

北京工商大学课程设计报告

- 课程名称：实训一
- 课题名称：模拟工作管理系统
- 班级：计算161
- 姓名：唐笑然
- 学号：1503010222
- 日期：2018年6月

内容

1. 摘要
 2. 选题说明
 3. 设计过程及目标
 4. 设计流程图
 5. 程序分析
 6. 实验心得
 7. 源程序
-

一、摘要

本设计实现了模拟工作管理系统的一些功能，分为雇主雇员两个模块，可以存储或读取雇主或雇员信息、添加新的雇主或雇员信息、查询全部雇主或雇员信息、按条件查找雇主或雇员信息、修改雇主或雇员信息、删除雇主或雇员信息、雇主菜单、雇员菜单、主菜单。雇主信息主要包括：雇主id、雇主姓名、工作名称、工作内容、工作要求、工作地点、工作时间、工作工资以及联系电话。雇员信息主要包括：雇员id、雇员姓名、雇员性别、雇员年龄、个人简历、理想工作类型、理想工作时间、理想工作工资以及联系电话。

二、选题说明

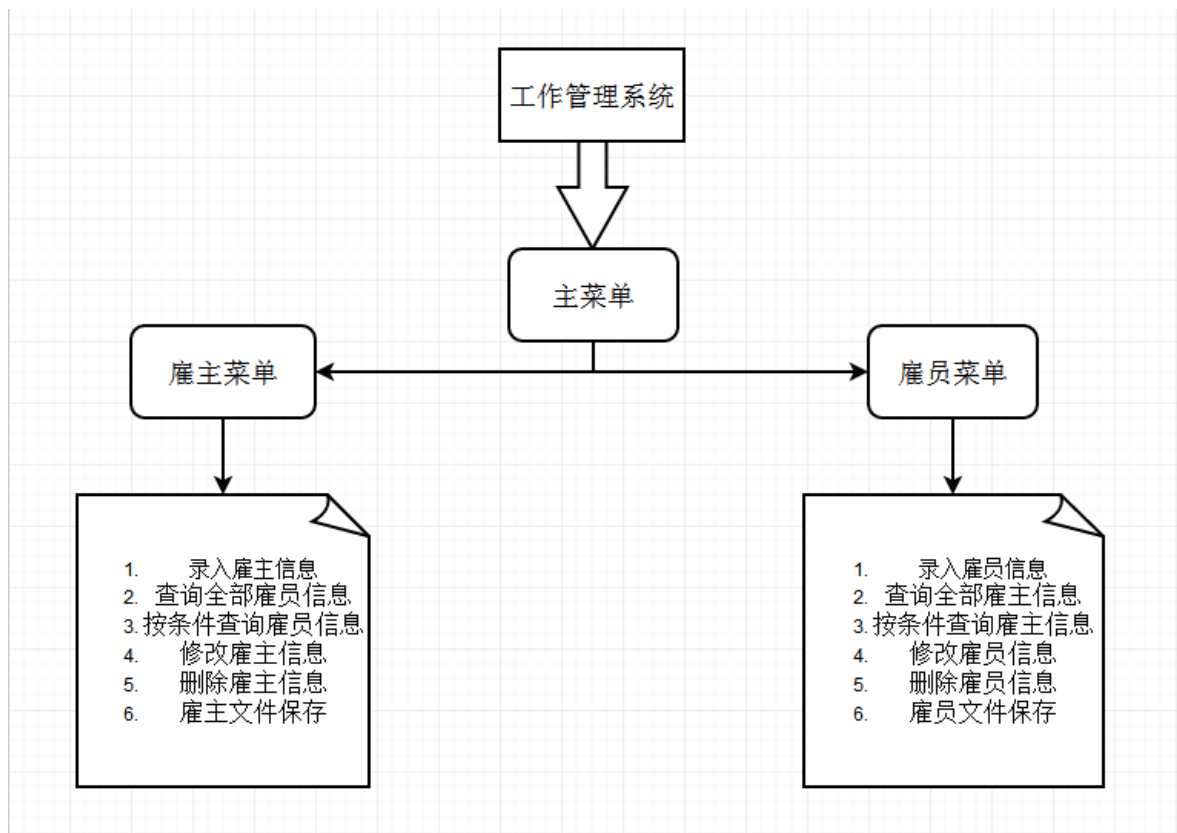
近年，就业越来越不方便，找工作变得很麻烦，因此工作管理系统变得更加重要。方便雇员找到自己理想的工作，也可以方便雇主寻找到合适的雇员。

三、设计过程及目标

运用文件读写操作实现文件读取与存储，添加、删除、修改以及查询信息功能中主要采用for循环、if/else判断语句等，基本信息以结构体数组的方法体现。菜单功能主要用switch来实现。

设计目标：实现简单的工作管理系统，可以实现简单的基本信息存储与读取，c程序控制台的简单界面，简单的信息添加、删除、修改以及查询功能。

四、设计流程图



五、程序分析

添加信息模块：用户可以新建新的雇主或雇员信息，并将新的信息添加到已有的雇主或雇员文件中。添加信息时，依次输入所需要的基本信息，按照提示选择继续添加或者返回菜单。

查询全部信息模块：用户可以通过此功能查看所有已经存入文件的信息，在程序添加了校验功能，如果信息为空会有提示选项。

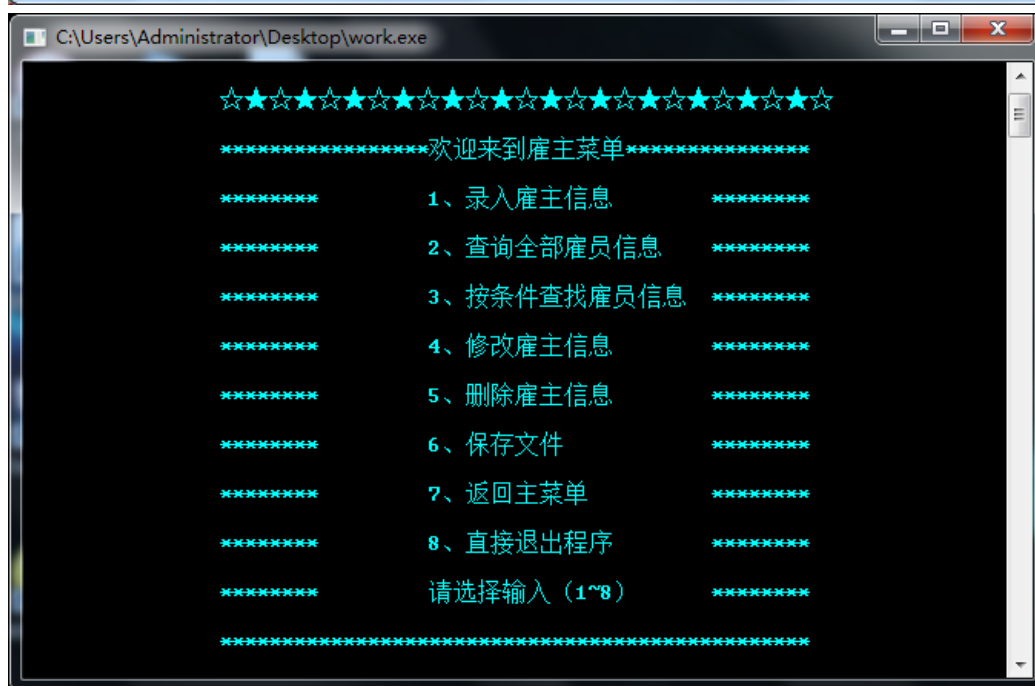
按条件查询信息模块：用户可以按条件精确查找，目前只可以单条件查找。例如可以通过查询id查找相应信息，雇员可以通过查找工作名称来查询相应的雇主信息。目前只可以精确输入以后查找，例如工作地点是“xxx饭店”，如果只输入“饭店”无法查到，必须全部对应上，这一点还有待改进。

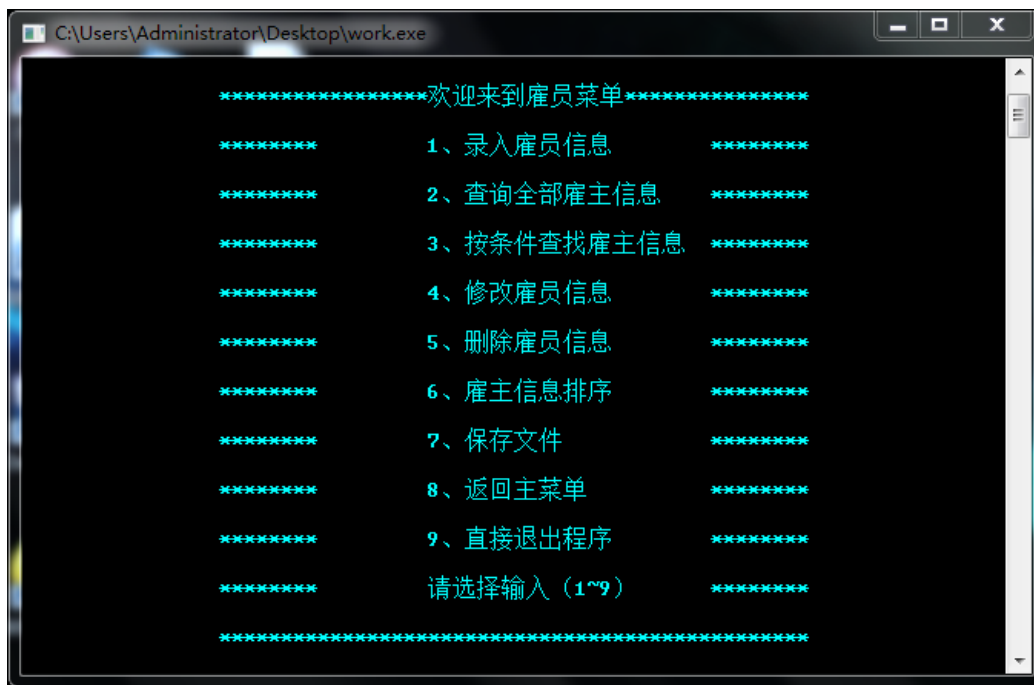
修改信息模块：用户可以通过此功能修改信息，可以选择修改哪个信息。

删除信息模块：用户可以删除已有的信息，删除信息后，后面的信息会依次往前移。

雇主信息排序模块：主要是以工资为基准排序，用户可以选择升序或者降序排列。

菜单模块：菜单模块分为三个，雇主菜单，雇员菜单以及主菜单，分别对应不同的功能。





六、实验心得

基本实现了程序最开始的设计目标，过程中也有很多不懂的地方，通过询问同学以及上网查找逐步弄懂并解决了问题，最终实现了工作管理系统的设计目标。

以前学C语言就是单纯的较为枯燥的写一个一个的平台上的题，这次实训课把C语言与实际问题结合起来，不仅巩固了C语言的基本知识，也加强了对实际问题的解决能力，提高了编程的兴趣，让编程思路更加清晰丰富。

这个题目不是我bb平台上提交的，之前bb平台上提交的三个程序分别是：模拟单项选择题、图书管理系统和模拟高速公路收费。一开始上实训课，拿到这些题目没有什么思路，然后参考网上的一些思路，模仿着写了这个三个题目，也学习了别人的代码的优点还有设计上要多方面考虑。最终选择答辩的时候，决定要写出自己的东西来，所以前面三个已经做完的作业全部放弃了，自己重新选了题目，开始设计思路，初步设定了一些应有的基本功能，然后自己写了代码，有了之前的模仿学习经历，自己写起来就容易了许多。

这个程序目前也存在一些缺陷有待解决。在查找过程中，还不能通过关键字来查找相应信息。按条件查找的时候，目前也是只能选择一个条件，不能多条件查找。修改信息的地方也是只能一条一条的修改，不能选择同时修改某几条。这些是还有待改进的地方，努力在之后的实验中改进。

七、源程序

```
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
#include<conio.h>

# define N 1000 //雇主数
# define n 2000 //雇员数

int count=0;
int num=0;
FILE *fp1;
FILE *fp2;

void Read_Er(); //导入雇主信息
void Read_Ee(); //导入雇员信息
void writefile_Er(); //雇主文件保存
void writefile_Ee(); //雇员文件保存
void input_Er(); //录入雇主信息
void input_Ee(); //录入雇员信息
void check_Er(); //查询全部雇主信息
void check_Ee(); //查询全部雇员信息
void find_Er(); //按条件查询雇主信息
```

```

void find_Ee(); //按条件查询雇员信息
void amend_Er(); //修改雇主信息
void amend_Ee(); //修改雇员信息
void delete_Er(); //删除雇主信息
void delete_Ee(); //删除雇员信息
void sort_Er(); //雇主信息排序
void mainMenu(); //主菜单
void menu_Er(); //雇主菜单
void menu_Ee(); //雇员菜单

struct Employer //雇主信息结构体
{
    int id_Er;//雇主id
    char name[5];//雇主名字
    char work[20];//工作名字
    char detail[150];//工作内容
    char require[50];//工作要求
    char location[100];//工作地点
    int time;//工作时间
    int salary;//工资
    char phone[20];//雇主联系电话
}Er[N];

struct Employee //雇员信息结构体
{
    int id_Ee;//雇员id
    char E_name[5];//雇员名字
    char sex[3];//雇员性别
    int age;//雇员年龄
    char resume[100];//个人简历
    char want[50];//想找的工作类型
    int w_time;//理想工作时间
    int w_salary;//理想工资
    char E_phone[20];//雇员联系电话
}Ee[n];

void Read_Er() //导入雇主信息
{
    char filename[20];
    if((fp1=fopen("Er_work.txt","r"))==NULL)
    {
        printf("无雇主信息文件，请新建一个文件\n");
        printf("输入文件名,回车键结束: ");
        scanf("%s",filename);
        if((fp1=fopen(filename,"w"))==NULL)
        {
            printf("新建失败");
            exit(0);
        }
        else
        {
            printf("新建文件成功! ");
            getchar();
            return;
        }
    }
    fseek(fp1,0,2); // 文件内部指针退回到离文件结尾0位置，即指针指向文件内部末尾处
    if(ftell(fp1)>0) // 使用fseek函数后再调用函数ftell()就能非常容易地确定文件的当前位置。
    {
        fseek(fp1,0,0);
        for(count=0;!feof(fp1)&&fread(&Er[count],sizeof(Employer),1,fp1);count++);
        return;
    }
}

void Read_Ee() //导入雇员信息
{

```

```

char filename[20];
if((fp2=fopen("Ee_work.txt", "r"))==NULL)
{
    printf("\n\n无雇员信息文件，请新建一个文件\n");
    printf("输入文件名,回车键结束: ");
    scanf("%s", filename);
    if((fp2=fopen(filename, "w"))==NULL)
    {
        printf("新建失败");
        exit(0);
    }
    else
    {
        printf("新建文件成功! ");
        getchar();
        return;
    }
}
fseek(fp2, 0, 2);           // 文件内部指针退回到离文件结尾0位置，即指针指向文件内部末尾处
if(ftell(fp2)>0)             // 使用fseek函数后再调用函数ftell()就能非常容易地确定文件的当前位置。
{
    fseek(fp2, 0, 0);
    for(num=0; !feof(fp2)&&fread(&Ee[num], sizeof(Employee), 1, fp2); num++);
    return;
}
}

void writefile_Er() //雇主文件保存
{
    system("cls");
    int i;
    int ch;
    if((fp1=fopen("Er_work.txt", "w"))==NULL)
    {
        printf("打开文件失败");
    }
    for(i=0; i<count; i++)
    {
        if(fwrite(&Er[i], sizeof(Employer), 1, fp1)!=1)
            printf("无法读入文件");
        //fwrite(&Er[i], sizeof(Employer), 1, fp1);
    }
    fclose(fp1);
    printf("\n\n\n\n\n\n\t\t\t文件保存成功\n\n\n");
    printf("\t\t\t\t按 1 返回雇主菜单，按 2 返回主菜单，按 3 退出程序\n\n");
    ch=getch();
    switch(ch)
    {
        case '1':
            menu_Er();
            break;
        case '2':
            mainMenu();
            break;
        case '3':
            exit(1);
    }
}

void writefile_Ee() //雇员文件保存
{
    system("cls");
    int i;
    int ch;
    if((fp2=fopen("Ee_work.txt", "w"))==NULL)
    {
        printf("打开文件失败");
    }

```

```

}
for(i=0;i<num;i++)
{
    if(fwrite(&Ee[i],sizeof(Employee),1,fp2)!=1)
        printf("无法读入文件");
    //fwrite(&Ee[i],sizeof(Employee),1,fp2);
}
fclose(fp2);
printf("\n\n\n\n\n\t\t\t文件保存成功\n\n\n");
printf("\t\t\t按 1 返回雇员菜单, 按 2 返回主菜单, 按 3 退出程序\n\n");
ch=getch();
switch(ch)
{
    case '1':
        menu_Ee();
        break;
    case '2':
        mainMenu();
        break;
    case '3':
        exit(1);
}
}

void input_Er() //录入雇主信息
{
    system("cls");
    Er[count].id_Er=count+1;
    printf("\n-----录入雇主信息-----");
    printf("\n您的id: %d\n",Er[count].id_Er);
    printf("\n请输入您的名字: ");
    scanf("%s",Er[count].name);
    printf("\n请输入工作名称: ");
    scanf("%s",Er[count].work);
    printf("\n请输入简要工作内容 (不超过150字): ");
    scanf("%s",Er[count].detail);
    printf("\n请输入基本工作要求 (不超过50字): ");
    scanf("%s",Er[count].require);
    printf("\n请输入详细工作地点 (不超过100字): ");
    scanf("%s",Er[count].location);
    printf("\n请输入工作时间 (以小时为单位): ");
    scanf("%d",&Er[count].time);
    printf("\n请输入工作工资: ");
    scanf("%d",&Er[count].salary);
    printf("\n请输入您的联系电话: ");
    scanf("%s",Er[count].phone);
    count++;
    printf("\n\n是否继续录入工作信息? \n按 1 继续录入, 按 2 返回雇主菜单, 按 3 返回主菜单\n");
    switch(getch())
    {
        case '1':
            input_Er();
            break;
        case '2':
            menu_Er();
            break;
        case '3':
            mainMenu();
            break;
    }
}

void input_Ee() //录入雇员信息
{
    system("cls");
    Ee[num].id_Ee=num+1;
    printf("\n-----录入雇员信息-----");

```

```

printf("\n您的id: %d\n",Ee[num].id_Ee);
printf("\n请输入您的名字: ");
scanf("%s",Ee[num].E_name);
printf("\n请输入您的性别: ");
scanf("%s",Ee[num].sex);
printf("\n请输入您的年龄: ");
scanf("%d",&Ee[num].age);
printf("\n请输入个人简介（不超过100字）: ");
scanf("%s",Ee[num].resume);
printf("\n请输入想找的工作类型（不超过50字）: ");
scanf("%s",Ee[num].want);
printf("\n请输入理想工作时间（以小时为单位）: ");
scanf("%d",&Ee[num].w_time);
printf("\n请输入理想工作工资: ");
scanf("%d",&Ee[num].w_salary);
printf("\n请输入您的联系电话: ");
scanf("%s",Ee[num].E_phone);
num++;
printf("\n\n是否继续录入工作信息? \n按 1 继续录入, 按 2 返回雇员菜单, 按 3 返回主菜单\n");
switch(getch())
{
    case '1':
        input_Ee();
        break;
    case '2':
        menu_Ee();
        break;
    case '3':
        mainMenu();
        break;
}
}

void check_Er() //查询全部雇主信息
{
    system("cls");
    int i;
    if(count==0)
    {
        printf("雇主信息为空! \n");
        printf("是否立刻录入雇主信息? 按 1 录入雇主信息, 按 2 返回雇员菜单, 按 3 返回主菜单\n");
        switch(getch())
        {
            case '1':
                input_Er();
                break;
            case '2':
                menu_Ee();
                break;
            case '3':
                mainMenu();
                break;
        }
    }
    else
    {
        for(i=0;i<count;i++)
        {
            printf("雇主id: %d\n",Er[i].id_Er);
            printf("雇主姓名: %s\n",Er[i].name);
            printf("工作名称: %s\n",Er[i].work);
            printf("工作内容: %s\n",Er[i].detail);
            printf("基本工作要求: %s\n",Er[i].require);
            printf("工作地点: %s\n",Er[i].location);
            printf("工作时间: %d\n",Er[i].time);
            printf("工作工资: %d\n",Er[i].salary);
            printf("联系电话: %s\n",Er[i].phone);

```



```

        printf("\n");
    }
    printf("\n\n按 1 返回雇员菜单, 按 2 返回主菜单\n");
    switch(getch())
    {
        case '1':
            menu_Ee();
            break;
        case '2':
            mainMenu();
            break;
    }
}

void check_Ee() //查询全部雇员信息
{
    system("cls");
    int i;

    if(num==0)
    {
        printf("雇员信息为空! \n");
        printf("是否立刻录入雇员信息? 按 1 录入雇员信息, 按 2 返回雇主菜单, 按 3 返回主菜单\n");
        switch(getch())
        {
            case '1':
                input_Ee();
                break;
            case '2':
                menu_Er();
                break;
            case '3':
                mainMenu();
                break;
        }
    }
    else
    {
        for(i=0;i<num;i++)
        {
            printf("雇员id: %d\n",Ee[i].id_Ee);
            printf("雇员姓名: %s\n",Ee[i].E_name);
            printf("雇员性别: %s\n",Ee[i].sex);
            printf("雇员年龄: %d\n",Ee[i].age);
            printf("个人简历: %s\n",Ee[i].resume);
            printf("理想工作类型: %s\n",Ee[i].want);
            printf("理想工作时间: %d\n",Ee[i].w_time);
            printf("理想工作工资: %d\n",Ee[i].w_salary);
            printf("联系电话: %s\n",Ee[i].E_phone);
            printf("\n");
        }
        printf("\n\n按 1 返回雇主菜单, 按 2 返回主菜单\n");
        switch(getch())
        {
            case '1':
                menu_Er();
                break;
            case '2':
                mainMenu();
                break;
        }
    }
}

void menu_Er() //雇主菜单
{

```

```

system("cls");
char x;

printf("\n\n\n");
printf("\t\t☆☆☆☆☆☆☆☆☆☆☆☆☆☆☆☆\n\n");
printf("\t\t*****欢迎来到雇主菜单*****\n\n");
printf("\t\t*****\t 1、录入雇主信息\t*****\n\n");
printf("\t\t*****\t 2、查询全部雇员信息\t*****\n\n");
printf("\t\t*****\t 3、按条件查找雇员信息\t*****\n\n");
printf("\t\t*****\t 4、修改雇主信息\t*****\n\n");
printf("\t\t*****\t 5、删除雇主信息\t*****\n\n");
printf("\t\t*****\t 6、保存文件\t*****\n\n");
printf("\t\t*****\t 7、返回主菜单\t*****\n\n");
printf("\t\t*****\t 8、直接退出程序\t*****\n\n");
printf("\t\t*****\t 请选择输入 (1~8) \t*****\n\n");
printf("\t\t*****\n\n");

x=getch();
switch(x)
{
    case '1':
        input_Er();
        break;
    case '2':
        check_Ee();
        break;
    case '3':
        find_Ee();
        break;
    case '4':
        amend_Er();
        break;
    case '5':
        delete_Er();
        break;
    case '6':
        writefile_Er();
        break;
    case '7':
        mainMenu();
        break;
    case '8':
        exit(1);
}
}

void menu_Ee() //雇员菜单
{
    system("cls");
    char x;

    printf("\n\n\n");
    printf("\t\t☆☆☆☆☆☆☆☆☆☆☆☆☆☆☆☆\n\n");
    printf("\t\t*****欢迎来到雇员菜单*****\n\n");
    printf("\t\t*****\t 1、录入雇员信息\t*****\n\n");
    printf("\t\t*****\t 2、查询全部雇主信息\t*****\n\n");
    printf("\t\t*****\t 3、按条件查找雇主信息\t*****\n\n");
    printf("\t\t*****\t 4、修改雇员信息\t*****\n\n");
    printf("\t\t*****\t 5、删除雇员信息\t*****\n\n");
    printf("\t\t*****\t 6、雇主信息排序\t*****\n\n");
    printf("\t\t*****\t 7、保存文件\t*****\n\n");
    printf("\t\t*****\t 8、返回主菜单\t*****\n\n");
    printf("\t\t*****\t 9、直接退出程序\t*****\n\n");
    printf("\t\t*****\t 请选择输入 (1~9) \t*****\n\n");
    printf("\t\t*****\n\n");

    x=getch();

```

```

switch(x)
{
    case '1':
        input_Ee();
        break;
    case '2':
        check_Er();
        break;
    case '3':
        find_Er();
        break;
    case '4':
        amend_Ee();
        break;
    case '5':
        delete_Ee();
        break;
    case '6':
        sort_Er();
        break;
    case '7':
        writefile_Ee();
        break;
    case '8':
        mainMenu();
        break;
    case '9':
        exit(1);
}
}

void mainMenu() //主菜单
{
    system("cls");
    char x;

    printf("\n\n\n");
    printf("\t\t☆☆☆☆☆☆☆☆☆☆☆☆☆☆☆☆☆☆\n\n");
    printf("\t\t*****欢迎来到工作管理系统主菜单*****\n\n");
    printf("\t\t*****\t 1、我是雇主! \t*****\n\n");
    printf("\t\t*****\t 2、我是雇员! \t*****\n\n");
    printf("\t\t*****\t 3、直接退出程序\t*****\n\n");
    printf("\t\t*****\t 请选择输入 (1~3) \t*****\n\n");
    printf("\t\t*****\n\n");

    x=getch();
    switch(x)
    {
        case '1':
            menu_Er();
            break;
        case '2':
            menu_Ee();
            break;
        case '3':
            exit(1);
    }
}

void delete_Er() //删除雇主信息
{
    system("cls");
    if(count==0)
    {
        printf("雇主信息为空! \n");
        printf("是否立刻录入雇主信息? 按 1 录入雇主信息, 按 2 返回雇主菜单, 按 3 返回主菜单\n");
        switch(getch())

```

```

{
    case '1':
        input_Er();
        break;
    case '2':
        menu_Er();
        break;
    case '3':
        mainMenu();
        break;
}
}
else
{
    int id;
    int i;
    int j;
    printf("请输入要删除的雇主id: \n");
    scanf("%d",&id);
    printf("\n");
    for(i=0;i<count;i++)
    {
        if(Er[i].id_Er==id)
        {
            break;
        }
    }
    for(j=i;j<=count-1;j++)
    {
        Er[j].id_Er=Er[j+1].id_Er-1;
        for(int k=0;k<5;k++)
        {
            Er[j].name[k]=Er[j+1].name[k];
        }
        for(int k=0;k<20;k++)
        {
            Er[j].work[k]=Er[j+1].work[k];
        }
        for(int k=0;k<150;k++)
        {
            Er[j].detail[k]=Er[j+1].detail[k];
        }
        for(int k=0;k<50;k++)
        {
            Er[j].require[k]=Er[j+1].require[k];
        }
        for(int k=0;k<100;k++)
        {
            Er[j].location[k]=Er[j+1].location[k];
        }
        Er[j].time=Er[j+1].time;
        Er[j].salary=Er[j+1].salary;
        for(int k=0;k<20;k++)
        {
            Er[j].phone[k]=Er[j+1].phone[k];
        }
    }
    count--;
    printf("删除成功!\n");
    printf("\n\n按 1 返回雇主菜单, 按 2 返回主菜单\n");
    switch(getch())
    {
        case '1':
            menu_Er();
            break;
        case '2':
            mainMenu();

```

```

        break;
    }
}

void delete_Ee()    //删除雇员信息
{
    system("cls");
    if(num==0)
    {
        printf("雇员信息为空! \n");
        printf("是否立刻录入雇员信息? 按 1 录入雇员信息, 按 2 返回雇员菜单, 按 3 返回主菜单\n");
        switch(getch())
        {
            case '1':
                input_Ee();
                break;
            case '2':
                menu_Ee();
                break;
            case '3':
                mainMenu();
                break;
        }
    }
    else
    {
        int id;
        int i;
        int j;
        printf("请输入要删除的雇员id: \n");
        scanf("%d",&id);
        printf("\n");
        for(i=0;i<num;i++)
        {
            if(Ee[i].id_Ee==id)
            {
                break;
            }
        }
        for(j=i;j<=num-1;j++)
        {
            Ee[j].id_Ee=Ee[j+1].id_Ee-1;
            for(int k=0;k<5;k++)
            {
                Ee[j].E_name[k]=Ee[j+1].E_name[k];
            }
            for(int k=0;k<3;k++)
            {
                Ee[j].sex[k]=Ee[j+1].sex[k];
            }
            Ee[j].age=Ee[j+1].age;
            for(int k=0;k<100;k++)
            {
                Ee[j].resume[k]=Ee[j+1].resume[k];
            }
            for(int k=0;k<50;k++)
            {
                Ee[j].want[k]=Ee[j+1].want[k];
            }
            Ee[j].w_time=Ee[j+1].w_time;
            Ee[j].w_salary=Ee[j+1].w_salary;
            for(int k=0;k<20;k++)
            {
                Ee[j].E_phone[k]=Ee[j+1].E_phone[k];
            }
        }
    }
}

```

```

num--;
printf("删除成功!\n");
printf("\n\n按 1 返回雇员菜单, 按 2 返回主菜单\n");
switch(getch())
{
    case '1':
        menu_Ee();
        break;
    case '2':
        mainMenu();
        break;
}
}
}

void amend_Er() //修改雇主信息
{
    system("cls");
    if(count==0)
    {
        printf("雇主信息为空! \n");
        printf("是否立刻录入雇主信息? 按 1 录入雇主信息, 按 2 返回雇主菜单, 按 3 返回主菜单\n");
        switch(getch())
        {
            case '1':
                input_Er();
                break;
            case '2':
                menu_Er();
                break;
            case '3':
                mainMenu();
                break;
        }
    }
    else
    {
        int id;
        int i;
        printf("请输入要修改的雇主id: \n");
        scanf("%d",&id);
        printf("\n");
        for(i=0;i<count;i++)
        {
            if(Er[i].id_Er==id)
            {
                printf("雇主id: %d\n",Er[i].id_Er);
                printf("雇主姓名: %s\n",Er[i].name);
                printf("工作名称: %s\n",Er[i].work);
                printf("工作内容: %s\n",Er[i].detail);
                printf("基本工作要求: %s\n",Er[i].require);
                printf("工作地点: %s\n",Er[i].location);
                printf("工作时间: %d\n",Er[i].time);
                printf("工作工资: %d\n",Er[i].salary);
                printf("联系电话: %s\n",Er[i].phone);
                break;
            }
        }
        printf("\n请选择您要修改的雇主信息: \n");
        printf("1、雇主姓名\n");
        printf("2、工作名称\n");
        printf("3、工作内容\n");
        printf("4、工作要求\n");
        printf("5、工作地点\n");
        printf("6、工作时间\n");
        printf("7、工作工资\n");
        printf("8、联系电话\n");
    }
}

```

```
switch(getch())
{
    case '1':
    {
        printf("请输入新的雇主姓名: \n");
        scanf("%s",Er[i].name);
        break;
    }
    case '2':
    {
        printf("请输入新的工作名称: \n");
        scanf("%s",Er[i].work);
        break;
    }
    case '3':
    {
        printf("请输入新的工作内容: \n");
        scanf("%s",Er[i].detail);
        break;
    }
    case '4':
    {
        printf("请输入新的工作要求: \n");
        scanf("%s",Er[i].require);
        break;
    }
    case '5':
    {
        printf("请输入新的工作地点: \n");
        scanf("%s",Er[i].location);
        break;
    }
    case '6':
    {
        printf("请输入新的工作时间: \n");
        scanf("%d",&Er[i].time);
        break;
    }
    case '7':
    {
        printf("请输入新的工作工资: \n");
        scanf("%d",&Er[i].salary);
        break;
    }

    case '8':
    {
        printf("请输入新的联系电话: \n");
        scanf("%s",Er[i].phone);
        break;
    }
}
printf("修改成功! \n");
printf("修改后的雇主信息如下: \n");
printf("\n雇主id: %d\n",Er[i].id_Er);
printf("雇主姓名: %s\n",Er[i].name);
printf("工作名称: %s\n",Er[i].work);
printf("工作内容: %s\n",Er[i].detail);
printf("基本工作要求: %s\n",Er[i].require);
printf("工作地点: %s\n",Er[i].location);
printf("工作时间: %d\n",Er[i].time);
printf("工作工资: %d\n",Er[i].salary);
printf("联系电话: %s\n",Er[i].phone);

printf("\n\n按 1 返回雇主菜单, 按 2 返回主菜单\n");
switch(getch())
{
```

```

        case '1':
            menu_Er();
            break;
        case '2':
            mainMenu();
            break;
    }
}
}

void amend_Ee() //修改雇员信息
{
    system("cls");
    if(num==0)
    {
        printf("雇员信息为空! \n");
        printf("是否立刻录入雇员信息? 按 1 录入雇员信息, 按 2 返回雇员菜单, 按 3 返回主菜单\n");
        switch(getch())
        {
            case '1':
                input_Ee();
                break;
            case '2':
                menu_Ee();
                break;
            case '3':
                mainMenu();
                break;
        }
    }
    else
    {
        int id;
        int i;
        printf("请输入要修改的雇员id: \n");
        scanf("%d",&id);
        printf("\n");
        for(i=0;i<num;i++)
        {
            if(Ee[i].id_Ee==id)
            {
                printf("雇员id: %d\n",Ee[i].id_Ee);
                printf("雇员姓名: %s\n",Ee[i].E_name);
                printf("雇员性别: %s\n",Ee[i].sex);
                printf("雇员年龄: %d\n",Ee[i].age);
                printf("个人简历: %s\n",Ee[i].resume);
                printf("理想工作类型: %s\n",Ee[i].want);
                printf("理想工作时间: %d\n",Ee[i].w_time);
                printf("理想工作工资: %d\n",Ee[i].w_salary);
                printf("联系电话: %s\n",Ee[i].E_phone);
                break;
            }
        }
        printf("\n请选择您要修改的雇员信息: \n");
        printf("1、雇员姓名\n");
        printf("2、雇员性别\n");
        printf("3、雇员年龄\n");
        printf("4、个人简历\n");
        printf("5、理想工作类型\n");
        printf("6、理想工作时间\n");
        printf("7、理想工作工资\n");
        printf("8、联系电话\n");
        switch(getch())
        {
            case '1':
                {
                    printf("请输入新的雇员姓名: \n");

```



```

        scanf("%s",Ee[i].E_name);
        break;
    }
    case '2':
    {
        printf("请输入新的雇员性别: \n");
        scanf("%s",Ee[i].sex);
        break;
    }
    case '3':
    {
        printf("请输入新的雇员年龄: \n");
        scanf("%d",&Ee[i].age);
        break;
    }
    case '4':
    {
        printf("请输入新的个人简历: \n");
        scanf("%s",Ee[i].resume);
        break;
    }
    case '5':
    {
        printf("请输入新的理想工作类型: \n");
        scanf("%s",Ee[i].want);
        break;
    }
    case '6':
    {
        printf("请输入新的理想工作时间: \n");
        scanf("%d",&Ee[i].w_time);
        break;
    }
    case '7':
    {
        printf("请输入新的理想工作工资: \n");
        scanf("%d",&Ee[i].w_salary);
        break;
    }

    case '8':
    {
        printf("请输入新的联系电话: \n");
        scanf("%s",Ee[i].E_phone);
        break;
    }
}
printf("修改成功! \n");
printf("修改后的雇主信息如下: \n");
printf("雇员id: %d\n",Ee[i].id_Ee);
printf("雇员姓名: %s\n",Ee[i].E_name);
printf("雇员性别: %s\n",Ee[i].sex);
printf("雇员年龄: %d\n",Ee[i].age);
printf("个人简历: %s\n",Ee[i].resume);
printf("理想工作类型: %s\n",Ee[i].want);
printf("理想工作时间: %d\n",Ee[i].w_time);
printf("理想工作工资: %d\n",Ee[i].w_salary);
printf("联系电话: %s\n",Ee[i].E_phone);

printf("\n\n按 1 返回雇员菜单, 按 2 返回主菜单\n");
switch(getch())
{
    case '1':
        menu_Ee();
        break;
    case '2':
        mainMenu();

```

```

        break;
    }
}

void find_Er() //按条件查询雇主信息
{
    system("cls");
    if(count==0)
    {
        printf("雇主信息为空! \n");
        printf("是否立刻录入雇主信息? 按 1 录入雇主信息, 按 2 返回雇员菜单, 按 3 返回主菜单\n");
        switch(getch())
        {
            case '1':
                input_Er();
                break;
            case '2':
                menu_Ee();
                break;
            case '3':
                mainMenu();
                break;
        }
    }
    else
    {
        int i;
        printf("\n\n请选择您想查找的雇主信息: \n");
        printf("1、雇主id\n");
        printf("2、雇主姓名\n");
        printf("3、工作名称\n");
        printf("4、工作地点\n");
        printf("5、工作时间\n");
        printf("6、工作工资\n");
        printf("7、联系电话\n");
        printf("\n");
        switch(getch())
        {
            case '1':
                {
                    printf("按雇主id查找\n");
                    int new_id;
                    scanf("%d",&new_id);
                    for(i=0;i<count;i++)
                    {
                        if(Er[i].id_Er==new_id)
                        {
                            printf("雇主id: %d\n",Er[i].id_Er);
                            printf("雇主姓名: %s\n",Er[i].name);
                            printf("工作名称: %s\n",Er[i].work);
                            printf("工作内容: %s\n",Er[i].detail);
                            printf("基本工作要求: %s\n",Er[i].require);
                            printf("工作地点: %s\n",Er[i].location);
                            printf("工作时间: %d\n",Er[i].time);
                            printf("工作工资: %d\n",Er[i].salary);
                            printf("联系电话: %s\n",Er[i].phone);
                            printf("\n");
                        }
                    }
                    break;
                }
            case '2':
                {
                    printf("按雇主姓名查找\n");
                    char new_name[5];
                    scanf("%s",new_name);

```

```

for(i=0;i<count;i++)
{
    if(strcmp(new_name,Er[i].name)==0)
    {
        printf("雇主id: %d\n",Er[i].id_Er);
        printf("雇主姓名: %s\n",Er[i].name);
        printf("工作名称: %s\n",Er[i].work);
        printf("工作内容: %s\n",Er[i].detail);
        printf("基本工作要求: %s\n",Er[i].require);
        printf("工作地点: %s\n",Er[i].location);
        printf("工作时间: %d\n",Er[i].time);
        printf("工作工资: %d\n",Er[i].salary);
        printf("联系电话: %s\n",Er[i].phone);
        printf("\n");
    }
}
break;
}
case '3':
{
    printf("按雇主工作名称查找\n");
    char new_work[20];
    scanf("%s",new_work);
    for(i=0;i<count;i++)
    {
        if(strcmp(new_work,Er[i].work)==0)
        {
            printf("雇主id: %d\n",Er[i].id_Er);
            printf("雇主姓名: %s\n",Er[i].name);
            printf("工作名称: %s\n",Er[i].work);
            printf("工作内容: %s\n",Er[i].detail);
            printf("基本工作要求: %s\n",Er[i].require);
            printf("工作地点: %s\n",Er[i].location);
            printf("工作时间: %d\n",Er[i].time);
            printf("工作工资: %d\n",Er[i].salary);
            printf("联系电话: %s\n",Er[i].phone);
            printf("\n");
        }
    }
    break;
}
case '4':
{
    printf("按雇主工作地点查找\n");
    char new_location[100];
    scanf("%s",new_location);
    for(i=0;i<count;i++)
    {
        if(strcmp(new_location,Er[i].location)==0)
        {
            printf("雇主id: %d\n",Er[i].id_Er);
            printf("雇主姓名: %s\n",Er[i].name);
            printf("工作名称: %s\n",Er[i].work);
            printf("工作内容: %s\n",Er[i].detail);
            printf("基本工作要求: %s\n",Er[i].require);
            printf("工作地点: %s\n",Er[i].location);
            printf("工作时间: %d\n",Er[i].time);
            printf("工作工资: %d\n",Er[i].salary);
            printf("联系电话: %s\n",Er[i].phone);
            printf("\n");
        }
    }
    break;
}
case '5':
{
    printf("按雇主工作时间查找\n");

```

```

int new_time;
scanf("%d",&new_time);
for(i=0;i<count;i++)
{
    if(new_time==Er[i].time)
    {
        printf("雇主id: %d\n",Er[i].id_Er);
        printf("雇主姓名: %s\n",Er[i].name);
        printf("工作名称: %s\n",Er[i].work);
        printf("工作内容: %s\n",Er[i].detail);
        printf("基本工作要求: %s\n",Er[i].require);
        printf("工作地点: %s\n",Er[i].location);
        printf("工作时间: %d\n",Er[i].time);
        printf("工作工资: %d\n",Er[i].salary);
        printf("联系电话: %s\n",Er[i].phone);
        printf("\n");
    }
}
break;
}
case '6':
{
    printf("按雇主工作工资查找\n");
    int new_salary;
    scanf("%d",&new_salary);
    for(i=0;i<count;i++)
    {
        if(new_salary==Er[i].salary)
        {
            printf("雇主id: %d\n",Er[i].id_Er);
            printf("雇主姓名: %s\n",Er[i].name);
            printf("工作名称: %s\n",Er[i].work);
            printf("工作内容: %s\n",Er[i].detail);
            printf("基本工作要求: %s\n",Er[i].require);
            printf("工作地点: %s\n",Er[i].location);
            printf("工作时间: %d\n",Er[i].time);
            printf("工作工资: %d\n",Er[i].salary);
            printf("联系电话: %s\n",Er[i].phone);
            printf("\n");
        }
    }
    break;
}
case '7':
{
    printf("按雇主联系电话查找\n");
    char new_phone[20];
    scanf("%s",new_phone);
    for(i=0;i<count;i++)
    {
        if(new_phone==Er[i].phone)
        {
            printf("雇主id: %d\n",Er[i].id_Er);
            printf("雇主姓名: %s\n",Er[i].name);
            printf("工作名称: %s\n",Er[i].work);
            printf("工作内容: %s\n",Er[i].detail);
            printf("基本工作要求: %s\n",Er[i].require);
            printf("工作地点: %s\n",Er[i].location);
            printf("工作时间: %d\n",Er[i].time);
            printf("工作工资: %d\n",Er[i].salary);
            printf("联系电话: %s\n",Er[i].phone);
            printf("\n");
        }
    }
    break;
}
}

```

```

printf("\n\n按 1 返回雇员菜单, 按 2 返回主菜单\n");
switch(getch())
{
    case '1':
        menu_Ee();
        break;
    case '2':
        mainMenu();
        break;
}
}
}

void find_Ee() //按条件查找雇员信息
{
    system("cls");
    if(num==0)
    {
        printf("雇员信息为空! \n");
        printf("是否立刻录入雇员信息? 按 1 录入雇员信息, 按 2 返回雇主菜单, 按 3 返回主菜单\n");
        switch(getch())
        {
            case '1':
                input_Ee();
                break;
            case '2':
                menu_Er();
                break;
            case '3':
                mainMenu();
                break;
        }
    }
    else
    {
        int i;
        char choice;
        printf("\n\n请选择您要查找的雇员信息: \n");
        printf("1、雇员id\n");
        printf("2、雇员姓名\n");
        printf("3、雇员性别\n");
        printf("4、雇员年龄\n");
        printf("5、理想工作类型\n");
        printf("6、理想工作时间\n");
        printf("7、理想工作工资\n");
        printf("8、联系电话\n");
        printf("\n");
        choice=getch();
        switch(choice)
        {
            case '1':
                {
                    printf("按雇员id查找\n");
                    int new_id;
                    scanf("%d",&new_id);
                    for(i=0;i<num;i++)
                    {
                        if(Ee[i].id_Ee==new_id)
                        {
                            printf("雇员id: %d\n",Ee[i].id_Ee);
                            printf("雇员姓名: %s\n",Ee[i].E_name);
                            printf("雇员性别: %s\n",Ee[i].sex);
                            printf("雇员年龄: %d\n",Ee[i].age);
                            printf("个人简历: %s\n",Ee[i].resume);
                            printf("理想工作类型: %s\n",Ee[i].want);
                            printf("理想工作时间: %d\n",Ee[i].w_time);
                        }
                    }
                }
            }
        }
    }
}

```

```

        printf("理想工作工资: %d\n", Ee[i].w_salary);
        printf("联系电话: %s\n", Ee[i].E_phone);
    }
}
break;
}
case '2':
{
    printf("按雇员姓名查找\n");
    char new_name[5];
    scanf("%s", new_name);
    printf("\n");
    for(i=0; i<num; i++)
    {
        if(strcmp(new_name, Ee[i].E_name)==0)
        {
            printf("雇员id: %d\n", Ee[i].id_Ee);
            printf("雇员姓名: %s\n", Ee[i].E_name);
            printf("雇员性别: %s\n", Ee[i].sex);
            printf("雇员年龄: %d\n", Ee[i].age);
            printf("个人简历: %s\n", Ee[i].resume);
            printf("理想工作类型: %s\n", Ee[i].want);
            printf("理想工作时间: %d\n", Ee[i].w_time);
            printf("理想工作工资: %d\n", Ee[i].w_salary);
            printf("联系电话: %s\n", Ee[i].E_phone);
        }
    }
    break;
}
case '3':
{
    printf("按雇员性别查找\n");
    char new_sex[3];
    scanf("%s", new_sex);
    for(i=0; i<num; i++)
    {
        if(strcmp(new_sex, Ee[i].sex)==0)
        {
            printf("雇员id: %d\n", Ee[i].id_Ee);
            printf("雇员姓名: %s\n", Ee[i].E_name);
            printf("雇员性别: %s\n", Ee[i].sex);
            printf("雇员年龄: %d\n", Ee[i].age);
            printf("个人简历: %s\n", Ee[i].resume);
            printf("理想工作类型: %s\n", Ee[i].want);
            printf("理想工作时间: %d\n", Ee[i].w_time);
            printf("理想工作工资: %d\n", Ee[i].w_salary);
            printf("联系电话: %s\n", Ee[i].E_phone);
        }
    }
    break;
}
case '4':
{
    printf("按雇员年龄查找\n");
    int new_age;
    scanf("%d", &new_age);
    for(i=0; i<num; i++)
    {
        if(new_age==Ee[i].age)
        {
            printf("雇员id: %d\n", Ee[i].id_Ee);
            printf("雇员姓名: %s\n", Ee[i].E_name);
            printf("雇员性别: %s\n", Ee[i].sex);
            printf("雇员年龄: %d\n", Ee[i].age);
            printf("个人简历: %s\n", Ee[i].resume);
            printf("理想工作类型: %s\n", Ee[i].want);
            printf("理想工作时间: %d\n", Ee[i].w_time);

```

```

        printf("理想工作工资: %d\n", Ee[i].w_salary);
        printf("联系电话: %s\n", Ee[i].E_phone);
    }
}
break;
}
case '5':
{
    printf("按雇员理想工作类型查找\n");
    char new_want[50];
    scanf("%s", new_want);
    for(i=0; i<num; i++)
    {
        if(strcmp(new_want, Ee[i].want)==0)
        {
            printf("雇员id: %d\n", Ee[i].id_Ee);
            printf("雇员姓名: %s\n", Ee[i].E_name);
            printf("雇员性别: %s\n", Ee[i].sex);
            printf("雇员年龄: %d\n", Ee[i].age);
            printf("个人简历: %s\n", Ee[i].resume);
            printf("理想工作类型: %s\n", Ee[i].want);
            printf("理想工作时间: %d\n", Ee[i].w_time);
            printf("理想工作工资: %d\n", Ee[i].w_salary);
            printf("联系电话: %s\n", Ee[i].E_phone);
        }
    }
    break;
}
case '6':
{
    printf("按雇员理想工作时间查找\n");
    int new_w_time;
    scanf("%d", &new_w_time);
    for(i=0; i<num; i++)
    {
        if(new_w_time==Ee[i].w_time)
        {
            printf("雇员id: %d\n", Ee[i].id_Ee);
            printf("雇员姓名: %s\n", Ee[i].E_name);
            printf("雇员性别: %s\n", Ee[i].sex);
            printf("雇员年龄: %d\n", Ee[i].age);
            printf("个人简历: %s\n", Ee[i].resume);
            printf("理想工作类型: %s\n", Ee[i].want);
            printf("理想工作时间: %d\n", Ee[i].w_time);
            printf("理想工作工资: %d\n", Ee[i].w_salary);
            printf("联系电话: %s\n", Ee[i].E_phone);
        }
    }
    break;
}
case '7':
{
    printf("按雇员理想工作工资查找\n");
    int new_w_salary;
    scanf("%d", &new_w_salary);
    for(i=0; i<num; i++)
    {
        if(new_w_salary==Ee[i].w_salary)
        {
            printf("雇员id: %d\n", Ee[i].id_Ee);
            printf("雇员姓名: %s\n", Ee[i].E_name);
            printf("雇员性别: %s\n", Ee[i].sex);
            printf("雇员年龄: %d\n", Ee[i].age);
            printf("个人简历: %s\n", Ee[i].resume);
            printf("理想工作类型: %s\n", Ee[i].want);
            printf("理想工作时间: %d\n", Ee[i].w_time);
            printf("理想工作工资: %d\n", Ee[i].w_salary);

```

```

        printf("联系电话: %s\n", Ee[i].E_phone);
    }
}
break;
}
case '8':
{
    printf("按雇员联系电话查找\n");
    char new_E_phone[20];
    scanf("%s", new_E_phone);
    for(i=0; i<num; i++)
    {
        if(strcmp(new_E_phone, Ee[i].E_phone)==0)
        {
            printf("雇员id: %d\n", Ee[i].id_Ee);
            printf("雇员姓名: %s\n", Ee[i].E_name);
            printf("雇员性别: %s\n", Ee[i].sex);
            printf("雇员年龄: %d\n", Ee[i].age);
            printf("个人简历: %s\n", Ee[i].resume);
            printf("理想工作类型: %s\n", Ee[i].want);
            printf("理想工作时间: %d\n", Ee[i].w_time);
            printf("理想工作工资: %d\n", Ee[i].w_salary);
            printf("联系电话: %s\n", Ee[i].E_phone);
        }
    }
    break;
}
}

printf("\n\n按 1 返回雇主菜单, 按 2 返回主菜单\n");
switch(getch())
{
    case '1':
        menu_Er();
        break;
    case '2':
        mainMenu();
        break;
}
}
}

void sort_Er() //雇主信息排序
{
    system("cls");
    if(count==0)
    {
        printf("雇主信息为空! \n");
        printf("是否立刻录入雇主信息? 按 1 录入雇主信息, 按 2 返回雇员菜单, 按 3 返回主菜单\n");
        switch(getch())
        {
            case '1':
                input_Er();
                break;
            case '2':
                menu_Ee();
                break;
            case '3':
                mainMenu();
                break;
        }
    }
    else
    {
        printf("-----按 1 工资升序排列, 按 2 工资降序排列-----\n");
        switch(getch())
        {

```



```

case '1':
{
    for(int i=0;i<count-1;i++)
    {
        for(int j=0;j<count-1-i;j++)
        {
            if(Er[j].salary>Er[j+1].salary)
            {
                int temp;
                temp=Er[j].salary;
                Er[j].salary=Er[j+1].salary;
                Er[j+1].salary=temp;

                temp=Er[j].id_Er;
                Er[j].id_Er=Er[j+1].id_Er;
                Er[j+1].id_Er=temp;
                for(int k=0;k<5;k++)
                {
                    char temp[5];
                    temp[k]=Er[j].name[k];
                    Er[j].name[k]=Er[j+1].name[k];
                    Er[j+1].name[k]=temp[k];
                }
                for(int k=0;k<20;k++)
                {
                    char temp[20];
                    temp[k]=Er[j].work[k];
                    Er[j].work[k]=Er[j+1].work[k];
                    Er[j+1].work[k]=temp[k];
                }
                for(int k=0;k<150;k++)
                {
                    char temp[150];
                    temp[k]=Er[j].detail[k];
                    Er[j].detail[k]=Er[j+1].detail[k];
                    Er[j+1].detail[k]=temp[k];
                }
                for(int k=0;k<50;k++)
                {
                    char temp[50];
                    temp[k]=Er[j].require[k];
                    Er[j].require[k]=Er[j+1].require[k];
                    Er[j+1].require[k]=temp[k];
                }
                for(int k=0;k<100;k++)
                {
                    char temp[100];
                    temp[k]=Er[j].location[k];
                    Er[j].location[k]=Er[j+1].location[k];
                    Er[j+1].location[k]=temp[k];
                }
                temp=Er[j].time;
                Er[j].time=Er[j+1].time;
                Er[j+1].time=temp;
                for(int k=0;k<20;k++)
                {
                    char temp[20];
                    temp[k]=Er[j].phone[k];
                    Er[j].phone[k]=Er[j+1].phone[k];
                    Er[j+1].phone[k]=temp[k];
                }
            }
        }
    }
}
printf("工资升序排列后的雇主信息: \n");
for(int i=0;i<count;i++)
{

```

```

        printf("雇主id: %d\n", Er[i].id_Er);
        printf("雇主姓名: %s\n", Er[i].name);
        printf("工作名称: %s\n", Er[i].work);
        printf("工作内容: %s\n", Er[i].detail);
        printf("基本工作要求: %s\n", Er[i].require);
        printf("工作地点: %s\n", Er[i].location);
        printf("工作时间: %d\n", Er[i].time);
        printf("工作工资: %d\n", Er[i].salary);
        printf("联系电话: %s\n", Er[i].phone);
        printf("\n");
    }
    break;
}
case '2':
{
    for(int i=0; i<count-1; i++)
    {
        for(int j=0; j<count-1-i; j++)
        {
            if(Er[j].salary<Er[j+1].salary)
            {
                int temp;
                temp=Er[j].salary;
                Er[j].salary=Er[j+1].salary;
                Er[j+1].salary=temp;

                temp=Er[j].id_Er;
                Er[j].id_Er=Er[j+1].id_Er;
                Er[j+1].id_Er=temp;
                for(int k=0; k<5; k++)
                {
                    char temp[5];
                    temp[k]=Er[j].name[k];
                    Er[j].name[k]=Er[j+1].name[k];
                    Er[j+1].name[k]=temp[k];
                }
                for(int k=0; k<20; k++)
                {
                    char temp[20];
                    temp[k]=Er[j].work[k];
                    Er[j].work[k]=Er[j+1].work[k];
                    Er[j+1].work[k]=temp[k];
                }
                for(int k=0; k<150; k++)
                {
                    char temp[150];
                    temp[k]=Er[j].detail[k];
                    Er[j].detail[k]=Er[j+1].detail[k];
                    Er[j+1].detail[k]=temp[k];
                }
                for(int k=0; k<50; k++)
                {
                    char temp[50];
                    temp[k]=Er[j].require[k];
                    Er[j].require[k]=Er[j+1].require[k];
                    Er[j+1].require[k]=temp[k];
                }
                for(int k=0; k<100; k++)
                {
                    char temp[100];
                    temp[k]=Er[j].location[k];
                    Er[j].location[k]=Er[j+1].location[k];
                    Er[j+1].location[k]=temp[k];
                }
                temp=Er[j].time;
                Er[j].time=Er[j+1].time;
                Er[j+1].time=temp;
            }
        }
    }
}

```

```

        for(int k=0;k<20;k++)
        {
            char temp[20];
            temp[k]=Er[j].phone[k];
            Er[j].phone[k]=Er[j+1].phone[k];
            Er[j+1].phone[k]=temp[k];
        }
    }
}

printf("工资降序排列后的雇主信息: \n");
for(int i=0;i<count;i++)
{
    printf("雇主id: %d\n",Er[i].id_Er);
    printf("雇主姓名: %s\n",Er[i].name);
    printf("工作名称: %s\n",Er[i].work);
    printf("工作内容: %s\n",Er[i].detail);
    printf("基本工作要求: %s\n",Er[i].require);
    printf("工作地点: %s\n",Er[i].location);
    printf("工作时间: %d\n",Er[i].time);
    printf("工作工资: %d\n",Er[i].salary);
    printf("联系电话: %s\n",Er[i].phone);
    printf("\n");
}
break;
}

}

printf("\n\n按 1 返回雇员菜单, 按 2 返回主菜单\n");
switch(getch())
{
    case '1':
        menu_Ee();
        break;
    case '2':
        mainMenu();
        break;
}
}

}

int main()
{
    system("color 0B");
    Read_Er();
    Read_Ee();
    mainMenu();
}

```