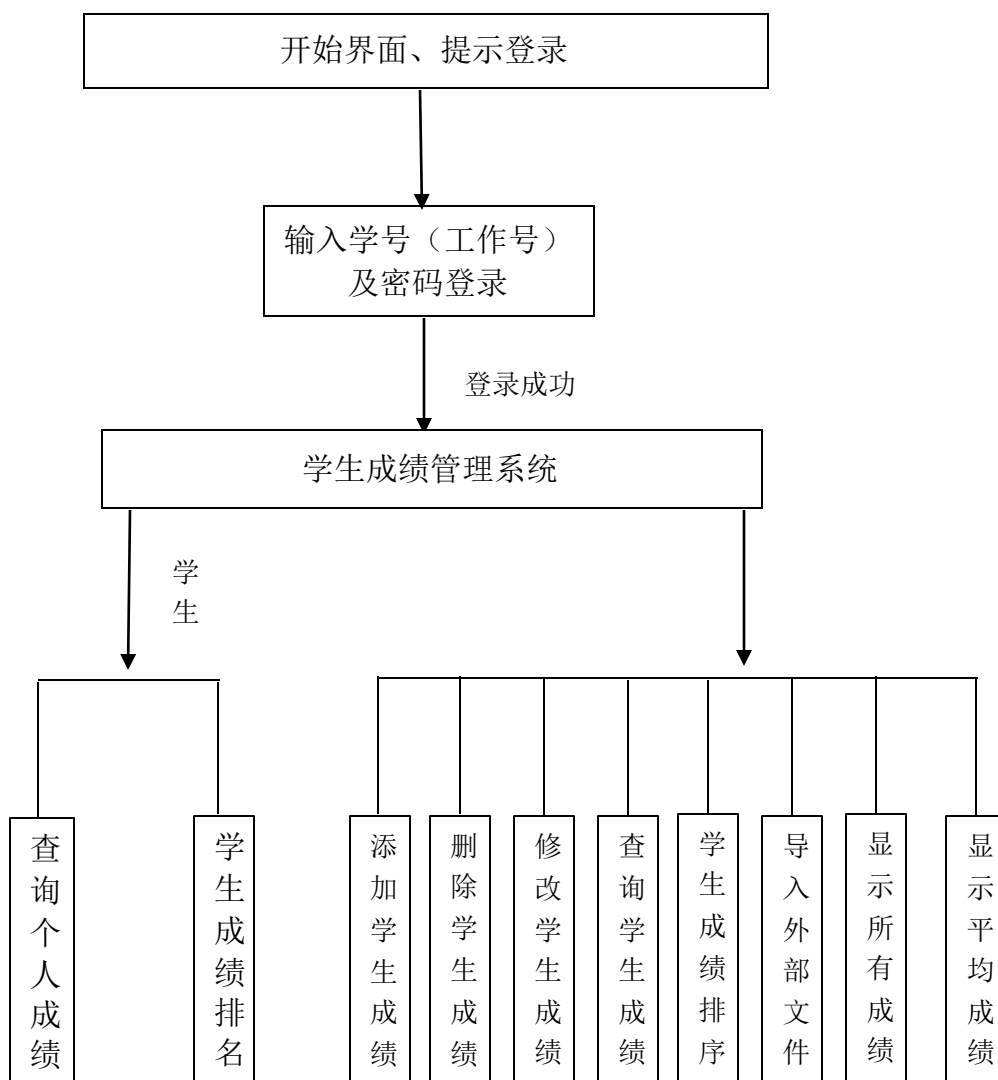
系统功能模块划分与设计:

首先, 点击“学生成绩管理系统”的应用程序, 将进入提示界面, 输入身份, 进入相应的登录模块。登录成功之后, 便进入了学生/教师不同的功能模块。

学生功能模块主要包括了“个人成绩查询”和“学生成绩排序”;

教师功能模块所包括的功能较多, 主要包括: 增加学生信息及成绩、删除学生信息及成绩、修改学生成绩、查询学生成绩、成绩排名查询、导入外部文件、显示所有成绩、显示平均成绩等等。因而, 本学生成绩管理系统功能模块划分合理, 学生与老师的模块区别明显, 并且能够实现较多的功能。

系统功能模块图：



3. 系统详细设计

① 数据文件：

本系统数据是计算机上存储的文件，但注意有以下要求：

- 1) 考虑到数据文件为二进制文件时，无法从外部修改文件，使得操作过于复杂，因此选用文本文件作为系统的数据文件；
- 2) 为了确保录入信息和成绩时能够正确录入，需保证存储的数据按照“姓名”“学号”“微积分成绩”“线性代数成绩”“离散数学成绩”“英语成绩”“密码”的顺序存入文本文件。
- 3) 为了方便自己测试数据，本系统有默认数据文件，学生的信息及成绩存储在默认文本文件“G:\student.txt”中，教师的信息存储在默认文件“G:\teacher.txt”中。
- 4) 为了方便导入其他文件，在教师功能模块内特意设置了“导入外部文件”

的功能，要求正确输入数据文件在计算机中存储的路径及文件名，即可将数据导入成绩管理系统。

② 类层次图：

学生成绩管理系统共使用了 6 个类，其中主要实现功能的是 Student 和 Teacher 类。

现对 6 个类进行分析：

1) Person 类：

系统中的抽象类，包含两个纯虚函数，作为整个系统的基础。不能具体化，但是派生出 School_Member 类（虚基类）。不能实现具体的功能。

2) School_Member 类：

顾名思义，表示学校的成员，主要包括学生、教师和管理人员。School_Member 作为虚基类，数据成员有姓名和学号（工作号），是学生和教师的基础。因此，School_Member 派生出 Student 和 Teacher 类，之后 Manage 多重继承了 Student 和 Teacher 类。School_Member 类为虚基类的原因是为了避免 Manage 在继承 Student 和 Teacher 过程中产生二义性。

3) Student 类：

学生类主要包含了一些小功能，为后面 Teacher 类功能的实现打基础。实现的功能包括：修改单科成绩；计算 GPA；获取单科成绩及 GPA；显示单科成绩及 GPA；等等。

4) Teacher 类：

Teacher 类作为学校的成员，和 Student 类同样公有继承了虚基类 School_Member 类。Teacher 类表示学校中的教师，实现了教师模块中的所有功能：对学生成绩的添加、修改和删除，以及查询学生成绩、学生成绩排序、导入外部文件、显示所有成绩、显示平均成绩等功能。

【注意】Student 类和 Teacher 类的联系：Teacher 类的保护成员是 Student 类的对象数组，体现了实际生活中“一名教师管理多名学生”的模式。

5) Manage 类：

Manage 类，主要功能是对学生的管理。Manage 类同时继承了 Student 类和 Teacher 类，School_Member 作为虚基类有效避免了 Manage 类成员函数的二义性。

Manage 类主要实现了学生的登录功能：将包含学生学号和密码的文件读入系统，学生进入系统前必须保证学号和密码完全匹配才能登入系统。

6) Director 类

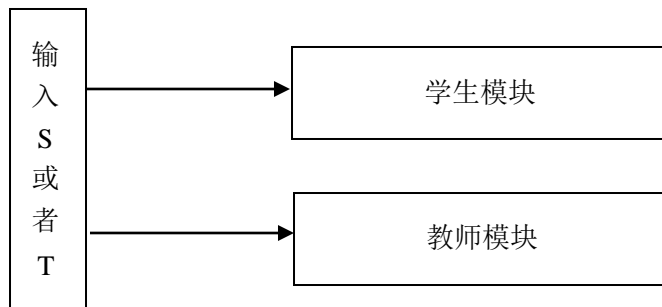
Director 类，主要体现在对教师的管理。Director 类的私有成员包含 Teacher 类的对象数组，体现了“一个领导管理多名教师的模式”。

Director 类实现的功能主要是将包含教师信息的文本文件导入系统中，并且确认教师在登入系统前所输入的工作号和密码完全匹配才能进入成绩管理系统。

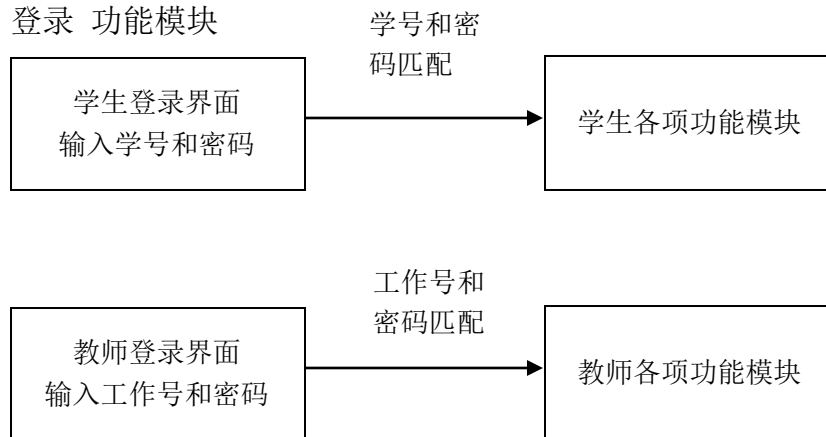
【注意】Manage 类和 Director 类共同实现了系统使用者（学生和教师）登录系统的功能，对保护学生隐私、以及保护学生成绩不被非法修改具有重要的意义。

③ 界面设计与各功能模块实现：

1) 开始界面、提示登录 功能模块

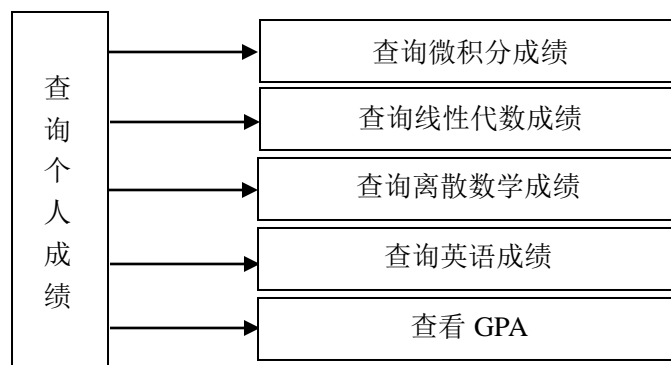


2) 登录 功能模块

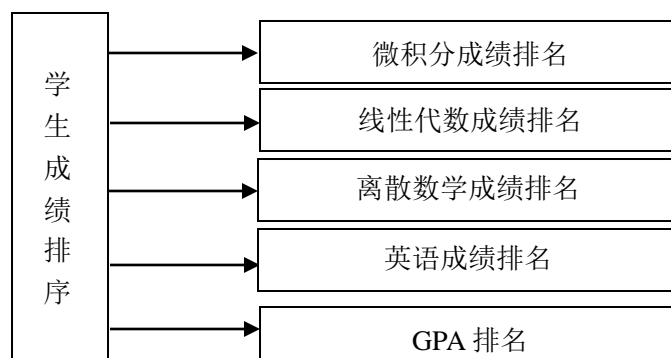


3) 学生各项功能模块

a) 查询个人成绩 功能模块

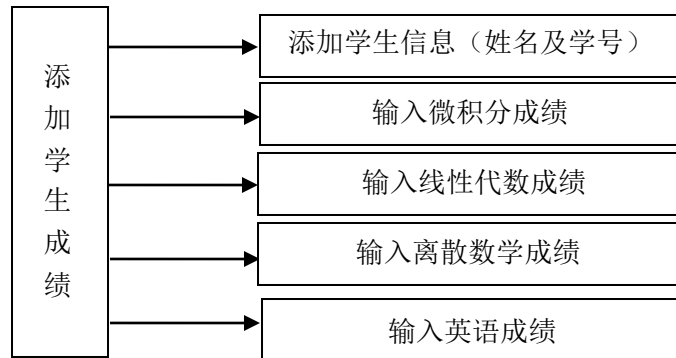


b) 学生成绩排序 功能模块

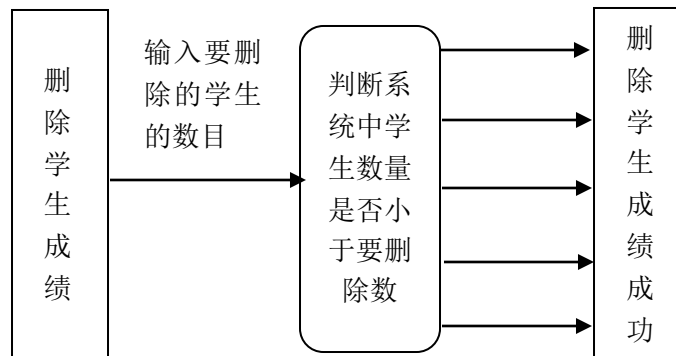


4) 教师各项功能模块

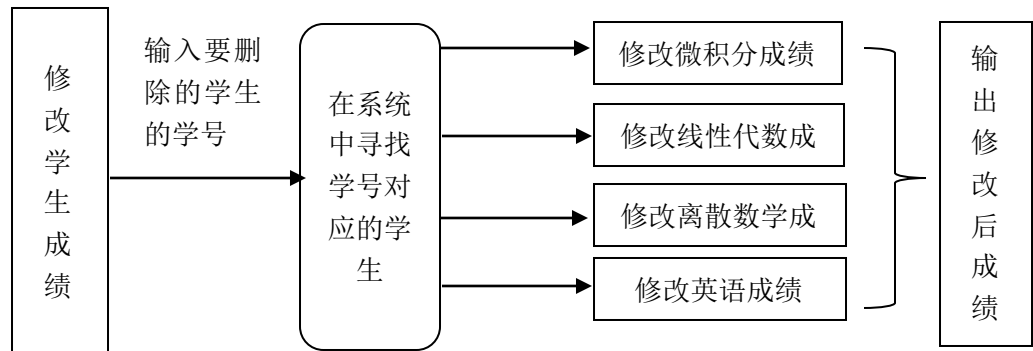
a) 添加学生成绩 功能模块



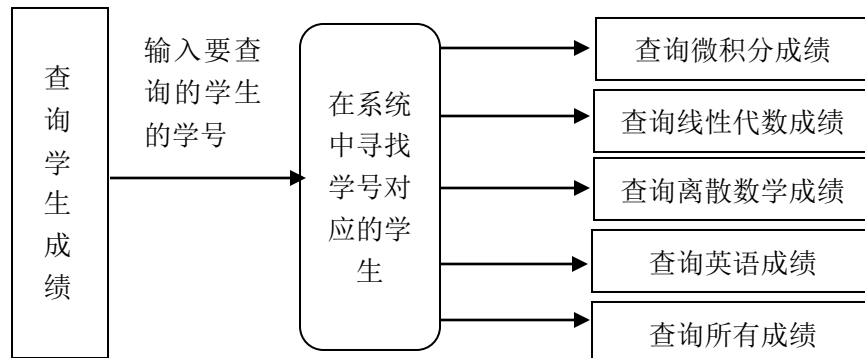
b) 删除学生成绩 功能模块



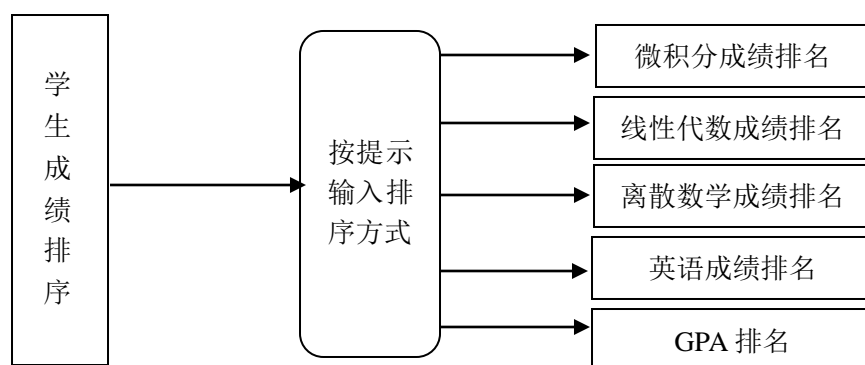
c) 修改学生成绩 功能模块



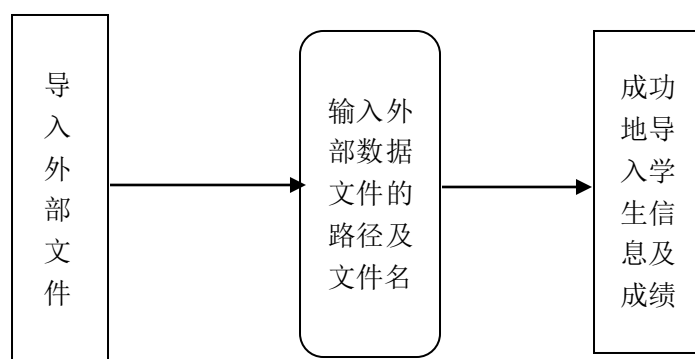
d) 查询学生成绩 功能模块



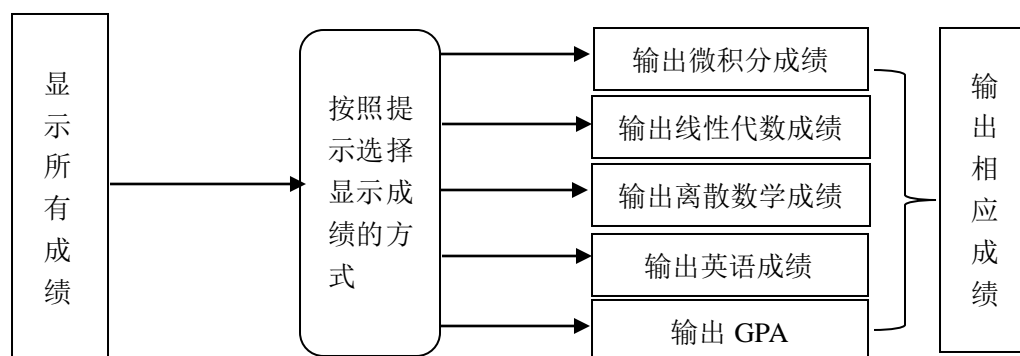
e) 学生成绩排序 功能模块



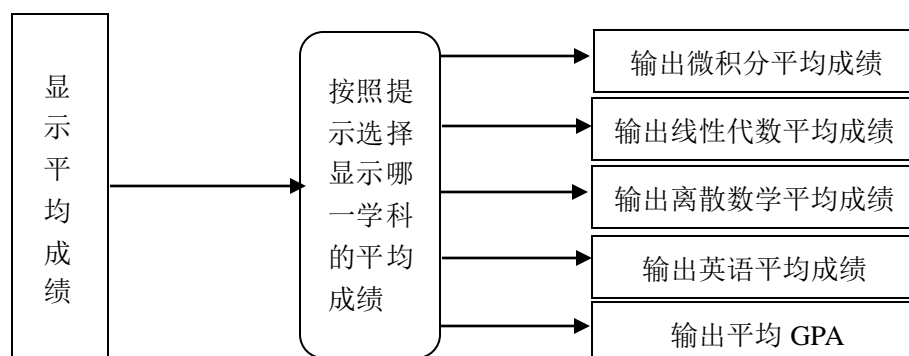
f) 导入外部文件 功能模块



g) 显示所有成绩 功能模块



h) 显示平均成绩 功能



4. 系统调试

编写完所有程序之后，开始第一次调试。Visual Studio 2008 显示共有 102 条错误，一一 Debug 之后，发现很大一部分 bug 是编程时输入的错误，并不是真正意义上的 bug。剩下的五六个错误是与系统的逻辑错误以及关于类的概念不太清晰。下面是在这次编程中出现的一些问题。

① 关于字符串。

一开始考虑学生和教师的名字用字符串指针，但发现自己对字符串指针的使用有些生疏，容易出错。第一次 Debug 之后发现许多错误与字符串有关，于是决定使用 C++特有的 string 类（字符串类），极大地简化了编程中关于字符串的操作。

②关于输入密码登录。

起初觉得密码登录可能太过繁琐，担心自己的能力不够。但后来在编程中思考，觉得学生成绩管理系统如果不设置密码完全不能保证数据的安全。于是增加了学生和教师的登录功能。但在调试的过程中，发现读入学生信息及成绩时，总是不能正确的读入。最终发现是“从文件中读取学生信息及成绩”函数的形参顺序与数据文件中的顺序不一致，导致姓名等均无法正确读入。纠正了这一点之后，登录功能模块就可以正常运行了。

③关于系统自动退出。

在调试过程中发现登入学生成绩管理系统之后，完成一项功能输出结果之后，界面上就显示“请按任意键继续...”，说明系统已经自动退出，导致进行下一步操作时不得不再次打开系统，很不方便。思前想后，决定给每个功能性的成员函数作简单的递归。也就是在每个成员函数的最后调用一次本成员函数，实现了无穷递归，除非选择“退出”，将会一直在系统内完成各项任务，这样可以保证完成一项功能后系统返回上一级或者返回最初的菜单选项，极大地提高了系统的效率。

④关于栈溢出。

另一个 bug 困扰了我一段时间，每次调试时，都会提示“...0xC00000FD: Stack Overflow...”不知为何栈溢出了。但是调试了很多地方，尤其是指针和数组容易出错之处，但仍然没有解决这个问题。之后，到网上一查，得到以下原因：可能是一次性申请了过大的内存，超出了默认的 1M。于是我才意识到，一开始为了实现系统的强大功能，我给每个 Teacher 的对象申请了 3000 个 Student 类的对象，又给 Director 类的对象申请了 100 个 Teacher 类的对象。确实一次性申请了很大的内存。联系生活实际之后，将 Student 类的对象数量减少至 500，将 Teacher 类的对象减少至 20。系统不再显示该处有 Bug。

⑤关于功能的完善

实现了基本功能之后，感觉作为学生成绩管理系统，功能性还是不够强，于是又增加了“导入外部文件”“显示平均成绩”等功能，对教师未来的教学以及评估学生的整体水准具有重要的意义。

以上是在系统调试过程中遇到的主要问题，总体来看，调试过程还是比较顺利的。并且遇见了之前从未见到的 bug，也使我 debug 的能力有一定的提高，总之获益匪浅。

5. 使用说明书

学生成绩管理系统 使用说明书

目录

第一部分 系统说明	
1. 背景	17
2. 系统概况	17
第二部分 系统使用	
1. 进入界面	17
2. 登录系统 ·	
①学生登录系统	17
②教师登录系统	18
3. 学生功能模块	
①查询个人成绩	19
②学生成绩排序	20
③退出系统	21
4. 教师功能模块	
①添加学生成绩	21
②删除学生成绩	22
③修改学生成绩	22
④查询学生成绩	23
⑤学生成绩排序	23
⑥导入外部文件	25
⑦显示所有成绩	25
⑧显示平均成绩	26
⑨退出系统	27
第三部分	
注意事项	27

2014011216 刘前

2015.7.21

第一部分 系统说明

1. 背景

目前，学生的学习成绩在很大程度上体现了该生的在校表现，因而，对学生成绩的管理及进行的相关数据处理具有重要的意义。然而，现今学校的学生规模很大，课程也很丰富，学生成绩的数据也十分庞杂，导致学生成绩管理是一个非常繁琐与复杂的工作。为了减轻人的工作负担，运用计算机程序、设计出一个学生成绩管理系统应运而生。

2. 系统概况

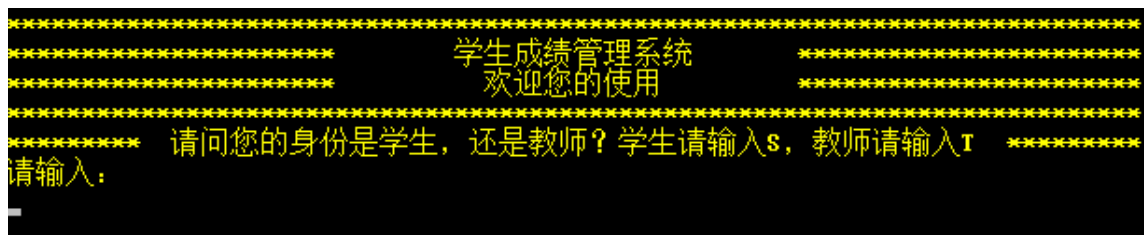
学生成绩管理系统的目的就在于方便学校对学生的成绩进行处理和应用，使教务人员从复杂的成绩管理中解脱出来。计算机具有运算速度快、处理能力强、可靠性高、存储量大等特点，这是传统的人工管理所不能企及的。通过学生成绩管理系统，可以实现对学生成绩的各项处理，并且提高成绩管理的效率和水平；同时，对学生成绩的深入分析可以判断学生的学习方式，甚至能据此针对性的提出一些学习的建议，这对学校教务教学工作的发展具有重要的推动作用。

本系统的操作过程主要根据学生和教师两个不同的身份进行操作，在进入系统之前要选择自己的身份并且输入学号/工作号及密码，在系统确认学号/工作号完全匹配准确无误之后，才能进入系统，根据不同身份进行操作。

第二部分 系统使用

1. 进入界面：

点击“学生成绩管理系统.exe”文件之后，即可进入下面的界面：



根据个人身份，学生登录时输入“S”或“s”（表示 Student），教师登录时输入“T”或“t”（表示 Teacher）。点击回车键即可进入登录界面。

2. 登录系统

①学生：

输入“S”后，进入学生登录界面（如下图）：

```
*****请您先登录系统*****
请输入您的学号:
学号: _
```

此时需要输入学号和密码。以默认数据文件“G:\\student.txt”中的张飞同学为例。张飞同学学号为 2014011216，密码为 1321232。

```
*****请您先登录系统*****
请输入您的学号:
学号:2014011216
密码:1321232_
```

回车键确认之后，登录成功！即可进入学生成绩管理系统欢迎界面：

```
*****
*****张飞同学,您好!*****
*****欢迎使用学生成绩管理系统*****
*****菜 单*****
*****1.查询个人成绩*****
*****2.学生成绩排序*****
*****3.退出系统*****
*****
*****
请输入您要执行的操作序号
```

【注意】如果学号和密码不匹配，系统会显示：

```
请重新登录:
*****请您先登录系统*****
请输入您的学号:
学号:
```

此时需确认学号和密码，重新登录。

②教师登录:

输入“T”/“t”之后，进入教师登录系统。步骤与学生登录系统相似，只需要正确输入工作号及密码即可。以微积分教师 Mike 为例：

```
*****请您先登录系统*****
请输入您的工作号:
工作号:201401121
密码:235862
```

```

*****
*****      微积分 Mike老师,您好!      *****
*****      欢迎使用学生成绩管理系统      *****
*****      菜 单      *****
*****      1.添加学生成绩      *****
*****      2.删除学生成绩      *****
*****      3.修改学生成绩      *****
*****      4.查询学生成绩      *****
*****      5.学生成绩排序      *****
*****      6.导入外部文件      *****
*****      7.显示所有成绩      *****
*****      8.显示平均成绩      *****
*****      9.退出      系统      *****
*****
请输入您要执行的操作序号

```

即可进入教师管理系统。

3. 学生功能模块

①查询个人成绩:

在提示“请输入您要执行的操作序号”中输入 1,即可进入“查询个人成绩”模块。进入之前,为了尽力保护您的隐私,需要再次输入密码。之后即可得到:

```

*****
*****      张飞同学,您好!      *****
*****      欢迎使用学生成绩管理系统      *****
*****      菜 单      *****
*****      1.查询个人成绩      *****
*****      2.学生成绩排序      *****
*****      3.退出      系统      *****
*****
请输入您要执行的操作序号
1
*****查询个人成绩*****
请再次输入您的密码
1321232
*****
1.查询微积分成绩
2.查询线性代数成绩
3.查询离散数学成绩
4.查询英语成绩
5.查看GPA
6.退出
请输入您的查询方式
_

```

根据提示,输入需要的查询方式。

例:输入 1 时,

```

请输入您的查询方式
1
姓名      学号      微积分
张飞      2014011216  94

```

输入 5 时，

姓名	学号	微积分	线性代数	离散数学	英语	GPA
张飞	2014011216	94	92	85	88	90.6429

②学生成绩排序：

```

*****
*****      张飞同学,您好!      *****
*****      欢迎使用学生成绩管理系统      *****
*****      菜  单      *****
*****      1.查询个人成绩      *****
*****      2.学生成绩排序      *****
*****      3.退出      系统      *****
*****
请输入您要执行的操作序号
2
*****个人成绩排名*****
请输入您的选项
1.微积分成绩排名
2.线性代数成绩排名
3.离散数学成绩排名
4.英语成绩排名
5.GPA排名
6.退出

```

学生成绩排序共分为“微积分成绩排名”“线性代数成绩排名”“离散数学成绩排名”“英语成绩排名”“GPA 排名”等。按提示输入任意选项即可得到相应的结果。

【注意】为了保护其他同学隐私，学生查询排名时，只会显示学号不会显示姓名。

例如：

输入 5，可得：

5
按照GPA从高到低排序<显示所有>，结果为：

排名	学号	微积分	线性代数	离散数学	英语	GPA
1	2014011216	94	92	85	88	90.6429
2	2014011245	85	74	85	96	83.4286
3	2014011258	75	89	64	85	78.0714
4	2014011215	63	89	84	75	76.6429
5	2014011256	85	69	84	56	76.0714
6	2014011265	98	74	58	28	72.5714
7	2014012553	58	93	72	58	71
8	2014013324	58	63	84	95	70.2857
9	2014011211	85	69	41	58	67.1429
10	2014011259	66	33	69	98	61.7857
11	2014011162	65	36	84	64	60.6429
12	2014025533	45	85	69	12	56.8571
13	2014011284	33	99	24	65	54.5
14	2014011243	26	95	64	22	53.2857
15	2014011233	25	96	41	28	49.1429
16	2014011285	25	65	26	86	45.3571
17	2014011285	26	14	28	32	23.8571

③退出系统

直接从当前界面退出学生成绩管理系统。

4. 教师功能模块

①添加学生成绩:

输入“1”，回车键确认之后，系统会先自动显示出系统内目前学生成绩数目：

```

*****
*****          微积分 Mike老师,您好!          *****
*****      欢迎使用学生成绩管理系统          *****
*****          菜 单          *****
*****      1.添加学生成绩          *****
*****      2.删除学生成绩          *****
*****      3.修改学生成绩          *****
*****      4.查询学生成绩          *****
*****      5.学生成绩排序          *****
*****      6.导入外部文件          *****
*****      7.显示所有成绩          *****
*****      8.显示平均成绩          *****
*****      9.退出          系统          *****
*****
请输入您要执行的操作序号
1
*****添加学生信息及成绩*****
学生成绩管理系统中现已存入17条学生成绩信息
现请输入第18条学生成绩信息请输入学生姓名:

```

按照提示，输入依次输入学生姓名、学号、微积分成绩、线性代数成绩、离散数学成绩、英语成绩，系统就会将该生录入。

例如：

依次输入冯哲、2014011121、98、59、87、49，可以得到：

```

1
*****添加学生信息及成绩*****
学生成绩管理系统中现已存入17条学生成绩信息
现请输入第18条学生成绩信息请输入学生姓名:
冯哲
请输入学生学号:
2014011121
请添加微积分成绩:
98
请添加线性代数成绩:
59
请添加离散数学成绩:
87
请添加英语成绩:
49
姓名          学号          微积分          线性代数          离散数学          英语          GPA
冯哲          2014011121          98          59          87          49          77.5
信息及成绩添加完毕!
请输入您要执行的操作序号

```

②删除学生成绩:

输入“2”后进入删除学生成绩模块:

a) 请先输入要删除的学生数目, 系统会判断是否超出了系统内的存储量。

b) 输入要删除的学生的学号, 即可从系统中删除该生信息。

```

*****
*****      微积分 Mike老师,您好!      *****
*****      欢迎使用学生成绩管理系统      *****
*****      菜 单      *****
*****      1.添加学生成绩      *****
*****      2.删除学生成绩      *****
*****      3.修改学生成绩      *****
*****      4.查询学生成绩      *****
*****      5.学生成绩排序      *****
*****      6.导入外部文件      *****
*****      7.显示所有成绩      *****
*****      8.显示平均成绩      *****
*****      9.退出 系统      *****
*****
请输入您要执行的操作序号
2
您需要删除多少名同学? 请输入数字
1
*****删除学生信息及其成绩*****
请输入您要删除的学生的学号
2014011216
学号为2014011216的学生成绩及个人信息均已删除完毕!
请输入您要执行的操作序号
  
```

③修改学生信息:

输入“3”, 进入更改学生信息模块。

```

*****      欢迎使用学生成绩管理系统      *****
*****      菜 单      *****
*****      1.添加学生成绩      *****
*****      2.删除学生成绩      *****
*****      3.修改学生成绩      *****
*****      4.查询学生成绩      *****
*****      5.学生成绩排序      *****
*****      6.导入外部文件      *****
*****      7.显示所有成绩      *****
*****      8.显示平均成绩      *****
*****      9.退出 系统      *****
*****
请输入您要执行的操作序号
3
*****【注意不允许批量修改】*****
*****更改学生信息*****
请输入您要修改的学生的学号
2014011216
该学号学生原来的成绩是:
姓名      学号      微积分      线性代数      离散数学      英语      GPA
张飞      2014011216      94      92      85      88      90.6429
您现在要执行的操作是: 修改学生微积分成绩
请输入微积分成绩
  
```

根据提示，依次修改微积分、线性代数、离散数学、英语的成绩。

例：修改张飞同学的英语成绩改为 90 分。结果如下：

```
*****更改学生信息*****
请输入您要修改的学生的学号
2014011216
该学号学生原来的成绩是：
姓名      学号      微积分      线性代数      离散数学      英语      GPA
张飞      2014011216      94      92      85      88      90.6429
您现在要执行的操作是：修改学生微积分成绩
请输入微积分成绩
94
您现在要执行的操作是：修改学生线性代数成绩
请输入线性代数成绩
92
您现在要执行的操作是：修改学生离散数学成绩
请输入离散数学成绩
85
您现在要执行的操作是：修改学生英语成绩
请输入英语成绩
90
张飞的同学们的成绩更改完毕！
修改后的信息如下：
姓名      学号      微积分      线性代数      离散数学      英语      GPA
张飞      2014011216      94      92      85      90      90.9286
请输入您要执行的操作序号
```

④查询学生成绩：

输入“4”，进入查询学生成绩功能模块。

输入要查询的学号，会出现 6 个选项。根据需要输入相应选项，即可得到结果。

例：需要查询学号为 2014011211 的学生的所有成绩时，如下：

```
请输入您要执行的操作序号
4
*****查询学生成绩*****
请输入您要查询的学生的学号：
2014011211
*****
1. 查询微积分成绩
2. 查询线性代数成绩
3. 查询离散数学成绩
4. 查询英语成绩
5. 查看GPA
6. 退出
请输入您的查询方式
5
姓名      学号      微积分      线性代数      离散数学      英语      GPA
刘爽      2014011211      85      69      41      58      67.1429
```

⑤学生成绩排序：

输入“5”，进入学生成绩排序功能模块，出现以下选项。任选其中之一，借得到相应结果。

```

***** 微积分 Mike老师,您好! *****
***** 欢迎使用学生成绩管理系统 *****
***** 菜 单 *****
***** 1.添加学生成绩 *****
***** 2.删除学生成绩 *****
***** 3.修改学生成绩 *****
***** 4.查询学生成绩 *****
***** 5.学生成绩排序 *****
***** 6.导入外部文件 *****
***** 7.显示所有成绩 *****
***** 8.显示平均成绩 *****
***** 9.退出 系统 *****
*****
请输入您要执行的操作序号
5
*****学生成绩排名*****
请输入您的选项
1.微积分成绩排名
2.线性代数成绩排名
3.离散数学成绩排名
4.英语成绩排名
5.GPA排名
6.退出

```

例如:

微积分成绩排名:

按照微积分成绩从高到低排序, 结果为:

排名	学号	微积分
1	2014011265	98
2	2014011216	94
3	2014011256	85
4	2014011211	85
5	2014011245	85
6	2014011258	75
7	2014011259	66
8	2014011162	65
9	2014011215	63
10	2014012553	58
11	2014013324	58
12	2014025533	45
13	2014011284	33
14	2014011243	26
15	2014011285	26
16	2014011285	25
17	2014011233	25

GPA 排名:

按照GPA从高到低排序<显示所有>，结果为：

排名	学号	微积分	线性代数	离散数学	英语	GPA
1	2014011216	94	92	85	88	90.6429
2	2014011245	85	74	85	96	83.4286
3	2014011258	75	89	64	85	78.0714
4	2014011215	63	89	84	75	76.6429
5	2014011256	85	69	84	56	76.0714
6	2014011265	98	74	58	28	72.5714
7	2014012553	58	93	72	58	71
8	2014013324	58	63	84	95	70.2857
9	2014011211	85	69	41	58	67.1429
10	2014011259	66	33	69	98	61.7857
11	2014011162	65	36	84	64	60.6429
12	2014025533	45	85	69	12	56.8571
13	2014011284	33	99	24	65	54.5
14	2014011243	26	95	64	22	53.2857
15	2014011233	25	96	41	28	49.1429
16	2014011285	25	65	26	86	45.3571
17	2014011285	26	14	28	32	23.8571

⑥导入外部文件：

根据提示，输入要导入的数据文件的正确路径。例如：

```

6
*****从文件导入学生信息及成绩*****
请输入您要导入的文件地址和文件名
G:\student.txt
G:\student.txt 打开成功，正在从文件导入...
***第1条学生信息及成绩导入成功***
***第2条学生信息及成绩导入成功***
***第3条学生信息及成绩导入成功***
***第4条学生信息及成绩导入成功***
***第5条学生信息及成绩导入成功***
***第6条学生信息及成绩导入成功***
***第7条学生信息及成绩导入成功***
***第8条学生信息及成绩导入成功***
***第9条学生信息及成绩导入成功***
***第10条学生信息及成绩导入成功***
***第11条学生信息及成绩导入成功***
***第12条学生信息及成绩导入成功***
***第13条学生信息及成绩导入成功***
***第14条学生信息及成绩导入成功***
***第15条学生信息及成绩导入成功***
***第16条学生信息及成绩导入成功***
***第17条学生信息及成绩导入成功***
本次共导入17条学生信息

```

⑦显示所有成绩：

输入“7”，即可进入显示所有成绩功能模块。

```

***** 欢迎使用学生成绩管理系统 *****
***** 菜 单 *****
***** 1. 添加学生成绩 *****
***** 2. 删除学生成绩 *****
***** 3. 修改学生成绩 *****
***** 4. 查询学生成绩 *****
***** 5. 学生成绩排序 *****
***** 6. 导入外部文件 *****
***** 7. 显示所有成绩 *****
***** 8. 显示平均成绩 *****
***** 9. 退出系统 *****
*****
请输入您要执行的操作序号
?
*****显示系统内所有学生成绩*****
1. 显示所有学生信息 (包括姓名和学号)
2. 显示所有学生信息及微积分成绩
3. 显示所有学生信息及线性代数成绩
4. 显示所有学生信息及离散数学成绩
5. 显示所有学生信息及英语成绩
6. 显示所有学生的各科成绩
7. 退出
请输入您的选项:

```

根据需要，输入选项，即可得到相应结果。

例如：

输入“6”，得到：

```

请输入您的选项:
6
所有学生信息及各科成绩如下:
姓名      学号      微积分      线性代数      离散数学      英语      GPA
张飞      2014011216  94          92            85            88        90.6429
张三      2014011256  85          69            84            56        76.0714
李四      2014011285  25          65            26            86        45.3571
马武      2014011284  33          99            24            65        54.5
徐帅      2014011243  26          95            64            22        53.2857
刘爽      2014011211  85          69            41            58        67.1429
姜鹤      2014011233  25          96            41            28        49.1429
何柳      2014011285  26          14            28            32        23.8571
刘伟      2014012553  58          93            72            58        71
马赫      2014011259  66          33            69            98        61.7857
王刚      2014011258  75          89            64            85        78.0714
龙俊亨    2014011245  85          74            85            96        83.4286
项少玲    2014011162  65          36            84            64        60.6429
闫平      2014011265  98          74            58            28        72.5714
汪廷栋    2014013324  58          63            84            95        70.2857
师素芳    2014025533  45          85            69            12        56.8571
安明玉    2014011215  63          89            84            75        76.6429
*****

```

⑧显示平均成绩:

根据不同学科教师的需要，设置了该功能。输入“8”进入该功能模块。根据需要选择不同科目的平均成绩或者平均 GPA。

例如：

```

请输入您要执行的操作序号
8
*****学生平均成绩*****
请问您想查询哪一门<总分>的平均成绩?
1.微积分
2.线性代数
3.离散数学
4.英语
5.GPA
请输入: 1
平均成绩为:59.5294
请输入您要执行的操作序号
8
*****学生平均成绩*****
请问您想查询哪一门<总分>的平均成绩?
1.微积分
2.线性代数
3.离散数学
4.英语
5.GPA
请输入: 2
平均成绩为:72.6471
请输入您要执行的操作序号

```

⑨退出系统。

直接从教师功能模块退出，退出学生成绩管理系统。

第三部分 注意事项

- ① 使用该系统时尽量不要一次性调用过多功能，可能会影响性能。
- ② 部分功能暂不支持返回上一级菜单的功能。
- ③ 注意“导入外部文件”时，数据文件内的数据必须按照“姓名”“学号”“微积分成绩”“线性代数成绩”“离散数学成绩”“英语成绩”“密码”的数据存入，否则会造成文件导入错误。

6. 总结及编程体会

整个系统是自己构思、设计并使之实现的，因而当最终得到自己第一个具有实际应用意义的小系统的时候，还是比较兴奋的。其实，一开始拿到老师布置的题目的时候，脑子里完全没有想法，而且害怕自己满足不了要求。作业中要求“至少定义四个类”，但我起初认为只要学生和教师两个类就差不多够用了。但是，经过仔细思考，认为虚基类和抽象类都应该有。**Person** 表示一个人，作为抽象类；**School_Member** 学校中的人员，作为虚基类；**Student** 和 **Teacher** 类作为主体，再加上管理和领导类 **Manage** 和 **Director** 就可以实现一切功能了。

有了明确的思路之后，就要考虑系统具有哪些功能，并且如何实现这些功能。

在实现功能的过程中几个主要的 Bug 耗费了一些时间，但总体进行比较顺利。在此次课程设计中，不得不说遇到了很多问题，有的上网查找相关资料，有的问题直接向助教请教，确实收获了不少新的知识。但也给了我足够的警示作用：自己平时课后对知识和技巧的掌握还很欠缺，语法方面也不够规范。

总之，此次编写“学生成绩管理系统”，让我对编程有了更直接的体验，也积累了一定的 debug 经验。更重要的是，我感觉得到自己的编程能力提高了，而且自己对编程也有了更浓厚的兴趣。希望通过更深入的学习和实验，我的编程能力能够稳步提高。

附录：源程序清单

```
//头文件"Class.h"主要包含了所有的类及其成员函数
#include<iostream>
#include<fstream>//文件流
#include<string>//字符串统一用C++的string类
#include<iomanip>// I/O格式控制
#define calculus_score 5 //设置微积分学分
#define linear_score 4 //设置线性代数学分
#define discrete_score 3 //设置离散数学学分
#define english_score 2 //设置英语学分
#define M 500//默认最大学生人数
#define N 20//默认教师人数
using namespace std;
//Person类作为抽象类
class Person
{
public:
    virtual string GetName()=0;
    virtual int GetID()=0;
};
//School_Member类作为虚基类，公有继承Person类，并派生出Student和Teacher类;
class School_Member: public Person
{
public:
    School_Member(string name,int id);
    School_Member();
    string GetName();//取姓名
    int GetID();//取学号或工作号
protected:
    string Name;
```

```

    int ID;
};
School_Member::School_Member(string name,int id)
{
    Name=name;
    ID=id;
}
School_Member::School_Member()
{
    Name=" ";
    ID=0;
}
string School_Member::GetName()
{
    return Name;
}
int School_Member::GetID()
{
    return ID;
}
//学生类，公有继承虚基类School_Member
class Student:virtual public School_Member
{
public:
    Student(string name,int id,double calculus,double linear,double discrete,double
english,int code);
    Student();
    bool IsEmpty();//学生状态函数，true表示学生存在，false表示学生信息为空
    //后面个函数分科修改学生成绩
    void Alter_calculus();//修改微积分成绩
    void Alter_linear();//修改线性代数成绩
    void Alter_discrete();//修改离散数学成绩
    void Alter_english();//修改英语成绩
    int GetCode();//取密码（登录时匹配）
    //后面个函数为了得到学生的各科成绩及GPA
    double GetCalculus();//取微积分成绩
    double GetLinear_algebra();//取线性代数成绩
    double GetDicrete_Math();//取离散数学成绩
    double GetEnglish();//取英语成绩
    double GetGPA();//取GPA，包含GPA计算
    void Delete();    //删除学生信息
    void From_File(string name,int id,double caculus,double linear,
        double discrete, double english,int code);//从文件导入学生信息及成绩
    //为了满足后面不同函数显示成绩的需要，以下个函数满足显示学生成绩的

```

需要

```

void Display_Calculus();//显示微积分成绩函数
void Display_Linear();//显示线性代数成绩函数
void Display_Discrete();//显示离散数学成绩函数
void Display_English();//显示英语成绩函数
void Display_All();//显示所有成绩及GPA
friend ostream & operator <<(ostream &out, Student &s);
    //操作符重载，输出学生对象即可输出该生的所有信息及成绩
protected://保护成员
    double Calculus,Linear_algebra,Discrete_math,English;
    int Code;
};
Student::Student(string name,int id,double calculus,double linear,double
discrete,double english, int code)
{
    School_Member::School_Member(name,id);
    Calculus=calculus;
    Linear_algebra=linear;
    Discrete_math=discrete;
    English=english;
    Code=code;
}
Student::Student()
{
    School_Member::School_Member();
    Calculus=0.0;
    Linear_algebra=0.0;
    Discrete_math=0.0;
    English=0.0;
    Code=0;
}
bool Student::IsEmpty()
{
    if(ID==0)
        return true;
    else return false;
}
//下面个修改成绩
void Student::Alter_calculus()
{
    cout<<"您现在要执行的操作是： 修改学生微积分成绩"<<endl;
    double calculus;
    cout<<"请输入微积分成绩"<<endl;
    cin>>calculus;
}

```

```
    Calculus=calculus;
}
void Student::Alter_linear()
{
    cout<<"您现在要执行的操作是：修改学生线性代数成绩"<<endl;
    double linear;
    cout<<"请输入线性代数成绩"<<endl;
    cin>>linear;
    Linear_algebra=linear;
}
void Student::Alter_discrete()
{
    cout<<"您现在要执行的操作是：修改学生离散数学成绩"<<endl;
    double discrete;
    cout<<"请输入离散数学成绩"<<endl;
    cin>>discrete;
    Discrete_math=discrete;
}
void Student::Alter_english()
{
    cout<<"您现在要执行的操作是：修改学生英语成绩"<<endl;
    double english;
    cout<<"请输入英语成绩"<<endl;
    cin>>english;
    English=english;
}
//获取密码
int Student::GetCode()
{
    return Code;
}
//下面个获取各科成绩
double Student::GetCalculus()
{
    return Calculus;
}
double Student::GetLinear_algebra()
{
    return Linear_algebra;
}
double Student::GetDicrete_Math()
{
    return Discrete_math;
}
```

```
double Student::GetEnglish()
{
    return English;
}
//获取GPA（包含了GPA的计算）
double Student::GetGPA()
{
    double sum_grade,sum_score,GPA;
    sum_grade=Calculus*calculus_score+Linear_algebra*linear_score
        +Discrete_math*discrete_score+English*english_score;
    sum_score=calculus_score+linear_score+discrete_score+english_score;
    GPA=sum_grade/sum_score;
    return GPA;
}
void Student::Delete()
{
    Name=" ";
    ID=0;
    Calculus=0.0;
    Linear_algebra=0.0;
    Discrete_math=0.0;
    English=0.0;
}
void Student::From_File(string name, int id, double calculus, double linear, double
discrete, double english,int code)
{
    Name=name;
    ID=id;
    Calculus=calculus;
    Linear_algebra=linear;
    Discrete_math=discrete;
    English=english;
    Code=code;
}
//后面个显示学生成绩及GPA
void Student::Display_Calculus()
{
    cout<<setw(15)<<Name<<setw(12)<<ID<<setw(9)<<Calculus<<endl;
}
void Student::Display_Linear()
{
    cout<<setw(15)<<Name<<setw(12)<<ID<<setw(9)<<Linear_algebra<<endl;
```



```

}
void Student::Display_Discrete()
{
    cout<<setiosflags(ios::left)<<setw(15)<<Name<<setw(12)<<ID<<setw(9)<<Discrete_math<<endl;
}
void Student::Display_English()
{
    cout<<setiosflags(ios::left)<<setw(15)<<Name<<setw(12)<<ID<<setw(9)<<English<<endl;
}
void Student::Display_All()
{
    cout<<setiosflags(ios::left)<<setw(15)<<Name<<setw(12)<<ID<<setw(9)
        <<Calculus<<setw(9)<<Linear_algebra<<setw(9)<<Discrete_math
        <<setw(9)<<English<<setw(9)<<GetGPA()<<endl;
}
ostream & operator << (ostream &out,Student &s)//“<<”操作符重载
{
    cout<<setiosflags(ios::left)<<setw(15)<<s.GetName()<<setw(12)<<s.GetID()<<setw(9)
        <<s.GetCalculus()<<setw(9)<<s.GetLinear_algebra
        (<<setw(9)<<s.GetDiscrete_Math()
        <<setw(9)<<s.GetEnglish()<<setw(9)<<s.GetGPA()<<endl;
    return out;
}
//教师类公有继承虚基类School_Member
class Teacher: virtual public School_Member
{
public:
    Teacher(string name,int id,string subject,int code);//构造函数
    Teacher();
    int GetCode();//获取密码
    string GetSubject();//获取教师教的学科
    void From_File(string name,int id,string subject,int code);//从文件导入教师信息
    void In_Student_File();//向系统导入默认文件中的学生信息及成绩
    void Add();//增加学生信息及成绩
    void Delete();//删除学生信息及成绩
    void Change();//修改学生信息及成绩
    void Search_Student();//学生界面查询学生成绩
    void Search_Teacher();//教师界面查询学生成绩
    void Sort_Teacher();//教师和学生界面内“学生成绩排序”
    double GetAverage();//取各科成绩的平均值

```

```

void A_S_D();//显示系统内所有学生所有科目成绩
void All_Student_Display();//为了多次调用，将“操作提示”单独作为一个成员
函数
void In_File_Data();//从系统外文件导入学生信息及成绩
int GetNum();//取系统内存入的学生数目
friend ostream & operator <<(ostream &out, Teacher &t);//操纵流重载，输出
教师信息
protected:
    Student STU[M];//学生类对象作为Teacher类的成员
    int Num;
    int Code;
    string Subject;
};
Teacher::Teacher(string name,int id,string subject,int code)
{
    School_Member::School_Member(name,id);
    Code=code;
    Subject=subject;
}
Teacher::Teacher()
{
    School_Member::School_Member();
}
int Teacher::GetCode();//取密码
{
    return Code;
}
string Teacher::GetSubject();//取科目
{
    return Subject;
}
void Teacher::From_File(string name,int id,string subject,int code)
{
    Name=name;
    ID=id;
    Subject=subject;
    Code=code;
}
void Teacher::In_Student_File();//导入默认文件内的学生信息
{
    ifstream in_File;
    string name;
    int i=0,id,code;
    double calculus, linear, discrete, english;

```

```

in_File.open("G:\\student.txt",ios::in);
if(in_File==0)
{
    cout<<"文件打开失败！"<<endl;
    exit(0);
}
else//从文件将学生信息及成绩逐一导入
{
    Num=0;
    while(in_File&&!in_File.eof())
    {
        in_File>>name>>id>>calculus>>linear>>discrete>>english>>code;
        STU[i].From_File(name,id,calculus,linear,discrete,english,code);
        Num++;
        i++;
    }
    in_File.close();//关闭文件
}
}
void Teacher::Add()//增加学生信息及成绩
{
    int i,id;i=Num;
    Student ST;
    string _name;
    double calculus, linear, discrete,english;
    cout<<"*****添加学生信息及成绩*****"<<endl;
    cout<<"学生成绩管理系统中现已存入"<<i<<"条学生成绩信息"<<endl;
    cout<<"现请输入第"<<i+1<<"条学生成绩信息";
    cout<<"请输入学生姓名： "<<endl;    cin>>_name;
    cout<<"请输入学生学号： "<<endl;    cin>>id;
    cout<<"请添加微积分成绩： "<<endl; cin>>calculus;
    cout<<"请添加线性代数成绩： "<<endl;    cin>>linear;
    cout<<"请添加离散数学成绩： "<<endl;    cin>>discrete;
    cout<<"请添加英语成绩： "<<endl;    cin>>english;
    ST.From_File(_name,id,calculus,linear,discrete,english,id);
    cout<<setiosflags(ios::left)<<setw(15)<<"姓名"<<setw(12)<<"学号"
    "<<setw(9)<<"微积分"
        <<setw(9)<<"线性代数"<<setw(9)<<"离散数学"<<setw(9)<<"英语"
    "<<setw(9)<<"GPA"<<endl;
    cout<<ST;
    cout<<"信息及成绩添加完毕！"<<endl;
}
void Teacher::Delete()//删除学生信息及成绩
{

```

```

int id,i;
cout<<"*****删除学生信息及其成绩*****"<<endl;
cout<<"请输入您要删除的学生的学号"<<endl;//通过学号删除学生
cin>>id;
for(i=0;i<M;i++)
{
    if(!STU[i].IsEmpty()&&STU[i].GetID()==id)
    {
        STU[i].Delete();
        cout<<"学号为"<<id<<"的学生成绩及个人信息均已删除完毕！"
"<<endl;
        Num--;
    }
    break;
}
}
void Teacher::Change()//修改学生信息及成绩
{
    In_Student_File();int i,id;
    cout<<"*****更改学生信息*****"<<endl;
    cout<<"请输入您要修改的学生的学号"<<endl;
    cin>>id;
    for(i=0;i<Num;i++)
    {
        if(id==STU[i].GetID())
        {
            cout<<"该学号学生原来的成绩是： "<<endl;
            cout<<setiosflags(ios::left)<<setw(15)<<"姓名"<<setw(12)<<"学号"
"<<setw(9)<<"微积分"
                <<setw(9)<<"线性代数"<<setw(9)<<"离散数学"<<setw(9)<<"英语"
"<<setw(9)<<"GPA"<<endl;
            cout<<STU[i];
            STU[i].Alter_calculus();//修改微积分成绩
            STU[i].Alter_linear();//修改线性代数成绩
            STU[i].Alter_discrete();//修改离散数学成绩
            STU[i].Alter_english();//修改英语成绩
            cout<<STU[i].GetName()<<"的同学的成绩更改完毕！ "<<endl;
            cout<<"修改后的信息如下： "<<endl;
            cout<<setiosflags(ios::left)<<setw(15)<<"姓名"<<setw(12)<<"学号"
"<<setw(9)<<"微积分"
                <<setw(9)<<"线性代数"<<setw(9)<<"离散数学"<<setw(9)<<"英语"
"<<setw(9)<<"GPA"<<endl;
            cout<<STU[i];
        }
    }
}

```

```

        break;
    }
    if(i==M)
        cout<<"对不起！您输入的学号有误！请查证！"<<endl;
}
void Teacher::Search_Student()//查询学生成绩（学生和教师界面共用）
{
    int i,command;
    int code;
    In_Student_File();
    cout<<"*****查询个人成绩*****"<<endl;
    cout<<"请再次输入您的密码"<<endl;
    cin>>code;
    cout<<"*****"<<endl;
    cout<<"1.查询微积分成绩"<<endl;
    cout<<"2.查询线性代数成绩"<<endl;
    cout<<"3.查询离散数学成绩"<<endl;
    cout<<"4.查询英语成绩"<<endl;
    cout<<"5.查看GPA"<<endl;
    cout<<"6.退出"<<endl;
    for(i=0;i<M;i++)
    {
        if(STU[i].GetCode()==code)
        {
            cout<<"请输入您的查询方式"<<endl;
            cin>>command;
            switch(command)
            {
                case 1:
                    cout<<setiosflags(ios::left)<<setw(15)<<"姓名"<<setw(12)<<"学号"
"<<setw(9)<<"微积分"<<endl;
                    STU[i].Display_Calculus();break;
                case 2:
                    cout<<setiosflags(ios::left)<<setw(15)<<"姓名"<<setw(12)<<"学号"
"<<setw(9)<<"线性代数"<<endl;
                    STU[i].Display_Linear();break;
                case 3:
                    cout<<setiosflags(ios::left)<<setw(15)<<"姓名"<<setw(12)<<"学号"
"<<setw(9)<<"离散数学"<<endl;
                    STU[i].Display_Discrete();break;
                case 4:
                    cout<<setiosflags(ios::left)<<setw(15)<<"姓名"<<setw(12)<<"学号"
"<<setw(9)<<"英语"<<endl;
                    STU[i].Display_English();break;
            }
        }
    }
}

```

```

        case 5:
            cout<<setiosflags(ios::left)<<setw(15)<<"姓名"<<setw(12)<<"学号"
            "<<setw(9)<<"微积分"
                <<setw(9)<<"线性代数"<<setw(9)<<"离散数学"<<setw(9)<<"
            英语"<<setw(9)<<"GPA"<<endl;
            cout<<STU[i];break;
        case 6:exit(0);break;
        default:
            cout<<"对不起！您的输入有误"<<endl;
            system("cls");
            Search_Student();
        }
    }
}

void Teacher::Search_Teacher()//查询学生成绩（学生和教师界面共用）
{
    int i,command;
    int id;
    In_Student_File();
    cout<<"*****查询学生成绩*****"<<endl;
    cout<<"请输入您要查询的学生的学号："<<endl;
    cin>>id;
    cout<<"*****"<<endl;
    cout<<"1.查询微积分成绩"<<endl;
    cout<<"2.查询线性代数成绩"<<endl;
    cout<<"3.查询离散数学成绩"<<endl;
    cout<<"4.查询英语成绩"<<endl;
    cout<<"5.查看GPA"<<endl;
    cout<<"6.退出"<<endl;
    for(i=0;i<M;i++)
    {
        if(STU[i].GetID()==id)
        {
            cout<<"请输入您的查询方式"<<endl;
            cin>>command;
            switch(command)
            {
                case 1:
                    cout<<setiosflags(ios::left)<<setw(15)<<"姓名"<<setw(12)<<"学号"
                    "<<setw(9)<<"微积分"<<endl;
                    STU[i].Display_Calculus();break;
                case 2:
                    cout<<setiosflags(ios::left)<<setw(15)<<"姓名"<<setw(12)<<"学号"

```

```

"<<setw(9)<<"线性代数"<<endl;
    STU[i].Display_Linear();break;
    case 3:
        cout<<setiosflags(ios::left)<<setw(15)<<"姓名"<<setw(12)<<"学号"
"<<setw(9)<<"离散数学"<<endl;
        STU[i].Display_Discrete();break;
    case 4:
        cout<<setiosflags(ios::left)<<setw(15)<<"姓名"<<setw(12)<<"学号"
"<<setw(9)<<"英语"<<endl;
        STU[i].Display_English();break;
    case 5:
        cout<<setiosflags(ios::left)<<setw(15)<<"姓名"<<setw(12)<<"学号"
"<<setw(9)<<"微积分"
        <<setw(9)<<"线性代数"<<setw(9)<<"离散数学"<<setw(9)<<"
        英语"<<setw(9)<<"GPA"<<endl;
        cout<<STU[i];break;
    case 6:exit(0);break;
    default:
        cout<<"对不起！您的输入有误"<<endl;
        system("cls");
        Search_Teacher();
    }
}
}
}
// 【注意】为在一定程度上保护个人隐私，显示排名时不显示姓名，只显示学
号
void Teacher::Sort_Teacher()//教师和学生界面内“学生成绩排序功能”
{
    int command;
    Student *Stu[M],*s;//使用指向类的对象的指针，用来成绩的排序
    In_Student_File();//先从文件中导入学生信息及成绩
    for(int z=0;z<Num;z++)//指针指向对象
        Stu[z]=&STU[z];
    cout<<"*****学生成绩排名*****"<<endl;
    cout<<"请输入您的选项"<<endl;
    cout<<"1.微积分成绩排名"<<endl;
    cout<<"2.线性代数成绩排名"<<endl;
    cout<<"3.离散数学成绩排名"<<endl;
    cout<<"4.英语成绩排名"<<endl;
    cout<<"5.GPA排名"<<endl;
    cout<<"6.退出"<<endl;
    cin>>command;
    if(command==1)//微积分排名

```

```

{
    int i,j,flag;
    for(i=1; i<Num; i++)
    {
        flag=0;
        for(j=0;j<Num-i;j++)
            if(Stu[j]->GetCalculus()<Stu[j+1]->GetCalculus())
            {
                s=Stu[j];Stu[j]=Stu[j+1];Stu[j+1]=s;
                flag=1;
            }
        if(!flag) break;
    }
    cout<<"按照微积分成绩从高到低排序，结果为："<<endl;
    cout<<setiosflags(ios::left)<<setw(6)<<"排名"<<setw(12)<<"学号"
    "<<setw(9)<<"微积分"<<endl;
    for(i=0;i<Num;i++)
    {
        cout<<setw(6)<<i+1;

        cout<<setw(12)<<Stu[i]->GetID()<<setw(9)<<Stu[i]->GetCalculus()<<endl;
    }
}
else if(command==2)//线性代数排名
{
    int i,j,flag;
    for(i=1; i<Num; i++)
    {
        flag=0;
        for(j=0;j<Num-i;j++)
            if(Stu[j]->GetLinear_algebra()<Stu[j+1]->GetLinear_algebra())
            {
                s=Stu[j];Stu[j]=Stu[j+1];Stu[j+1]=s;
                flag=1;
            }
        if(!flag) break;
    }
    cout<<"按照线性代数成绩从高到低排序，结果为："<<endl;
    cout<<setiosflags(ios::left)<<setw(6)<<"排名"<<setw(12)<<"学号"
    "<<setw(9)<<"线性代数"<<endl;
    for(i=0;i<Num;i++)
    {
        cout<<setw(6)<<i+1;;

```



```

        cout<<setw(12)<<Stu[i]->GetID()<<setw(9)<<Stu[i]->GetLinear_algebra()<<endl;
    }
}
else if(command==3)//离散数学排名
{
    int i,j,flag;
    for(i=1; i<Num; i++)
    {
        flag=0;
        for(j=0;j<Num-i;j++)
            if(Stu[j]->GetDicrete_Math()<Stu[j+1]->GetDicrete_Math())
            {
                s=Stu[j];Stu[j]=Stu[j+1];Stu[j+1]=s;
                flag=1;
            }
        if(!flag) break;
    }
    cout<<"按照离散数学成绩从高到低排序，结果为："<<endl;
    cout<<setiosflags(ios::left)<<setw(6)<<"排名"<<setw(12)<<"学号"
    "<<setw(9)<<"离散数学"<<endl;
    for(i=0;i<Num;i++)
    {
        cout<<setw(6)<<i+1;

        cout<<setw(12)<<Stu[i]->GetID()<<setw(9)<<Stu[i]->GetDicrete_Math()<<endl;
    }
}
else if(command==4)//英语成绩排名
{
    int i,j,flag;
    for(i=1; i<Num; i++)
    {
        flag=0;
        for(j=0;j<Num-i;j++)
            if(Stu[j]->GetEnglish()<Stu[j+1]->GetEnglish())
            {
                s=Stu[j];Stu[j]=Stu[j+1];Stu[j+1]=s;
                flag=1;
            }
        if(!flag) break;
    }
    cout<<"按照英语成绩从高到低排序，结果为："<<endl;
    cout<<setiosflags(ios::left)<<setw(6)<<"排名"<<setw(12)<<"学号"
    "<<setw(9)<<"英语"<<endl;
}

```

```

        for(i=0;i<Num;i++)
        {
            cout<<setw(6)<<i+1;
            cout<<setw(12)<<Stu[i]->GetID()<<setw(9)<<Stu[i]->GetEnglish()<<endl;
        }
    }
    else if(command==5)//GPA排名
    {
        int i,j,flag;
        for(i=1; i<Num; i++)
        {
            flag=0;
            for(j=0;j<Num-i;j++)
                if(Stu[j]->GetGPA()<Stu[j+1]->GetGPA())
                {
                    s=Stu[j];Stu[j]=Stu[j+1];Stu[j+1]=s;
                    flag=1;
                }
            if(!flag) break;
        }
        cout<<"按照GPA从高到低排序(显示所有), 结果为: "<<endl;
        cout<<setiosflags(ios::left)<<setw(6)<<"排名"<<setw(12)<<"学号"
        "<<setw(9)<<"微积分"
        <<setw(9)<<"线性代数"<<setw(9)<<"离散数学"<<setw(9)<<"英语"
        "<<setw(9)<<"GPA"<<endl;
        for(i=0;i<Num;i++)
        {
            cout<<setw(6)<<i+1;

            cout<<setw(12)<<Stu[i]->GetID()<<setw(9)<<Stu[i]->GetCalculus()<<setw(9)<<Stu[i]->GetLinear_algebra()

            <<setw(9)<<Stu[i]->GetDicrete_Math()<<setw(9)<<Stu[i]->GetEnglish()<<setw(9)
            <<Stu[i]->GetGPA()<<endl;
        }
    }
    else if(command==6)
        exit(0);
    else cout<<"对不起, 您输入有误! "<<endl;
}

double Teacher::GetAverage()//获取学生单科或总GPA的平均成绩
{
    int i,command;
    double sum=0,average;

```

```

In_Student_File();
cout<<"*****学生平均成绩*****"<<endl;
cout<<"请问您想查询哪一门(总分)的平均成绩?"<<endl;
cout<<"1.微积分"<<endl<<"2.线性代数"<<endl<<"3.离散数学"<<endl<<"4.英
语"<<endl<<"5.GPA"<<endl;
cout<<"请输入： "; cin>>command;
if(command==1)
{
    for(i=0;i<Num;i++)
        sum+=STU[i].GetCalculus();
    average=sum/Num;
    return average;//微积分平均成绩
}
else if(command==2)
{
    sum=0.0;
    for(i=0;i<Num;i++)
        sum+=STU[i].GetLinear_algebra();
    average=sum/Num;
    return average;//线性代数平均成绩
}
else if(command==3)
{
    sum=0.0;
    for(i=0;i<Num;i++)
        sum+=STU[i].GetDicrete_Math();
    average=sum/Num;
    return average;//离散数学平均成绩
}
else if(command==4)
{
    sum=0.0;
    for(i=0;i<Num;i++)
        sum+=STU[i].GetEnglish();
    average=sum/Num;
    return average;//英语平均成绩
}
else if(command==5)
{
    sum=0.0;
    for(i=0;i<Num;i++)
        sum+=STU[i].GetGPA();
    average=sum/Num;
    return average;//平均GPA
}

```

```

    }
    else return 0;
}
void Teacher::A_S_D()//显示所有成绩函数的主体
{
    cout<<"请输入您的选项: "<<endl;
    int i,command;
    In_Student_File();//导入学生信息及成绩
    cin>>command;
    //包含单科显示和整体显示
    if(command==1)
    {
        cout<<"所有学生信息（包括姓名和学号）如下:"<<endl;
        cout<<setiosflags(ios::left)<<setw(15)<<"姓名"<<setw(12)<<"学号"<<endl;
        for(i=0;i<Num;i++)

            cout<<setiosflags(ios::left)<<setw(12)<<STU[i].GetName()<<setw(9)<<STU[i].Get
            ID()<<endl;
            for(int i=0;i<50;i++)
            {cout<<"*";}
            cout<<endl;
        }
        else if(command==2)
        {
            cout<<"所有学生信息及微积分成绩如下: "<<endl;
            cout<<setiosflags(ios::left)<<setw(15)<<"姓名"<<setw(12)<<"学号
            "<<setw(9)<<"微积分"<<endl;
            for(i=0;i<Num;i++)
                STU[i].Display_Calculus();
            for(int i=0;i<50;i++)
            {cout<<"*";}
            cout<<endl;
        }
        else if(command==3)
        {
            cout<<"所有学生信息及线性代数成绩如下: "<<endl;
            cout<<setiosflags(ios::left)<<setw(15)<<"姓名"<<setw(12)<<"学号
            "<<setw(9)<<"线性代数"<<endl;
            for(i=0;i<Num;i++)
                STU[i].Display_Linear();
            for(int i=0;i<50;i++)
            {cout<<"*";}
            cout<<endl;
        }
    }
}

```

```

else if(command==4)
{
    cout<<"所有学生信息及离散数学成绩如下： "<<endl;
    cout<<setiosflags(ios::left)<<setw(15)<<"姓名"<<setw(12)<<"学号"
"<<setw(9)<<"离散数学"<<endl;
    for(i=0;i<Num;i++)
        STU[i].Display_Discrete();
    for(int i=0;i<50;i++)
    {cout<<"*";}
    cout<<endl;
}
else if(command==5)
{
    cout<<"所有学生信息及英语成绩如下： "<<endl;
    cout<<setiosflags(ios::left)<<setw(15)<<"姓名"<<setw(12)<<"学号"
"<<setw(9)<<"英语"<<endl;
    for(i=0;i<Num;i++)
    {

        STU[i].Display_Linear();
    }
    for(int i=0;i<50;i++)
    {cout<<"*";}
    cout<<endl;
}
else if(command==6)
{
    cout<<"所有学生信息及各科成绩如下： "<<endl;
    cout<<setiosflags(ios::left)<<setw(15)<<"姓名"<<setw(12)<<"学号"
"<<setw(9)<<"微积分"
        <<setw(9)<<"线性代数"<<setw(9)<<"离散数学"<<setw(9)<<"英语"
"<<setw(9)<<"GPA"<<endl;
    for(i=0;i<Num;i++)
        cout<<STU[i];
    for(int i=0;i<50;i++)
    {cout<<"*";}
    cout<<endl;
}
else if(command==7)
    exit(0);
else
{
    cout<<"请按提示正确输入！ "<<endl;
    cin.get();
}

```

```

    }
    A_S_D();//循环调用，避免重新登录系统，提高系统效率
}
void Teacher::All_Student_Display();//显示所有成绩的提示部分
{
    cout<<"*****显示系统内所有学生成绩*****"<<endl;
    cout<<"1.显示所有学生信息（包括姓名和学号）"<<endl;
    cout<<"2.显示所有学生信息及微积分成绩"<<endl;
    cout<<"3.显示所有学生信息及线性代数成绩"<<endl;
    cout<<"4.显示所有学生信息及离散数学成绩"<<endl;
    cout<<"5.显示所有学生信息及英语成绩"<<endl;
    cout<<"6.显示所有学生的各科成绩"<<endl;
    cout<<"7.退出"<<endl;
    A_S_D();
}
void Teacher::In_File_Data();//从外部文件导入学生信息及成绩
{
    ifstream in_File;
    char name[20];
    int i=0,id,code;
    char File_Name[20];
    double calculus, linear, discrete, english;
    cout<<"*****从文件导入学生信息及成绩*****"<<endl;
    cout<<"请输入您要导入的文件地址和文件名"<<endl;
    cin>>File_Name;//输入要导入的文件的准确地址&文件名
    in_File.open(File_Name,ios::out);
    if(in_File==0)
    {
        cout<<"文件打开失败！"<<endl;
        exit(0);
    }
    else
    {
        Num=0;
        cout<<File_Name<<"打开成功，正在从文件导入..."<<endl;
        while(in_File&&!in_File.eof())
        {
            in_File>>name>>id>>calculus>>linear>>discrete>>english>>code;
            STU[i].From_File(name,id,calculus,linear,discrete,english,code);
            cout<<"***第"<<i+1<<"条学生信息及成绩导入成功***"<<endl;
            Num++;
            i++;
        }
    }
}

```

```

        cout<<"本次共导入"<<Num<<"条学生信息"<<endl;
        in_File.close();//关闭文件
    }
}
int Teacher::GetNum()//获取目前系统中学生的数目
{
    return Num;
}
ostream & operator <<(ostream &out, Teacher &t)//操作符重载
{
    cout<<setiosflags(ios::left)<<setw(15)<<t.GetName()<<setw(12)<<t.GetID()<<setw(9)
        <<t.GetSubject()<<endl;
    return out;
}
//Manage派生类，主要负责学生的登录，具体学生功能的实现在Teacher类中已包含
class Manage: virtual public Student,virtual public Teacher
{
public:
    Manage(string name,int ID,double caculus,double linear,double discrete,double english,int code);
    Manage();
    void Login_Student();//学生登录成员函数
};
Manage::Manage(string name,int ID,double caculus,double linear,double discrete,double english,int code)
{
    Student::Student(name,ID,caculus,linear,discrete,english,code);
};

Manage::Manage()
{
    Student::Student();
}

void Manage::Login_Student()
{
    int i,id,code;
    Teacher::In_Student_File();
    cout<<"*****请您先登录系统*****"<<endl;
    cout<<"请输入您的学号: "<<endl;
    cout<<"学号:";
    cin>>id;
}

```

```

    cout<<"密码:";
    cin>>code;
    for(i=0;i<Teacher::GetNum();i++)
    {
        if(Teacher::STU[i].GetID()==id&&Teacher::STU[i].GetCode()==code)//学生学
        号和密码匹配之后才能进入系统
        {
            cout<<"恭喜您成功登陆!"<<endl;
            system("cls");//清屏进入学生模块的欢迎界面

            cout<<"*****"
            *****"<<endl;
            cout<<"*****"
            "<<Teacher::STU[i].GetName()<<"同学,您好!
            *****"<<endl;

            break;
        }
    }
    if(i==Teacher::GetNum())
    {
        cout<<"对不起,您的用户名或密码有误"<<endl;
        system("cls");
        cout<<"请重新登录: "<<endl;
        Login_Student();
    }
}
class Director//Director类主要负责教师登录系统
{
public:
    Director(string name);
    Director();
    void In_Teacher_File();//从默认文件导入教师信息
    void Login_Teacher();//教师登录
private:
    int Count;//教师数量
    Teacher TEA[N];//教师类的对象数组
    string Name;
};
Director::Director(string name)
{
    Name=name;
}
Director::Director()
{

```



```
        Name=" ";
    }
    void Director::In_Teacher_File()
    {
        ifstream in_File;
        string name;
        int i=0,id,code;
        string subject;
        in_File.open("G:\\teacher.txt",ios::in);
        if(in_File==0)
        {
            cout<<"文件打开失败！"<<endl;
            exit(0);
        }
        else
        {
            Count=0;
            while(in_File&&!in_File.eof())
            {
                in_File>>name>>id>>subject>>code;
                TEA[i].From_File(name,id,subject,code);
                Count++;
                i++;
            }
            in_File.close();
        }
    }

    void Director::Login_Teacher()
    {
        int i,id,code;
        In_Teacher_File();
        system("cls");
        cout<<"*****请您先登录系统*****"<<endl;
        cout<<"请输入您的工作号："<<endl;
        cout<<"工作号:";
        cin>>id;
        cout<<"密码:";
        cin>>code;
        for(i=0;i<Count;i++)
        {
            if(TEA[i].GetID()==id&&TEA[i].GetCode()==code)
            {
                cout<<"恭喜您成功登陆!"<<endl;
            }
        }
    }
}
```

```

        system("cls");

        cout<<"*****"
        *****"<<endl;
        cout<<"*****"          "<<TEA[i].GetSubject()<<"
        "<<TEA[i].GetName()<<"老师,您好! *****"<<endl;
            break;
        }
    }
    if(i==Count)
    {
        cout<<"对不起, 您的用户名或密码有误"<<endl;
        system("cls");
        cout<<"请重新登录: "<<endl;
        Login_Teacher();
    }
}

```

//学生成绩管理系统.cpp

```

#include"Class.h"
using namespace std;
void Student_Continue()
{
    int command;Teacher T;
    cout<<"请输入您要执行的操作序号"<<endl;
    cin>>command;
    switch(command)
    {
        case 1:
            T.Search_Student();break;
        case 2:
            T.Sort_Teacher();break;
        case 3:
            exit(0);break;
        default:
            cout<<"对不起, 您的输入有误! "<<endl;
    }
    Student_Continue();
}
void Start_Student()
{
    Manage MA;
}

```

```

    MA.Login_Student();
    cout<<"*****"
*****"<<endl;
    cout<<"*****"
*****"<<endl;
    cout<<"*****"
*****"<<endl;
    cout<<"*****"
*****"<<endl;
    cout<<"*****"
*****"<<endl;
    for(int i=0;i<76;i++)
        cout<<"*";
    cout<<endl;
    Student_Continue();
}
void Continue_Teacher()
{
    Teacher T;
    int command;
    cout<<"请输入您要执行的操作序号"<<endl;
    cin>>command;
    switch(command)
    {
    case 1:
        T.In_Student_File();
        T.Add();
        break;
    case 2:
        int number;
        T.In_Student_File();
        cout<<"您需要删除多少名同学？ "<<"请输入数字"<<endl;
        cin>>number;
        if(number>T.GetNum())
            cout<<"对不起！ 系统中现在只存入了"<<T.GetNum()<<"位学生的信
息及成绩"<<endl;
        else
        {
            for(int i=0;i<number;i++)
                T.Delete();
        }
        break;
    case 3:
        cout<<"***** 【注意不允许批量修改】 *****"<<endl;

```

欢迎使用学生成绩管理系统

菜 单

1.查询个人成绩

2.学生成绩排序

3.退出 系统

```

        T.Change();break;
    case 4:
        T.Search_Teacher();break;
    case 5:
        T.Sort_Teacher();break;
    case 6:
        T.In_File_Data();break;
    case 7:
        T.All_Student_Display();break;
    case 8:
        cout<<"平均成绩为:"<<T.GetAverage()<<endl; break;
    case 9:
        exit(0);break;
    default:cout<<"对不起，您的输入有误！"<<endl;
    }
    Continue_Teacher();
}
void Start_Teacher()
{
    Director A;
    A.Login_Teacher();
    cout<<"*****
*****"<<endl;
    cout<<"*****
*****"<<endl;
    cout<<"*****
*****"<<endl;
    cout<<"*****
*****"<<endl;
    cout<<"*****
*****"<<endl;
    cout<<"*****
*****"<<endl;
    cout<<"*****
*****"<<endl;
    cout<<"*****
*****"<<endl;
    cout<<"*****
*****"<<endl;
    cout<<"*****
*****"<<endl;
    cout<<"*****
*****"<<endl;
    for(int i=0;i<76;i++)

```

欢迎使用学生成绩管理系统

菜 单

- 1.添加学生成绩
- 2.删除学生成绩
- 3.修改学生成绩
- 4.查询学生成绩
- 5.学生成绩排序
- 6.导入外部文件
- 7.显示所有成绩
- 8.显示平均成绩
- 9.退出 系统

```

        cout<<"*";
    cout<<endl;
    Continue_Teacher();
}
int main()
{
    system("color 0E");
    char c;
    for(int i=0;i<76;i++)
        cout<<"*";
    cout<<endl;
    cout<<"*****"
*****"<<endl;
    cout<<"*****"
*****"<<endl;

    for(int i=0;i<76;i++)
        cout<<"*";
    cout<<endl;
    cout<<"***** 请问您的身份是学生，还是教师？学生请输入S，教师
请输入T *****"<<endl;
    cout<<"请输入： "<<endl;
    cin>>c;
    if(c=='S' || c=='s')
    {
        system("cls");
        Start_Student();
    }
    else if(c=='T' || c=='t')
    {
        system("cls");
        Start_Teacher();
    }
    else
    {
        system("cls");
        cout<<"对不起输入有误！"<<endl;
        main();
    }
}

```

学生成绩管理系统

欢迎您的使用

数据文件（测试用）

G:\\student.txt

张飞 2014011216 94 92 85 88 1321232
张三 2014011256 85 69 84 56 3121313
李四 2014011285 25 65 26 86 9523541
马武 2014011284 33 99 24 65 2654221
徐帅 2014011243 26 95 64 22 2358621
刘爽 2014011211 85 69 41 58 2358634
姜鹤 2014011233 25 96 41 28 3524681
何柳 2014011285 26 14 28 32 2014011
刘伟 2014012553 58 93 72 58 2332632
马赫 2014011259 66 33 69 98 1564646
王刚 2014011258 75 89 64 85 8456325
龙俊亨 2014011245 85 74 85 96 5311112
项少玲 2014011162 65 36 84 64 2153630
闫平 2014011265 98 74 58 28 3226626
汪廷栋 2014013324 58 63 84 95 2356238
师素芳 2014025533 45 85 69 12 3665965
安明玉 2014011215 63 89 84 75 5623335

G:\\teacher.txt

Mike 201401121 微积分 235862
Sam 201401125 线性代数 541852
谢峰 201401121 离散数学 132326
Sarah 2014011523 英语 3331332

二、图书管理系统

目 录

1. 系统需求分析	56
2. 总体设计	59
3. 详细设计	62
4. 系统调试	67
5. 使用说明书	68
6. 总结	83
附录：源程序清单	83
数据文件（测试用）	111

1. 系统需求分析

书籍是记录人类知识的载体，读书是人类获取知识的重要途径之一。随着生活条件的逐步提高，人们越来越把读书作为重要的生活部分。大到全国各地的中小学和高校，小到各个公司甚至是一个小乡村的文化站，都有大量的图书信息亟待处理。而这些书籍及其借阅、归还的信息由人力处理太过繁琐和复杂，而且很容易出错，造成书籍的丢失。于是，为了减轻人的工作负担，目前迫切地需要用计算机编程设计出一个图书管理系统。

（1）系统设计目的与意义

设计图书管理系统的目的就在于方便图书管理员和使用图书的读者（用户）对图书进行各种操作。比如，管理员需要更新图书的信息，读者的借阅和还书也要有图书管理系统处理。这样就能够将大量的原本属于图书管理员的工作交由图书管理系统来处理。计算机的存储量大，运算处理的速度和精确度都远远高于人。

通过图书管理系统，可以对图书的各项处理，并且提高了处理的效率和水平。因而，处理能力提高了，图书管理系统对提高图书馆的馆藏也会由此增加。这使得用户能够接触到更多更广的书籍，对知识的普及和公众文化素质的提高也有积极的意义。

（2）系统功能需求：

功能需求

图书管理系统对不同的使用者有不同的功能限制。

③ 对于读者（用户）而言：

每位读者的全部个人信息及登录密码都存储在系统数据文件中，进入系统前需要登录。当通过登录获得权限之后，系统就可以实现针对读者的功能。功能主要包括：查询系统中的图书、借阅书籍、归还书籍以及查看目前自己的信息及借阅情况等。

根据读者自己输入的不同选项，系统将进入不同的功能模块，不同的功能模块又细分成其他小的功能模块。

④ 对于图书管理系统的管理员，同样需要登录（输入管理员密码），通过登录获得权限之后，可以进行对图书管理系统中存入的书籍信息及读者（用户）信息进行删改、查询等许多操作。

根据不同的需求，系统需要实现以下功能：

一、针对书籍的功能：

- 1) 考虑到图书数量很大，因而直接找到实体书籍很不方便。因此，将所有图书的信息录入系统，并且设计查询功能，只需在系统中输入一定量的已知信息，即可找到相应书的全部信息。根据已知信息的不同，查询功能模块下又分为按照书名、ISBN 号、出版社等许多不同的查询方式，极大地方便对图书的了解和管理。
- 2) 考虑到图书馆的书籍会不断更新，有新书不断补充，也有一些旧书不能继续向外借阅，甚至有一些书籍因为意外而丢失。因此，对书籍的管理还需要包括增加和删除书籍，确保图书管理系统中

的图书及时地更新。

3) 为了方便管理员更好地掌握所有图书的情况，还设计了“显示全部图书信息”的功能。

4) 同时，考虑到日后图书管理系统会被更先进的管理系统所替代，因此设计了“清除全部图书信息”的功能，确保信息不被泄露。

二、针对读者（用户）的功能：

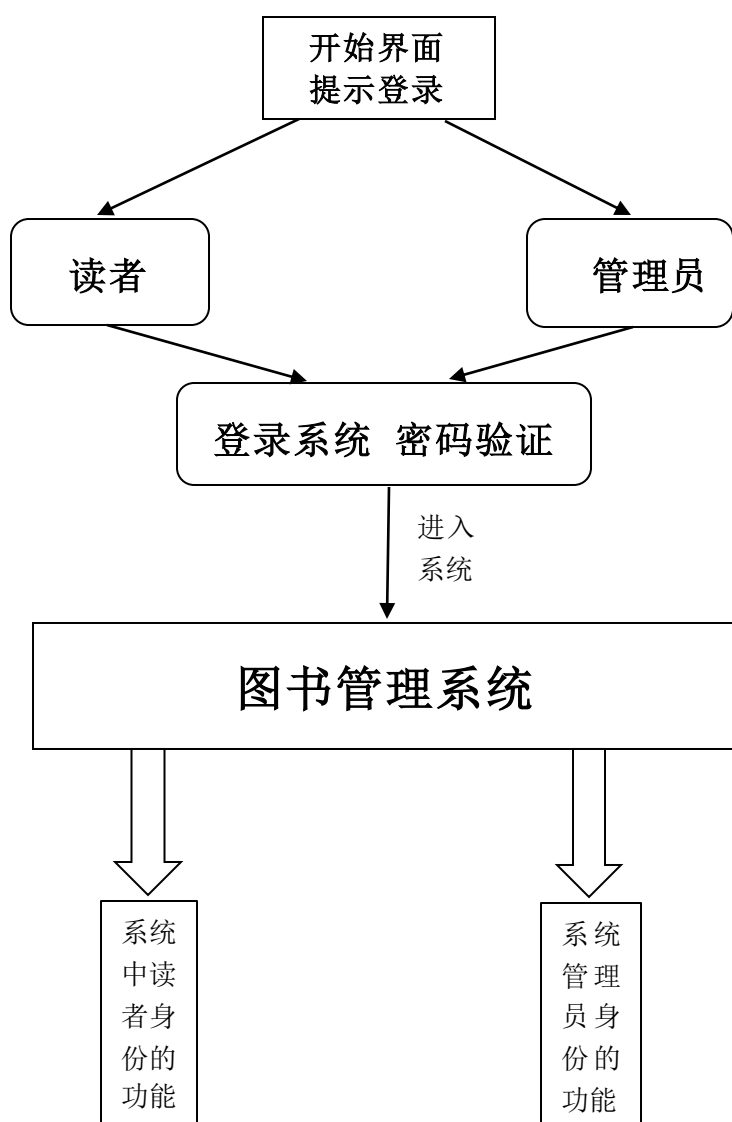
1) 考虑到图书管理系统的使用者主要是借阅和还书的读者，因而需要对读者的基本信息进行录入。因而，为了更方便地了解读者的情况，需要实现查询读者的功能。

2) 读者也会有流动变化，因而，“添加读者”和“删除读者”也是两个必不可少的功能，确保系统中存入的读者信息是准确的。

3) 为了方便管理员掌握所有读者（用户）的信息以及借阅情况，还设计了“显示全部读者信息”的功能。

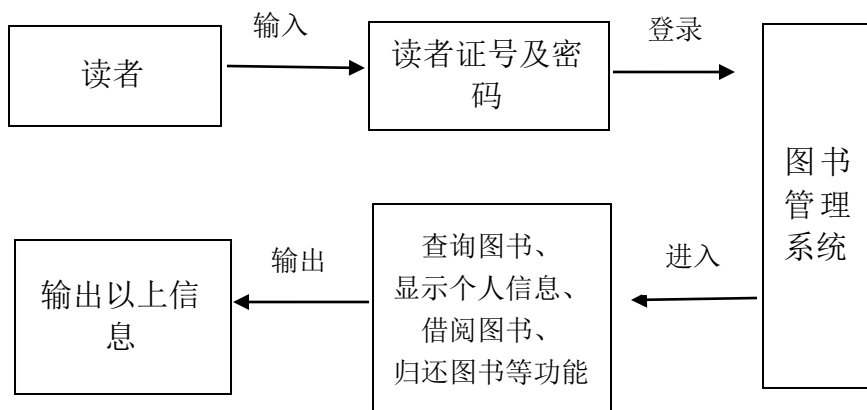
4) 同时，考虑到日后图书管理系统可能被更先进的管理系统所替代，因而设计了“清除全部读者信息”的功能，防止读者的隐私被泄露。

系统流程图

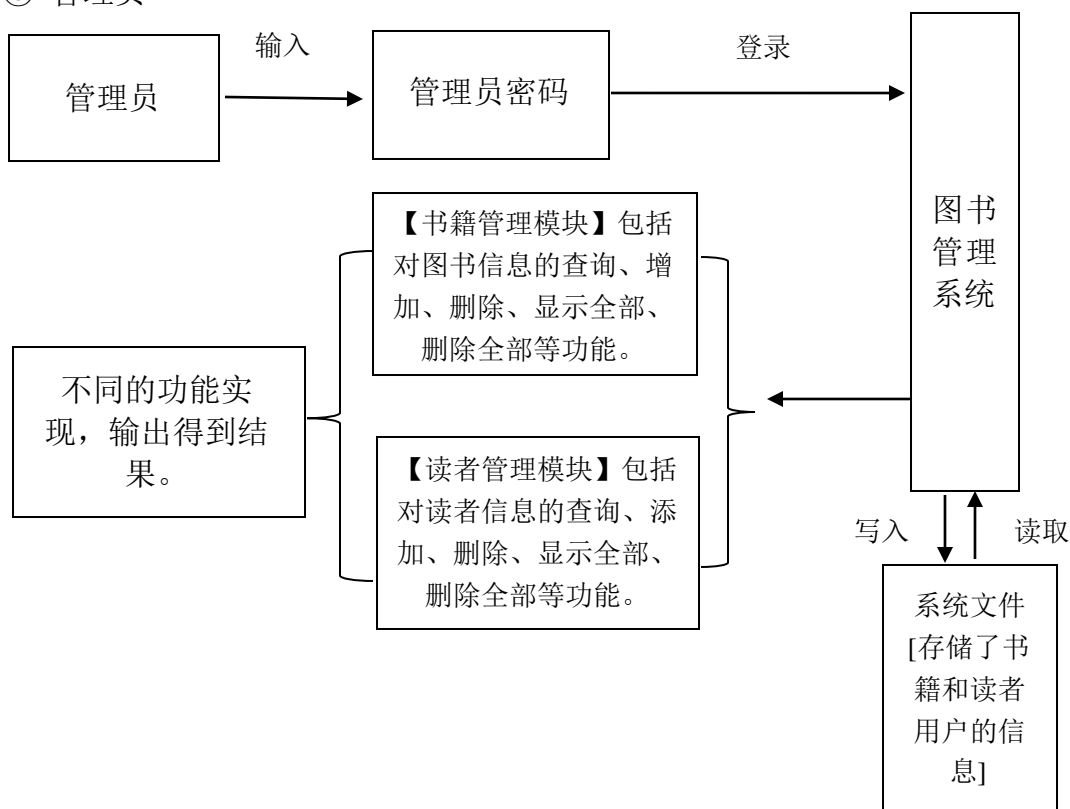


图书管理系统根据不同使用者，实现的功能也不同，下面的流程图分别针对读者身份和管理员身份。

④ 对读者的功能



⑤ 管理员



(3) 输入与输出要求:

进入图书管理系统之前都要登录。读者需要先输入读者证号及密码，通过验证之后方可进入系统，以此保护系统和读者个人信息的安全。管理员唯一，因而只需输入管理员密码进行登录。系统执行的操作及功能由使用者输入的选项决定，根据提示输入不同的选项，即可实现读者和管理员两个不同身份的不同功能。

在读者和管理员这两大不同的功能板块中：

①读者能够以不同方式进行查询图书信息，输出内容为图书的书名、作者、出版社、所属学科、ISBN 号、价格等诸多条目信息；也可以输出个人账户下的信息及借阅图书情况。读者也可以选择输入关于书籍的一些信息（比如 ISBN 号、出版社等信息）对书籍进行借阅和归还操作。

②对于管理员，在管理书籍的模块，也要输入选项实现不同功能，除了查询功能，管理员也可以选择增加和删除书籍信息；其中，增加信息需要将图书的所有信息按照提示依次输入。同样地，对于读者管理，也有类似的查询、增加和删除功能。

③在输出图书或读者信息时，对输出格式有严格的限制，从而使得输出界面更加整齐。可读性更高。

2. 系统总体设计

系统功能分析：

5) 系统进入界面：

非常人性化地提示系统使用者（读者或管理员）进行选择身份，不同的身份进入不同的登陆界面。

6) 登录界面：

读者输入读者证号和密码，系统在确认两者匹配无误之后，显示“登录成功”，进入“读者”功能模块。管理员直接输入管理员密码，密码正确，即显示“登录成功”，进入“管理员”功能模块。

【注意】

读者选用读者证号登录是因为姓名可能相同，但读者证号是一一对应的。因此当读者数目较多，系统规模比较大的情况下不宜用“姓名”登录。

7) 读者板块功能分析：

读者在图书管理系统中实现的功能主要是查询并借阅和归还书籍，以及显示目前自己的借阅情况。实现的功能不多，只实现读者个体的功能。

c) 图书查询：

图书查询分为两大查询方式，即精确查询和模糊查询。精确查询指的是输入 ISBN 号进行查找书籍；而模糊查询又细分为按照作者、出版社、所属学科等不同的查询方式。最终目的是帮助读者在众多书籍中能够尽快找到所需图书。

d) 显示个人信息及借阅情况：

当读者需要了解自己目前借阅了哪些书籍时，可以选择该功能，系统会输出该读者的信息及借阅的书籍详细信息，帮助读者及时还书。

c) 借阅书籍：

输入所要借阅的书籍的 ISBN 号，系统在判断确认读者目前借阅书籍数目小于最大借阅量之后，会将该书借出。同时系统内书籍管理会将已经借出的书籍记录为状态“0”，表示应急借出，不能再外借。

d) 归还书籍：

同样输入需要归还的书籍，系统在判断确认该书在读者已经借阅的图书内之后，将书从读者个人借阅的书里删除，并反馈到系统内将给图书的状态改为“0”，即该书在架上，能继续借出。

8) 管理员板块功能分析:

管理员实现的功能模块分为“图书管理”和“读者管理”两个部分。每个部分根据不同的需要实现更小的功能。下面是细化的功能分析:

一、图书管理功能模块

i) 图书查询:

同读者的图书查询功能一样,图书查询分为两大查询方式,即精确查询和模糊查询。精确查询指的是输入 ISBN 号进行查找书籍;而模糊查询又细分为按照作者、出版社、所属学科等不同的查询方式。最终目的是帮助管理员在众多书籍中能够尽快找到所需图书,以方便图书维护。

j) 添加图书:

考虑到图书的流动性(新书入库),需要系统进行更新。此情况下管理员需要根据系统界面的提示,按步骤录入书籍的各条目信息,最终将该书录入系统中。

k) 删除图书:

考虑到图书的流动性(丢失或过于陈旧不宜外借),需要系统进行更新。此情况下管理员需要根据系统界面的提示,输入图书的信息,并将图书从能够借阅的图书中删除。

l) 显示全部图书:

一次性输出所有图书的所有信息,这些图书信息存储在系统数据文件之中,将数据导入系统后,为了方便管理者对所有图书进行维护,因而设计了这一功能;

m) 删除全部图书:

此功能会将系统内已经导入的所有图书的信息全部清除,防止相关信息泄露。

二、读者管理功能模块

功能实现时,与对书籍的管理类似,只是将对象改为读者。主要功能也是查询、添加、删除、显示全部和全部清除等。

n) 读者查询:

读者查询也分为两大查询方式,即精确查询和模糊查询。精确查询指的是输入读者证号进行查找书籍;而模糊查询又细分为按照姓名、性别等不同的查询方式。最终目的是帮助管理员在众多的读者用户能够尽快找到目标读者,以方便系统维护。

o) 添加读者:

考虑到读者的流动性(新用户),需要系统具有更新的功能。此情况下管理员需要根据系统界面的提示,按步骤录入读者的信息,最终将读者存于系统之中。

p) 删除读者:

考虑到图书的流动性(注销读者证号),需要系统进行更新。此情况下管理员需要根据系统界面的提示,输入读者证号,并将该读者从系统中删除。

q) 显示全部读者:

一次性输出所有读者的所有信息及借阅情况，读者个人信息存储在系统数据文件之中，将数据导入系统后，为了方便管理者对所有读者进行维护，因而设计了这一功能。

r) 删除全部读者：

此功能会将系统内已经导入的所有读者（用户）的信息全部清除，防止个人隐私被泄露。

系统功能模块划分与设计：

首先，点击“图书管理系统”的应用程序，将进入提示界面，输入身份信息，进入相应的登录模块。登录成功之后，便进入了读者/管理员不同的功能模块。

读者功能模块主要包括了“图书查询”“显示个人信息”“借阅图书”“归还图书”等诸多功能。

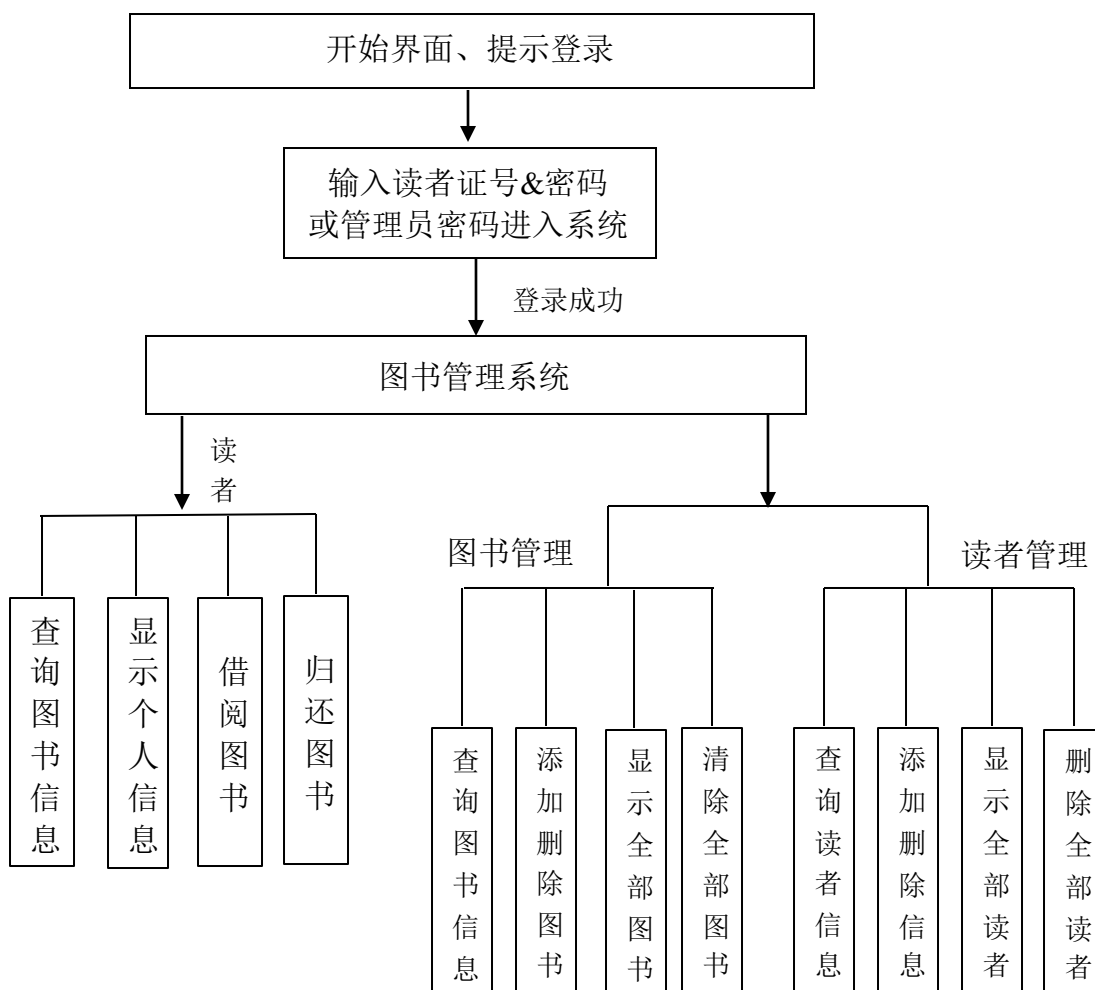
管理员功能模块所包括的功能较多，主要包括：

【图书管理】：查询、添加、删除、显示全部和删除全部书籍。

【读者管理】：查询、添加、删除、显示全部和删除全部读者。

因而，本图书管理系统功能模块划分合理，读者和管理员的模块区别明显，并且能够实现较多的功能。

系统功能模块图：



3. 系统详细设计

① 数据文件:

本系统数据是计算机上存储的文件，但注意有以下要求：

- 5) 考虑到数据文件为二进制文件时，无法从外部修改文件，使得操作过于复杂，因此选用文本文件作为系统的数据文件；
- 6) 为了确保录入图书和读者信息时能够正确录入，需保证存储的图书信息按照“书名”“作者”“出版社”“所属学科”“ISBN 号”“价格”“出版年份”“出版月份”的顺序存入文本文件；而存储的读者信息则需要按照“姓名”“读者证号”“性别”“密码”的顺序存储。
- 7) 为了方便自己测试数据，本系统有默认数据文件，图书的详细信息存储在默认文本文件“E: \\book.txt”中，所有读者的信息存储在默认文件“E:\\readertxt”中。
- 4) 系统数据文件写在程序内部，如果需要修改数据文件的路径，需要在程序内进行修改。

② 类层次图：

图书管理系统共使用了 6 个类，其中主要实现功能的是 **Manage_Book** 和 **Manage_Reader** 类。顾名思义，即对图书和读者的管理。

现对 6 个类进行分析：

7) Base 类:

系统中的抽象类，包含一个纯虚函数，作为整个系统的基础。不能具体化，但是派生出 **Book** 类和 **Reader** 两个类。但不能实现具体的功能。

8) Book 类:

顾名思义，表示图书管理系统中需要的图书，具体在系统中体现为书籍的各条信息，存储在文件中。**Book** 类的每一个对象代表不同的图书。图书是整个系统的核心。每位读者都可以借阅书籍，因而 **Book** 类的对象数组作为 **Reader** 类的成员。同时 **Manage_Book** 类也是以 **Book** 类更多的对象作为成员。**Manage_Reader** 类对 **Reader** 类的管理实际上也是间接地对 **Book** 类执行一些函数。因此，**Book** 类是整个系统的基础。

9) Manage_Book 类:

Manage_Book 负责实现有关 **Book** 类的各项功能。包括从所有书籍中根据已知信息查询书籍、增加书籍信息、删除图书信息以及显示图书信息。在系统功能的实现中起到了重要的作用。

10) Reader 类:

Reader 类则表示图书管理系统中的读者，具体体现为读者的个人信息及借阅图书情况。**Reader** 的成员包括 **Book** 类的对象数组，同时

Reader 类的对象数组又作为 Manage_Reader 类的私有成员。起到了一定的纽带作用。

11) Manage_Reader 类:

Manage_Reader 类，主要功能是对读者的管理。主要功能有：根据读者的部分信息查询读者、增加或删除读者、显示全部读者信息以及清除所有读者信息等。使得管理员更加方便地掌握图书借阅和归还情况。

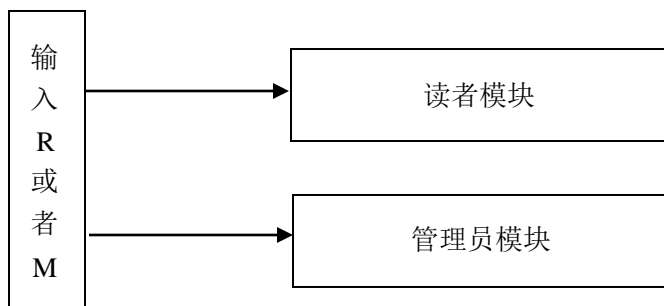
【注意】Manage_Book 类和 Manage_Reader 类分别实现图书管理系统对图书和读者的管理，这也与系统的两大功能板块相对应，使得系统的功能更加完善。并且两者共同派生出 Manage 类。

12) Manage 类

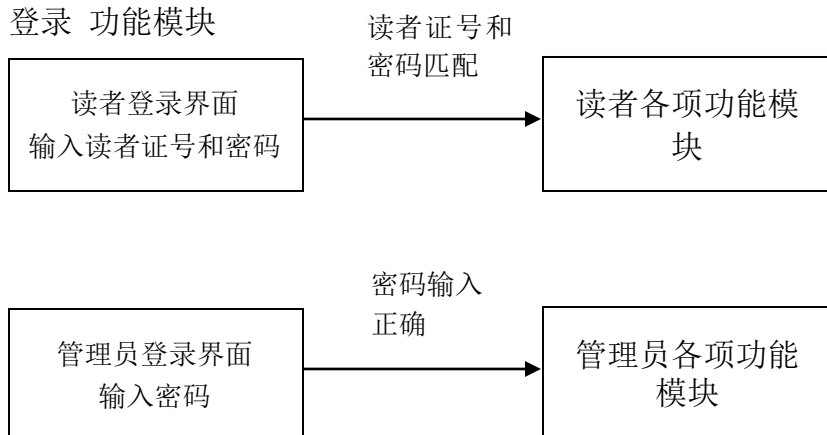
Manage 实现的功能主要是读者和管理员的登录，因为 Manage 多重继承了 Manage_Book 和 Manage_Reader，因而可以访问到所有的 Book 和 Reader 对象。读者登录时读者证号和密码必须完全匹配才能登录成功；同样地，管理员也需要输入正确密码才能进入各功能模块。

界面设计与各功能模块实现:

5) 开始界面、提示登录 功能模块

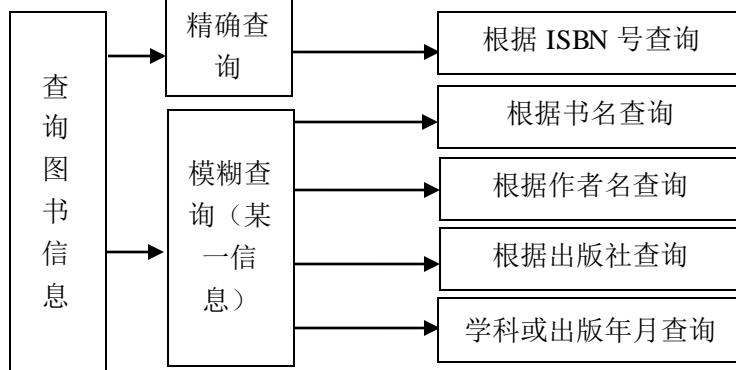


6) 登录 功能模块

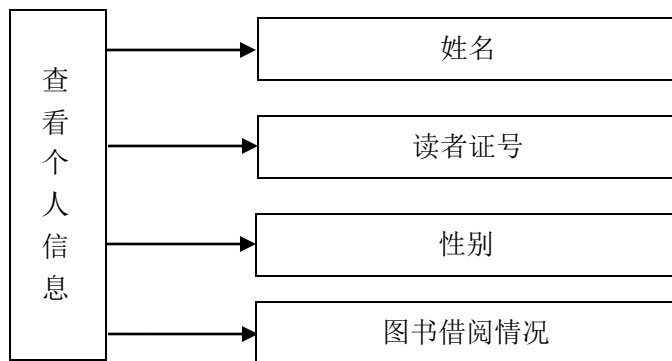


7) 读者各项功能模块

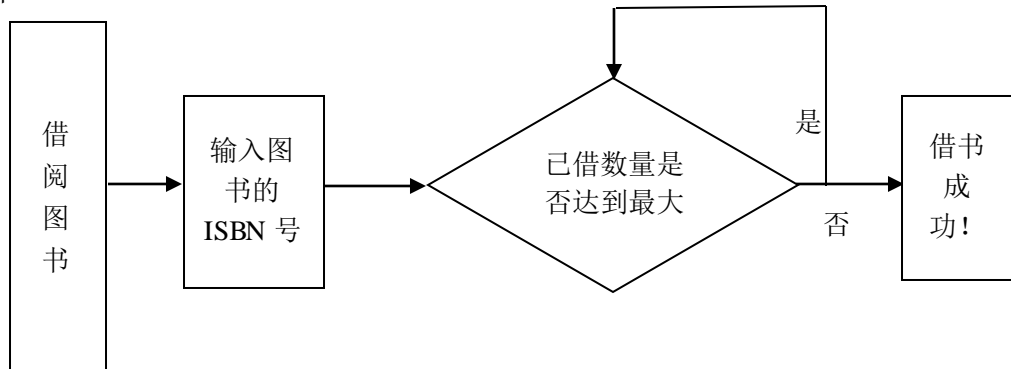
a) 查询图书信息功能模块



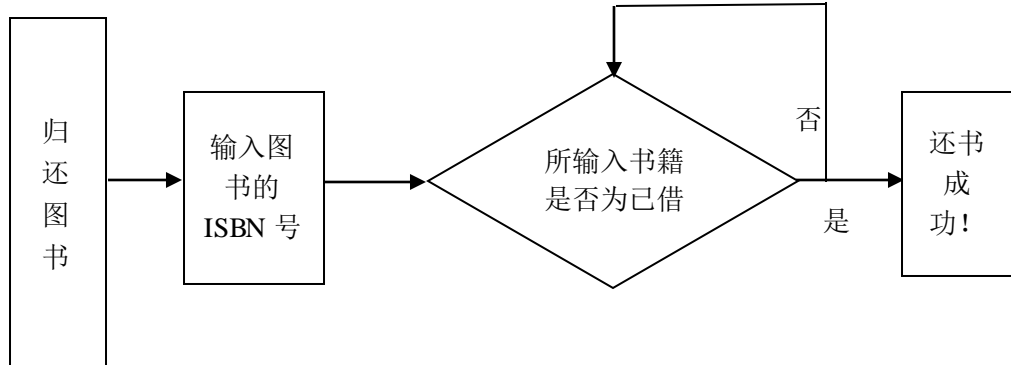
b) 查看个人信息 功能模块



c) 借阅图书



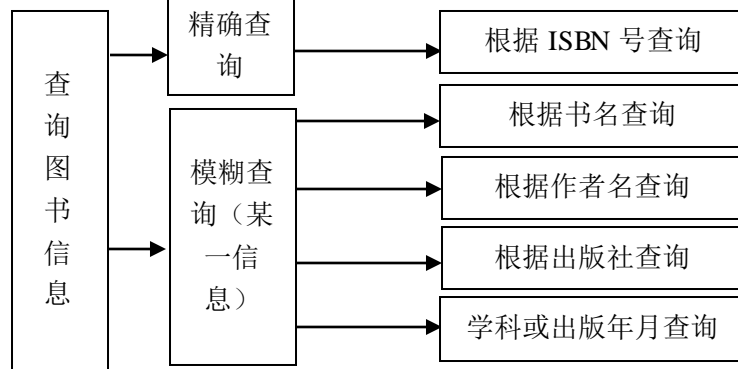
d) 归还图书



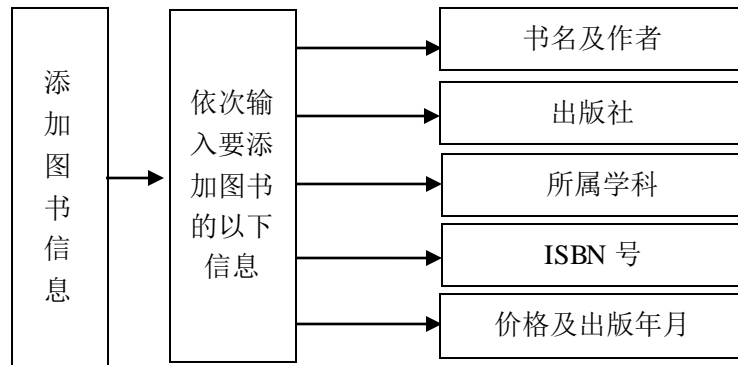
4) 管理员各项功能模块

一、管理图书 功能模块

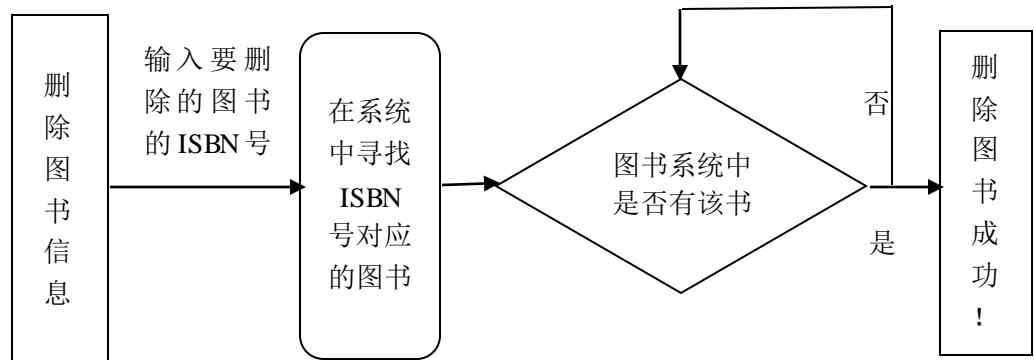
a) 查询图书信息功能模块



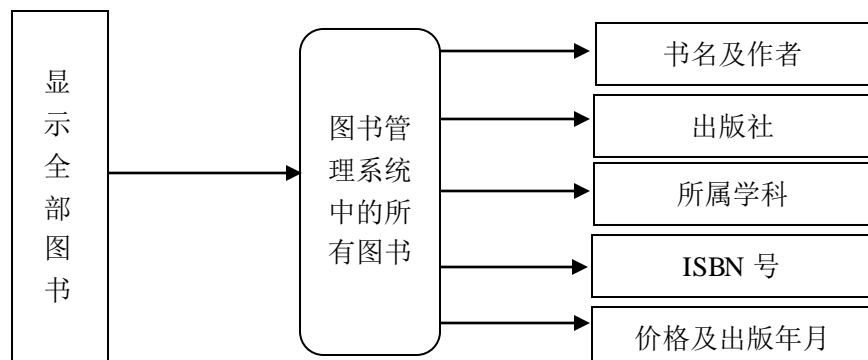
b) 添加图书 功能模块



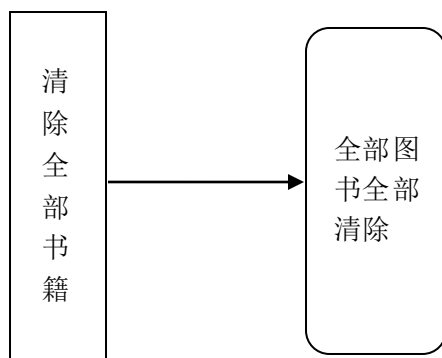
c) 删除图书 功能模块



d) 显示全部图书 功能模块

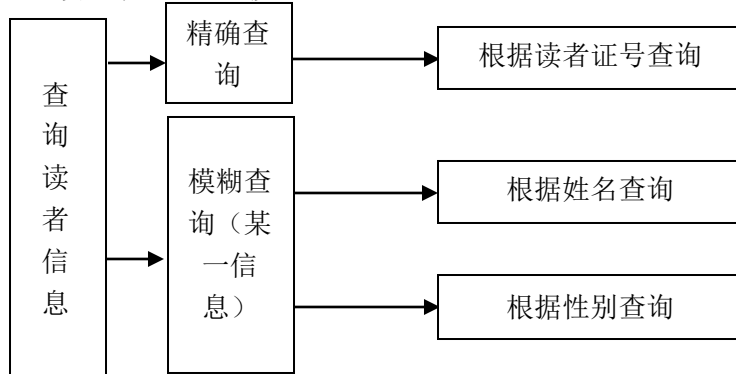


e) 清除全部书籍 功能模块

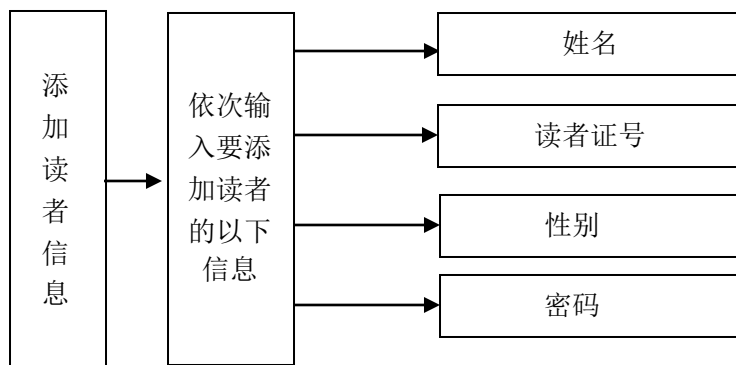


二、管理读者 功能模块

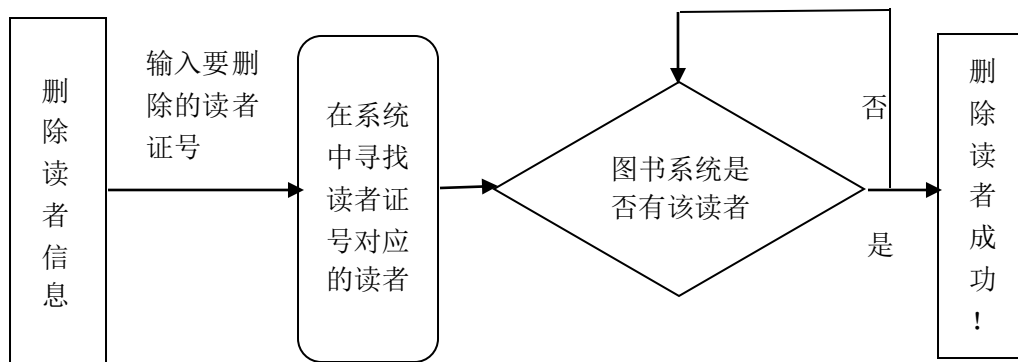
a) 查询读者信息功能模块



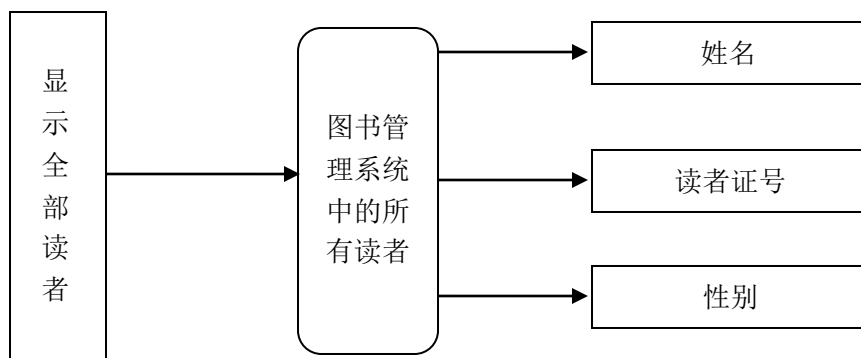
b) 添加读者 功能模块



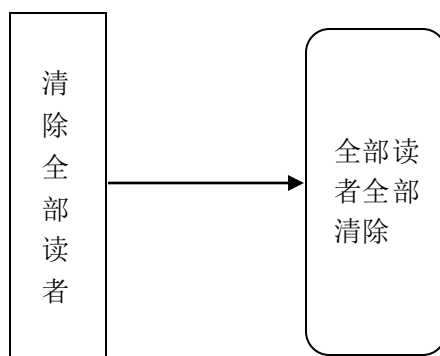
c) 删除读者 功能模块



d) 显示全部读者 功能模块



e) 清除全部读者 功能模块



4. 系统调试

编写完所有程序之后，进行系统调试。Visual Studio 2008 显示共有 50 多条错误，这与前一个学生成绩管理系统相比显著减少。Debug 之后，发现其实很多 bug 是编程时输入的错误，剩下的一些是需要认真思考和修改的。下面是在这次编程中出现的一些问题。

① 关于字符串。

基于之前对学生成绩管理系统的设计的经验，这次编程决定直接使用 C++ 特有的 `string` 类（字符串类）来表示一些变量。比如 `Book` 类和 `Reader` 类的很多私有（保护）成员都使用的是 `string` 类，极大地简化了编程中关于字符串的操作。

② 关于输入密码登录。

和学生成绩管理系统一样，图书管理系统如果不设置密码完全不能保证信息和数据的安全。于是增加了读者和管理员的登录功能。有了之前设计学生成绩管理系统登录模块的经验，实现这一功能不是很难。

③ 关于操作符重载。

在操作符重载时，我直接借用了学生成绩管理系统中的重载，结果调试时未能通过，查找书籍和资料之后，将操作符重载函数设置为友元函数之后，调试才通过。

④ 关于系统功能的完善。

实现了基本功能之后，感觉作为图书管理系统，功能性还是不够强，于是又增加了“显示全部图书（读者）”“删除全部图书（读者）”等功能，对图书管理和读者管理具有重要的意义。

以上是在系统调试过程中遇到的主要问题，总体来看，调试过程比之前顺利了许多，也使我 debug 的能力有一定的提高，感觉收获颇丰。

5. 使用说明书

图书管理系统

使用说明书

目录

第一部分 系统说明	
1. 背景	69
2. 系统概况	69
第二部分 系统使用	
1. 进入界面	69
2. 登录系统	
①读者登录系统	69
②管理员登录系统	70
3. 读者功能模块	
①查询图书信息	71
②查看个人信息	74
③借阅图书	74
④归还图书	74
⑤退出系统	74
4. 管理员功能模块	
A) 图书管理功能模块	
①查询图书信息	75
②添加图书信息	78
③删除图书信息	78
④显示全部图书	79
⑤清除全部图书	79
B) 读者管理功能模块	
①查询读者信息	80
②添加读者信息	81
③删除读者信息	81
④显示全部读者	82
⑤清除全部读者	82
第三部分	
注意事项	83

2014011216 刘前

2015.7.31

第一部分 系统说明

1. 背景

随着生活条件的逐步提高，人们越来越把读书作为重要的生活部分。大到全国各地的中小学和高校，小到各个公司甚至是一个小乡村的文化站，都有大量的图书信息亟待处理。而这些书籍及其借阅、归还的信息由人力处理太过繁琐和复杂，而且很容易出错，造成书籍的丢失。于是，为了减轻人的工作负担，目前迫切地需要用计算机编程设计出一个图书管理系统。

2. 系统概况

图书管理系统的设计目的就在于方便图书管理员和使用图书的读者（用户）对图书进行各种操作。比如，管理员需要更新图书的信息，读者的借阅和还书也要有图书管理系统处理。这样就能够将大量的原本属于图书管理员的工作交由图书管理系统来处理。计算机的存储量大，运算处理的速度和精确度都远远高于人。图书管理系统作为工具，可以对图书的各项处理，并且提高了处理的效率和水平。因而，处理能力提高了，图书管理系统对提高图书馆的馆藏也会由此增加。这使得用户能够接触到更多更广的书籍，对知识的普及和公众文化素质的提高也有积极的意义。

第二部分 系统使用

1. 进入界面：

点击“图书管理系统.exe”文件或者运行程序之后，即可进入下面的界面：



根据个人身份，读者登录时输入“R”或“r”（表示 Reader），管理员登录时输入“M”或“m”（表示 Manager）。点击回车键即可进入登录界面。

2. 登录系统

①读者：

输入“R”后，进入学生登录界面（如下图）：

```

*****
*****欢迎使用图书管理系统*****
***** 读者登录请输入“R”，管理员登录请输入“M” *****
*****
****请输入: R
*****读者登录*****
请输入用户ID及密码:
****ID:

```

此时需要输入读者证号和密码。以默认数据文件“E:\\reader.txt”中的张三为例。张三读者证号为 2014011216，密码为 235862。

```

*****
*****欢迎使用图书管理系统*****
***** 读者登录请输入“R”，管理员登录请输入“M” *****
*****
****请输入: R
*****读者登录*****
请输入用户ID及密码:
****ID:2014011216
**密码: 235862

```

回车键确认之后，登录成功！即可进入图书管理系统欢迎界面：

```

***** 张三先生，您好！欢迎使用图书管理系统！ *****
***** 欢迎使用图书管理系统 *****
***** 菜单 *****
***** 1. 查询书籍 *****
***** 2. 个人信息 *****
***** 3. 借阅书籍 *****
***** 4. 归还书籍 *****
***** 5. 退出系统 *****
****请输入您的选项:

```

【注意】如果读者证号和密码不匹配，系统会显示：

```

对不起，ID或密码有误！请重新登录：
*****读者登录*****
请输入用户ID及密码:
****ID: _

```

此时需确认读者证号和密码，重新登录。

②管理员登录:

输入“M”/“m”之后，进入管理员登录系统。只需要正确输入管理员密码即可。

```

*****
***** 欢迎使用图书管理系统 *****
***** 读者登录请输入“R”，管理员登录请输入“M” *****
*****
****请输入: M
请输入管理员密码:
LIUQIAN_

```

```

管理员您好! 欢迎使用图书管理系统!
*****
***** 菜单 *****
***** 1.图书管理 *****
***** 2.读者管理 *****
*****
****请输入选项:

```

登录成功之后，即可进入管理员功能模块。

3. 读者功能模块

①查询图书信息:

在提示“请输入您的选项”中输入 1，即可进入“查询书籍”模块。之后即可得到:

```

***** 张三先生, 您好! 欢迎使用图书管理系统! *****
***** 欢迎使用图书管理系统 *****
*****
***** 菜单 *****
***** 1. 查询书籍 *****
***** 2. 个人信息 *****
***** 3. 借阅书籍 *****
***** 4. 归还书籍 *****
***** 5. 退出系统 *****
****请输入您的选项: 1

```

```

***** 张三先生, 您好! 欢迎使用图书管理系统! *****
***** 欢迎使用图书管理系统 *****
*****
***** 菜单 *****
***** 1. 查询书籍 *****
***** 2. 个人信息 *****
***** 3. 借阅书籍 *****
***** 4. 归还书籍 *****
***** 5. 退出系统 *****
****请输入您的选项: 1
*****
***** 查找书目 *****
***** 1. 精确查询 *****
***** 2. 模糊查询 *****
****请输入您的查询方式:

```

根据提示，输入需要的查询方式。

输入 1 时，为“精确查询”，输入图书的 ISBN 号即可。

```

***** 张三先生, 您好! 欢迎使用图书管理系统! *****
***** 欢迎使用图书管理系统 *****
***** 菜单 *****
***** 1. 查询书籍 *****
***** 2. 个人信息 *****
***** 3. 借阅书籍 *****
***** 4. 归还书籍 *****
***** 5. 退出系统 *****
***请输入您的选项: 1
***** 查找书目 *****
***** 1. 精确查询 *****
***** 2. 模糊查询 *****
***请输入您的查询方式: 1
***** 精确查询 *****
***请输入您要查询的书的ISBN号
978-7-5086-4937-5
***查询结果:
=====
书名/题目          作者          出版社          学科          ISBN号          价格          出版日期
-----
世界第一好懂的哲学课  [日]小川仁志  中信出版社      哲学          978-7-5086-4937-5  35.00元      2015年2 月
=====
继续查询请输入Y, 退出输入N

```

输入 2 时，进入“模糊查询”，再次选择不同的选项：

```

***** 张三先生, 您好! 欢迎使用图书管理系统! *****
***** 欢迎使用图书管理系统 *****
***** 菜单 *****
***** 1. 查询书籍 *****
***** 2. 个人信息 *****
***** 3. 借阅书籍 *****
***** 4. 归还书籍 *****
***** 5. 退出系统 *****
***请输入您的选项: 1
***** 查找书目 *****
***** 1. 精确查询 *****
***** 2. 模糊查询 *****
***请输入您的查询方式: 2
***** 模糊查询 *****
***** 1. 根据书名查询 *****
***** 2. 根据作者查询 *****
***** 3. 根据出版社查询 *****
***** 4. 根据学科查询 *****
***** 5. 根据出版年月查询 *****
***请输入您的选项: 1

```

可以看出，模糊查询板块又细分为 5 个功能模块：

以下是五个模块的测试结果：

①根据书名查询：

```

***** 查找书目 *****
***** 1. 精确查询 *****
***** 2. 模糊查询 *****
***请输入您的查询方式: 2
***** 模糊查询 *****
***** 1. 根据书名查询 *****
***** 2. 根据作者查询 *****
***** 3. 根据出版社查询 *****
***** 4. 根据学科查询 *****
***** 5. 根据出版年月查询 *****
***请输入您的选项: 1
***请输入书名:
政治哲学
***查询结果:
=====
未查到相关书目
继续查询请输入Y, 退出输入N

```


②根据作者查询:

```

***** 查找书目 *****
***** 1.精确查询 *****
***** 2.模糊查询 *****
***** 请输入您的查询方式:2 *****
***** 模糊查询 *****
***** 1.根据书名查询 *****
***** 2.根据作者查询 *****
***** 3.根据出版社查询 *****
***** 4.根据学科查询 *****
***** 5.根据出版年月查询 *****
***** 请输入您的选项: 2 *****
***** 请输入作者名: *****
***** 江春莲 *****
***** 查询结果: *****

```

书名/题目	作者	出版社	学科	ISBN号	价格	出版日期
数学教育初论	江春莲	科学出版社	教育	978-7-03-041790-9	128.00元	2015年1月

继续查询请输入Y, 退出输入N

③根据出版社查询:

```

***** 查找书目 *****
***** 1.精确查询 *****
***** 2.模糊查询 *****
***** 请输入您的查询方式:2 *****
***** 模糊查询 *****
***** 1.根据书名查询 *****
***** 2.根据作者查询 *****
***** 3.根据出版社查询 *****
***** 4.根据学科查询 *****
***** 5.根据出版年月查询 *****
***** 请输入您的选项: 3 *****
***** 请输入准确的出版社名称: *****
***** 清华大学出版社 *****
***** 查询结果: *****

```

书名/题目	作者	出版社	学科	ISBN号	价格	出版日期
过程控制技术及其应用	徐湘元	清华大学出版社	技术	978-7-322-37268-4	39.00元	2013年12月

继续查询请输入Y, 退出输入N

④根据所属学科查询:

```

***** 查找书目 *****
***** 1.精确查询 *****
***** 2.模糊查询 *****
***** 请输入您的查询方式:2 *****
***** 模糊查询 *****
***** 1.根据书名查询 *****
***** 2.根据作者查询 *****
***** 3.根据出版社查询 *****
***** 4.根据学科查询 *****
***** 5.根据出版年月查询 *****
***** 请输入您的选项: 4 *****
***** 【注意】当前分类有: 哲学、社会、自然、教育、艺术、技术, 请输入要查询的类别: *****
***** 艺术 *****
***** 查询结果: *****

```

书名/题目	作者	出版社	学科	ISBN号	价格	出版日期
环境艺术设计概论	王烨	中国电力出版社	艺术	978-7-5123-6532-2	59.00元	2014年12月
思考的艺术	[英]Buzan&Tony	化学工业出版社	艺术	978-7-122-22298-5	45.00元	2014年11月
中国树木盆景艺术	黄映泉	安徽科学技术出版社	艺术	978-7-5337-6473-9	85.00元	2014年12月

继续查询请输入Y, 退出输入N

⑤根据出版日期查询:

```

***** 查找书目 *****
***** 1.精确查询 *****
***** 2.模糊查询 *****
***** 请输入您的查询方式:2 *****
***** 模糊查询 *****
***** 1.根据书名查询 *****
***** 2.根据作者查询 *****
***** 3.根据出版社查询 *****
***** 4.根据学科查询 *****
***** 5.根据出版年月查询 *****
***** 请输入您的选项: 5 *****
***** 请输入出版年份: 2015 *****
***** 请输入出版月份: 2 *****
***** 查询结果: *****

```

书名/题目	作者	出版社	学科	ISBN号	价格	出版日期
世界第一好懂哲学课	[日]小川仁志	中信出版社	哲学	978-7-5086-4937-5	35.00元	2015年2月
政治哲学	[美]史密	北京联合出版公司	哲学	978-7-5502-3895-5	39.80元	2015年2月
宋代经学哲学研究	王心竹	上海科技出版社	哲学	978-7-5439-6389-4	48.00元	2015年2月
中国教育费皮书	周洪宇	湖北教育出版社	教育	978-7-5351-9910-2	62.00元	2015年2月
自然形态的城市设计	苏黎	东南大学出版社	自然	978-7-5641-5320-5	45.00元	2015年2月
日食与自然灾害	赵得秀	西北工业大学出版社	自然	978-7-5612-4235-3	20.00元	2015年2月

继续查询请输入Y, 退出输入N

②查看个人信息:

```

***** 张三先生, 您好! 欢迎使用图书管理系统! *****
***** 欢迎使用图书管理系统 *****
***** 菜单 *****
***** 1. 查询书籍 *****
***** 2. 个人信息 *****
***** 3. 借阅书籍 *****
***** 4. 归还书籍 *****
***** 5. 退出系统 *****

***请输入您的选项: 2
请再次输入您的密码:
235862

-----
姓名      读者证号      性别
张三      2014011216      男
目前借阅的所有书籍有:
没有借阅书籍!

-----
请按任意键继续. . .

```

为了保障个人信息安全, 在查看个人信息之前, 仍需要输入密码。在显示个人基本信息的同时, 还会显示该用户目前借阅图书的情况。

③借阅书籍:

```

***** 张三先生, 您好! 欢迎使用图书管理系统! *****
***** 欢迎使用图书管理系统 *****
***** 菜单 *****
***** 1. 查询书籍 *****
***** 2. 个人信息 *****
***** 3. 借阅书籍 *****
***** 4. 归还书籍 *****
***** 5. 退出系统 *****

***请输入您的选项: 3
请再次输入您的密码[注意借阅每本书之前都会要输入密码]:
235862

-----
请输入您要借的书籍的ISBN号: 978-7-5086-4937-5
借书 *****
ISBN号为978-7-5086-4937-5的图书借阅成功!
目前借阅的所有书籍有:
书名/题目      作者      出版社      学科      ISBN号      价格      出版日期
世界第一好懂      [日]小川仁志      中信出版社      哲学      978-7-5086-4937-5      35.00元      2015年2月

-----
继续借阅书籍? 是请输入A; 还书? 是请输入B; 退出请输入N

```

进入借书功能模块之后, 按照系统提示, 即可借阅图书。

④归还书籍:

```

***** 张三先生, 您好! 欢迎使用图书管理系统! *****
***** 欢迎使用图书管理系统 *****
***** 菜单 *****
***** 1. 查询书籍 *****
***** 2. 个人信息 *****
***** 3. 借阅书籍 *****
***** 4. 归还书籍 *****
***** 5. 退出系统 *****

***请输入您的选项: 4
请再次输入您的密码[注意归还每本书之前都会要输入密码]:
235862
对不起, 您当前未借阅任何书籍!
请按任意键继续. . .

```

归还图书时, 系统会判断该用户是否借阅了图书。如果借阅了图书, 就会要求输入要退还的书本 ISBN 号, 判断 ISBN 号对应的书是否在已借书的列表内, 如果在列表中, 即可成功归还书籍。

③退出系统

直接从当前界面退出图书管理系统。

4. 管理员功能模块

一、图书管理功能模块

①查询图书信息：

在提示“请输入您的选项”中输入 1，即可进入“查询书籍”模块。之后即可得到：

```

***** 张三先生，您好！欢迎使用图书管理系统！ *****
***** 欢迎使用图书管理系统 *****
***** 菜单 *****
***** 1. 查询书籍 *****
***** 2. 个人信息 *****
***** 3. 借阅书籍 *****
***** 4. 归还书籍 *****
***** 5. 退出系统 *****
***请输入您的选项：1
  
```

```

***** 张三先生，您好！欢迎使用图书管理系统！ *****
***** 欢迎使用图书管理系统 *****
***** 菜单 *****
***** 1. 查询书籍 *****
***** 2. 个人信息 *****
***** 3. 借阅书籍 *****
***** 4. 归还书籍 *****
***** 5. 退出系统 *****
***请输入您的选项：1
***** 查找书目 *****
***** 1. 精确查询 *****
***** 2. 模糊查询 *****
***请输入您的查询方式：
  
```

- 根据提示，输入需要的查询方式。

输入 1 时，为“精确查询”，输入图书的 ISBN 号即可。

```

***** 张三先生，您好！欢迎使用图书管理系统！ *****
***** 欢迎使用图书管理系统 *****
***** 菜单 *****
***** 1. 查询书籍 *****
***** 2. 个人信息 *****
***** 3. 借阅书籍 *****
***** 4. 归还书籍 *****
***** 5. 退出系统 *****
***请输入您的选项：1
***** 查找书目 *****
***** 1. 精确查询 *****
***** 2. 模糊查询 *****
***请输入您的查询方式:1
***** 精确查询 *****
***请输入您要查询的书的ISBN号
978-7-5086-4937-5
***查询结果：
  
```

书名/题目	作者	出版社	学科	ISBN号	价格	出版日期
世界第一好懂的哲学课	【日】小川仁志	中信出版社	哲学	978-7-5086-4937-5	35.00元	2015年2 月

继续查询请输入Y，退出输入N

输入 2 时，进入“模糊查询”，再次选择不同的选项：

```

***** 张三先生，您好！欢迎使用图书管理系统！ *****
***** 欢迎使用图书管理系统 *****
***** 菜单 *****
***** 1. 查询书籍 *****
***** 2. 个人信息 *****
***** 3. 借阅书籍 *****
***** 4. 归还书籍 *****
***** 5. 退出系统 *****
***请输入您的选项：1
***** 查找书目 *****
***** 1.精确查询 *****
***** 2.模糊查询 *****
***请输入您的查询方式:2
***** 模糊查询 *****
***** 1.根据书名查询 *****
***** 2.根据作者查询 *****
***** 3.根据出版社查询 *****
***** 4.根据学科查询 *****
***** 5.根据出版年月查询 *****
***请输入您的选项：

```

可以看出，模糊查询板块又细分为 5 个功能模块：

以下是五个模块的测试结果：

①根据书名查询：

```

***** 查找书目 *****
***** 1.精确查询 *****
***** 2.模糊查询 *****
***请输入您的查询方式:2
***** 模糊查询 *****
***** 1.根据书名查询 *****
***** 2.根据作者查询 *****
***** 3.根据出版社查询 *****
***** 4.根据学科查询 *****
***** 5.根据出版年月查询 *****
***请输入您的选项：1
***请输入书名：
政治哲学
***查询结果：
-----
未查到相关书目
继续查询请输入Y，退出输入N

```

②根据作者查询：

```

***** 查找书目 *****
***** 1.精确查询 *****
***** 2.模糊查询 *****
***请输入您的查询方式:2
***** 模糊查询 *****
***** 1.根据书名查询 *****
***** 2.根据作者查询 *****
***** 3.根据出版社查询 *****
***** 4.根据学科查询 *****
***** 5.根据出版年月查询 *****
***请输入您的选项：2
***请输入作者名：
江春莲
***查询结果：
-----
书名/题目      作者      出版社      学科      ISBN号      价格      出版日期
数学教育初论    江春莲    科学出版社    教育    978-7-03-041790-9    128.00元    2015年1 月
继续查询请输入Y，退出输入N

```

③根据出版社查询:

查找书目						
1.精确查询						
2.模糊查询						
***请输入您的查询方式:2						
模糊查询						
1.根据书名查询						
2.根据作者查询						
3.根据出版社查询						
4.根据学科查询						
5.根据出版年月查询						
***请输入您的选项:3						
请输入准确的出版社名称:						
清华大学出版社						
查询结果:						
书名/题目	作者	出版社	学科	ISBN号	价格	出版日期
过程控制技术及其应用	徐湘元	清华大学出版社	技术	978-7-322-37268-4	39.00元	2013年12月
继续查询请输入Y, 退出输入N						

④根据所属学科查询:

查找书目						
1.精确查询						
2.模糊查询						
***请输入您的查询方式:2						
模糊查询						
1.根据书名查询						
2.根据作者查询						
3.根据出版社查询						
4.根据学科查询						
5.根据出版年月查询						
***请输入您的选项:4						
【注意】当前分类有:哲学、社会、自然、教育、艺术、技术, 请输入要查询的类别:						
艺术						
查询结果:						
书名/题目	作者	出版社	学科	ISBN号	价格	出版日期
环境艺术设计概论	王烨	中国电力出版社	艺术	978-7-5123-6532-2	59.00元	2014年12月
思考的艺术	[英]Buzan&Tony	化学工业出版社	艺术	978-7-122-22298-5	45.00元	2014年11月
中国树木盆景艺术	黄映泉	安徽科学技术出版社	艺术	978-7-5337-6473-9	85.00元	2014年12月
继续查询请输入Y, 退出输入N						

⑤根据出版日期查询

查找书目						
1.精确查询						
2.模糊查询						
***请输入您的查询方式:2						
模糊查询						
1.根据书名查询						
2.根据作者查询						
3.根据出版社查询						
4.根据学科查询						
5.根据出版年月查询						
***请输入您的选项:5						
请输入出版年份: 2015						
请输入出版月份: 2						
查询结果:						
书名/题目	作者	出版社	学科	ISBN号	价格	出版日期
世界第一好懂的哲学课	[日]小川仁志	中信出版社	哲学	978-7-5086-4937-5	35.00元	2015年2月
政治哲学	[美]史密斯	北京联合出版公司	哲学	978-7-5502-3895-5	39.80元	2015年2月
宋代经济学哲学研究	王心竹	上海科技出版社	哲学	978-7-5439-6389-4	48.00元	2015年2月
中国教育黄皮书	周洪宇	湖北教育出版社	教育	978-7-5351-9910-2	62.00元	2015年2月
自然形态的城市设计	苏毅	东南大学出版社	自然	978-7-5641-5320-5	45.00元	2015年2月
日食与自然灾害	赵得秀	西北工业大学出版社	自然	978-7-5612-4235-3	20.00元	2015年2月
继续查询请输入Y, 退出输入N						

②添加图书信息：

输入“2”，进入添加图书信息模块。

```

管理员您好！欢迎使用图书管理系统！
*****
***** 菜单 *****
***** 1.图书管理 *****
***** 2.读者管理 *****
*****
****请输入选项：1
*****
***** 欢迎使用图书管理模块 *****
***** 菜单 *****
***** 1.查询图书 *****
***** 2.添加图书 *****
***** 3.删除图书 *****
***** 4.全部图书 *****
***** 5.清除所有图书 *****
*****
****请输入选项：2
*****
***** 添加图书 *****
*****
请按照以下步骤添加书目：
****请输入书名：飞向蓝天
****请输入作者：马云天
****请输入出版社：中信出版社
****请输入学科类别：
【注意】当前分类有：哲学、社会、自然、教育、艺术、技术，请输入相应类别
自然
****请输入ISBN号：978-7-5895-7989-6
****请输入价格：58.00元
****请输入年份：2014
****请输入月份：8
书目添加成功！

书名/题目      作者      出版社      学科      ISBN号      价格      出版日期
-----
飞向蓝天      马云天      中信出版社      自然      978-7-5895-7989-6      58.00元      2014年8 月

继续添加图书么？是请输入Y，否请输入N

```

按照系统提示的步骤，完成图书信息的录入。

③删除图书信息：

输入“3”，进入删除图书信息模块。

```

管理员您好！欢迎使用图书管理系统！
*****
***** 菜单 *****
***** 1.图书管理 *****
***** 2.读者管理 *****
*****
****请输入选项：1
*****
***** 欢迎使用图书管理模块 *****
***** 菜单 *****
***** 1.查询图书 *****
***** 2.添加图书 *****
***** 3.删除图书 *****
***** 4.全部图书 *****
***** 5.清除所有图书 *****
*****
****请输入选项：3
****请输入您要删除的图书的ISBN号：978-7-5086-4973-5
图书管理系统中没有该书！
****继续删除请输入Y，退出请输入N
Y
****请输入您要删除的图书的ISBN号：978-7-5086-4937-5
《世界第一好懂的哲学课》删除完毕！
请按任意键继续...

```

【注意】

执行这一功能时输入要删除的图书的 ISBN 号，如果 ISBN 号输入错误，系统会提示“图书管理系统中没有该书！”输入的 ISBN 正确时，即可将该书成功删除！

④显示全部图书:

输入“4”，进入显示全部图书模块。如下图所示:

管理员您好! 欢迎使用图书管理系统!

```

*****
菜单
1.图书管理
2.读者管理
*****
请输入选项: 1
*****
欢迎使用图书管理模块
*****
菜单
1.查询图书
2.添加图书
3.删除图书
4.全部图书
5.清除所有图书
*****
请输入选项:4
*****
全部图书
*****

```

书名/题目	作者	出版社	学科	ISBN号	价格	出版日期
世界第一好懂的哲学课	[日]小川仁志	中信出版社	哲学	978-7-5086-4937-5	35.00元	2015年2月
政治哲学	[美]史密斯	北京联合出版公司	哲学	978-7-5502-3895-5	39.80元	2015年2月
宋代经济学研究	王心竹	上海科技出版社	哲学	978-7-5439-6389-4	48.00元	2015年2月
中国教育黄皮书	周洪宇	湖北教育出版社	教育	978-7-5351-9910-2	62.00元	2015年2月
数学教育初论	江春莲	科学出版社	教育	978-7-03-041790-9	128.00元	2015年1月
环境艺术设计概论	王辉	中国电力出版社	艺术	978-7-5123-6532-2	59.00元	2014年12月
思考的艺术	[英]Buzan&Tony	化学工业出版社	艺术	978-7-122-22298-5	45.00元	2014年11月
中国树木盆景艺术	黄映泉	安徽科学技术出版社	艺术	978-7-5337-6473-9	85.00元	2014年12月
自然形态的城市设计	苏毅	东南大学出版社	自然	978-7-5641-5320-5	45.00元	2015年2月
自然的历史	[美]Tom·Baione	重庆大学出版社	自然	978-7-5624-8116-4	78.00元	2014年2月
自然与自然灾害	赵得秀	西北工业大学出版社	自然	978-7-5612-4235-3	20.00元	2015年2月
自然与道德	史军	北京人民文学出版社	自然	978-7-03-041868-5	69.00元	2014年6月
社会思潮怎么看	王燕文	江苏人民出版社	社会	978-7-214-13522-3	48.00元	2014年11月
社会调查与统计	董海全	武汉大学出版社	社会	978-7-307-15088-1	40.00元	2015年1月
世界社会主义五百年	中共宣传部	武汉出版社	社会	978-7-5099-0489-3	29.00元	2013年8月
过程控制技术及其应用	徐湘元	清华大学出版社	技术	978-7-322-37268-4	39.00元	2013年12月
厌氧生物技术	王凯军	化学工业出版社	技术	978-7-122-12538-5	98.00元	2015年4月
京东技术解密	京东研发体系	电子工业出版社	技术	978-7-121-24691-3	65.00元	2011年12月
京东技术解密	京东研发体系	电子工业出版社	技术	978-7-121-24691-3	65.00元	2011年12月

所有图书信息非常整齐地罗列出来。具有很强的可读性。

⑤删除全部图书:

输入“5”，进入删除全部图书功能模块，如图:

管理员您好! 欢迎使用图书管理系统!

```

*****
菜单
1.图书管理
2.读者管理
*****
请输入选项: 1
*****
欢迎使用图书管理模块
*****
菜单
1.查询图书
2.添加图书
3.删除图书
4.全部图书
5.清除所有图书
*****
请输入选项:5
*****
删除所有图书信息
《世界第一好懂的哲学课》删除完毕
《政治哲学》删除完毕
《宋代经济学研究》删除完毕
《中国教育黄皮书》删除完毕
《数学教育初论》删除完毕
《环境艺术设计概论》删除完毕
《思考的艺术》删除完毕
《中国树木盆景艺术》删除完毕
《自然形态的城市设计》删除完毕
《自然的历史》删除完毕
《自然与自然灾害》删除完毕
《自然与道德》删除完毕
《社会思潮怎么看》删除完毕
《社会调查与统计》删除完毕
《世界社会主义五百年》删除完毕
《过程控制技术及其应用》删除完毕
《厌氧生物技术》删除完毕
《京东技术解密》删除完毕
《京东技术解密》删除完毕
书库已空!
请按任意键继续. . .

```


二、读者管理功能模块

①查询读者信息：

在提示“请输入您的选项”中输入 1，即可进入“查询读者”模块。之后即可得到：

```

管理员您好！欢迎使用图书管理系统！
*****
***** 菜单 *****
***** 1.图书管理 *****
***** 2.读者管理 *****
*****
****请输入选项：2
***** 欢迎使用读者管理模块 *****
***** 菜单 *****
***** 1.查询读者 *****
***** 2.添加读者 *****
***** 3.删除读者 *****
***** 4.显示全部读者 *****
***** 5.清除全部读者 *****
*****
****请输入选项：
1

```

根据提示，输入需要的查询方式。

输入 1 时，为“精确查询”，输入读者的读者证号即可。

```

*****查找用户*****
1.精确查询
2.模糊查询
请输入您的查询方式：1
请输入您要查询的用户的ID
2014011216
查询结果：
姓名      读者证号      性别
张三      2014011216    男
目前借阅的所有书籍有：
没有借阅书籍！
请按任意键继续...

```

输入 2 时，进入“模糊查询”，再次选择不同的选项：

```

*****查找用户*****
1.精确查询
2.模糊查询
请输入您的查询方式：2
1.根据姓名查询：
2.根据性别查询：
请输入您的选项：

```

可以看出，模糊查询板块又细分为 2 个功能模块：

以下是两个模块的测试结果：

①根据姓名查询：

```

*****查找用户*****
1.精确查询
2.模糊查询
请输入您的查询方式：2
1.根据姓名查询：
2.根据性别查询：
请输入您的选项：1
请输入您要查询的读者姓名：李四
查询结果为：
姓名      读者证号      性别
李四      2014011211    男
目前借阅的所有书籍有：
没有借阅书籍！
请按任意键继续...

```


②根据性别查询：

例如，所有的女读者：

```

*****查找用户*****
1.精确查询
2.模糊查询
请输入您的查询方式：2
1.根据姓名查询：
2.根据性别查询：
请输入您的选项：2
请输入您要查询的性别（男/女）：女
查询结果为：
姓名      读者证号      性别
何柳      2014011212    女
目前借阅的所有书籍有：
没有借阅书籍！
姓名      读者证号      性别
项少玲    2014012153    女
目前借阅的所有书籍有：
没有借阅书籍！
姓名      读者证号      性别
闫平      2014001535    女
目前借阅的所有书籍有：
没有借阅书籍！
姓名      读者证号      性别
师素芳    2014678632    女
目前借阅的所有书籍有：
没有借阅书籍！
请按任意键继续...

```

②添加读者信息：

输入“2”，进入添加读者信息模块。

```

*****添加新读者*****
请按照以下步骤添加读者
请输入姓名：柳叶子
请输入登录ID：2014011111
请输入性别：女
请设置密码：585288
读者信息添加成功！
姓名      读者证号      性别
柳叶子    2014011111    女
继续添加读者么？是请输入Y，否请输入N

```

按照系统提示的步骤，完成读者信息的录入。

③删除读者信息：

输入“3”，进入删除读者信息模块。

```

*****欢迎使用读者管理模块*****
*****菜单*****
*****1.查询读者*****
*****2.添加读者*****
*****3.删除读者*****
*****4.显示全部读者*****
*****5.清除全部读者*****
*****请输入选项：*****
3
*****删除用户*****
请输入您要删除的读者用户的ID：
2014011215
图书管理系统中无该用户！
继续删除请输入“Y”，退出请输入“N”
Y
*****删除用户*****
请输入您要删除的读者用户的ID：
2014011216
用户张三所有信息删除完毕！
请按任意键继续...

```

【注意】

执行这一功能时输入要删除的读者证号，如果读者证号输入错误，系统会提示“图书管理系统中没有该用户！”输入的读者证号正确时，即可将该读者成功删除！

④显示全部读者：

输入“4”，进入显示全部读者模块。如下图所示：

```

管理员您好！欢迎使用图书管理系统！
*****
***** 菜单 *****
***** 1.图书管理 *****
***** 2.读者管理 *****
*****
*****请输入选项：2*****
***** 欢迎使用读者管理模块 *****
***** 菜单 *****
***** 1.查询读者 *****
***** 2.添加读者 *****
***** 3.删除读者 *****
***** 4.显示全部读者 *****
***** 5.清除全部读者 *****
*****请输入选项：4*****
显示所有读者用户及借阅的所有书籍
-----
姓名      读者证号      性别
张三      2014011216      男
目前借阅的所有书籍有：
没有借阅书籍！
-----
姓名      读者证号      性别
李四      2014011211      男
目前借阅的所有书籍有：
没有借阅书籍！
-----
姓名      读者证号      性别
吕武      2014011125      男
目前借阅的所有书籍有：
没有借阅书籍！
-----
姓名      读者证号      性别
何柳      2014011212      女
目前借阅的所有书籍有：
没有借阅书籍！
-----

```

所有读者信息非常整齐地罗列出来。具有很强的可读性。

⑤删除全部读者：

输入“5”，进入删除全部读者功能模块，如图：

```

*****
***** 菜单 *****
***** 1.图书管理 *****
***** 2.读者管理 *****
*****
*****请输入选项：2*****
***** 欢迎使用读者管理模块 *****
***** 菜单 *****
***** 1.查询读者 *****
***** 2.添加读者 *****
***** 3.删除读者 *****
***** 4.显示全部读者 *****
***** 5.清除全部读者 *****
*****请输入选项：5*****
*****删除所有用户信息*****
正在删除第1位用户的信息
正在删除第2位用户的信息
正在删除第3位用户的信息
正在删除第4位用户的信息
正在删除第5位用户的信息
正在删除第6位用户的信息
正在删除第7位用户的信息
正在删除第8位用户的信息
正在删除第9位用户的信息
所有用户信息删除完毕！
请按任意键继续...

```

第三部分 注意事项

- ① 使用该系统时尽量不要一次性调用过多功能，可能会影响性能。
- ② 部分功能暂不支持返回上一级菜单的功能。
- ③ 因为从系统数据文件读入图书和读者信息的要求较高，数据文件内的图书信息和读者信息必须按照顺序存入，否则会造成文件导入错误。

6. 总结及编程体会

图书管理系统完成后，心里十分激动。尽管自己设计的系统的功能性和界面美观性不是很优秀，但个人觉得自己在 C++ 语言编程方面有了很大的进步。编写程序之前的不自信和胆怯也减少了许多。

有了学生成绩管理系统的编程经验，图书管理系统的编程速度比之前快了许多。“思路清晰”是我在编程中体会的最重要的一环。在编程之前，一定要理清楚自己设计的程序要实现哪些功能，这些功能的实现需要哪些对象和条件，整理好彼此之间的逻辑关系，剩下的就是编程实现了。编程是按照之前的思路一步步进行的，因此在编程之前至少心里要有个大概的骨架，这样编程时才会有比较清晰的层次感。

还有一点，就是体会到了计算机强大的处理能力。尽管目前编出的系统存储功能有限，但也可以体会到计算机的迅捷和准确性。因此，我需要加强程序设计的训练，为之后更深入的学习和理解做准备。

附录：源程序清单

```
//头文件“Class.h”包含了所有的类及成员函数
#include<iostream>
#include<iomanip> //控制输出格式
#include<cstdlib>
#include<string> //本程序中所有的字符串使用C++自带的string类
#include<fstream> //文件流操作
#define M 500 //定义系统中书籍最大容纳量
#define N 10 //每位读者借阅图书数目的上限
#define Q 20 //系统的读者用户数量
using namespace std;
void BF() //（Beautify）用于美化输出界面的函数[功能是输出“-”，具有层次感]
{
    for(int i=0;i<115;i++) //此处输出“-”，因而最终exe文件输出框要改变默认长度，改为
        cout<<"-";
    cout<<endl;
}
```

```

class Base//基类，派生出Book和Reader类
{
public:
    Base(){};
};
class Book:virtual public Base//Book类继承虚基类
{
public:
    Book(string name,string author, string publisher,string subject,string isbn,string
price,int year,int month);//构造函数
    Book();//默认构造函数
    void From_File(string name,string author, string publisher,string subject,string
isbn,string price,int year,int month);//从文件中按顺序导入图书信息
    string GetName();//获取图书的书名
    string GetSubject();//获取图书所属的学科
    string GetISBN();//获取图书ISBN号
    string GetAuthor();//获取图书作者的姓名
    string GetPublisher();//获取相应出版社
    string GetPrice();//获取图书价格
    int GetYear();//获取出版年份
    int GetMonth();//获取出版月份
    int GetStatus();//获取图书状态（通过图书的状态表示是否可以借阅）
    void Delete();//从图书库里删除该书
    friend ostream & operator << (ostream &out, Book const &book);//通过友元函
数进行“<<”操作符重载，一次性输出Book对象的所有信息
private://私有成员，避免遭到更改
    string Name,Author,Publisher,Price,Subject,ISBN;
    int Year,Month;
    int status;//指示图书系统里是否有该书
};
Book::Book(string name,string author,string publisher,string subject,string isbn,
string price,int year,int month)
{
    Name=name;        Author=author;   Publisher=publisher;
    Subject=subject;   ISBN=isbn;        Price=price;
    Year=year;         Month=month;      status=1;
}
Book::Book();//默认构造函数，没有图书信息
{
    Name=" ";          Author=" ";       Publisher=" ";
    Subject=" ";        ISBN=" ";         Price=" ";
    Year=0;              Month=0;          status=0;
}
void Book::From_File(string name,string author, string publisher,string subject,string

```

```
isbn,string price,int year,int month)
{
    Name=name;      Author=author;  Publisher=publisher;
    Subject=subject; ISBN=isbn;      Price=price;
    Year=year;       Month=month;     status=1;
}
string Book::GetName()
{
    return Name;
}
string Book::GetSubject()
{
    return Subject;
}
string Book::GetISBN()
{
    return ISBN;
}
string Book::GetAuthor()
{
    return Author;
}
string Book::GetPublisher()
{
    return Publisher;
}
string Book::GetPrice()
{
    return Price;
}
int Book::GetYear()
{
    return Year;
}
int Book::GetMonth()
{
    return Month;
}
int Book::GetStatus()
{
    return status;
}
void Book::Delete()
{
```

```

        status=0;//将状态变量置为,表示该图书已经不存在系统中
    }
    ostream & operator << (ostream &out, Book &book)//输出图书的所有信息
    {
        cout<<setiosflags(ios::left)<<setw(25)<<book.GetName()<<setw(20)<<book.Get
        Author()<<setw(20)<<book.GetPublisher()<<setw(8)<<book.GetSubject()

        <<setw(20)<<book.GetISBN()<<setw(10)<<book.GetPrice()<<setw(4)<<book.Get
        Year()<<"年"<<setw(2)<<book.GetMonth()<<"月"<<endl;
        return out;
    }
class Manage_Book//管理图书类, 保护成员中包括Book类的对象数组
{
public:
    Manage_Book(){};//默认构造函数
    void In_Book_File();//将图书信息从文件中导入系统
    void Add();//添加图书信息
    void Search();//查询图书
    void Display();//显示系统内所有图书
    void Delete();//删除图书信息
    void All_Clear();//删除系统内所有图书
    int Borrow(string isbn);//以ISBN号为形参借阅书籍
    void Return(string isbn);//以ISBN号为形参归还书籍
    void Show(string isbn);//以ISBN号为形参显示图书信息
    int GetNum();//获取系统内目前的图书数目
protected:
    int Num;
    Book book[M];//Book类的对象数组作为保护成员
};
void Manage_Book::In_Book_File()
{
    ifstream In_Book;
    string name,author, publisher,subject,isbn,price;
    int year,month;
    In_Book.open("E:\\book.txt",ios::out);//打开文件& E盘内book.txt文件作为
    默认的图书信息导入路径
    if(In_Book==0)
    {
        cout<<"*****系统内图书加载失败! 请联系工作人员
        *****"<<endl;
        exit(0);
    }
    else
    {

```

```

    Num=0;
    while(In_Book&&!In_Book.eof())&&Num<M)
    {

        In_Book>>name>>author>>publisher>>subject>>isbn>>price>>year>>month;//
        导入图书信息

        book[Num].From_File(name,a author,publisher,subject,isbn,price,year,month);//
        存入对象数组

        Num++;
    }
    In_Book.close();//关闭文件
}
}
void Manage_Book::Add()//添加图书信息
{
    if(Num==M)
    {
        cout<<"*****对不起，系统存书已满，无额外存储空间！请尝试其他
        功能*****"<<endl;
        exit(0);
    }
    else
    {
        Num=Num+1;
        int i=0,year,month;      char command;
        In_Book_File();//导入图书信息
        string name,author,publisher,subject,isbn,price;
        cout<<"*****"                      添加图书
        *****"<<endl;
        cout<<"请按照以下步骤添加书目："<<endl;
        cout<<"*****请输入书名："      cin>>name;
        cout<<"*****请输入作者："      cin>>author;
        cout<<"*****请输入出版社："    cin>>publisher;
        cout<<"*****请输入学科类别："<<endl<<"【注意】当前分类有：哲学、
        社会、自然、教育、艺术、技术，请输入相应类别"<<endl;
        cin>>subject;
        cout<<"*****请输入ISBN号："      cin>>isbn;
        cout<<"*****请输入价格："      cin>>price;
        cout<<"*****请输入年份："      cin>>year;
        cout<<"*****请输入月份："      cin>>month;
        book[Num-
1].From_File(name,author,publisher,subject,isbn,price,year,month);//新的Book类对
象

```

```

        cout<<"书目添加成功！"<<endl;
        BF();//输出一行“-”进行美化
        cout<<setiosflags(ios::left)<<setw(25)<<"书名/题目"<<setw(20)<<"作者"
        "<<setw(20)<<"出版社"<<setw(8)<<"学科"
            <<setw(20)<<"ISBN号"<<setw(10)<<"价格"<<setw(4)<<"出版日期"
        "<<endl;
        BF();//输出一行“-”进行美化
        cout<<book[Num-1];
        BF();//输出一行“-”进行美化
        cout<<"继续添加图书么？是请输入Y，否请输入N"<<endl;
        cin>>command;
        if(command=='s' | command=='S')
            Add();
        else exit(0);
    }
}
void Manage_Book::Search()//查询/搜索图书
{
    int command,i=0,flag=0;
    char Next;
    string a;
    In_Book_File();//导入图书信息文件
    cout<<"*****"
    *****"<<endl;
    cout<<"*****"
    *****"<<endl;
    cout<<"*****"
    *****"<<endl;
    cout<<"****请输入您的查询方式:";
    cin>>command;
    if(command==1)
    {
        cout<<"*****"
        *****"<<endl;
        cout<<"****请输入您要查询的书的ISBN号"<<endl;
        cin>>a;
        cout<<"****查询结果："<<endl;
        BF();//美化
        cout<<setiosflags(ios::left)<<setw(25)<<"书名/题目"<<setw(20)<<"作者"
        "<<setw(20)<<"出版社"<<setw(8)<<"学科"
            <<setw(20)<<"ISBN号"<<setw(10)<<"价格"<<setw(4)<<"出版日期"
        "<<endl;
        BF();//美化
        for(i=0;i<Num;i++)

```

查找书目

1.精确查询

2.模糊查询

精确查询


```

{
    if(book[i].GetISBN()==a)
        cout<<book[i];
    flag=1;
    break;
}
BF();//美化
if(flag==0)
    cout<<"未查到此书！"<<endl;
cout<<"继续查询请输入Y，退出输入N"<<endl;
cin>>Next;
if(Next=='Y' || Next=='y')
    Search();
else exit(0);
}
else if(command==2)
{
    int command2;
    cout<<"*****"
    *****"<<endl;
    cout<<"*****"
    *****"<<endl;
    cout<<"*****"
    *****"<<endl;
    cout<<"*****"
    *****"<<endl;
    cout<<"*****"
    *****"<<endl;
    cout<<"*****"
    *****"<<endl;
    cout<<"*****"
    *****"<<endl;
    cout<<"****请输入您的选项：";
    cin>>command2;
    switch(command2)
    {
    case 1://根据书名查询
        flag=0;
        cout<<"****请输入书名："<<endl;    cin>>a;
        cout<<"****查询结果："<<endl;
        for(i=0;i<Num;i++)
        {
            if(book[i].GetName()==a)
            {
                BF();//美化
                cout<<setiosflags(ios::left)<<setw(25)<<"书名/题目

```

模糊查询

1.根据书名查询

2.根据作者查询

3.根据出版社查询

4.根据学科查询

5.根据出版年月查询

```

"<<setw(20)<<"作者"<<setw(20)<<"出版社"<<setw(8)<<"学科"
    <<setw(20)<<"ISBN号"<<setw(10)<<"价格"
"<<setw(4)<<"出版日期"<<endl;
    cout<<book[i];
    BF();//美化
    flag=1;
}
break;
}
BF();//美化
if(flag==0)
    cout<<"未查到相关书目"<<endl;
cout<<"继续查询请输入Y，退出输入N"<<endl;
cin>>Next;
if(Next=='Y' || Next=='y')
    Search();
else exit(0);
break;
case 2://根据作者查询
    flag=0;
    cout<<"****请输入作者名: "<<endl;    cin>>a;
    cout<<"****查询结果: "<<endl;
    BF();//美化
    cout<<setiosflags(ios::left)<<setw(25)<<"书名/题目"<<setw(20)<<"作
者"<<setw(20)<<"出版社"<<setw(8)<<"学科"
        <<setw(20)<<"ISBN号"<<setw(10)<<"价格"<<setw(4)<<"出
版日期"<<endl;
    BF();//美化
    for(i=0;i<Num;i++)
    {
        if(book[i].GetAuthor()==a)
            cout<<book[i];
        flag=1;
    }
    BF();//美化
    if(flag==0)
        cout<<"未查到相关书目"<<endl;
    cout<<"继续查询请输入Y，退出输入N"<<endl;
    cin>>Next;
    if(Next=='Y' || Next=='y')
        Search();
    else exit(0);
    break;
case 3://根据出版社查询

```

```

flag=0;
cout<<"请输入准确的出版社名称: "<<endl;      cin>>a;
cout<<"查询结果: "<<endl;
BF();//美化
cout<<setiosflags(ios::left)<<setw(25)<<"书名/题目"<<setw(20)<<"作者"
<<setw(20)<<"出版社"<<setw(8)<<"学科"
<<setw(20)<<"ISBN号"<<setw(10)<<"价格"<<setw(4)<<"出版日期"
<<endl;
BF();//美化
for(i=0;i<Num;i++)
{
    if(book[i].GetPublisher()==a)
        cout<<book[i];
    flag=1;
}
BF();//美化
if(flag==0)
    cout<<"未查到相关书目"<<endl;
cout<<"继续查询请输入Y，退出输入N"<<endl;
cin>>Next;
if(Next=='Y' || Next=='y')
    Search();
else exit(0);
break;
case 4://根据学科查询
flag=0;
cout<<"【注意】当前分类有：哲学、社会、自然、教育、艺术、技术，请输入要查询的类别:"<<endl;
cin>>a;
cout<<"查询结果: "<<endl;
BF();//美化
cout<<setiosflags(ios::left)<<setw(25)<<"书名/题目"<<setw(20)<<"作者"
<<setw(20)<<"出版社"<<setw(8)<<"学科"
<<setw(20)<<"ISBN号"<<setw(10)<<"价格"<<setw(4)<<"出版日期"
<<endl;
BF();//美化
for(i=0;i<Num;i++)
{
    if(book[i].GetSubject()==a)
        cout<<book[i];
    flag=1;
}
BF();//美化
if(flag==0)

```

```

        cout<<"未查到相关书目"<<endl;
    cout<<"继续查询请输入Y，退出输入N"<<endl;
    cin>>Next;
    if(Next=='Y' || Next=='y')
        Search();
    else exit(0);
    break;
case 5://根据出版年月查询
    flag=0;
    int year,month;
    cout<<"请输入出版年份： ";    cin>>year;
    cout<<"请输入出版月份： ";    cin>>month;
    cout<<"查询结果： "<<endl;
    BF();//美化
    cout<<setiosflags(ios::left)<<setw(25)<<"书名/题目"<<setw(20)<<"作者"
    <<setw(20)<<"出版社"<<setw(8)<<"学科"
    <<setw(20)<<"ISBN号"<<setw(10)<<"价格"<<setw(4)<<"出版日期"
    <<endl;
    BF();//美化
    for(i=0;i<Num;i++)
    {
        if(book[i].GetYear()==year&&book[i].GetMonth()==month)
            cout<<book[i];
        flag=1;
    }
    BF();//美化
    if(flag==0)
        cout<<"未查到相关书目"<<endl;
    cout<<"继续查询请输入Y，退出输入N"<<endl;
    cin>>Next;
    if(Next=='Y' || Next=='y')
        Search();
    else exit(0);
    break;
default://输入错误处理
    system("cls");
    cout<<"对不起您的输入有误！请重新输入： "<<endl;
    Search();
}
}
else
{
    cout<<"对不起，您的输入有误！请重新输入： "<<endl;
    Search();
}

```

```

    }
}
void Manage_Book::Display()//显示全部图书
{
    int i=0;
    In_Book_File();
    cout<<"*****"全部图书
*****"<<endl;
    BF();//美化
    cout<<setiosflags(ios::left)<<setw(25)<<"书名/题目"<<setw(20)<<"作者
"<<setw(20)<<"出版社"<<setw(8)<<"学科"
    <<setw(20)<<"ISBN号"<<setw(10)<<"价格"<<setw(4)<<"出版日期"<<endl;
    BF();//美化
    i=0;
    while(i<Num&&book[i].GetStatus()==1)
    {
        cout<<book[i];
        i++;
    }
    BF();//美化
}
void Manage_Book::Delete()//删除图书信息
{
    string isbn;
    int i=0,flag=0;
    In_Book_File();//导入图书信息
    char command;
    cout<<"请输入您要删除的图书的ISBN号： ";
    cin>>isbn;
    for(i=0;i<Num;i++)
    {
        if(book[i].GetISBN()==isbn)
        {
            book[i].Delete();
            cout<<"《"<<book[i].GetName()<<"》 "<<"删除完毕！"<<endl;
            flag=1;
        }
        break;
    }
    if(flag==0)
    {
        cout<<"图书管理系统中没有该书！"<<endl;
        cout<<"****继续删除请输入Y，退出请输入N"<<endl;
        cin>>command;
    }
}

```

```

        if(command=='Y')
            Delete();
        else exit(0);
    }
}
void Manage_Book::All_Clear()//清除所有图书信息
{
    In_Book_File();
    cout<<"删除所有图书信息"<<endl;
    for(int i=0;i<Num;i++)
    {
        cout<<"《"<<book[i].GetName()<<"》 "<<"删除完毕"<<endl;
        book[i].Delete();
    }
    cout<<"书库已空！"<<endl;
}
int Manage_Book::Borrow(string isbn)//根据ISBN号将书籍借出
{
    int i=0,flag=0;
    In_Book_File();
    cout<<"*****"
*****"<<endl;
    for(i=0;i<Num;i++)
    {
        if(book[i].GetISBN()==isbn)
        {
            flag=1;return 1;
        }
    }
    if(flag==0)
        return 0;
    else return 0;
}
void Manage_Book::Return(string isbn)//根据ISBN号将数据收回（读者归还）
{
    int i=0,flag=0;
    In_Book_File();//导入全部图书信息
    cout<<"*****"
*****"<<endl;
    for(i=0;i<Num;i++)//循环寻找要还的书
    {
        if(book[i].GetISBN()==isbn)
        {
            flag=1;
            cout<<"还书成功！"<<endl;

```

借书

还书

```

    }
}
if(flag==0)
{
    cout<<"对不起操作有误！"<<endl;
    exit(0);
}
}
void Manage_Book::Show(string isbn)//根据ISBN号显示书籍信息
{
    int i=0,flag=0;
    In_Book_File();//导入书籍信息
    for(i=0;i<Num;i++)
    {
        cout<<setiosflags(ios::left)<<setw(25)<<"书名/题目"<<setw(20)<<"作者"
        "<<setw(20)<<"出版社"<<setw(8)<<"学科"
        <<setw(20)<<"ISBN号"<<setw(10)<<"价格"<<setw(4)<<"出版日期"
        "<<endl;
        if(book[i].GetISBN()==isbn)
        {
            cout<<book[i];
            flag=1;
            break;
        }
    }
    if(flag==0)
        cout<<"对不起，没有找到此书！"<<endl;
}
int Manage_Book::GetNum()
{
    return Num;
}
class Reader:virtual public Base
{
public:
    Reader(string name,string id,string gender,string code);//构造函数
    Reader();//默认构造函数
    void From_File(string name,string id,string gender,string code);//从文件中导入
    读者信息
    int GetStatus();//获取读者在系统中的状态，表示在系统内，表示已从系统
    中清除
    string GetCode();//获取读者登录密码
    string GetName();//获取读者姓名
    string GetID();//获取读者ID

```

```

    string GetGender();//获取读者姓名
    void Search();//读者查询书籍信息
    void Borrow();//读者借书
    void Return();//读者还书
    void Display_Keeping();//显示目前借阅的书籍
    void Delete();//删除读者信息
    friend ostream & operator << (ostream &out, Reader const &r);//友元函数操作
    符重载，“<<”直接输出读者信息
protected://保护成员
    string Name,ID,Gender,Code,Mybook[N];//图书的全部信息以及已经借阅的图
    书的ISBN号
    int Status,Num_Keeping;//状态变量和目前借阅的图书数目
};
Reader::Reader(string name,string id,string gender,string code)//构造函数
{
    Name=name;    ID=id;    Gender=gender;
    Code=code;    Status=1;    Num_Keeping=0;
}
Reader::Reader();//默认构造函数
{
    Name=" ";    ID=" ";    Gender=" ";
    Code="0";    Status=0;    Num_Keeping=0;
}
void Reader::From_File(string name,string id,string gender,string code)//从文件中读
    入读者信息
{
    Name=name;    ID=id;    Gender=gender;
    Code=code;    Status=1;    Num_Keeping=0;
}
int Reader::GetStatus();//获取读者用户状态
{
    return Status;
}
string Reader::GetCode();//获取读者密码
{
    return Code;
}
string Reader::GetName();//获取读者姓名
{
    return Name;
}
string Reader::GetID();//获取读者ID
{
    return ID;
}

```



```
}
string Reader::GetGender()//获取读者性别
{
    return Gender;
}
void Reader::Search()//读者的查询图书功能
{
    Manage_Book MB;
    MB.Search();
}
void Reader::Borrow()//借书功能
{
    if(Num_Keeping==N)
    {
        cout<<"对不起，您的借书书目已经达到最高上限"<<N<<"本"<<"不能继续借书。"<<endl;
        exit(0);
    }
    else
    {
        string isbn;
        cout<<"请输入您要借的书籍的ISBN号： ";
        cin>>isbn;
        Manage_Book MB;
        if(MB.Borrow(isbn)==1)
        {
            Num_Keeping++;
            Mybook[Num_Keeping-1]=isbn;//将该书作为读者借阅的下一本书存到读者借阅书本的string数组里
            cout<<"ISBN号为"<<isbn<<"的图书借阅成功！"<<endl;
            Display_Keeping();
        }
        else cout<<"对不起，该书不在系统内或已被借出，借书未成功！"<<endl;
    }
}
void Reader::Return()//还书功能
{
    if(Num_Keeping==0)
    {
        cout<<"对不起，您当前未借阅任何书籍!"<<endl;
        exit(0);
    }
    else
```

```

    {
        string isbn;
        Manage_Book MB;
        cout<<"请输入您要还的图书的ISBN号： ";
        cin>>isbn;
        MB.Return(isbn);
    }
}

void Reader::Display_Keeping()//显示目前借阅的书籍
{
    Manage_Book MB;
    cout<<"目前借阅的所有书籍有： "<<endl;
    if(Num_Keeping==0)
        cout<<"没有借阅书籍！ "<<endl;
    else
    {
        for(int i=0;i<Num_Keeping;i++)
            MB.Show(Mybook[i]);
    }
}

void Reader::Delete()//删除读者信息
{
    Status=0;
}

ostream & operator << (ostream &out, Reader &r)//友元函数操作符重载，“<<”直接输出读者信息
{
    cout<<setiosflags(ios::left)<<setw(15)<<"姓名"<<setw(15)<<"读者证号"
    "<<setw(8)<<"性别"<<endl;
    cout<<setiosflags(ios::left)<<setw(15)<<r.GetName()<<setw(15)<<r.GetID()<<set
    w(8)<<r.GetGender()<<endl;
    return out;
}

class Manage_Reader//管理读者，类的保护成员中包含Reader类的对象数组
{
public:
    Manage_Reader(){};//构造函数
    void In_Reader_File();//导入读者信息
    void Add();//添加读者信息
    void Search();//查询搜索读者
    void Display();//显示所有读者的信息
    void Delete();//删除读者信息
    void All_Clear();//清除所有读者的信息
    void ShowOneself();//显示某一读者自身的信息

```

```

    void Borrow();//借书
    void Return();//还书
    int GetCount();//获取目前系统存储的读者数目
protected:
    Reader reader[Q];//Reader类的对象数组作为保护成员
    int Count;
};
void Manage_Reader::In_Reader_File();//导入读者信息
{
    ifstream In_Reader;
    int i=0;
    string name,id,gender,code;
    In_Reader.open("E:\\reader.txt",ios::out);//读者信息默认存储在文本文件
E:\\reader.txt中
    if(In_Reader==0)
    {
        cout<<"系统用户加载失败！"<<endl;
        exit(0);
    }
    else
    {
        Count=0;
        while(In_Reader&&!In_Reader.eof())&&Count<Q//按照顺序依次读入文件
中的读者信息
        {
            In_Reader>>name>>id>>gender>>code;
            reader[i].From_File(name,id,gender,code);
            Count++;
            i++;
        }
        In_Reader.close();//关闭文件
    }
}
void Manage_Reader::Add();//添加读者信息
{
    In_Reader_File();//导入读者信息
    if(Count==Q)
    {
        cout<<"对不起，暂不支持新用户注册"<<endl;
        exit(0);
    }
    else
    {
        Count++;
    }
}

```

```

    string name,id,gender,code;
    char command;
    cout<<"*****添加新读者*****"<<endl;
    cout<<"请按照以下步骤添加读者"<<endl;//依次输入读者信息
    cout<<"请输入姓名: ";    cin>>name;
    cout<<"请输入登录ID: ";  cin>>id;
    cout<<"请输入性别: ";    cin>>gender;
    cout<<"请设置密码: ";    cin>>code;
    reader[Count-1].From_File(name,id,gender,code);
    cout<<"读者信息添加成功!"<<endl;
    cout<<reader[Count-1];
    cout<<"继续添加读者么? 是请输入Y, 否请输入N"<<endl;
    cin>>command;
    if(command=='S' | command=='s')
        Add();//继续添加读者信息
    else exit(0);
}
}
void Manage_Reader::Search()
{
    int command,i=0,flag=0;
    string a;
    In_Reader_File();//导入读者信息
    cout<<"*****查找用户*****"<<endl;
    cout<<"1.精确查询"<<endl;
    cout<<"2.模糊查询"<<endl;
    cout<<"请输入您的查询方式: ";
    cin>>command;
    switch(command)
    {
    case 1://根据ID查找读者用户
        cout<<"请输入您要查询的用户的ID"<<endl;    cin>>a;
        cout<<"查询结果: "<<endl;
        for(i=0;i<Count;i++)
            if(reader[i].GetID()==a)
            {
                cout<<reader[i];
                reader[i].Display_Keeping();
                flag=1;
            }
        if(flag==0)
            cout<<"未查到该用户!"<<endl;
        break;
    case 2:

```

```
int command2;
cout<<"1.根据姓名查询: "<<endl;
cout<<"2.根据性别查询: "<<endl;
cout<<"请输入您的选项: ";
cin>>command2;
if(command2==1)//根据姓名查找读者用户
{
    cout<<"请输入您要查询的读者姓名: ";
    cin>>a;
    cout<<"查询结果为: "<<endl;
    for(i=0;i<Count;i++)
    {
        if(reader[i].GetName()==a)
        {
            cout<<reader[i];
            reader[i].Display_Keeping();
            flag=1;
        }
    }
    if(flag==0)
        cout<<"未查到该用户! "<<endl;
}
else if(command2==2)//根据性别查询读者信息
{
    cout<<"请输入您要查询的性别（男/女）: ";
    cin>>a;
    cout<<"查询结果为: "<<endl;
    for(i=0;i<Count;i++)
    {
        if(reader[i].GetGender()==a)
        {
            cout<<reader[i];
            reader[i].Display_Keeping();
            flag=1;
        }
    }
    if(flag==0)
        cout<<"未查到相关用户! "<<endl;
}
else
{
    cout<<"对不起，您的输入有误! "<<endl;
    Search();
};
```

```
        break;
    default:
        cout<<"对不起，您的输入有误！"<<endl;
        Search();//重新输入查询
    }
}

void Manage_Reader::Display()//显示所有的读者信息
{
    int i=0,k=0;
    In_Reader_File();//从文件中导入读者的信息
    cout<<"显示所有读者用户及借阅的所有书籍"<<endl;
    while(k<Count&&reader[k].GetStatus()==1)
    {
        BF();
        cout<<reader[k];
        reader[k].Display_Keeping();
        BF();
        k++;
    }
}

void Manage_Reader::Delete()//删除指定的读者信息
{
    string id;    int i=0,flag=0;
    In_Reader_File();//从文件中导入读者信息
    char command;
    cout<<"*****删除用户*****"<<endl;
    cout<<"请输入您要删除的读者用户的ID: "<<endl;
    cin>>id;
    for(i=0;i<Count;i++)
    {
        if(reader[i].GetID()==id&&reader[i].GetStatus()==1)
        {
            reader[i].Delete();//删除用户信息
            cout<<"用户"<<reader[i].GetName()<<"所有信息删除完毕！"<<endl;
            flag=1;
        }
        break;
    }
    if(flag==0)
    {
        cout<<"图书管理系统中无该用户!"<<endl;
        cout<<"继续删除请输入“Y”，退出请输入“N”"<<endl;
        cin>>command;
        if(command=='Y')
```

```

        Delete();
    else exit(0);
}
}
void Manage_Reader::All_Clear()//清除全部用户信息
{
    In_Reader_File();
    cout<<"***删除所有用户信息*****"<<endl;
    for(int i=0;i<Count;i++)
    {
        cout<<"正在删除第"<<i+1<<"位用户的信息"<<endl;
        reader[i].Delete();//逐一删除读者信息
    }
    cout<<"所有用户信息删除完毕！"<<endl;
}
void Manage_Reader::ShowOneself()//显示个人信息
{
    string code;
    int i=0,flag=0;
    In_Reader_File();//导入读者信息
    cout<<"请再次输入您的密码： "<<endl;
    cin>>code;
    BF();
    for(i=0;i<Count;i++)
    {
        if(reader[i].GetCode()==code&&reader[i].GetStatus()==1)
        {
            cout<<reader[i];
            reader[i].Display_Keeping();
            flag=1;
        }
        break;
    }
    BF();
    if(flag==0)
    {
        cout<<"对不起您的输入有误！请重新输入！"<<endl;
        ShowOneself();//重新输入密码
    }
}
void Manage_Reader::Borrow()//借书
{
    string code;
    int i=0,flag=0;

```

```
char command;
In_Reader_File();
cout<<"请再次输入您的密码[注意借阅每本书之前都会要输入密码]:
"<<endl;
cin>>code;
BF();
for(i=0;i<Count;i++)
{
    if(reader[i].GetCode()==code)
    {
        reader[i].Borrow();
        flag=1;
    }
    break;
}
BF();
if(flag==0)
    cout<<"对不起您的输入有误！"<<endl;
cout<<"继续借阅书籍？是请输入A;还书？是请输入B;退出请输入N"<<endl;
cin>>command;
if(command=='A' || command=='a')
    Borrow();
else if(command=='B' || command=='b')
    Return();
else exit(0);
}
void Manage_Reader::Return()
{
    string code;
    int i=0,flag=0;
    char command;
    In_Reader_File();
    cout<<"请再次输入您的密码[注意归还每本书之前都会要输入密码]:
"<<endl;
    cin>>code;
    for(i=0;i<Count;i++)
    {
        if(reader[i].GetCode()==code)
        {
            reader[i].Return();
            flag=1;
        }
        break;
    }
}
```



```

    if(flag=0)
        cout<<"对不起您的输入有误！"<<endl;
    cout<<"继续归还书籍？是请输入Y，否请输入N"<<endl;
    cin>>command;
    if(command=='Y' || command=='y')
        Return();//还书
    else exit(0);
}
int Manage_Reader::GetCount()
{
    return Count;
}
class Manage:public Manage_Book,public Manage_Reader//多重继承，继承两个
管理类
{
public:
    Manage(){};//构造函数
    void Login_Reader();//管理读者的登录
    void Login_Manage();//管理管理员的登录
};
void Manage::Login_Reader();//读者登录
{
    int i=0,flag=0;
    string id,code;
    Manage_Reader::In_Reader_File();//从文件导入读者的信息
    cout<<"*****读者登录
"<<"*****"<<endl;
    cout<<"请输入用户ID及密码："<<endl;
    cout<<"****ID:";          cin>>id;
    cout<<"**密码：";  cin>>code;
    for(i=0;i<Count;i++)
    {

        if(Manage_Reader::reader[i].GetCode()==code&&Manage_Reader::reader[i].Get
ID()==id)
        {
            flag=1;
            cout<<"系统登录成功！"<<endl;
            system("cls");
            cout<<"*****
"<<Manage_Reader::reader[i].GetName();
            if(Manage_Reader::reader[i].GetGender()=="男")
                cout<<"先生";
            else cout<<"女士";

```

```

        cout<<"，您好！欢迎使用图书管理系统！
        *****"<<endl;
    }
    break;
}
if(flag==0)
{
    system("cls");
    cout<<"对不起，ID或密码有误!请重新登录："<<endl;
    Login_Reader();//重新登录
}
}
void Manage::Login_Manage()//管理员登录
{
    string code;
    cout<<"请输入管理员密码："<<endl;
    cin>>code;
    if(code=="LIUQIAN")//默认密码为LIUQIAN
    {
        cout<<"系统登录成功！"<<endl;
        system("cls");
        cout<<"管理员您好！欢迎使用图书管理系统！"<<endl;
    }
    else
    {
        system("cls");
        cout<<"密码不正确！请重新登录！"<<endl;
        Login_Manage();//重新登录
    }
}

//所有功能的实现
#include"Class.h"
using namespace std;
void Start_exe()//登陆界面
{
    for(int i=0;i<76;i++)
        cout<<"*";
    cout<<endl;
    cout<<"*****欢迎使用图书管理系统

```

```

*****"<<endl;
    cout<<"***** 读者登录请输入“R”，管理员登录请输入
“M”*****"<<endl;
    for(int i=0;i<76;i++)
        cout<<"*";
    cout<<endl;
}
void Start_Reader()//读者登录系统
{
    Manage A;
    A.Login_Reader();
    Manage_Book MB;
    Manage_Reader MR;
    int command;
    cout<<"***** 欢迎使用图书管理系统
*****"<<endl;
    cout<<"***** 菜单
*****"<<endl;
    cout<<"***** 1. 查询书籍
*****"<<endl;
    cout<<"***** 2. 个人信息
*****"<<endl;
    cout<<"***** 3. 借阅书籍
*****"<<endl;
    cout<<"***** 4. 归还书籍
*****"<<endl;
    cout<<"***** 5. 退出系统
*****"<<endl;
    cout<<"***请输入您的选项: ";
    cin>>command;
    switch(command)
    {
    case 1:
        MB.Search();break;//查询功能
    case 2:
        MR.ShowOneself(); break;//查看个人信息
    case 3:
        MR.Borrow();break;//借书功能
    case 4:
        MR.Return();break;//还书功能
    case 5:
        exit(0);break;
    default:
        system("cls");

```

```

        Start_Reader();//重新进入读者登录系统
        cout<<"请重新输入选项: "<<endl;
    }
}
void Start_Manage();//管理员登录
{
    Manage A;
    A.Login_Manage();
    Manage_Book MB;
    Manage_Reader MR;
    int command,command2;
    cout<<"*****"
*****"<<endl;
    cout<<"*****"
*****"<<endl;
    cout<<"*****"
*****"<<endl;
    cout<<"****请输入选项: ";
    cin>>command;
    if(command==1)//图书管理
    {
        cout<<"*****"
*****"<<endl;
        cout<<"*****"
*****"<<endl;
        cout<<"*****"
*****"<<endl;
        cout<<"*****"
*****"<<endl;
        cout<<"*****"
*****"<<endl;
        cout<<"*****"
*****"<<endl;
        cout<<"*****"
*****"<<endl;
        cout<<"*****"
*****"<<endl;
        cout<<"****请输入选项:";
        cin>>command2;
        if(command2==1)
            MB.Search();//查询功能
        else if(command2==2)
            MB.Add();//增加图书
        else if(command2==3)
            MB.Delete();//删除图书
        else if(command2==4)

```

菜单

1.图书管理

2.读者管理

欢迎使用图书管理模块

菜单

1.查询图书

2.添加图书

3.删除图书

4.全部图书

5.清除所有图书

```

        MB.Display();//显示全部书籍
    else if(command2==5)
        MB.All_Clear();//清除全部书籍
    else
    {
        system("cls");
        cout<<"对不起，你的输入有误:"<<endl;
        Start_Manage();
    }
}
else if(command=2)//读者（用户管理）
{
    cout<<"***** 欢迎使用读者管理模块
*****"<<endl;
    cout<<"***** 菜单
*****"<<endl;
    cout<<"***** 1.查询读者
*****"<<endl;
    cout<<"***** 2.添加读者
*****"<<endl;
    cout<<"***** 3.删除读者
*****"<<endl;
    cout<<"***** 4.显示全部读者
*****"<<endl;
    cout<<"***** 5.清除全部读者
*****"<<endl;
    cout<<"*****请输入选项："<<endl;
    cin>>command2;
    if(command2==1)
        MR.Search();//查询读者
    else if(command2==2)
        MR.Add();//添加读者
    else if(command2==3)
        MR.Delete();//删除读者
    else if(command2==4)
        MR.Display();//显示全部读者
    else if(command2==5)
        MR.All_Clear();//清除全部读者
    else
    {
        system("cls");
        cout<<"对不起，你的输入有误:"<<endl;
        Start_Manage();//管理员登录
    }
}

```

```

    }
    else
    {
        system("cls");
        cout<<"对不起，你的输入有误:"<<endl;
        Start_Manage();//管理员登录
    }
}
int main()
{
    system("color 0B");
    char c;
    Start_exe();//登陆界面
    cout<<"****请输入: ";
    cin>>c;
    if(c=='R' || c=='r')
        Start_Reader();//读者的功能
    else if(c=='M' || c=='m')
        Start_Manage();//管理员的功能
    else
    {
        system("cls");
        main();
        cout<<"对不起您的输入有误！"<<endl;
    }
}

```

数据文件（测试用）：

E:\\book.txt

世界第一好懂的哲学课 [日]小川仁志 中信出版社 哲学 978-7-5086-4937-5
35.00 元 2015 02

政治哲学 [美]史密斯 北京联合出版公司 哲学 978-7-5502-3895-5 39.80 元 2015
02

宋代经学哲学研究 王心竹 上海科技出版社 哲学 978-7-5439-6389-4 48.00 元
2015 02

中国教育黄皮书 周洪宇 湖北教育出版社 教育 978-7-5351-9910-2 62.00 元
2015 02

数学教育初论 江春莲 科学出版社 教育 978-7-03-041790-9 128.00 元 2015 01

环境艺术设计概论 王烨 中国电力出版社 艺术 978-7-5123-6532-2 59.00 元
2014 12

思考的艺术 [英]Buzan&Tony 化学工业出版社 艺术 978-7-122-22298-5 45.00 元
2014 11

中国树木盆景艺术 黄映泉 安徽科学技术出版社 艺术 978-7-5337-6473-9 85.00 元 2014 12

自然形态的城市设计 苏毅 东南大学出版社 自然 978-7-5641-5320-5 45.00 元 2015 2

自然的历史 [美]Tom • Baione 重庆大学出版社 自然 978-7-5624-8116-4 78.00 元 2014 02

日食与自然灾害 赵得秀 西北工业大学出版社 自然 978-7-5612-4235-3 20.00 元 2015 02

自然与道德 史军 北京科学出版社 自然 978-7-03-041868-5 69.00 元 2014 06

社会思潮怎么看 王燕文 江苏人民出版社 社会 978-7-214-13522-3 48.00 元 2014 11

社会调查与统计 董海军 武汉大学出版社 社会 978-7-307-15088-1 40.00 元 2015 01

世界社会主义五百年 中共宣传部 党建读物出版社 社会 978-7-5099-0489-3 29.00 元 2013 08

过程控制技术及其应用 徐湘元 清华大学出版社 技术 978-7-322-37268-4 39.00 元 2013 12

厌氧生物技术 王凯军 化学工业出版社 技术 978-7-122-12538-5 98.00 元 2015 04

京东技术解密 京东研发体系 电子工业出版社 技术 978-7-121-24691-3 65.00 元 2011 12

E:\\reader.txt

张三 2014011216 男 235862

李四 2014011211 男 sfc532

马武 2014011125 男 162as2

何柳 2014011212 女 645210

项少玲 2014012153 女 8sda25

汪廷栋 2014011266 男 821ssz

闫平 2014001535 女 895ad2

师素芳 2014678632 女 1a2zx3

安明玉 2014263236 男 584966

附录：评分表

课程名称： 计算机程序设计基础 2

项 目	评 价	
设计方案的合理性与创新性	3×2	
设计与调试结果	4×2	
设计说明书的质量	1×2	
程序基本要求涵盖情况	4×2	
程序代码编写素养情况	2×2	
课程设计周表现情况	1×2	
综合成绩	15×2	

教师签名： _____

日 期： _____