**浙江树人学院**

**《程序设计基础(C)》**

(课程设计指导书)

**2025年3月**

**信息科技学院 计算机教研室**

**生产实习(课程设计)任务书**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 课题或任务 | | | 程序设计基础（C） | 周次 | 第3周 |
| 专业 | | | 计算机科学 | 班级 | 2024级计算机专业班级 |
| 指导教师 | | | 朱斌、洪峰、徐萍、华冠萍、赵丽、李君伟 | | |
| **实习任务与目的** | 通过本次实训，掌握C语言的主要知识点：控制语句、数组、函数、结构体、文件等，掌握C程序的结构化程序设计方法，能使用C语言开发简单的应用程序。 | | | | |
| 实习时间、地点 | 2025年3月3 日－2025年3 月 7日  计算机241\242班，20-507  计算机243\244班，20-606 | | | | |
| **实习具体内容** | （题目一）图书馆借阅系统：图书入库、借书、还书、遗失赔偿等功能  图书基本信息：书名、索引号、作者、出版社、单价、库存数、借出数……  学生基本信息：学号、姓名、电话、借书书名、借书索引号、借书日期……  用户：？？  （题目二）电影购票系统：影片入库管理、用户浏览、购票、退票等功能  影片信息：片名、片号、导演、时长、主演、票价、时间、房间、座位号……  购票信息：用户名、电话、座位号…..  用户：？？  （题目三）外卖点餐服务系统：菜单入库、菜单修改、菜单浏览、加入购物车、下单、模拟支付、营业额分析  菜单信息：菜名、菜号、类别、单价、总数量  订单信息：订单号、客户电话、地址、菜名、份数、总价…..  用户：？？  （题目四）酒店预定服务系统：酒店基本信息入库与编辑、酒店预定、入住、模拟结帐、营业客分析  （题目五）其他题目可与指导老师协商自拟 | | | | |
| **实习任务安排** | 基本任务 | 1. 选题，每人选择一个题目完成，一人一题，独立完成。 2. 理解题意，明确设计目标与任务，功能模块中务必包括5个基本模块：数据的增加、删除、修改、查询、排序，文件存储做为附加功能**。** 3. 用C语言实现课题设计、编码、调试、运行、验收。 4. 指导老师检查、验收设计结果，并提问。 | | | |
| 扩展任务 | 1、软件开发：了解软件开发思想和基本方法，规范设计过程和文档  2、菜单显示：能利用C语言的图形知识，设计得更漂亮、便捷  3、表格显示：为所显示的记录内容加上表格，表格线条利用画线等功能实现 | | | |
| **实习要求** | * 1. 明确设计任务，能口述并文字描述清楚自己的题目要求、期望结果；   2. 清楚设计过程中的主要知识运用，不清楚的及时与指导老师沟通；   3. 能进行总体设计，即清楚课题中的数据与文件设计、功能设计、函数模块设计；   4. 能对每个模块即函数进行详细设计编码；   5. 能对所有的函数模块进行联调；   6. 编码仔细，风格统一，尽量减少代码中的语法错误；   7. 调试数据多样，注意特殊情况和边界数据处理，尽量减少代码的逻辑错误； | | | | |
| **实习(设计)报告编写要求** | 1. 设计任务书 2. 总体设计（数据、文件设计，　功能设计，　函数模块设计） 3. 详细设计（每个模块功能说明，主要流程图，函数构成） 4. 调试与测试（描述几个典型模块的调试数据与测试方案及结果） 5. 部分主要代码（主要功能模块代码、注释、运行结果） 6. 设计体会与建议 7. 参考文献 | | | | |
| **成绩评定** | 由指导老师检查以上任务的完成情况，每天记录考勤，最后进行现场检查与问答情况，并批改实习报告。  成绩评定：最终系统60分（增加、删除、修改、查询、排序模块得分各占18%，其他个性化功能得分占10%） + 平时40分（实训报告内容完备性60%、格式规范性30%、创新性10%）。 | | | | |

1. **明确设计任务**
2. **目的**
   * + 1. 熟悉系统的组成及各个功能模块的作用
       2. 掌握C程序的组成
       3. 掌握主函数的功能
       4. 掌握菜单的显示方法
       5. 掌握菜单的处理方法
       6. 掌握函数的调试方法
3. **内容**
   * + 1. 由指导老师分配题目、分析题义
       2. 指导老师讲解文件函数的用法；
       3. 与指导老师讨论，进一步明确题意，分析选题的主要知识点和算法
       4. 学生根据分析，复习知识点：switch语句、循环语句、结构体、函数、结构体数组
       5. 编写主界面
4. **指导**
   * + 1. **实验环境**

可用C-Free或VC++或VS.net

* + - 1. **复习知识点：**

**（1）Switch语句、循环、函数、结构体、结构体数组、文件**

**（2）**熟练掌握数组元素：增、删、改、查、排序、统计；

**第二天 总体设计**

**目的**

* + - 1. 基本完成总体设计：数据文件设计、功能设计、模块设计；
      2. 编程设计主界面，若程序无法实现，需用草图画出；

**内容**

* + - * 1. 分析选题，描述功能模块、设计主界面；
        2. 定义主要数据、设计函数模块及函数间的调用关系。

**指导**

1. “学籍管理系统”的功能模块图参考；

学籍管理系统

删除学籍信息

修改学籍信息

添加学籍信息

查询学籍信息

统计学籍信息

排序学籍信息

按学号删除

按姓名删除

查询全部

按姓名查询

按生日查询

按地址查询

统计同年出生的

统计同一专业的

按专业名称排名

按年龄大小排名

1. “成绩管理系统”的功能模块图参考；

成绩管理系统

删除成绩信息

修改成绩信息

添加成绩信息

查询成绩信息

统计成绩信息

排序成绩信息

按学号删除

按课程名删除

查询全部

按姓名查询

按课程名查询

按学号查询

统计某课程的平均成绩

统计某学生的平均成绩

按总成绩高低排名

按平均值大小将课程排名

1. 定义结构体、结构体数组；

**typedef struct //日期结构体**

**{**

**int year;**

**int month;**

**int day;**

**} Date;**

**typedef struct //学籍结构体**

**{**

**char xh[20];//学号**

**char xm[10];//姓名**

**char lxdh[12]; //联系电话**

**char jtdz[30]; //家庭地址**

**Date birthday; //生日**

**char zy[15]; //专业**

**} XJ;**

**XJ xsxj[100]; //用XJ定义一个数组存放成绩；**

**typedef struct //成绩结构体**

**{**

**char xh[20]; //学号**

**char xm[10]; //姓名**

**Date birthday; //出生日期**

**char kcmc[15]; //课程名称**

**int cj; //成绩**

**} CJ;**

**CJ xscj[100]; //用结构体CJ定义一个数组存放成绩**

**第三天 详细设计**

**(一)　目的**

分解功能模块，分析函数间的调用流程；

编写各个函数代码，逐个实现功能模块；

**(二)　内容**

编写各个处理函数模块的代码；

**(三)　指导**

1. “学籍管理系统”完成模块：添加学籍信息、删除学籍信息、修改学籍信息；
2. “成绩管理系统”完成模块：添加成绩信息、删除成绩信息、修改成绩信息；

**第四天 详细设计**

**(一)　目的**

1、分解功能模块，分析函数间的调用流程；

2、编写各个函数代码，逐个实现功能模块；

**(二)　内容**

1. 结构体定义，结构体数组定义；

2. 完成各个处理函数模块的代码实现；

**(三)　指导**

1. “学籍管理系统”完成模块：查询学籍信息、统计学籍信息、排序学籍信息；
2. “成绩管理系统”完成模块：查询成绩信息、统计成绩信息、排序成绩信息；
3. **测试联调，检验评分**
4. **目的**
   * + 1. 为程序增加文件存储功能；
       2. 将所有程序模块进行联合调试
       3. 消除程序中存在的测试错误及改进程序的不足
       4. 学生演示并答疑，老师检查结果并评分
5. **内容**
   * + 1. 使用单步和断点调试方法发现函数中存在的错误。
       2. 使程序符合设计要求，达到对应的设计效果。
6. **指导**
   * + 1. 常用结构体数据的文件读写函数：fread( )、fwrite( )。
       2. 实现系统中的文件操作接口。

**附：部分参考代码**

　　“学籍管理系统”和“成绩管理系统”可以由以下函数模块组成，分别介绍如下：

|  |  |
| --- | --- |
| 函数名（功能模块）或其他 | 功能 |
| main() | 主函数 |
| menu\_select() | 菜单选择 |
| handle\_menu() | 菜单处理 |
| s\_read() | 读取记录 |
| s\_save() | 保存记录 |
| xhexist() | 判断某学号是否存在 |
| s\_add() | 添加记录 |
| modi\_xh() | 按学号修改记录 |
| modi\_xm() | 按姓名修改记录 |
| del\_xh() | 按学号删除记录 |
| del\_xm() | 按姓名删除记录 |
| sear\_all() | 查询所有记录 |
| sear\_xm() | 按姓名查询记录 |
| sear\_xh() | 按学号查询记录 |
| sear\_sr() | 按生日查询记录 |
| sear\_jtdz() | 按家庭地址查询记录 |
| sort\_xh() | 按学号排序 |
| sort\_cj() | 按成绩排序 |
| count1() | 统计？？？ |
| 其它函数 |  |

总体设计思想：为了简化程序，对所有记录操作通过结构体数组完成，当程序开始执行时把数据文件一次性读入到结构体数组中，以后对数据的操作都通过结构体数组来完成。“添加、修改、删除”时进行写文件操作，“查找、排序”操作进行读文件。

**（一）主控模块**

#include <stdio.h>

#include <stdlib.h>

#include <conio.h>

#include <ctype.h>

#include <string.h>

typedef struct //日期结构体

{

int year;

int month;

int day;

}Date;

struct XJ //学籍结构体

{

char xh[20];//学号

char xm[10];//姓名

char lxdh[12]; //联系电话

char jtdz[30]; //家庭地址

Date birthday; //生日

char zy[15]; //专业

};

#define LEN sizeof(struct XJ)

struct XJ stud[100]; //用XJ定义一个数组存放学籍

int reccount=0; /\*定义全局变量存放当前记录、记录个数\*/

int menu\_select(); /\*函数声明：菜单选择\*/

void handle\_menu(); /\*函数声明：菜单处理\*/

void s\_read( ); /\*函数声明：读记录\*/

void s\_save( ); /\*函数声明：保存记录\*/

int xhexist(char num[]); /\*函数声明：判断学号是否存在\*/

void s\_add( ); /\*函数声明：追加记录\*/

void sear\_xh( ); /\*函数声明：按学号查询\*/

void sear\_xm( ); /\*函数声明：按姓名查询\*/

void sear\_all( ); /\*函数声明：查询所有记录\*/

void sear\_sr( ); /\*函数声明：按输入的出生年、月或日期查询记录\*/

void sear\_jtdz( ); /\*函数声明：按输入的地址查询记录\*/

void modi\_xh( ); /\*函数声明：按学号修改\*/

void modi\_xm( ); /\*函数声明：按姓名修改\*/

void del\_xh( ); /\*函数声明：按学号删除\*/

void del\_xm( ); /\*函数声明：按姓名删除\*/

void sort\_xh( ); /\*函数声明：按学号排序\*/

void sort\_cj( ); /\*函数声明：按成绩排序\*/

void main() /\*程序入口

{

// s\_read( ); /\*调用读记录函数\*/

handle\_menu(); /\*调用菜单处理函数\*/

}

/\*以下定义菜单处理函数\*/

void handle\_menu()

{

int sel,flag;

while(1)

{

switch(menu\_select()) /\*用来调用菜单选择函数\*/

{

case 0:

printf("exit,bye!\n");

// s\_save( ); //退出前，调用一次函数将数组中的数组保存至文件

exit(0);

case 1 :

s\_add( ); /\*调用追加记录函数\*/

break;

case 2:

flag=1;

while (flag)

{

/\*以下显示修改记录子菜单\*/

printf("\n [修改记录子菜单]\n\n");

printf(" +------------------------+\n");

printf(" | 1. 按学号修改 |\n");

printf(" | 2. 按姓名修改 |\n");

printf(" | 0. 返回 |\n");

printf(" +------------------------+\n");

printf(" 请选择（ 0~2:");

scanf("%d",&sel);

switch(sel)

{

case 0:

flag=0; break;

case 1:

//modi\_xh( );

break;

case 2:

//modi\_xm( );

break;

}

}

break;

case 3:

flag=1;

while (flag)

{

/\*以下显示删除记录子菜单\*/

printf("\n [删除记录子菜单]\n\n");

printf(" +------------------------+\n");

printf(" | 1. 按学号删除 |\n");

printf(" | 2. 按姓名删除 |\n");

printf(" | 0. 返回 |\n");

printf(" +------------------------+\n");

printf(" 请选择 0~2:");

scanf("%d",&sel);

switch(sel)

{

case 0: flag=0; break;

// case 1: del\_xh( ); break;

// case 2: del\_xm( ); break;

}

}

break;

case 4:

flag=1;

while (flag)

{

/\*以下显示查询子菜单\*/

printf("\n [查询记录子菜单]\n\n");

printf(" +------------------------+\n");

printf(" | 1. 按学号查询 |\n");

printf(" | 2. 按姓名查询 |\n");

printf(" | 3. 查询全部 |\n");

printf(" | 0. 返回 |\n");

printf(" +------------------------+\n");

printf(" 请选择( 0~3:");

scanf("%d",&sel);

switch(sel)

{

case 0: flag=0; break;

// case 1: sear\_xh( ); break;

// case 2: sear\_xm( ); break;

case 3:sear\_all( ); break;

}

}

break;

case 5:

flag=1;

while (flag)

{

/\*以下显示排序记录子菜单\*/

printf("\n [排序记录子菜单]\n\n");

printf(" +------------------------+\n");

printf(" | 1. 按学号排序 |\n");

printf(" | 2. 按成绩排序 |\n");

printf(" | 0. 返回 |\n");

printf(" +------------------------+\n");

printf(" 请选择 0~2:");

scanf("%d",&sel);

switch(sel)

{

case 0: flag=0; break;

// case 1: sort\_xh( ); break;

// case 2: sort\_cj( ); break;

}

}

break;

}

}

}

/\*以下定义选择菜单函数\*/

int menu\_select()

{

char s[2];

int cn;

/\*以下显示主菜单\*/

printf("\n\n [主菜单]\n");

printf("\n");

printf(" +--------------------+\n");

printf(" | 1-追加记录 |\n");

printf(" | 2-修改记录 |\n");

printf(" | 3-删除记录 |\n");

printf(" | 4-查询记录 |\n");

printf(" | 5-排序记录 |\n");

printf(" | 0-退出 |\n");

printf(" +--------------------+\n");

while(1)

{

printf("\n 请选择 (0~5):");

scanf("%d",&cn);

printf("\n");

if (cn<0 || cn>5)

printf("\n input error!");

else

break;

}

return cn;

}

**（二）主要功能模块**

　1、s\_read()函数：

当第一次使用本系统时，应自动建立数据文件“st\_table”，以后则能从“st\_table”文件中读取记录并存入结构体数组中。

void s\_read( )

{

FILE \*fp;

int i;

if((fp=fopen("st\_table","rb"))==NULL)

fp=fopen("st\_table","wb");

i=0;

reccount=0;

while (!feof(fp))

{

if(fread(&stud[i],LEN,1,fp) != 1) break;

reccount++;

i++;

}

fclose(fp);

printf("记录总数=%d \n",reccount);

}

　2、s\_save()函数：

当结构体数组中的数据发生变化时，如添加记录、修改记录、删除记录等，将通过该函数把最新数据重新写回到磁盘文件“st\_table”中。

/\*以下定义写记录到文件函数\*/

void s\_save( )

{

FILE \*fp;

int i;

fp=fopen("st\_table","wb");

for(i=0;i<reccount;i++)

if(fwrite(&stud[i],LEN,1,fp)!=1)

printf("文件写错误！\n");

fclose(fp);

}

　3、xhexist()函数：

用来判断某学号是否存在，若存在则返回函数值1，否则返回0，因为本系统要求学号作为唯一的关键字，即不允许出现重复的学号，当执行添加、修改操作时通过调用本函数则保证学号的唯一性。

/\*以下定义判断学号是否存在函数\*/

int xhexist( char num[])

{

int i;

int flag=0;

for (i=0;i<reccount; i++)

if (strcmp(num,stud[i].xh)==0)

{

flag=1;

break;

}

return (flag);

}

4、s\_add()函数：

　 用来实现向结构体数组中添加一条新记录，在输入时对学号的重复进行判断。

void s\_add( )

{

int count = reccount;

//依次输入需添加的每一项内容

while (1)

{

printf("学号:");

scanf("%s",stud[count].xh);

if (xhexist(stud[count].xh)==1)

printf("学号已存在!\n");

else

break;

}

printf("姓名:");

scanf("%s",stud[reccount].xm);

printf("联系电话:");

scanf("%s",stud[reccount].lxdh);

printf("家庭地址:");

scanf("%s",stud[reccount].jtdz);

reccount++;

printf("总记录数=%d\n", reccount);

}

5、void modi\_xh( )函数：

用来实现按学号修改记录，首先输入要修改的学号，若存在则显示原数据，然后提示输入新数据，输入时对学号的重复进行判断，若已存在则不允许输入；否则提示学号不存在。

6、void modi\_xm( )函数：

用来实现按姓名修改记录，首先输入要修改的姓名，若存在则显示原数据，然后提示输入新数据，输入时对学号的重复进行判断，若已存在则不允许输入，并能修改多个相同姓名的记录；否则提示姓名不存在。

7、void del\_xh( )函数：

用来实现按学号删除记录，首先输入要删除的学号，若存在则显示原数据，然后提示是否真的删除，若输入Y则进行删除操作，删除通过记录上移的方法实现；否则提示学号不存在。

8、void del\_xm( )函数：

用来实现按姓名删除记录，首先输入要删除的姓名，若存在则显示原数据，然后提示是否真的删除，若输入Y则进行删除操作，删除通过记录上移的方法实现，并可以删除多个相同姓名的记录；否则提示姓名不存在。

8、 void sear\_xh( )函数：

用来实现按学号查询记录，首先输入要查询的学号，若存在则显示原数据；否则提示学号不存在。

9、void sear\_xm( )函数：

用来实现按姓名查询记录，首先输入要查询的姓名，若存在则显示原数据，并可以查询多个相同姓名的记录；否则提示姓名不存在。

10、void sear\_all( )函数：

用来实现显示全部记录。

11、void sear\_sr( )函数：

用来完成按生日查询，首先输入要查询的生日，可以输入四种数据，并分别得出相应查询结果：①输入年（格式4位：19\*\*），查询显示此年度出生的所有学生记录；②输入月（格式1至2位：1-12间的整数），查询显示这个月份出生的所有学生记录；③输入年月（格式6位：198708）查询显示本年和本月出生的所有学生记录；④输入月/日期（格式4位： 0621）,查询显示同月同日生的所有学生记录。

12、void sear\_jtdz( )函数：

用来完成按家庭地址查询，首先输入地址，格式可以是任意一个字符串，通过匹配字符串中的子串的方法，找出与家庭地址与输入地址相符的所有学生记录。

13、void sort\_xh( )函数：

用来实现按学号排序记录，要求采用起泡法进行排序。

14、void sort\_cj ( )函数：

用来实现按成绩进行排序，可以采用选择排序或冒泡排序算法。

15、其它，自己设计；