

综合性设计实验 任务书

【实验目的】

- (1) 熟练掌握 C 语言中流程控制、数组、函数、指针、结构体等程序设计的基础知识和调试技术。
- (2) 建立模块化结构程序设计的思想，培养对问题进行分析、综合运用、编程实现的综合能力。理解计算思维的思想和方法，综合运用 C 语言知识分析问题和利用计算机求解问题的综合能力

【实验环境】

- (1) 操作系统
- (2) 开发工具及编程语言

【实验要求】

- (1) 功能完备，实现用户需求。
- (2) 用户界面友好易用。
- (3) 模块化程序设计，规范化编程。
- (4) 合理的数据设计。在应用系统设计中，相对独立的模块间以数据相互连接，使各模块间的耦合性较低，方便系统运行，提高系统安全性。
- (5) 保证信息合法性，规范录入数据，对账号等规范保存。
- (6) 保证实用性，根据用户需求设计各种功能，并能够处理一些特殊情况的要求，此外，尽可能预留空间，以便扩充功能。
- (7) 驼峰命名、合理注释，保证源程序可读性。对系统常量等数据要求规范处理，对于常用的功能和算法如排序做成自己的库函数，使用头文件的方式实现分离编译。
- (8) 对实验结果的处理包括程序测试、结果分析、结论及改进方案等内容。
- (9) 程序实现的正确性、鲁棒性和代码质量。
- (10) 测试用例的覆盖率和测试结果的准确性。
- (11) 实验报告的完整性和质量，要包括程序运行截图或运行结果文件。
- (12) 实验总结的深度和心得体会的真实性。

【提交的设计成果】

- (1) **实验报告**。详细描述用户需求分析、可行性分析、算法设计、数据结构设计、函数设计等，辅以流程图描述。（综合性设计实验报告的具体格式详见模板）
- (2) 调试通过，能够正常运行的**程序文件**（**源代码文件.c**、**数据文件**和**可执行文件.exe**）。要求运行演示，演示时采用答辩形式，由同学边演示边讲解编程设计实现思路。提交的源代码要求具备良好的可读性。

【实验内容】

一、必做题目： 西山湖教务管理系统

● 数据：

(1) 学生自然信息：

学号、姓名、所在专业、毕业否、出生日期、3 门课程的学分(高数 3 学分、数字逻辑 2 学分、离散数学 4 学分)及对应成绩和绩点……

（数据文件预存至少 20 个学生的完备自然信息）。

- 学号：学号是唯一区分学生的关键字。
- 绩点：0 – 5。成绩低于 60 分的绩点为 0，60 分对应绩点 1.0，61 分对应绩点 1.1，……，99 分对应绩点 4.9，100 分对应绩点 5.0。
- 学分：绩点为 0 的对应学分记为 0，绩点为 1.0–5.0 之间的记为该课程学分。

(2) 教师自然信息：

工号、姓名……

(3) 课程自然信息：

课程号、课程名……

试编写程序，提供友好的用户界面，包括菜单选择进入学生用户、教师用户或管理员界面进行各种查询和编辑、以及系统帮助文件和退出等。

● 系统首页：

(1) 系统欢迎首页信息提示、登录功能。

- 欢迎首页（系统名称、版本，以及**系统设计者**等相关信息），提供登录验证功能，账户和密码验证成功则进入各自身份对应的功能模块。
- 用户（教务管理员、教师、学生）账号和密码。基于高校师生员工数据管

理的特殊性，不允许师生用户自由注册，仅由管理员分配账号（教师：6位工号；学生：12位学号）和初始密码，然后该账户可以根据自己初始密码更改新密码。

- 设置新的用户账号和密码，是教务管理员专有功能。

- (2) 提供友好的欢迎界面，用菜单选项分别进入学生用户查询界面、教师用户工作界面和后台教务管理员界面。用户通过键入数字或字母实现菜单功能选择，进入子模块。

● 学生用户模块：

（为学生用户提供友好的功能选择界面，通过键入数字或者字母实现功能选择，进而进入相应查询功能）：

基于个人信息保护，学生用户只能查询本人具体信息和本专业排名最高最低、优秀率等总体信息，不可查询他人具体个人信息。

- (1) 密码验证进入学生查询界面。
- (2) 菜单显示学生信息查询及退出系统等功能，用户通过键入数字或者字母实现功能选择，进而进入相应查询功能。
- (3) 查询学生本人的全部自然信息。
- (4) 查询学生本人的总分在本专业的排名。
- (5) 查询某课程的最高分、最低分和平均分（查询总分最高最低时如果有相同成绩的学生要全部显示）。
- (6) 查询某课程平均成绩以及高于、低于和等于平均成绩的人数。
- (7) 查询某课程成绩为不及格[0~60)、及格[60~70)、中等[70~80)、良好[80~90)、优秀[90~100]的人数和所占百分比。
- (8) 查询学生本人每学期学业预警：低于该学期指定学分的则学业预警，如第一学期这三门课总学分低于7学分则学业预警。
- (9) 其他功能。

● 教师用户模块

（为教师用户提供友好的功能选择界面，通过键入数字或者字母实现功能选择，进而进入相应成绩录入等功能）：

教师用户与教务管理员分别完成不同管理功能，

教师只能在有效期内编辑自己任课课程的学生成绩数据并提交。

【学有余力的同学完成】

- (1) 密码验证进入教师用户界面。
- (2) 菜单显示浏览、增删改及退出系统等功能，用户通过键入数字或者字母实现功能选择，进而进入相应功能
- (3) 查询浏览教师任课的某课程的所有信息。
- (4) 在有效期内编辑自己任课课程的考核计算方式，以及学生成绩数据并提交
- (5) 其他功能。

● 教务管理员模块

（为教务管理员提供友好的功能选择界面，通过键入数字或者字母实现功能选择，进而进入相应管理功能）：

【学有余力的同学完成】

- (1) 密码验证进入管理员界面。
- (2) 菜单显示课程设置和学生教务信息的浏览、增删改及退出系统等功能，用户通过键入数字或者字母实现功能选择，进而进入相应功能
- (3) 浏览某专业全体学生的所有信息及总排名，根据学生的总分进行排序输出。
- (4) 单科课程排序输出。
- (5) 对学生信息的增、删、改。
- (6) 对课程信息及相应教师信息的增、删、改。
- (7) 其他功能。

● 数据测试

对系统设计进行调试、修订、完善、测试。测试各级菜单是否正确、测试数据的读取和存储是否正常、测试退出菜单后各项功能是否正常运行、测试评估界面是否合理，是否友好易用。

【注意：】

- (1) 数据处理考虑文件的读取和写入。
- (2) 数据更新时，保障数据文件的及时刷新保存。

二、附加题目（任选）

目录

1. 银行帐户管理	5
2. 航班售票系统	6
3. 长整数运算	7
4. 计算器	7
5. 发牌	8
6. 时钟模拟	8
7. 俄罗斯方块游戏	9
8. 模拟指法练习程序（统计正确率及输入速度）	9
9. C 语言实现艺术清屏	10
10. 五子棋游戏	10
11. 其他	10

1. 银行帐户管理

要求建立二进制文件存储银行帐户信息，其中每个用户帐户信息中要求保存帐号、用户身份证号码、用户姓名、用户地址、帐户金额等，完成以下功能：

- 录入新帐户；
- 查询帐户情况，根据输入的帐号查询用户情况和帐户金额；
- 修改帐户信息，要求用户输入帐号，根据用户需要修改除了帐号之外的其余信息；
- 删除帐户：根据输入的帐号找到要删除的帐号信息以后，经确认后删除该帐号信息；

提示：程序中定义用户帐户结构体：

```
struct account
{
    char accounted[10];
    char customerid[30];
    char name[20];
    char address[30];
    float balance;
};
```

其中存储用户帐户信息的二进制文件中每个数据类型应为 struct account，在操作之前可以考虑先将文件中的用户帐户信息读出并建立成一个链表，然后进行查询、删除等操作，可节省读写文件的时间；当录入新帐户信息、修改帐户信息之后，同时更新相应文件的内容；

2. 航班售票系统

航班售票系统，要求完成以下功能：

- 假设该民航有若干航班，要求实现增加航班和取消某个航班的功能。
- 查询：根据用户提出的终到站查询航班号、售票情况等航班信息；也可根据航班号，列出该航班已订票的乘客名单。
- 订票：根据航班号为客户订票，如该航班有余票，则为客户订票；如该航班已满员，则显示相应信息。
- 退票：按乘客要求退出已预定的机票。

提示：

数据结构：可以使用链表表示航线表，对每个航班应包括以下信息：航班号、到达港、总座位数、余票额、乘客名单等；其中乘客名单应为一个单链表，每个乘客的信息有：乘客姓名、证件号码、座位号等，为方便查找，可考虑按乘客姓名排序；相应数据类型示范如下：

```
struct cnode{
char name[20]; //乘客姓名
char idcard[30]; //证件号码
int seatno; //座位号
struct cnode *clink; //下一乘客指针
};
struct pnode{
int no; //航班号
char destin[30]; //到达站
int limit,rest; //总座位数、余票额
struct cnode *clist; //乘客名单
struct pnode *plink; //下一航班指针
};
```

3. 长整数运算

C 中的 long int 所能表示的数据范围有限，现要求编程完成超过 long int 所能表示的数据范围以上的十进制正的长整数的加法和乘法运算；

提示：两个参与运算的长整数可用 char a[256], b[256] 表示，整个程序中要求完成以下几个函数的编写：

- int readlongint(char *x); 此函数用于读入一个正的长整数到 x 中，函数返回长整数的实际长度；要求输入时检查所读入的字符串是否是合法的长整数，如不是提示用户直到输入合法的长整数为止；
- char *addition(char *x, char *y); 此函数用于实现用字符串表示的长整数的加法运算，其计算结果保存在一个动态申请的字符数组空间（其长度为参与加法运算的两个长整数中较长的一个的长度加 1）中，函数返回该数组的指针。
- char *multiplacation(char *x, char *y); 此函数用于实现用字符串表示的两个长整数的乘法运算，可考虑先将乘数的从个位到最高位的每一位与被乘数相乘（这一步可利用 addition 函数实现），再进行向左偏移后相加完成；注意：此程序设计最关键的问题是对字符数组的下标定位和动态申请恰当的内存空间以保存计算结果，注意在乘法运算中回收不再使用的内存空间。

4. 计算器

编程实现计算器程序，完成实数的加、减、乘、除运算。注意运算符优先级别。

表达式要求采用中缀形式，例如：2.3+7.2*7

提示：表达式处理可参考第十章“逆波兰表达式”范例。按 END 键退出算术计算器程序。

检测是否有键按下，需要调用 kbhit() 库函数。

kbhit 的原数原型：int kbhit(void);

kbhit 函数功能：检测是否有键按下，如果有，则返回非 0 值（即真），否则返回 0（即假）。调用 kbhit() 函数的源程序必须包含 conio.h 文件。

具有延时功能的函数是 delay(unsigned milliseconds)。包含在 dos.h 头文件中。

Delay() 函数的功能是：将程序挂起 milliseconds 毫秒，即程序暂停或延时 milliseconds 毫秒。

当 kbhit 函数返回非 0 时，还要检测是否是 End 键。这需要调用 bioskey 库函数。但参数必须是 1，不能是 0。因为 bioskey(1) 虽然返回用户所按键值，但没有接受键盘输入，不会影响其它函数（如 scanf 函数）接受有效输入。

END 的键值定义为：#define END 0X4F00

bioskey(1) 判断是否是 END 的用法：

```
if (bioskey(1) == END)
```

```
{ 相关操作 }
```

5. 发牌

一副扑克牌 52 张共 4 种花色，用户可选择“1——发牌”将这 52 张牌随机发成 4 墩；或选择“0——退出”退出程序。

提示：用一个具有 52 个元素的数组（如 `char p[52][5]`）存放这 52 张牌的内容，如 `P_i` 表示黑桃、`H_i` 表示红桃、`C_i` 表示梅花、`D_i` 表示方块；程序中可以先初始化该数组，然后显示菜单等待用户选择输入，用户选择“发牌”，程序调用“发牌”函数；在发牌函数中，可使用 TC 中的产生随机数的函数（参看 `rand()`, `srand()`, `random()` 等函数的使用说明）一个随机数 `m`，选取一张牌 `{p[m]}`，将其与最前面的牌交换位置，即将选取的牌放在最前面；然后对数组 `p` 余下的牌重复选取操作，直到所有牌均重排列；最后按 13 行*4 列输出发牌结果。

6. 时钟模拟

图形化界面，屏幕上显示圆形的模拟时钟表盘，显示 12 个刻度。时钟上有秒针、分针和时针指示，随着时间推移，秒针、分针和时针在表盘上移动。

提示：在 `dos.h` 头文件中定义有如下结构类型

```
struct time{
unsigned char  ti_min; // Minutes
unsigned char  ti_hour; // Hours
unsigned char  ti_hund; // Hundredths of seconds
unsigned char  ti_sec; // Seconds
};
```

可直接利用该类型表示时间类型。

在 `dos.h` 头文件中定义库函数 `gettime(struct time *)`，该函数返回系统时钟。此程序可通过读取系统时钟调整秒针、分针和时针位置。

7. 俄罗斯方块游戏

用 C 语言实现常见的游戏-俄罗斯方块。应包含初始化新游戏，累积方块，消除方块，方块移动与变形，计分和调速等功能，方块界面和容器都可以自己设计。提示：可用以下数据结构记录一个点和一个方块形状：

```
typedef struct
{
    int x,y;
}Point;

typedef struct
{
    Point a;
    Point b;
    Point c;
    Point d;
    int FkColor;
}FkType; //方块类型
```

然后用一个 $m*n$ 的长方形区域作为容器。通过 C 的图形函数库画出界面。采用延时函数使方块自动下降，并采取相应算法实现方块叠加、消除整行，计分的功能。

8. 模拟指法练习程序（统计正确率及输入速度）

模拟指法测试程序：完成输入 N 行文字后，统计输入的正确率和输入速度。

要求：

程序运行后，屏幕首先输出提示信息，提示进入测试系统；先从键盘输入原文，自己定义原文最大长度；原文输入完成后，给出提示信息开始测试；依照原文从键盘输入，程序中统计输入的正确率和输入速度。输入完成后，屏幕显示最后统计结果。

9. C 语言实现艺术清屏

我们在编制程序时,经常要用到清屏处理(清除屏幕上的内容),如 dos 下的 `cls`, Turbo C 下的 `clrscr()` 等都具有清屏功能,但这些均为一般意义的清屏,并未显示其清屏规律。但有时为了达到清屏的艺术美观,往往对清屏有一些具体要求,如:开幕清屏;闭幕清屏;上清屏;下清屏;中清屏。为此,要求用 C 语言编制了几个子函数,用于程序中时,既可达到清屏的目的,有能增加屏幕的艺术美观。

提示:程序运行后,屏幕上输出满屏字符用于演示程序;主函数中调用几个清屏子函数,要求给出 5 个子函数:中心清屏(从中间向两边),矩形边清屏(从两边向中间),开屏式清屏(类似于展开扇形),闭幕式清屏(扇形收拢),自下清屏。调用一个子函数后,要恢复满屏再调用另一个,要求中间稍有延时。

10. 五子棋游戏

图形化界面,玩家可以输入 F1-F5 选择人机对战模式,双人对战模式,保存当前游戏信息,读取游戏信息以及退出游戏选项。程序画出棋盘,可以通过键盘选择位置,进行下子。如果是人机对战模式,在玩家下完子后,电脑自动算出位置进行下子。当某一方到五子连成一线,能提示玩家某方获胜。

提示:棋盘位置行列等信息可以定义成常量。下子可以数组表示。首先编写相应程序画出棋盘棋子等。然后实现响应键盘输入,定位下子。最后实现将棋盘上的棋子位置信息写入文件以及从文件中读出。

11. 其他