

## 课程设计题目

(适用于光电 22 级)

- |             |               |
|-------------|---------------|
| 1、订水信息管理系统  | 16、学生饮用水管理系统  |
| 2、酒店房间预订系统  | 17、学生选修课程管理系统 |
| 3、电影票预订系统   | 18、超市商品管理系统   |
| 4、职工工资管理系统  | 19、小区停车管理系统   |
| 5、图书信息管理系统  | 20、社区老人服务管理系统 |
| 6、学生信息管理系统  | 21、社区儿童活动管理系统 |
| 7、学生成绩管理系统  | 22、校园运动会管理系统  |
| 8、个人通讯录管理系统 | 23、校园勤工俭学排班系统 |
| 9、家庭财务管理系统  | 24、机房排课管理系统   |
| 10、酒店菜式管理系统 | 25、选择题标准化考试系统 |
| 11、学生宿舍管理系统 | 26、航班信息管理系统   |
| 12、校园课室管理系统 | 27、实验设备管理系统   |
| 13、学生水费管理系统 | 28、医院预约挂号系统   |
| 14、学生电费管理系统 | 29、电影院售票系统    |
| 15、学生饭卡管理系统 | 30、高速公路收费联网系统 |

说明:

- 1) 每个小组可以从上面的题目中选题, 但是每个题目只允许最多两个组选取。
- 2) 鼓励学生根据兴趣自己拟题。
- 3) 鼓励学生结合自己的专业拟题, 为培育、申报大学生创新创业项目打好基础。

## 参考例子 1

(一) 题目: 车票管理系统

(二) 目的与要求

一车站每天有  $n$  个发车班次, 每个班次都有一班次号 (1、2、3,, $n$ ), 固定的发车时间, 固定的路线 (起始站、终点站), 大致的行车时间, 固定的额定载客量。如:

班次	发车时间	起点站	终点站	行车时间	额定载量	已定票人数
1	8:00	广州	深圳	2	45	30
2	6:30	广州	惠州	1	40	40
3	7:00	广州	佛山	1	40	20
4	10:00	广州	东莞	1	40	2

.....

1、功能要求: 用 c/c++设计一系统, 能提供下列服务:

- (1) 录入班次信息(信息用文件保存)
- (2) 浏览班次信息,可显示出所有班次当前状态。
- (3) 查询路线: 可按班次号查询 ,可按终点站查询
- (4) 售票和退票功能

A: 当查询出已定票人数小于额定载量且当前系统时间小于发车时间时才能售票, 自动更新已售票人数

B: 退票时, 输入退票的班次, 当本班车未发出时才能退票, 自动更新已售票人数

- (5) 系统安全方面的功能

2、其它要求:

- (1) 源程序要有适当的注释, 使程序容易阅读
- (2) 至少采用文本菜单界面
- (3) 学生可自动增加新功能模块
- (4) 写出课程设计报告

## 参考例子 2

### （一）题目:职工信息管理系统

### （二）目的与要求

#### 1. 目的:

- (1)要求学生熟练掌握 C 语言的基本知识和编程技能;
- (2)基本掌握结构化程序设计的基本思路和方法;

#### 2. 基本要求

- (1)系统以菜单方式工作
- (2)职工信息录入功能(职工信息用文件保存)——输入
- (3)职工信息浏览功能——输出
- (4)查询或排序功能: (至少一种查询方式)——算法,按工资查询,按学历查询等
- (5)系统进入画面(静态或动画)
- (6)职工信息删除功能
- (7)职工信息修改功能

#### 3.创新要求

在基本要求达到后,可进行创新设计,如对菜单进行美化。

### （三）信息描述

职工信息包括职工号、姓名、性别、出生年月、学历、职务、工资、住址、电话等(职工号不重复)。

### （四）功能描述

1. 录入职工信息并保存
2. 显示所有职工信息(以一定的格式)
3. 查询职工信息(以一种或多种方式)
4. 修改职工信息并保存
5. 删除职工信息

### （五）解决方案

1. 首先进行需求分析,搞清楚系统功能和任务。
2. 然后总体设计中确定模块结构,划分功能模块,将软件功能需求分配给所划分的最小单元模块。确定模块间的联系,确定数据结构、文件结构、数据库模式,确定测试方法与策略。
3. 详细设计中为每个模块确定采用的算法,选择适当的工具表达算法的过程(流程图),描述模块的详细过程;确定每一模块使用的数据结构;确定模块接口的细节,包括对系统外部的接口和用户界面,对系统内部其它模块的接口。
4. 依据分析编写 C 语言代码。

### （六）职工信息管理系统

#### 需求分析

#### 编写目的:

通过对用户需求进行调查分析,写出需求分析文档作为项目设计的基本准则要求,以及作

为系统分析员进行系统分析和测试人员进行测试时的手册

需求概述：

设计一个职工信息管理系统，使之能提供以下功能：

- 1) 录入职工信息并保存
- 2) 显示所有职工信息
- 3) 查询职工信息
- 4) 修改职工信息并保存
- 5) 删除职工信息

需求说明：

职工信息包括职工号、姓名、性别、出生年月、学历、职务、工资、住址、电话等，并且要求职工号不重复；

录入的职工信息要求用文件形式保存，并可以对其进行浏览、查询、修改、删除等基本操作；

职工信息的显式要求有一定的规范格式；

对职工信息应能够分别按工资及按学历两种方式进行，要求能返回所有符合条件的职工的信息；

对职工信息的修改应逐个地进行，一个职工信息的更改不应影响其他的职工记录；

所设计的系统应以菜单方式工作，应为用户提供清晰的使用提示，依据用户的选择来进行各种处理，并要求在此过程中能够尽可能的兼容用户使用过程中的异常情况

---

总体设计

编写目的：

根据需求分析文档，初步提出问题的解决方案，以及软件系统的体系结构和数据结构的设计方案并写出书面文档总体设计说明书，为下一步进行详细设计做准备。

总体设计：

1. 该系统可以按功能进行模块划分，其模块图如下：

系统模块图---略

其中：

信息输入模块完成职工信息的输入功能，输入信息包括职工号、姓名、性别、出生年月、学历、职务、工资、住址、电话等；

信息浏览模块完成已录入职工信息的显示；

信息查询模块完成职工信息的查询，查询时对应按工资查询和按学历查询两种方式；

信息修改模块完成职工信息的修改功能；

信息删除模块完成职工信息的删除功能；

菜单选择模块完成用户命令的接收，此模块也是职工信息管理系统的入口，用户所要进行的各种操作均需在此模块中进行选择并进而调用其他模块实现相应的功能；

2. 数据结构

本系统中主要的数据结构就是职工的信息，包含职工号、姓名、性别、出生年月、学历、职务、工资、住址、电话等，在处理过程中各项可以作为一个职工的不同属性来进行处理

3. 程序流程

系统的执行应从功能菜单的选择开始，依据用户的选择来进行后续的处理，直到用户选择退出系统为止，其间应对用户的选择做出判断及异常处理。系统的流程图如下：

程序流程图---略

---

## 详细设计

### 编写目的：

根据项目概要设计说明书，在项目概要设计的基础之上，对系统进行详细设计，以便下一步进行程序编码工作。

### 详细设计：

对应于总体设计时的系统功能模块图,各个功能模块的处理逻辑如下：

#### ① 信息输入模块：

打开职工信息文件；

```
while ( 继续输入 )
```

```
{
```

读入职工信息；

将读入信息添加到职工信息文件中；

提示是否继续输入；

```
}
```

关闭职工信息文件；

#### ② 信息浏览模块：

打开职工信息文件；

```
while ( 不到文件结束 )
```

```
{
```

从文件中顺序读出一个职工的信息；

按指定格式输出该职工的信息；

```
}
```

关闭职工信息文件；

#### ③ 信息查询模块：

```
while ( 仍继续查询 )
```

```
{
```

```
if( 按工资查询 )
```

```
{
```

读入查询的职工工资条件；

打开职工信息文件；

```
while( 不到文件结束 )
```

```
{
```

顺序读出一个职工的信息；

```
if( 该职工信息的工资符合条件 ) 输出该职工的信息；
```

```
}
```

关闭职工信息文件；

提示共有几条符合条件的记录；

```
}
```

```
else
```

```
if ( 按学历查询 )
```

```
{
```

读入学历查询的条件；

打开职工信息文件；

```
while( 不到文件结束 )
```

```

{
顺序读出一个职工的信息；
if( 该职工信息的学历符合条件 ) 输出该职工的信息；
}
关闭职工信息文件；
提示符合条件的记录数；
}
else 提示输入不合适；
提示是否继续查询；
}
④ 信息修改模块：
while( 继续修改 )
{
从给出信息文件中所有的记录情况；
提示并读入待修改的记录的职工的职工号；
打开职工信息文件；
查找并读出该职工的信息到一个中间变量中；
提示并读入要针对该职工的哪个属性进行修改；
根据用户的选择提示并读入所要修改属性的新值；
在该职工的中间变量中修改刚刚读入的属性值；
提示是否确定要进行修改；
如果用户确定则将中间变量重新写回到原职工信息文件中；
关闭职工信息文件；
}
⑤ 信息删除模块：
while( 继续删除 )
{
显示所有职工的信息；
提示并读入待删除的职工的职工号；
打开职工信息文件；
查找是否有符合条件的职工记录；
if( 有符合条件的记录 )
{
提示是否确认该删除操作；
if (用户确认此操作)
{
创建一个新的临时文件；
打开原信息文件；
while( 原信息文件中记录未读完 )
{
读出原信息文件中的下一条记录；
if( 此条记录不是待删除记录 ) 写入临时文件中；
}
关闭原信息文件；

```

```

关闭临时文件;
读方式打开临时文件;
写方式重建与原信息文件同名的新信息文件;
将临时文件中的内容写入到新信息文件中;
关闭新信息文件;
关闭临时文件;
删除临时文件;
}
}
else 提示没有符合条件的记录存在;
}
⑥ 菜单选择模块:
给出信息提示;
清屏;
绘制菜单(包含输入、显示、查询、修改、删除、退出);
提示菜单选择并读入到变量;
根据读入变量的值调用不同的模块函数;
数据结构:
性别: enum Sex{ male, female };
学历 : enum Education{ high, junior, college, master, doctor};
日期 : struct Date
{ int year;
int month;
int day;
};
职工信息: struct Info
{ char num[5]; /* 职工号 */
char name[8]; /* 姓名 */
enum Sex sex; /* 性别 */
struct Date birthday; /* 出生年月*/
enum Education education; /* 学历 */
char duty[8]; /* 职务 */
double wage; /* 工资 */
char addr[12]; /* 地址 */
char phone[8]; /* 电话 */
};

```

源程序:

```

#include <stdio.h>
#include <string.h>
#include <stdlib.h>
enum Sex{ male, female }; /* 性别 */
enum Education{ high, junior, college, master, doctor}; /* 学历 */

```

```

struct Date /* 日期 */
{
int year;
int month;
int day;
};
struct Info
{
char num[5]; /* 职工号 */
char name[8]; /* 姓名 */
enum Sex sex; /* 性别 */
struct Date birthday; /* 出生年月*/
enum Education education; /* 学历 */
char duty[8]; /* 职务 */
double wage; /* 工资 */
char addr[12]; /* 地址 */
char phone[8]; /* 电话 */
};
void menu();
int load(char * a) /* 计算参数所指定的文件中的记录的个数的函数 */
{
struct Info info;
FILE *fp;
int i;
if((fp=fopen(a,"rb"))==NULL)
{printf("\nCannot open file\n"); getch();
return 0;
}
for(i=0;!feof(fp);i++)
fread(&info,sizeof(struct Info),1,fp);
fclose(fp);
return(i-1); /*返回记录个数*/
}
void append() /* 信息添加函数 */
{
struct Info info;
FILE * fp;
int flag = 1;
char temp[10];
if((fp=fopen("inform.txt","ab")) == NULL)
if((fp=fopen("inform.txt","wb"))==NULL)
{
printf("\tCan not open the inform file!");
getch();
}
}

```



```

exit(1);
}
do
{
printf("\tnum:");gets(info.num);
printf("\tname:");gets(info.name);
printf("\tsex:");gets(temp);
if(!strcmp(temp,"female")) info.sex=female;
else info.sex=male;
printf("\tbirthday(yyyy/mm/dd):");scanf("%d/%d/%d",&info.birthday.year,&info.bi
rthday.month,&info.birthday.day); getchar();
printf("\teducation:");gets(temp);
if(!strcmp(temp,"doctor")) info.education=doctor;
else if(!strcmp(temp,"master")) info.education=master;
else if(!strcmp(temp,"college")) info.education=college;
else if(!strcmp(temp,"junior")) info.education=junior;
else info.education=high;
printf("\tduty:");gets(info.duty);
printf("\twage:");gets(temp); info.wage=atof(temp);
printf("\taddress:");gets(info.addr);
printf("\tphone:");gets(info.phone);
if(fwrite(&info, sizeof(info), 1, fp) != 1)
{
printf("\nfile input wrong!\n");
getch();
exit(1);
}
printf("Any more ? \t1).Yes\t2).No: [ ]\b\b");
scanf("%d",&flag); getchar();
}while(flag == 1);

fclose(fp);
menu();
}
void display() /* 职工信息显示函数 */
{
int amount; /* 信息文件中的记录总数 */
struct Info info;
FILE * fp;
int total = 0;
char temp[10], temp1[10];
amount=load("inform.txt");
if((fp=fopen("inform.txt","rb")) == NULL)
{

```

```

printf("\tCan not open the inform file!");
getch();
exit(1);
}
printf("%-5s%-8s%-8s%-11s%-10s%-8s%-8s%-12s%-8s\n", "nun", "name", "sex", "birthday",
"education", "duty", "wage", "address", "phone");
for(;total<amount;)
{
if(fread(&info, sizeof(info), 1, fp) != 1)
{
printf("\nfile read wrong!\n");
getch();
exit(1);
}
total++;
if(info.sex==male) strcpy(temp, "male");
else strcpy(temp, "female");
if(info.education==doctor) strcpy(temp1, "doctor");
else if(info.education==master) strcpy(temp1, "master");
else if(info.education==college) strcpy(temp1, "college");
else if(info.education==junior) strcpy(temp1, "junior");
else strcpy(temp1, "high");
printf("%-5s%-8s%-8s%-4d%-2d%-2d%-10s%-8s%-8.2lf%-12s%-8s\n", info.num, info.name, temp, info.birthday.year, info.birthday.month, info.birthday.day, temp1, info.duty, info.wage, info.addr, info.phone);
if((total != 0) && (total%10 ==0))
{
printf("\n\n\tPress any key to continue.....");
getch();
puts("\n\n");
printf("%-5s%-8s%-8s%-11s%-10s%-8s%-8s%-12s%-8s\n", "nun", "name", "sex", "birthday", "education", "duty", "wage", "address", "phone");
}
}
fclose(fp);
printf("\n\n\tThere are %d record in all!", total);
getch();
menu();
}

void search() /* 信息查询函数 */
{
int amount; /* 信息文件中的记录总数 */
int con=0; /* 已处理的文件记录总数 */
struct Info info;

```

```

FILE * fp;
int flag=-1, flag1=-1; /* flag 为 1 按工资查询, flag 为 2 按学历查询; flag1 为 0
表等于, 为 1 表大于, 为 2 表小于 */
int total=0; /*记录符合条件的记录的个数 */
char temp[10], temp1[10], degree[10], express[10];
char operator;
double value;
enum Education grade1;
int i=0; /* 读入表达式时用于字符串到数值的转换 及 控制学历的合法输入 */
amount=load("inform.txt");
do
{
printf("\n\nChoice your item that to search by: \t1).wage \t2).education:
[ ]\b\b");
scanf("%d",&flag); getchar();
}while(!(flag==1 ||flag==2));
if(flag==1) /* 按工资进行查询 */
{
flag1=-1;
do
{
printf("Please input the relative expression(>value or <value or =value):");
gets(express);
i=1;
do
{
temp[i-1]=express[i];
i++;
}while(express[i-1] != '\0');
operator = express[0];
value = atof(temp);
if(operator=='>') flag1=1;
else if(operator=='<') flag1=2;
else if(operator=='=') flag1=0;
else printf("\nthe express format must be >value or <value or =value, please input
again...\n");
}while(!(flag1==1 || flag1==2 || flag1==0));
if((fp=fopen("inform.txt", "rb")) == NULL)
{
printf("\tCan not open the inform file!");
getch();
exit(1);
}
total=0; /* 符合条件的记录数 */

```

```

for(con=0; con<amount; con++)
{
if(fread(&info, sizeof(info), 1, fp) != 1)
{
printf("\nfile read wrong!\n");
getch();
exit(1);
}
if(flag1==0 && info.wage==value)
{
total++;
if(total == 1)
printf("%-5s%-8s%-8s%-11s%-10s%-8s%-12s%-8s\n", "nun", "name", "sex",
", "birthday", "education", "duty", "wage", "address", "phone");
if(info.sex==male) strcpy(temp, "male");
else strcpy(temp, "female");
if(info.education==doctor) strcpy(temp1, "doctor");
else if(info.education==master) strcpy(temp1, "master");
else if(info.education==college) strcpy(temp1, "college");
else if(info.education==junior) strcpy(temp1, "junior");
else strcpy(temp1, "high");
printf("%-5s%-8s%-8s%-4d/%-2d/%-2d%-10s%-8s%-8.2lf%-12s%-8s\n", info.num, info.name, temp, info.birthday.year, info.birthday.month, info.birthday.day, temp1, info.duty, info.wage, info.addr, info.phone);
if((total != 0) && (total%10 ==0))
{
printf("\n\n\tPress any key to continue.....");
getch();
puts("\n\n");
printf("%-5s%-8s%-8s%-11s%-10s%-8s%-12s%-8s\n", "nun", "name", "sex",
", "birthday", "education", "duty", "wage", "address", "phone");
}
}
else if(flag1==1 && info.wage>value)
{
total++;
if(total == 1)
printf("%-5s%-8s%-8s%-11s%-10s%-8s%-12s%-8s\n", "nun", "name", "sex",
", "birthday", "education", "duty", "wage", "address", "phone");
if(info.sex==male) strcpy(temp, "male");
else strcpy(temp, "female");
if(info.education==doctor) strcpy(temp1, "doctor");
else if(info.education==master) strcpy(temp1, "master");
else if(info.education==college) strcpy(temp1, "college");
}
}
}

```

```

else if(info.education==junior) strcpy(temp1,"junior");
else strcpy(temp1,"high");
printf("%-5s%-8s%-8s%-4d/%-2d/%-2d%-10s%-8s%-8.2lf%-12s%-8s\n",info.num,info.name,temp,info.birthday.year,info.birthday.month,info.birthday.day,temp1,info.duty,info.wage,info.addr,info.phone);
if((total != 0) && (total%10 ==0))
{
printf("\n\n\tPress any key to continue.....");
getch();
puts("\n\n");
printf("%-5s%-8s%-8s%-11s%-10s%-8s%-8s%-12s%-8s\n","nun","name","sex","birthday","education","duty","wage","address","phone");
}
}
else if(flag1==2 && info.wage<value)
{
total++;
if(total == 1)
printf("%-5s%-8s%-8s%-11s%-10s%-8s%-8s%-12s%-8s\n","nun","name","sex","birthday","education","duty","wage","address","phone");
if(info.sex==male) strcpy(temp,"male");
else strcpy(temp,"female");
if(info.education==doctor) strcpy(temp1,"doctor");
else if(info.education==master) strcpy(temp1,"master");
else if(info.education==college) strcpy(temp1,"college");
else if(info.education==junior) strcpy(temp1,"junior");
else strcpy(temp1,"high");
printf("%-5s%-8s%-8s%-4d/%-2d/%-2d%-10s%-8s%-8.2lf%-12s%-8s\n",info.num,info.name,temp,info.birthday.year,info.birthday.month,info.birthday.day,temp1,info.duty,info.wage,info.addr,info.phone);
if((total != 0) && (total%10 ==0))
{
printf("\n\n\tPress any key to continue.....");
getch();
puts("\n\n");
printf("%-5s%-8s%-8s%-11s%-10s%-8s%-8s%-12s%-8s\n","nun","name","sex","birthday","education","duty","wage","address","phone");
}
}
}
fclose(fp);
}
else /* 按学历进行查询 */
{

```

```

i=0;
do
{
printf("Please input the education degree you want to search:");
gets(degree);
if(!strcmp(degree,"doctor")) grade1=doctor;
else if(!strcmp(degree,"master")) grade1=master;
else if(!strcmp(degree,"college")) grade1=college;
else if(!strcmp(degree,"junior")) grade1=junior;
else if(!strcmp(degree,"high")) grade1=high;
else { printf("\nThe education value is illegimate....\n"); i=1; }
}while(i==1);
if((fp=fopen("inform.txt","rb")) == NULL)
{
printf("\tCan not open the inform file!");
getch();
exit(1);
}
total=0; /* 符合条件的记录数 */
for(con=0; con<amount; con++)
{
if(fread(&info, sizeof(info), 1, fp) != 1)
{
printf("\nfile read wrong!\n");
getch();
exit(1);
}
if(info.education == grade1)
{
total++;
if(total == 1)
printf("%-5s%-8s%-8s%-11s%-10s%-8s%-8s%-12s%-8s\n", "num", "name", "sex",
"birthday", "education", "duty", "wage", "address", "phone");
if(info.sex==male) strcpy(temp, "male");
else strcpy(temp, "female");
if(info.education==doctor) strcpy(temp1, "doctor");
else if(info.education==master) strcpy(temp1, "master");
else if(info.education==college) strcpy(temp1, "college");
else if(info.education==junior) strcpy(temp1, "junior");
else strcpy(temp1, "high");
printf("%-5s%-8s%-8s%-4d%-2d%-2d%-10s%-8s%-8.2lf%-12s%-8s\n", info.num, info.name, temp, info.birthday.year, info.birthday.month, info.birthday.day, temp1, info.duty, info.wage, info.addr, info.phone);
if((total != 0) && (total%10 ==0))

```

```

{
printf("\n\n\tPress any key to continue.....");
getch();
puts("\n\n");
printf("%-5s%-8s%-8s%-11s%-10s%-8s%-8s%-12s%-8s\n", "nun", "name", "sex",
", "birthday", "education", "duty", "wage", "address", "phone");
}
}
}
fclose(fp);
}
printf("\n\n\tThere are %d record included!", total);
getch();
menu();
}
void modify() /* 信息修改函数 */
{
struct Node
{
char num[10];
struct Node * next;
};
int amount; /* 信息文件中的记录总数 */
int con=0; /* 已处理的文件记录总数 */
struct Node node;
struct Node * head=NULL, * pf=NULL, * pb=NULL;
struct Info info, info_temp;
FILE * fp;
int total=0;
long offset=0; /* 文件位置指针偏移量 */
char temp[10], temp1[10];
int flag=0; /* 指示是否继续进行修改，为 1 时继续 */
char ID[10]; /* 待修改的记录的员工号 */
int ord; /* 待修改的项目指示 */
int repeat; /* 在修改中用于重复的输入 */
int Yes=0; /* 指示是否保存修改 */
char choice;
amount=load("inform.txt");
if((fp=fopen("inform.txt", "rb")) == NULL)
{
printf("\tCan not open the inform file!");
getch();
exit(1);
}
}

```

```

printf("\n%-5s%-8s%-8s%-11s%-10s%-8s%-8s%-12s%-8s\n", "num", "name", "sex",
", "birthday", "education", "duty", "wage", "address", "phone");
for(con=0; con<amount; con++)
{
if(fread(&info, sizeof(info), 1, fp) != 1)
{
printf("\nfile read wrong!\n");
getch();
exit(1);
}
pb=(struct Node *)malloc(sizeof(struct Node)); /* 创建包含 num 信息的链表,
用于后续的查找 */
strcpy(pb->num, info.num);
if(total==0) head=pf=pb;
else pf->next=pb;
pb->next=NULL;
pf=pb;
total++;
if(info.sex==male) strcpy(temp, "male");
else strcpy(temp, "female");
if(info.education==doctor) strcpy(temp1, "doctor");
else if(info.education==master) strcpy(temp1, "master");
else if(info.education==college) strcpy(temp1, "college");
else if(info.education==junior) strcpy(temp1, "junior");
else strcpy(temp1, "high");
printf("%-5s%-8s%-8s%-4d/%-2d/%-2d%-10s%-8s%-8.2lf%-12s%-8s\n", info.num, info.name, temp, info.birthday.year, info.birthday.month, info.birthday.day, temp1, info.duty, info.wage, info.addr, info.phone);
if((total != 0) && (total%10 ==0))
{
printf("\n\n\tPress any key to continue.....");
getch();
puts("\n\n");
}
}
fclose(fp);
do
{
printf("\nPlease enter the num that you want to modify:");
gets(ID);
pf=head;
offset=0;
while(strcmp(pf->num, ID) && pf->next!=NULL)
{

```



```

pf=pf->next;
offset++;
}
if(!strcmp(pf->num,ID)) /* 待修改的记录存在 */
{
if((fp=fopen("inform.txt","rb+"))==NULL)
{
printf("\n\nfile open wrong!");
getch();
exit(1);
}
offset *= sizeof(struct Info); /* 使文件位置指针移动到待修改记录处 */
rewind(fp);
if(fseek(fp, offset, 0) != 0)
{
printf("\n\nfile operation wrong !");
getch();
exit(1);
}
fread(&info, sizeof(info), 1, fp);
fseek(fp,(long)(-1)*sizeof(struct Info), 1); /* 使文件位置指针返回到待修改记录处 */
printf("\nthe record you want to modify is:\n");
printf("%-5s%-8s%-8s%-11s%-10s%-8s%-8s%-12s%-8s\n","num","name","sex",
"birthday","education","duty","wage","address","phone");
printf("%-5s%-8s%-8s%-4d%-2d%-2d%-10s%-8s%-8.2lf%-12s%-8s\n",info.num,info.name,temp,info.birthday.year,info.birthday.month,info.birthday.day,temp1,info.duty,info.wage,info.addr,info.phone);
do
{
printf("\n\nyou can modify
by:1).num\t2).name\t3).sex\t4).birthday\t5).education");
printf("\n\t6).duty\t7).wage\t8).address\t9).phone\t10).all
data\t11).cancel and back");
printf("\nWhich item you need to modify: [ ]\b\b");
scanf("%d", &ord);
}while(ord>11 || ord<1);
switch(ord)
{
case 1: printf("\nnum:");
repeat=0;
do
{
gets(ID); getch();

```

```

pb=head;
while(strcmp(pb->num,ID) && pb->next !=NULL) pb=pb->next;
if(!strcmp(pb->num,ID)) printf("\nError:the num is already exist,
please input again...");
repeat=1;
}while(repeat);
strcpy(info.num, ID);
break;
case 2: printf("\nname:");gets(info.name); getchar(); break;
case 3: printf("\nsex:"); gets(temp); getchar();
if(!strcmp(temp,"male")) info.sex=male;
else if(!strcmp(temp,"female")) info.sex=female;
else printf("\ninput error! Can not modify the record ...");
break;
case 4:
printf("\nbirthday(yyyy/mm/dd):");scanf("%d/%d/%d",&info.birthday.year,&info.bi
rthday.month,&info.birthday.day);
getchar(); break;
case 5: printf("\neducation:"); gets(temp); getchar();
if(!strcmp(temp,"doctor")) info.education=doctor;
else if(!strcmp(temp,"master")) info.education=master;
else if(!strcmp(temp,"college")) info.education=college;
else if(!strcmp(temp,"junior")) info.education=junior;
else if(!strcmp(temp,"high")) info.education=high;
else printf("\ninput error! Can not modify the record...");
break;
case 6: printf("\nduty:"); gets(info.duty); getchar(); break;
case 7: printf("\nwage:"); scanf("%lf",info.wage); getchar(); break;
case 8: printf("\naddress:"); gets(info.addr); getchar(); break;
case 9: printf("\nphone:"); gets(info.phone); getchar(); break;
case 10:printf("\ntnum:");gets(info.num); getchar();
printf("\ntname:");gets(info.name); getchar();
printf("\ntsex:");gets(temp); getchar();
if(!strcmp(temp,"female")) info.sex=female;
else info.sex=male;
printf("\tbirthday(yyyy/mm/dd):");scanf("%d/%d/%d",&info.birthday.year,&info.
birthday.month,&info.birthday.day);
printf("\teducation:");gets(temp); getchar();
if(!strcmp(temp,"doctor")) info.education=doctor;
else if(!strcmp(temp,"master")) info.education=master;
else if(!strcmp(temp,"college")) info.education=college;
else if(!strcmp(temp,"junior")) info.education=junior;
else info.education=high;
printf("\tduty:");gets(info.duty); getchar();

```

```

printf("\n\twage:");gets(temp); info.wage=atof(temp); getchar();
printf("\n\taddress:");gets(info.addr); getchar();
printf("\n\tnum:");gets(info.phone); getchar();
break;
case 11:
default:printf("\n\n");
menu();
}
printf("\nAre you sure?\n\n\t1).Sure 2).Back without save [ ]\b\b"); /*是
否确定*/
scanf("%d",&Yes); getchar();
if(Yes==1) /* 将更改后的数据写回到文件中 */
if((fwrite(&info, sizeof(struct Info), 1, fp))==1) printf("\nthe
modification is successful ^_^");
else printf("\noh,the modification has not done ! ");
else {printf("\n%d\n",Yes);getch();}
fclose(fp);
}
else
{
printf("\n\nThe record you enter is not found!\n\tany key to continue...");
getch();
}
printf("\n\nDo you want to modify more records?:\t\t1).Yes\t2).No [ ]\b\b");
/* 是否继续修改其他记录 */
scanf("%d",&flag); getchar();
}while(flag == 1);
menu();
}
void delete() /* 信息删除函数 */
{
struct Node
{
char num[10];
struct Node * next;
};
int amount; /* 信息文件中的记录总数 */
int con=0; /* 已处理的文件记录总数 */
struct Node node;
struct Node * head=NULL, * pf=NULL, * pb=NULL;
struct Info info, info_temp;
FILE * fp, *fp1; /* fp 用于指示信息文件 inform, 而 fp1 用于指示临时文件 */
int total=0;
int offset=0; /* 文件位置指针偏移量 */

```

```

char temp[10], temp1[10];
int flag=0; /* 指示是否继续进行修改，为 1 时继续 */
char ID[10]; /* 待修改的记录的员工号 */
int ord; /* 待修改的项目指示 */
int repeat; /* 在修改中用于重复的输入 */
int Yes=0; /* 指示是否保存修改 */
amount=load("inform.txt");
if((fp=fopen("inform.txt","rb")) == NULL)
{
printf("\tCan not open the inform file!");
getch();
exit(1);
}
printf("%-5s%-8s%-8s%-11s%-10s%-8s%-12s%-8s\n","num","name","sex",
"birthday","education","duty","wage","address","phone");
for(con=0; con<amount; con++)
{
if(fread(&info, sizeof(info), 1, fp) != 1)
{
printf("\nfile read wrong!\n");
getch();
exit(1);
}
pb=(struct Node *)malloc(sizeof(struct Node)); /* 创建包含 num 信息的链表，
用于后续的查找 */
strcpy(pb->num,info.num);
if(total==0) head=pf=pb;
else pf->next=pb;
pb->next=NULL;
pf=pb;
total++;
if(info.sex==male) strcpy(temp,"male");
else strcpy(temp,"female");
if(info.education==doctor) strcpy(temp1,"doctor");
else if(info.education==master) strcpy(temp1,"master");
else if(info.education==college) strcpy(temp1,"college");
else if(info.education==junior) strcpy(temp1,"junior");
else strcpy(temp1,"high");
printf("%-5s%-8s%-8s%-4d/%-2d/%-2d%-10s%-8s%-8.2lf%-12s%-8s\n",info.num,info.na
me,temp,info.birthday.year,info.birthday.month,info.birthday.day,temp1,info.dut
y,info.wage,info.addr,info.phone);
if((total != 0) && (total%10 ==0))
{
printf("\n\n\tPress any key to continue.....");

```

```

getch();
puts("\n\n");
}
}
fclose(fp);
do
{
printf("Please enter the num that you want to delete:");
gets(ID);
pf=head;
while(strcmp(pf->num,ID) && pf->next!=NULL)
{
pf=pf->next;
offset++;
}
if(!strcmp(pf->num,ID)) /* 待删除的记录存在 */
{
printf("\nAre you sure to delete the record?:\t\t1).Yes\t2).No [ ]\b\b");
scanf("%d",&Yes);
if(Yes==1) /* 如果确定要删除该记录 */
{
if((fp=fopen("inform.txt","rb"))==NULL)
{
printf("\n\nfile open wrong!");
getch();
exit(1);
}
if((fp1=fopen("info_temp.txt","wb"))==NULL)
{
printf("\n\nfile open wrong!");
getch();
exit(1);
}
total=0;
amount=load("inform.txt");
for(con=0; con<amount; con++)
{
if(total==offset-1) fseek(fp,(long)sizeof(struct Info),1);
fread(&info, sizeof(struct Info),1,fp);
fwrite(&info, sizeof(struct Info), 1, fp1);
}
fclose(fp1);
fclose(fp);
if((fp1=fopen("info_temp.txt","rb"))==NULL)

```

```

{
printf("\n\nfile open wrong!");
getch();
exit(1);
}
if((fp=fopen("inform.txt","wb"))==NULL)
{
printf("\n\nfile open wrong!");
getch();
exit(1);
}
amount=load("info_temp.txt");
for(con=0; con<amount; con++)
{
fread(&info, sizeof(struct Info),1,fp1);
fwrite(&info, sizeof(struct Info), 1, fp);
}
fclose(fp);
fclose(fp1);
}
}
else /* 待删除的记录不存在 */
{
printf("\n\nThe record you enter is not found!\tany key to continue...");
getch();
}
printf("\n\nDo you want to continue?:\t\t1).Yes\t2).No [ ]\b\b");
scanf("%d",&flag); getch();
}while(flag == 1);
menu();
}
void menu() /* 菜单选择界面 */
{
int n, w1; /* n 记录选择的菜单, w1 记录选择是否合法 */
do
{
system("cls");/*清屏*/system("cls")*/
puts("\t\t\t Welcome to employee mangement system ");
puts("\t\t\t*****MENU*****\n\n");
puts("\t\t\t1.Append inform");
puts("\t\t\t2.Display inform");
puts("\t\t\t3.Search inform");
puts("\t\t\t4.Modify inform");
puts("\t\t\t5.Delete inform");

```

```

puts("\t\t\t\t6.Exit");
puts("\n\n\t\t*****\n");
printf("Choice your number(1-6): [ ]\b\b");
scanf("%d",&n);
getchar();
if(n<1||n>6) /*对选择的数字作判断*/
{ w1=1;getchar();}
else w1=0;
}
while(w1==1);
switch(n) /*菜单选择功能*/
{
case 1:append();break; /*信息添加模块*/
case 2:display();break; /*信息显示模块*/
case 3:search();break; /*信息查询模块*/
case 4:modify();break; /*信息修改模块*/
case 5:delete();break; /*信息删除模块*/
case 6:exit(0); /*退出*/
}
}
main() /* 主函数 */
{
menu();
}

```