

《程序设计技能实训 I 》

实训任务书

一、课程目标

1. 掌握线性链表的基本操作：包括创建、插入、删除、修改、遍历等
2. 理解文件操作：实现数据的持久化存储与读取
3. 培养综合编程能力：将 C 语言基础知识整合应用于实际项目开发
4. 强化调试技能：学会分析解决程序运行中的错误
5. 培养规范化编程习惯：包括代码注释、模块化设计等

二、评分标准

实训任务	必做/选做	分值	评分标准
学生成绩管理系统	必做	80 分	程序无词法和语法错误，能够正常运行。 按如下规则评分： 1.完成全部基本功能，得 50 分。每缺少一项基本功能，扣 10 分。扣完 50 分为止。 2.每增加一项基本功能以外的扩展功能，加 5 分。最多加 20 分。 3.代码质量。程序结构清晰、命名规范、注释充分、操作界面友好得 10 分。否则酌情扣分。
贪吃蛇	选做	20 分	
飞翔的小鸟 Flappy bird	选做	20 分	
五子棋 Gobang	选做	20 分	
走迷宫 Maze-master	选做	20 分	
推箱子 sokoban	选做	20 分	

三、实训进度安排

日期	任务
day 1	(1) 链表节点定义 (2)菜单 (2)插入学生信息 (4)输出学生信息； (5)修改指定学生信息； (6)删除指定学生信息；
day 2	(1)把学生信息保存到文件； (2)从文件导入学生信息； (3)完成扩展功能
day 3	(1)检查任务 1 完成情况

	(2)图形化项目基础讲解 (3)完成选做项目
day 4	完成选做项目并检查
day 5	未完成的同学继续完成

四、实训内容

(一) 学生成绩管理系统

要求用单链表保存学生信息。学生信息包括学号、姓名和成绩（至少 1 门课程）等。

1、功能需求如下：

- 1) 用户注册和登录
 - 1.添加数据验证（学号唯一性、成绩 0-100 范围等）
 - 2.从文件导入用户信息： userinfo.txt
 - 3.注册： adduser 函数和 encrypt 函数(加密密码)
 - 4.登录 login 函数， checkUserValid 函数： 验证账户信息
- 2) 显示菜单（成功登录后显示，用 displaymenu 函数显示菜单）
 - 1、Creat： 创建学生链表(Create_Stu_Doc()函数)
 - 2、Insert： 插入学生信息
 - 3、Modify： 修改学生信息
 - 4、Delete： 删除学生信息
 - 5、Print： 输出学生信息
 - 6、Load： 从文件导入学生信息
 - 7、Save： 保存学生信息
 - 0、Exit： 退出
- 3) 完善菜单所示基本功能
- 4) 菜单功能及其对应函数与作用

序号	模块名称	函数名	功能说明
1	登录	void login()	运行程序后，显示登录界面
2	加密函数	void encrypt(char *pwd)	用于加密密码
3	用户身份验证	int checkUserValid(sysuser psu)	校验用户信息的合法性,成功返回 1， 否则返回 0
4	注册	void adduser()	注册， 增加用户
5	菜单	void displaymenu()	成功登录后， 显示的菜单
6	创建学生链表	stud_node * Create_Stu_Doc()	新建链表
7	插入	stud_node * InsertDoc(stud_node * head, stud_node *stud)	插入学生信息, 一个一个信息插入
8	修改	stud_node * Modify(stud_node * head, struct stud_node stu);	修改学生信息, 先输入修改后学生的所有信息到结构变量中， 再在链表中查找对应学

			号,如果找到就修改链表对应节点的信息,并将链表保存为文件, 否则显示修改失败
9	删除	stud_node * DeleteDoc(stud_node * head, int num)	输入学号,删除链表中对应的节点
10	遍历	void Print_Stu_Doc(stud_node * head)	输出链表中所有节点的信息
11	保存	void savestudents(stud_node * head, const char *filename)	保存学生信息到文件
12	加载	stud_node * loadstudents(const char *filename,int *count)	从文件中读取数据

2. 扩展功能

- (1)实现多文件系统
 - (2)支持模糊查询（姓名部分匹配）
 - (3)奖学金评定
 - (4)使用二进制文件提高存储效率 //这个要对应保存和导入两个模块
 - (5)其它自定义功能
- 注：可以在菜单栏显示的 需列出来

3. 相关要求

(1)程序结构

链表节点定义示例:

```
struct stud_node{
    int num;
    char name[20];
    int score;
    stud_node *next;
};
```

基本功能函数定义示例:

```
void login();//登录
void encrypt(char *pwd);//加密函数
int checkUserValid(sysuser psu);//用户身份验证
void adduser();//注册, 增加用户
void displaymenu();//显示菜单
stud_node * Create_Stu_Doc(); /* 新建链表 */
stud_node * InsertDoc(stud_node * head, stud_node *stud); /* 插入 */
stud_node * Modify(stud_node * head, struct stud_node stu);/* 修改 */
stud_node * DeleteDoc(stud_node * head, int num); /* 删除 */
void Print_Stu_Doc(stud_node * head); /* 遍历 */
```

```
void savestudents(stud_node * head, const char *filename); //存入文件
stud_node * loadstudents(const char *filename, int *count); //从文件中读取数据
```

(2) 技术要求

- ① 使用带头节点的单链表存储数据
- ② 采用模块化编程（至少包含以下模块）
 - `list.c/h`：链表基本操作
 - `file.c/h`：文件读写操作
 - `ui.c/h`：用户界面与菜单
 - `main.c`：主程序入口

（二）选做项目

选做项目内容详见配套资源。

五、详细评分标准

总分构成：基础功能 50 分 + 扩展优化 20 分 + 代码规范和实训手册 10 分 = 80 分（管理系统）

附加分：小游戏 20 分（可选，计入总分上限 100 分）

5.1 基础功能评分（50 分）

功能模块	分值	考核要点	常见扣分值
1. main 框架 + displaymenu 菜单	4 分	主函数结构完整，菜单显示正常，switch 分支正确	缺少任意菜单项扣 1 分
2. 用户注册与登录	10 分	login 循环、adduser 注册、encrypt 加密、checkUserValid 验证	每个函数 2-3 分，函数缺失扣对应分
3. Create_Stu_Doc	6 分	循环输入创建链表，正确调用 InsertDoc	未复用 InsertDoc 扣 2 分
4. InsertDoc	8 分	按学号升序插入，空链表/头插/中插/尾插正确	头结点更新错误扣 3 分
5. Modify	6 分	按学号查找并修改，修改后保存文件	未保存到文件扣 2 分
6. DeleteDoc	6 分	头结点和非头结点删除，free 释放	内存泄漏扣 2 分
7. Print_Stu_Doc	4 分	空链表提示，格式对齐	无空链表判断扣 1 分
8. Save + Load	6 分	文件打开关闭正确，格式一致	每项 3 分，fclose 缺失扣 1 分

5.2 扩展功能评分（20 分）

扩展功能	分值	考核要点	评分说明
1. 多文件系统	5 分	拆分为 .h 头文件 + 多个 .cpp，结构清晰	未拆分则 0 分；拆分但不合理扣 2 分

2. 模糊查询	5 分	输入姓名关键词， strstr 匹配输出结果	未实现 0 分，仅精确 匹配扣 3 分
3. 奖学金评定	5 分	按成绩分等级（>=90 一等，>=80 二等，>=70 三等）	未实现 0 分，等级划 分不完整扣 2 分
4. 二进制文件	5 分	Save/Load 改 用 fread/fwrite 二进制读 写	未实现 0 分，仅改一 部分扣 2 分

5.3 小游戏评分（附加 20 分）

评分项	分值	考核要点	评分说明
代码完整性	5 分	无 编 译 错 误 ， WinMain 入口正确， 窗口创建成功	有编译错误 0 分
游戏可玩性	5 分	游戏规则完整，交互 流畅，能正常开始和 结束	不能正常运行则 0 分
界面美观度	5 分	图形绘制、颜色搭配、 布局合理	仅字符界面扣 3 分
创新扩展	5 分	计分板、难度等级、 音效等额外功能	视创新程度给分

5.4 代码规范和实训手册（10 分）

项目程序代码规范，有必要的注释 5 分

实训手册按照要求填写 5 分，未交一个手册扣 20 分。