计算机程序设计 II 实验课程设计报告

学号：23052023　　　姓名：冯凯

一、系统介绍

职工管理系统是一个用于管理公司或组织职工信息的应用程序。该系统旨在通过数字化和自动化的方式简化职工信息的管理过程，提高工作效率和信息的准确度。系统的主要功能包括增加职工、显示职工信息、删除职工、查找职工、更新职工信息、排序职工以及清空文档等。

二、系统设计

1. 项目各个文件的设计

职工管理系统.cpp	各功能的存放，用 while 和 switch 实现
wokerManager.h	功能的添加修改
worker.h	职工抽象类
manager.h	经理类
employee.h	普通职工类
boss.h	总裁类

2. 函数设计

设计多个头文件是为了以后维护和修改系统更加方便，也是为了熟练自己的编程能力而设计，此外多个头文件的作用还能够使主函数更加简洁美观，显而易见识别出每个文件的作用。

三、功能实现

在这里挑几个重要的函数拿出来介绍。即核心的代码模块加以介绍和解释。

功能 1：展示菜单

先在 workerManager.h 文件中添加 void Show_Menu();函数，接着在 workerManager.cpp 输入以下代码

```
//展示菜单
void WorkerManager::Show_Menu()
```

```
{
    cout << "**************************************" << endl;
    cout << "*******  欢迎使用职工管理系统！ *******" << endl;
    cout << "********    0.退出管理程序   ********" << endl;
    cout << "********    1.增加职工信息   ********" << endl;
    cout << "********    2.显示职工信息   ********" << endl;
    cout << "********    3.删除离职职工   ********" << endl;
    cout << "********    4.修改职工信息   ********" << endl;
    cout << "********    5.查找职工信息   ********" << endl;
    cout << "********    6.安排编号排序   ********" << endl;
    cout << "********    7.清空所有文档   ********" << endl;
    cout << "*************************************" << endl;
    cout << endl;
}
```

最后在职工管理系统.cpp 中调用 wm.Show_Menu();

功能 2：退出系统

先在 workerManager.h 文件中添加 void ExitSystem();函数，接着在 workerManager.cpp 输入以下代码

```
//退出系统
void WorkerManager::ExitSystem()
{
    cout << "欢迎下次使用！" << endl;
    system("pause");
    exit(0); //退出程序
}
```

最后在职工管理系统.cpp 中调用 wm.ExitSystem();

功能 3：增加职工

```
//添加职工
void WorkerManager::Add_Emp()
{
    cout << "请输入添加职工数量：" << endl;

    int addNum = 0; //保存用户的输入数量
    cin >> addNum;

    if (addNum > 0)
```

```
{
    //添加
    //计算添加新空间大小
    int newSize = this->m_EmpNum + addNum; //新空间人数 = 原来记录人数 + 新增人数

    //开辟新空间
    Worker** newSpace = new Worker * [newSize];

    //将原来空间下数据，拷贝到新空间下
    if (this->m_EmpArray != NULL)
    {
        for (int i = 0; i < this->m_EmpNum; i++)
        {
            newSpace[i] = this->m_EmpArray[i];
        }
    }

    //批量添加新数据
    for (int i = 0; i < addNum; i++)
    {
        int id;       //职工编号
        string name;  //职工姓名
        int dSelect;      //部门选择


        cout << "请输入第 " << i + 1 << " 个新职工编号：" << endl;
        cin >> id;

        cout << "请输入第 " << i + 1 << " 个新职工姓名：" << endl;
        cin >> name;

        cout << "请选择该职工岗位：" << endl;
        cout << "1、普通员工" << endl;
        cout << "2、经理" << endl;
        cout << "3、总裁" << endl;
        cin >> dSelect;

        Worker* worker = NULL;
        switch (dSelect)
        {
        case 1:
            worker = new Employee(id, name, 1);
            break;
        case 2:
                worker = new Manager(id, name, 2);
                break;
            case 3:
                worker = new Boss(id, name, 3);
                break;
            default:
                break;
            }
            //将创建职工职责，保存到数组中
            newSpace[this->m_EmpNum + i] = worker;

        }

        //释放原有的空间
        delete[] this->m_EmpArray;

        //更改新空间指向
        this->m_EmpArray = newSpace;

        //更新新职工人数
        this->m_EmpNum = newSize;

        //更新职工不为空标志
        this->m_FileIsEmpty = false;

        //提示添加成功
        cout << "添加成功！" << addNum << "名新职工" << endl;

        //保存数据到文件中
        this->save();
    }
    else
    {
        cout << "数据输入有误！" << endl;
    }

    //按任意键清屏
    system("pause");
    system("cls");
}
```

功能 4：显示职工

```
//显示职工
void WorkerManager::Show_Emp()
{
    //判断文件是否为空
    if (this->m_FileIsEmpty)
    {
        cout << "文件不存在或记录为空" << endl;
    }
    else
    {
        for (int i = 0; i < m_EmpNum; i++)
        {
            //利用多态调用程序接口
            this->m_EmpArray[i]->showInfo();
        }
    }
    //按任意键清屏
    system("pause");
    system("cls");
}
```

功能 5：删除职工

```
//删除职工函数
void WorkerManager::Del_Emp()
{
    if (this->m_FileIsEmpty)
    {
        cout << "文件不存在或记录为空！" << endl;
    }
    else
    {
        //按照职工编号删除
        cout << "请输入想要删除的职工编号：" << endl;
        int id = 0;
        cin >> id;

        int index = this->IsExist(id);

        if (index != -1) //说明职工存在，并且要删除掉 index 位置上的职工
        {
            for (int i = index; i < this->m_EmpNum - 1; i++)
            {
                //数据前移
                this->m_EmpArray[i] = this->m_EmpArray[i + 1];
            }
            this->m_EmpNum--; //更新数组中记录的人员个数
            //同步更新到文件中
            this->save();

            cout << "删除成功！" << endl;
        }
        else
        {
            cout << "删除失败，未找到职工！" << endl;
        }
    }

    //按任意键清屏
    system("pause");
    system("cls");
}


//判断职工是否存在  如果存在返回职工所在数组中的位置，不存在返回-1
int WorkerManager::IsExist(int id)
{
    int index = -1;

    for(int i = 0; i < this->m_EmpNum; i++)
    {
        if (this->m_EmpArray[i]->m_Id == id)
        {
            //找到职工
            index = i;

            break;
        }
    }

    return index;
}
```

功能 6：修改职工

```
//修改职工
void WorkerManager::Mod_Emp()
{
    if (this->m_FileIsEmpty)
    {
        cout << "文件不存在或记录为空！" << endl;
    }
    else
    {
        cout << "请输入修改职工的编号：" << endl;
        int id;
        cin >> id;

        int ret = this->IsExist(id);
        if (ret != -1)
        {
            //查找到编号职工

            delete this->m_EmpArray[ret];

            int newId = 0;
            string newName = "";
            int dSelect = 0;

            cout << "查到：" << id << "号职工，请输入新职工号：" << endl;
            cin >> newId;

            cout << "请输入新的姓名：" << endl;
            cin >> newName;

            cout << "请输入岗位：" << endl;
            cout << "1、普通职工" << endl;
            cout << "2、经理" << endl;
            cout << "3、总裁" << endl;

            cin >> dSelect;

            Worker* worker = NULL;
            switch (dSelect)
            {
            case1:
                worker = new Employee(newId, newName, dSelect);
                break;
            case2:
                worker = new Manager(newId, newName, dSelect);
                break;
            case3:
                worker = new Boss(newId, newName, dSelect);
                break;
            }

            //更新数据 到数组中
            this->m_EmpArray[ret] = worker;

            cout << "修改成功！" << endl;

            //保存到文件中
            this->save();
        }
        else
        {
            cout << "修改失败，查无此人！" << endl;
        }
    }

    //按任意键清屏
    system("pause");
    system("cls");
    }
```

功能 7：查找职工

```
//查找职工
void WorkerManager::Find_Emp()
{
    if (this->m_FileIsEmpty)
    {
        cout << "文件不存在或记录为空！" << endl;
    }
    else
    {
        cout << "请输入查找的方式：" << endl;
        cout << "1、按照职工编号查找" << endl;
        cout << "2、按照职工姓名查找" << endl;

        int select = 0;
```

```
cin >> select;

if (select == 1)
{
    //按照编号查
    int id;
    cout << "请输入查找的职工编号：" << endl;
    cin >> id;

    int ret = IsExist(id);
    if (ret != -1)
    {
        //找到职工
        cout << "查找成功！该职工信息如下：" << endl;
        this->m_EmpArray[ret]->showInfo();
    }
    else
    {
        cout << "查找失败，查无此人" << endl;
    }
}
else if (select == 2)
{
    //按照姓名查
    string name;
    cout << "请输入查找的姓名：" << endl;
    cin >> name;

    //加入判断是否查到的标志
    bool flag = false; //默认未找到职工

    for (int i = 0; i < m_EmpNum; i++)
    {
        if (this->m_EmpArray[i]->m_Name == name)
        {
            cout << "查找成功，职工编号："
                << this->m_EmpArray[i]->m_Id
                << "号职工信息如下：" << endl;

            flag = true;

            //调用显示信息接口
            this->m_EmpArray[i]->showInfo();
        }
```

```
            }
            if (flag == false)
            {
                cout << "查找失败，查无此人" << endl;
            }
        }
        else
        {
            cout << "输入选项有误！" << endl;
        }
    }

    //按任意键清屏
    system("pause");
    system("cls");
}
```

功能 8：排序职工

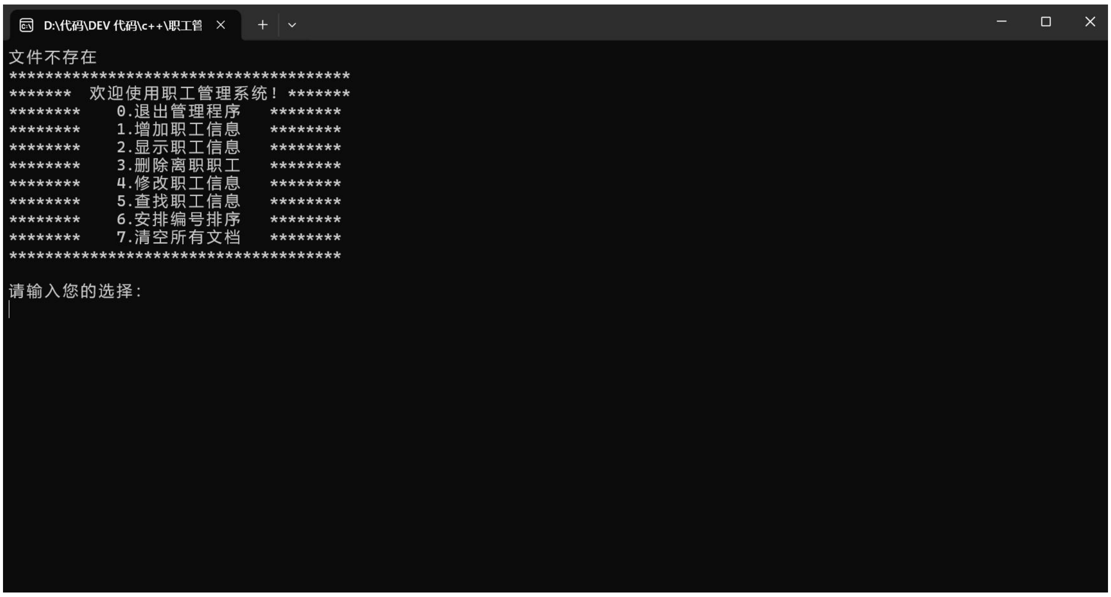
```
//按照职工编号排序
void WorkerManager::Sort_Emp()
{
    if (this->m_FileIsEmpty)
    {
        cout << "文件不存在或记录为空！" << endl;
        system("pause");
        system("cls");
    }
    else
    {
        cout << "请选择排序的方式：" << endl;
        cout << "1、按照职工号升序" << endl;
        cout << "2、按照职工号降序" << endl;

        int select = 0;
        cin >> select;
        for (int i = 0; i < m_EmpNum; i++)
        {
            int minOrMax = i; //声明最小值 或 最大值下标
            for (int j = i + 1; j < this->m_EmpNum; j++)
            {
                if (select == 1) //升序
                {
                    if (this->m_EmpArray[minOrMax]->m_Id > this->m_EmpArray[j]->m_Id)
                    {
                        minOrMax = j;
                    }
                }
                else  //降序
                {
                    if (this->m_EmpArray[minOrMax]->m_Id < this->m_EmpArray[j]->m_Id)
                    {
                        minOrMax = j;
                    }
                }
            }

            //判断一开始认定的最小值或最大值 是不是 计算的最小值或最大值，如果不是 交换数据
            if (i != minOrMax)
            {
                Worker* temp = this->m_EmpArray[i];
                this->m_EmpArray[i] = this->m_EmpArray[minOrMax];
                this->m_EmpArray[minOrMax] = temp;
            }
        }

        cout << "排序成功！排序后的结果为：" << endl;
        this->save(); //排序后结果保存到文件中
        this->Show_Emp(); //展示所以职工
    }



    }
```

功能 9：清空文档

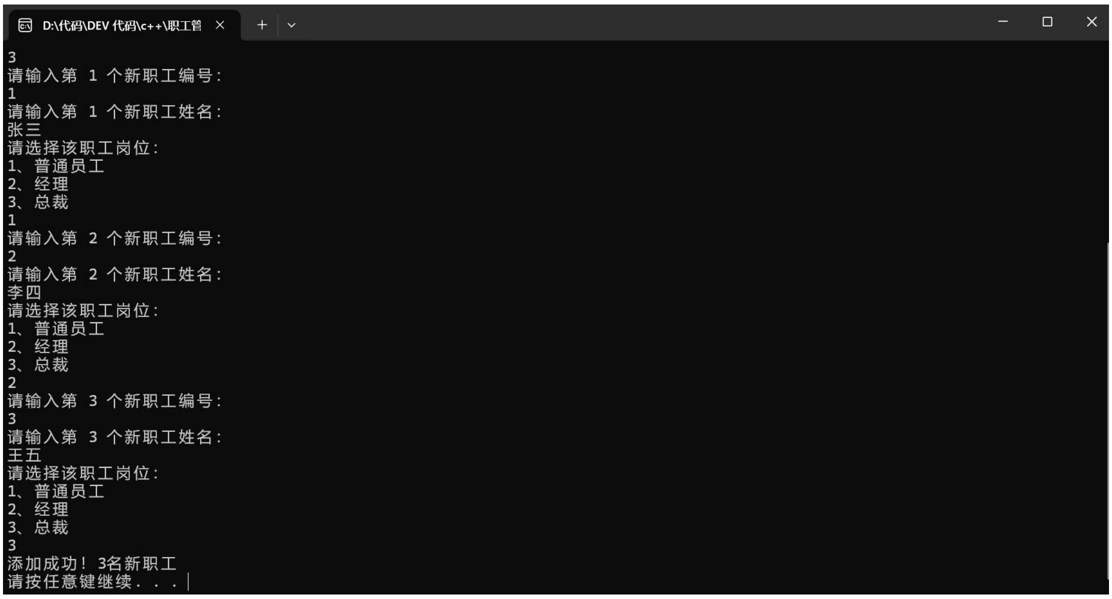
```
//清空文件
void WorkerManager::Clean_File()
{

    cout << "确定清空？" << endl;
    cout << "1、确定" << endl;
    cout << "2、返回" << endl;

    int select = 0;
    cin >> select;
    if (select == 1)
    {
        //清空文件
        ofstream ofs(FILENAME, ios::trunc); //删除文件后重新创建
        ofs.close();

        if (this->m_EmpArray != NULL)
        {
            //删除堆区的每个职工对象
            for (int i = 0; i < this->m_EmpNum; i++)
            {
                delete this->m_EmpArray[i];
                this->m_EmpArray[i] = NULL;
            }

            //删除堆区数组指针
            delete[] this->m_EmpArray;
            this->m_EmpArray = NULL;
            this->m_EmpNum = 0;
            this->m_FileIsEmpty = true;
        }

        cout << "清空成功！" << endl;
    }

    //按任意键清屏
    system("pause");
    system("cls");
}
```

四、运行界面

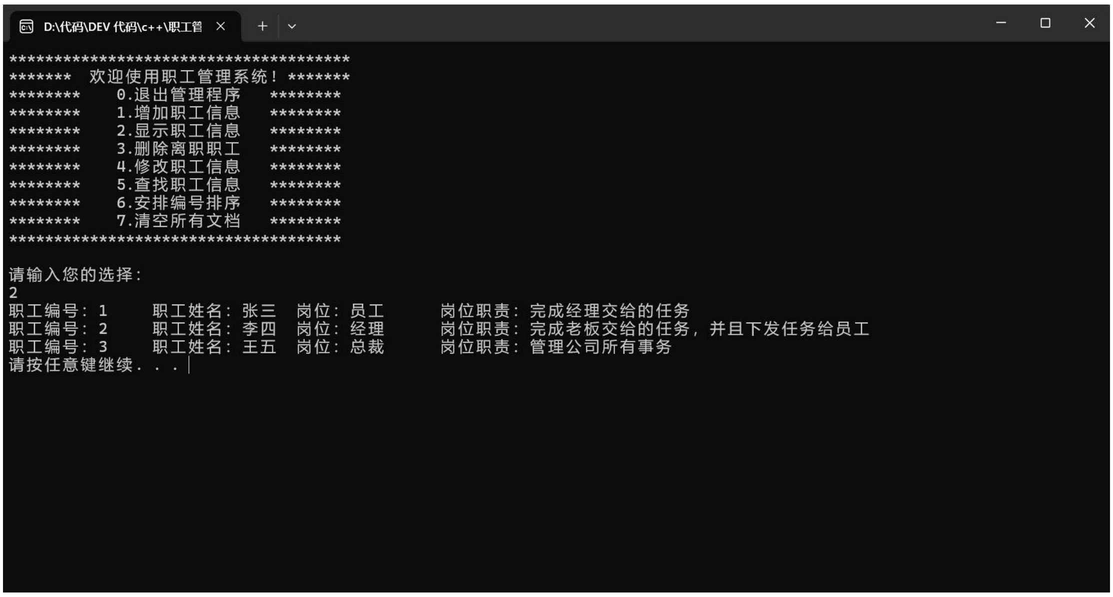
将你整个系统运行过程，逐条写清楚并截图呈现



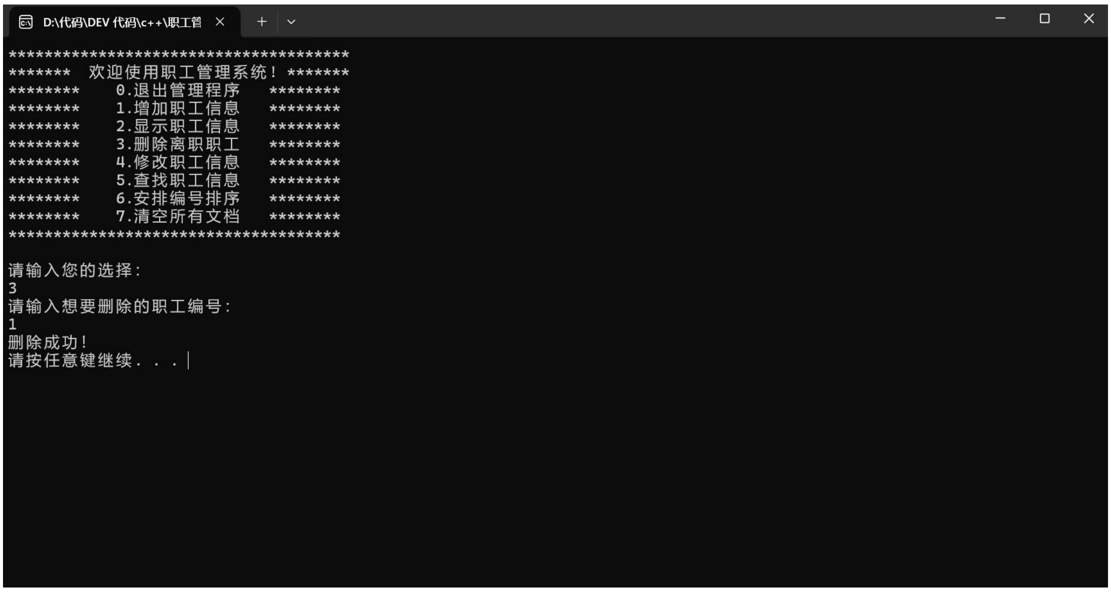
```
文件不存在
**************************************
*******  欢迎使用职工管理系统！*******
********    0.退出管理程序   ********
********    1.增加职工信息   ********
********    2.显示职工信息   ********
********    3.删除离职职工   ********
********    4.修改职工信息   ********
********    5.查找职工信息   ********
********    6.安排编号排序   ********
********    7.清空所有文档   ********
*************************************

请输入您的选择：
```



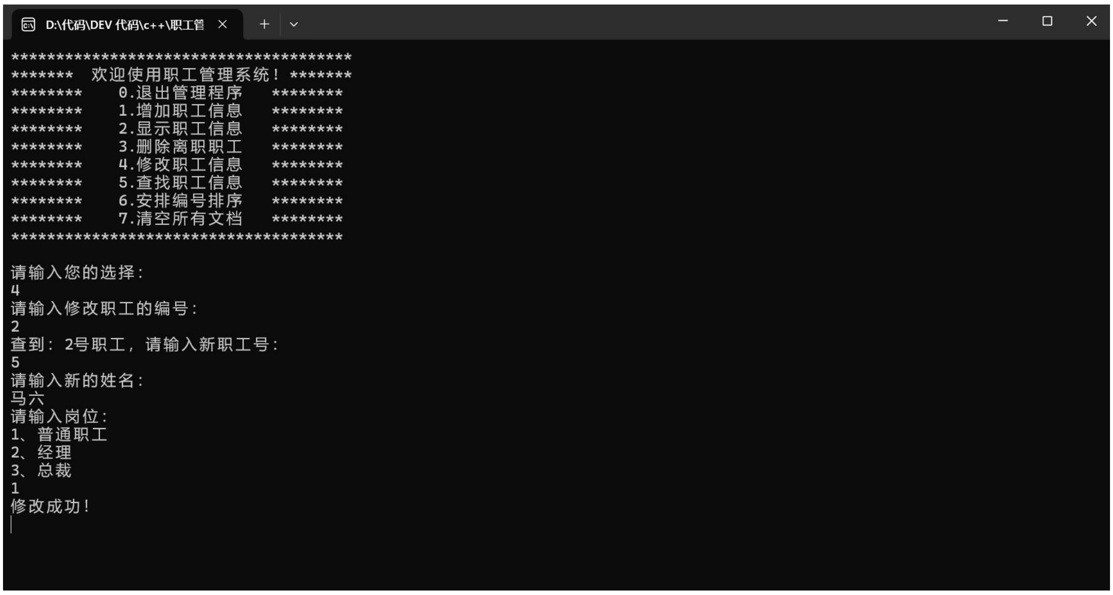
```
3
请输入第 1 个新职工编号：
1
请输入第 1 个新职工姓名：
张三
请选择该职工岗位：
1、普通员工
2、经理
3、总裁
1
请输入第 2 个新职工编号：
2
请输入第 2 个新职工姓名：
李四
请选择该职工岗位：
1、普通员工
2、经理
3、总裁
2
请输入第 3 个新职工编号：
3
请输入第 3 个新职工姓名：
王五
请选择该职工岗位：
1、普通员工
2、经理
3、总裁
3
添加成功！3名新职工
请按任意键继续. . .
```



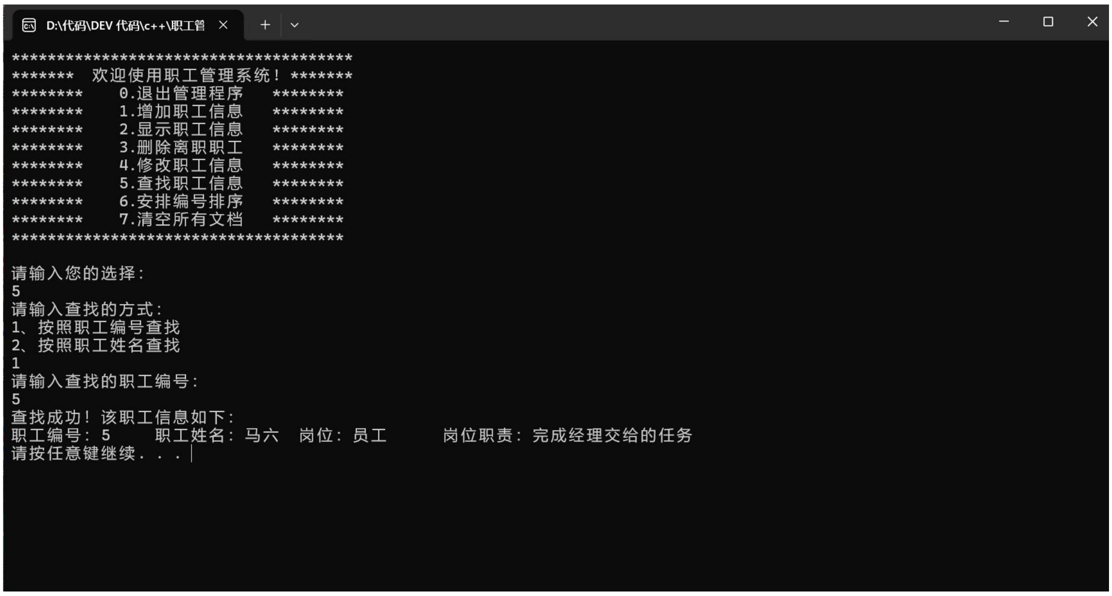
```
**************************************
*******  欢迎使用职工管理系统！*******
*********   0.退出管理程序   *********
*********   1.增加职工信息   *********
*********   2.显示职工信息   *********
*********   3.删除离职职工   *********
*********   4.修改职工信息   *********
*********   5.查找职工信息   *********
*********   6.安排编号排序   *********
*********   7.清空所有文档   *********
*************************************

请输入您的选择：
2
职工编号：1     职工姓名：张三  岗位：员工      岗位职责：完成经理交给的任务
职工编号：2     职工姓名：李四  岗位：经理      岗位职责：完成老板交给的任务，并且下发任务给员工
职工编号：3     职工姓名：王五  岗位：总裁      岗位职责：管理公司所有事务
请按任意键继续. . .
```



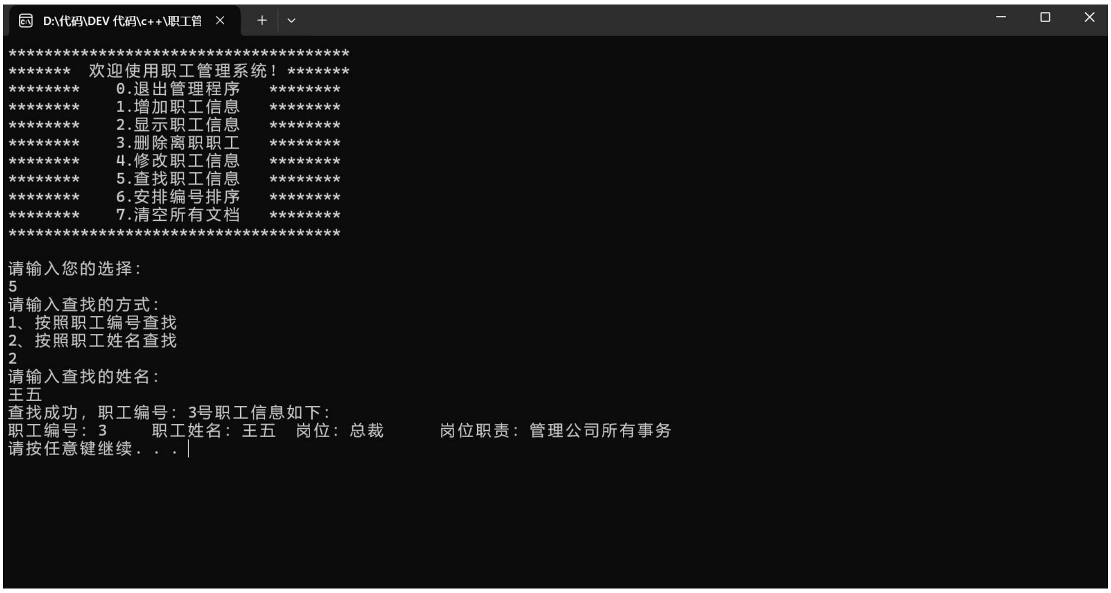
```
**************************************
*******  欢迎使用职工管理系统！*******
*********   0.退出管理程序   *********
*********   1.增加职工信息   *********
*********   2.显示职工信息   *********
*********   3.删除离职职工   *********
*********   4.修改职工信息   *********
*********   5.查找职工信息   *********
*********   6.安排编号排序   *********
*********   7.清空所有文档   *********
*************************************

请输入您的选择：
3
请输入想要删除的职工编号：
1
删除成功！
请按任意键继续. . .
```



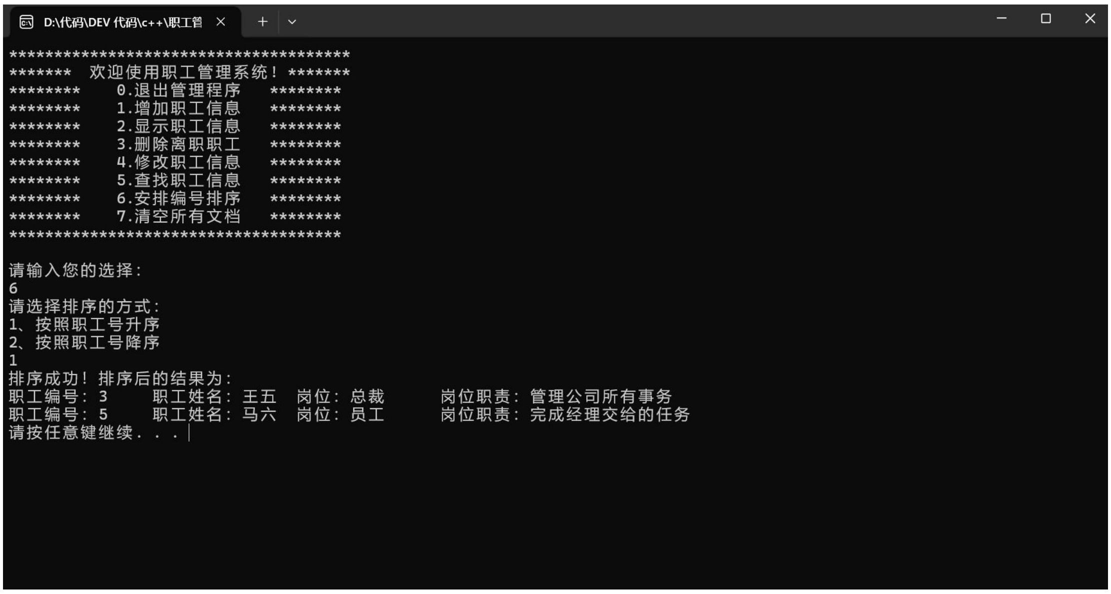
```
**************************************
*******  欢迎使用职工管理系统！*******
*********    0.退出管理程序   *********
*********    1.增加职工信息   *********
*********    2.显示职工信息   *********
*********    3.删除离职职工   *********
*********    4.修改职工信息   *********
*********    5.查找职工信息   *********
*********    6.安排编号排序   *********
*********    7.清空所有文档   *********
*************************************

请输入您的选择：
4
请输入修改职工的编号：
2
查到：2号职工，请输入新职工号：
5
请输入新的姓名：
马六
请输入岗位：
1、普通职工
2、经理
3、总裁
1
修改成功！
```



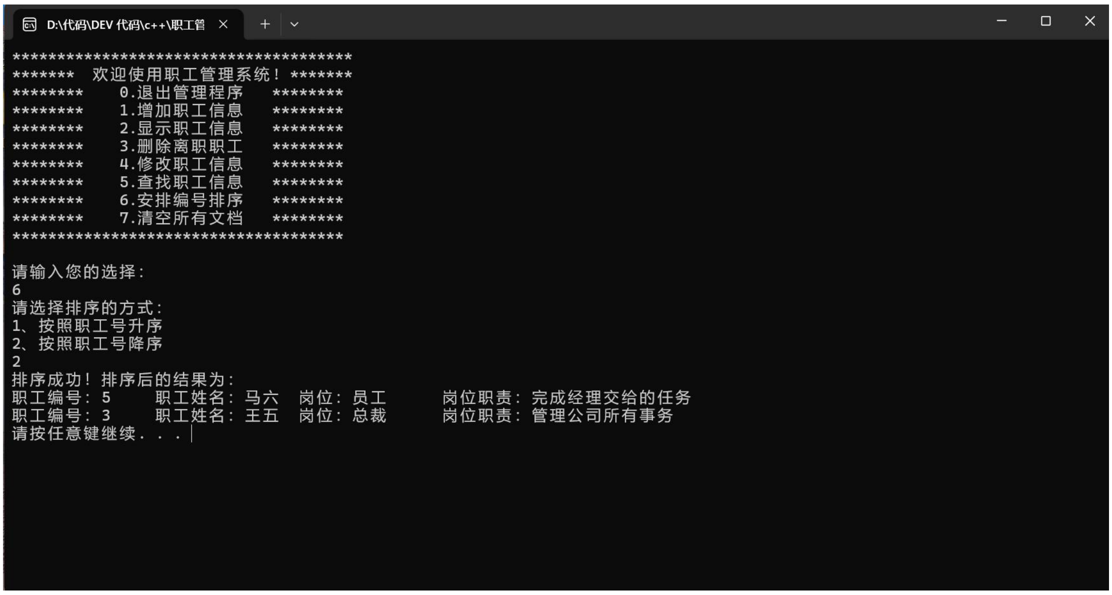
```
**************************************
*******  欢迎使用职工管理系统！*******
*********    0.退出管理程序   *********
*********    1.增加职工信息   *********
*********    2.显示职工信息   *********
*********    3.删除离职职工   *********
*********    4.修改职工信息   *********
*********    5.查找职工信息   *********
*********    6.安排编号排序   *********
*********    7.清空所有文档   *********
*************************************

请输入您的选择：
5
请输入查找的方式：
1、按照职工编号查找
2、按照职工姓名查找
1
请输入查找的职工编号：
5
查找成功！该职工信息如下：
职工编号：5     职工姓名：马六  岗位：员工      岗位职责：完成经理交给的任务
请按任意键继续. . .
```



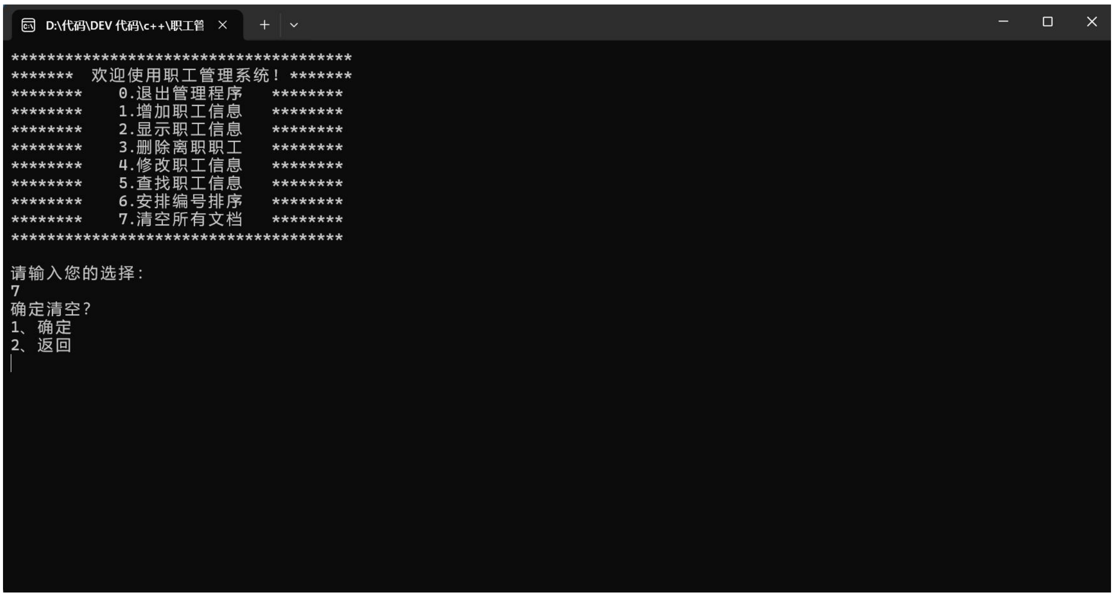
```
**************************************
*******  欢迎使用职工管理系统！*******
********    0.退出管理程序   ********
********    1.增加职工信息   ********
********    2.显示职工信息   ********
********    3.删除离职职工   ********
********    4.修改职工信息   ********
********    5.查找职工信息   ********
********    6.安排编号排序   ********
********    7.清空所有文档   ********
*************************************

请输入您的选择：
5
请输入查找的方式：
1、按照职工编号查找
2、按照职工姓名查找
2
请输入查找的姓名：
王五
查找成功，职工编号：3号职工信息如下：
职工编号：3     职工姓名：王五  岗位：总裁      岗位职责：管理公司所有事务
请按任意键继续. . .
```



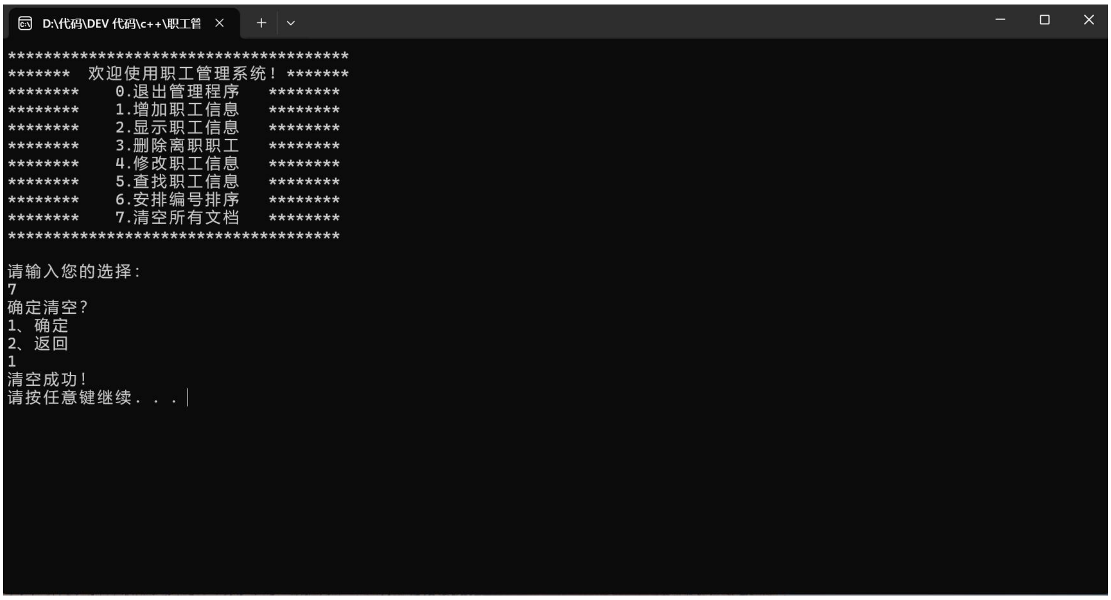
```
**************************************
*******  欢迎使用职工管理系统！*******
********    0.退出管理程序   ********
********    1.增加职工信息   ********
********    2.显示职工信息   ********
********    3.删除离职职工   ********
********    4.修改职工信息   ********
********    5.查找职工信息   ********
********    6.安排编号排序   ********
********    7.清空所有文档   ********
*************************************

请输入您的选择：
6
请选择排序的方式：
1、按照职工号升序
2、按照职工号降序
1
排序成功！排序后的结果为：
职工编号：3     职工姓名：王五  岗位：总裁      岗位职责：管理公司所有事务
职工编号：5     职工姓名：马六  岗位：员工      岗位职责：完成经理交给的任务
请按任意键继续. . .
```



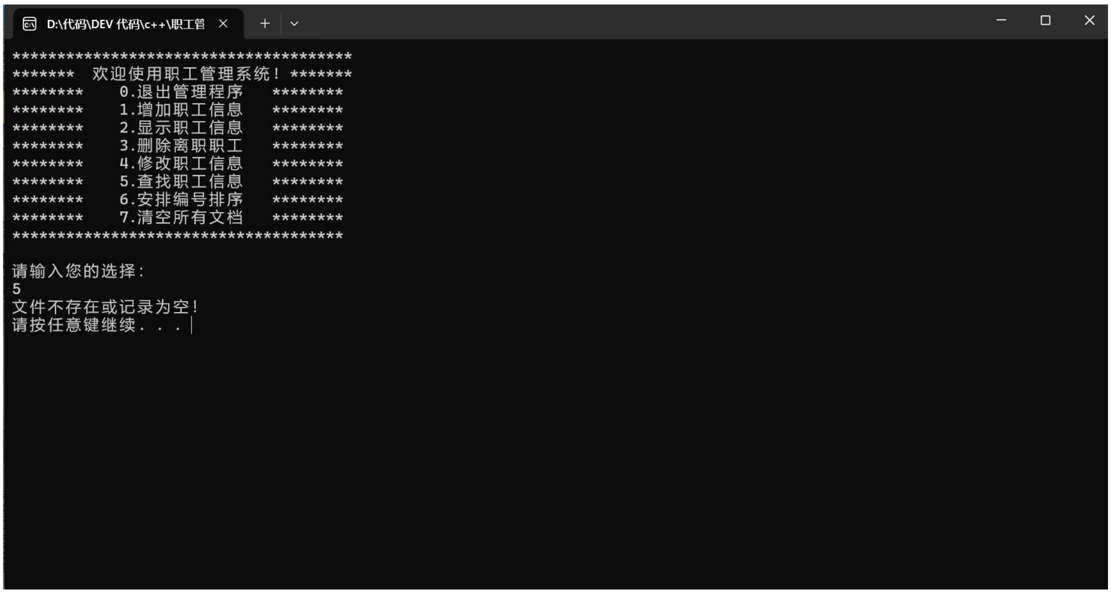
```
**************************************
*******  欢迎使用职工管理系统！*******
********    0.退出管理程序   ********
********    1.增加职工信息   ********
********    2.显示职工信息   ********
********    3.删除离职职工   ********
********    4.修改职工信息   ********
********    5.查找职工信息   ********
********    6.安排编号排序   ********
********    7.清空所有文档   ********
*************************************

请输入您的选择：
6
请选择排序的方式：
1、按照职工号升序
2、按照职工号降序
2
排序成功！排序后的结果为：
职工编号：5     职工姓名：马六  岗位：员工      岗位职责：完成经理交给的任务
职工编号：3     职工姓名：王五  岗位：总裁      岗位职责：管理公司所有事务
请按任意键继续. . .
```



```
**************************************
*******  欢迎使用职工管理系统！*******
********    0.退出管理程序   ********
********    1.增加职工信息   ********
********    2.显示职工信息   ********
********    3.删除离职职工   ********
********    4.修改职工信息   ********
********    5.查找职工信息   ********
********    6.安排编号排序   ********
********    7.清空所有文档   ********
*************************************

请输入您的选择：
7
确定清空？
1、确定
2、返回
```



```
**************************************
*******  欢迎使用职工管理系统！ *******
********    0.退出管理程序   ********
********    1.增加职工信息   ********
********    2.显示职工信息   ********
********    3.删除离职职工   ********
********    4.修改职工信息   ********
********    5.查找职工信息   ********
********    6.安排编号排序   ********
********    7.清空所有文档   ********
*************************************

请输入您的选择:
7
确定清空?
1、确定
2、返回
1
清空成功!
请按任意键继续. . .
```



```
**************************************
*******  欢迎使用职工管理系统！ *******
********    0.退出管理程序   ********
********    1.增加职工信息   ********
********    2.显示职工信息   ********
********    3.删除离职职工   ********
********    4.修改职工信息   ********
********    5.查找职工信息   ********
********    6.安排编号排序   ********
********    7.清空所有文档   ********
*************************************

请输入您的选择:
5
文件不存在或记录为空!
请按任意键继续. . .
```

五、总结

通过这次期末大作业我更加深刻的掌握了各个头文件和源文件之间的联系，能让我对以后编程更加分类和美观，方便自己与他人理解代码。职工管理系统项目的完成不仅能够提高实际职工信息管理的效率，还能帮助大一学生加深对编程和系统开发的理解。该系统通过模块化的设计，实现了各个功能的独立性和高效性。此外，系统还具有用户友好、数据安全和灵活扩展的特点，使其在实际应用中具有很高的实用价值。通过不断的完善和改进，该系统将变得更加完善和强大。