

图书馆管理系统

一. 功能简介

1. 主界面:

- a. 管理员
- b. 借书者
- c. 退出

(1). 管理员界面:

- a. 书籍管理
- b. 信息查询
- c. 修改密码
- d. 返回

a. 书籍管理界面:

- 1. 增加标题
- 2. 删除标题
- 3. 增加书目
- 4. 删除书目
- 返回

b. 信息查询界面:

- 5. 查看全部标题
- 6. 查看全部书目
- 7. 查看全部借书人
- 8. 查看全部借书信息
- 9. 查看全部预定信息
- 10. 查看某书籍书目信息
- 11. 查看某书籍借阅信息
- 12. 查看某借书人信息
- 13. 查看某书目信息
- 返回

c. 修改密码界面

(2). 借书者界面:

- a. 账号管理
- b. 图书借阅
- c. 返回

a. 账号管理界面:

- 14. 注册账号
- 15. 注销账号
- 16. 帐号信息

17. 修改密码

18. 缴纳罚金

返回

b. 图书借阅界面:

19. 查看全部书籍

20. 查找书籍

21. 借书

22. 还书

23. 预定书籍

24. 取消预定

返回

二. 具体过程

共有6个类: CTitle, CItem, CBorrower, CLoan, CReservation, 下面分别用A, B, C, D, E代表
还有一个类 date

共有6个文件: file1~file6, 分别存放前五个类信息和管理员密码

共有4个动态数组

A, B, C, D, E类各有一个链表, 链表有3或4个函数: 添加节点, 删除节点, 查找节点

注: 进入管理员界面需输入密码

功能 1: 输入信息——>判断标题原来不存在——>添加节点——>添加成功

功能 2: 输入书名——>查找标题——>删除包含的书目(B 删除节点)——>删除相关预(C
删除预定, E 删除节点)——>A 删除节点(无外借书时执行此功能)

功能 3: 输入书名——>判断标题存在——>A 添加书目——>B 添加节点

功能 4: 输入书目标号——>判断书目存在——>A 删除书目——>B 删除节点

功能 5: 显示 A 链表信息

功能 6: 显示 B 链表信息

功能 7: 显示 C 链表信息

功能 8: 显示 D 链表信息

功能 9: 显示 E 链表信息

功能 10: 输入书名——>显示某标题及所含书目信息

功能 11: 输入书名——>显示某标题及相关借书, 预定信息

功能 12: 输入借书者编号——>显示某借书人及相关借书, 预定信息

功能 13: 输入书目编号——>显示某书目具体信息(借出可显示借书者编号, 姓名)

功能 14: 输入信息——>设置密码——>C 添加节点

功能 15: 输入姓名, 密码——>判断无罚金, 无借书——>A 删除预定——>E 删除节点——>C 删除预定——>C 删除节点——>注销成功

功能 16: 输入姓名, 密码——>判断有无罚金(有则提示)——>显示基本信息及相关借书, 预定信息

功能 17: 输入姓名, 密码——>判断有无罚金(有则提示)——>输入新密码——>修改成功

功能 18: 输入姓名, 密码——>判断有无罚金(无则退出, 有则缴纳)

功能 19: 同功能 5

功能 20: 输入书名——>判断书籍存在——>显示标题及所含书目信息(包含书目是否借出信息)

功能 21: 输入书目编号——>判断书目存在且未借出——>输入姓名, 密码——>判断有无罚金(有则无法借书)——>B 标记借出——>C 添加借书——>D 添加节点——>借书成功

功能 22: 输入书目编号——>判断书目存在且借出——>B 标记未借出——>C 添加删除——>D 删除节点——>还书成功(不需输入姓名, 密码)

功能 23: 输入书名——>判断标题存在——>输入姓名, 密码——>判断有无罚金(有则无法预定)——>A 添加预定——>C 添加预定——>E 添加节点——>预定成功

功能 24: 输入预定编号——>判断预定存在——>A 删除预定——>C 删除预定——>E 删除节点——>删除预定成功(不需输入姓名, 密码)

三. 特色

1. 使用链表和动态数组

2. 使用命令 `system("color ab")`, `system("pause")`, `system("cls")`

3. 使用函数 `date date::gettime()`

{

{

```

        time_t t;
        struct tm *local;
        t=time(NULL);
        local=localtime(&t);
        date p(local->tm_year+1900, local->tm_mon+1, local->tm_mday);
        return p;
    }

```

获取当前时间

4. 重载运算符 << 输出类 date 对象