

高级语言程序设计

课程设计报告

|  |  |  |
| --- | --- | --- |
| 课程设计成绩（总分20） | | 分数 |
| 程序 | 程序设计完整，功能齐全（8-10分） |  |
| 程序设计基本完整，功能基本完成（0-7分） |
| 报告 | 结构清晰完整，报告格式正确，语言通顺（8-10分） |  |
| 格式不正确，封面或字体有误，语言欠通顺（0-7分） |
|  | 总分 |  |

|  |  |
| --- | --- |
| 课程名称 | 高级语言程序设计 |
| 课程编号 | 08306135 |
| 报告题目 | 企业管理系统 |
| 序 号 | 34 |
| 姓 名 | 李昀哲 |
| 学 号 | 20123101 |
| 所在学院 | 计算机工程与科学学院 |
| 专业名称 | 智能科学与技术 |

2020 年 11 月

目 录

[企业管理系统 3](#_Toc10495)

[1. 数据格式与功能描述 3](#_Toc27987)

[1.1 数据文件格式 3](#_Toc27341)

[1.2 功能描述 3](#_Toc2668)

[1.3 要求 5](#_Toc15647)

[2. 问题分析 5](#_Toc9028)

[2.1 链表创建、删除、修改等问题 5](#_Toc7492)

[2.2 文件输入输出问题 6](#_Toc29797)

[2.3 对系统的初步思路 6](#_Toc83)

[3.总体设计 6](#_Toc18060)

[3.1功能模块设计 6](#_Toc7173)

[3.2系统界面设计 7](#_Toc12444)

[3.3数据结构设计 7](#_Toc19329)

[3.4函数设计 8](#_Toc27077)

[4.详细设计 10](#_Toc5966)

[4.1各个函数的调用关系图 10](#_Toc13858)

[4.2各个函数的功能描述 10](#_Toc25095)

[5.功能测试 29](#_Toc2344)

[5.1 初始化界面 29](#_Toc1928)

[新系统界面 29](#_Toc5264)

[5.2查询员工信息 30](#_Toc27309)

[5.2.1全部员工信息 30](#_Toc28589)

[5.4删除员工和事务信息 33](#_Toc3543)

[5.5 修改员工信息 34](#_Toc8776)

[5.6员工排序 39](#_Toc29911)

[5.7 领导层级 39](#_Toc30579)

[5.8 查询事务信息 41](#_Toc17135)

[5.9 退出系统 42](#_Toc24573)

[6.0 保存的文件 42](#_Toc24064)

[6. 总 结 43](#_Toc10595)

企业管理系统

随着中小企业的不断发展，企业内部员工沟通就显得非常重要，通过一个企业管理系统能解决员工沟通困难的问题。它可以实现发布消息，查看消息，删除消息等功能。

本实验要求设计一个简单的企业管理系统，涉及函数、结构体、链表、文件等方面的知识，学习利用链表处理数据的方法，熟练掌握文件操作，构建综合程序设计的思路及框架，提高综合设计软件系统的能力。

1. 数据格式与功能描述
   1. 数据文件格式

数据文件分为2个，employee.txt与things.txt。employee.txt包括了所有员工的个人信息，具体包括员工编号、姓名、部门、工资、直系领导等。格式如图1-1所示。things.txt包括员工编号与分配任务。格式如图1-2所示。

具体的数据信息与内容可自行设定，但要保证直系领导存在传递关系。

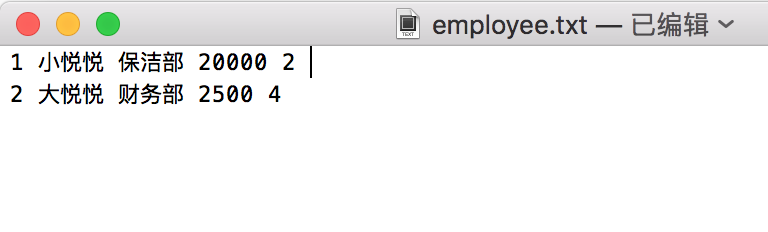


图1-1 员工信息

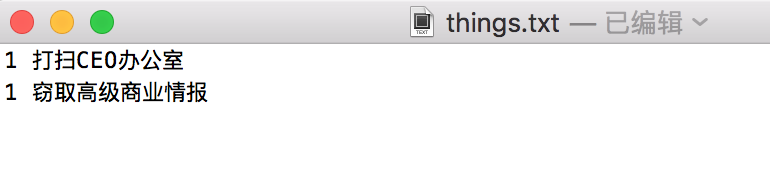


图1-2 事务信息

* 1. 功能描述

设计一个员工管理系统，利用单链表来处理员工信息与事务信息，要求实现如下系统功能。

1. 初始化菜单

程序开始运行后显示菜单，要求菜单能完全展示本系统的功能，尽力做到设计美观。

1. 选择待操作的文件

考虑到本系统有多个文件，需要提示用户输入来选择需要操作的文件是员工信息还是事务信息。

1. 查询员工信息与事务信息

选择员工信息文件后，按员工编号、姓名、所在部门查询员工所有信息。提示用户输入待查人员的员工编号、姓名、所在部门，显示符合条件的联系人所有信息。如果没有满足要求的联系人，输出提示信息。

可输出全部员工信息，若当前企业没有员工，输出提示信息。

选择事务信息文件后，按员工编号查询事务信息。提示用户输入待查事务的员工编号显示符合条件的事务信息。如果没有满足要求的事务，输出提示信息。

可输出全部事务信息，若当前企业没有事务，输出提示信息。

1. 删除员工信息与事务信息

按指定的员工编号、姓名删除员工信息中对应记录，删除时应删除该员工的所有信息并提示删除成功。若此员工不存在，输出提示信息。

1. 修改员工信息

修改指定员工的信息。提示用户输入要修改的员工编号，若此编号存在，则提示选择修改员工信息中的姓名、所在部门、工资或者只系领导。若该员工编号不存在，输出提示信息。

1. 增加员工信息与事务信息

通过控制台增加员工信息，提示用户输入待增加的员工编号，如果编号存在，提示编号已经存在；如果不存在，则逐项录入此员工的每一条信息。

通过控制台增加事务信息，提示用户输入分配给事务的员工编号，录入待增加的事务。

1. 员工排序

按照员工的工资进行排序，输出排序后员工信息。

1. 保存员工信息与事务信息

将当前员工信息与事务信息分别保存到txt文件中。

1. 输出管理层级顺序

根据员工信息文件中的直系领导数据，构建一个员工领导领导的领导的关系链表。用户可以通过输入员工编号来查询并输出此员工所有的直系领导。如果此员工没有直系领导，输出提示信息。

(10)退出系统

用户可通过输入特定指令来退出程序，在用户输入退出指令之前，程序不得自行退出。

* 1. 要求

(1)源程序编写要求

根据系统功能描述，采用模块化程序设计方法进行程序设计，要求程序结构清晰。上述各个功能模块要求分别用函数实现，在主函数中通过调用这些函数，完成系统功能的要求。代码书写规范，有简要的注释，给出数据和函数说明。

(2) 设计报告撰写要求

设计报告内容包括题目内容和要求、总体设计、详细设计、源代码、调试过程中的问题、总结等。

总体设计：对程序的整体设计思路进行描述，画出图书管理系统的总体功能模块图，说明系统使用的主要数据结构，给出包含上述功能的系统功能模块图。

详细设计：分析实现各函数功能的算法，描述函数的功能。

调试过程中的问题：记录程序编写和调试过程中遇到的各种问题，以及解决这些问题的途径和方法。

总结：回顾整个综合程序设计的过程，对学习到的设计方法和思路进行总结，写出个人体会。

1. 问题分析
   1. 链表创建、删除、修改等问题

由于对链表知识缺乏甚多，便在开始设计系统前在bilibili上查找有关链表学习视频，并跟着视频进行调试。

学习视频网址：<https://www.bilibili.com/video/BV1Vx411M7A8>

* 1. 文件输入输出问题

同样，对于文件输入输出基本没有了解的情况下，同样网上寻找相关资料针对输入流、输出流的用法进行学习。

* 1. 对系统的初步思路

通过分步骤实现多个基础子函数整合调试为一个完整系统，再进行问题分析debug。

企业员工、事务的数据为文本格式，由多个数据项组成。在程序设计中，可以用结构体来存储，由于系统功能涉及员工、事务信息插入和删除，使用链表结构存储信息更有利于插入和删除处理。因此，本实验将图书信息的存储在单链表的结点中，由数据域和指针域构成。员工数据部分包括：员工编号、姓名、性别、年龄、部门、薪资、直系领导序号共七个成员，事务数据部分包括：员工编号和员工事务信息；指针域存储其直接后继结点的地址。

员工信息的查找、添加、删除、修改、管理层级和事务信息的查询、添加都需要先查找，找到目标结点后再进行相应操作，因此如何编写找到目标节点的算法尤为重要。对员工根据工资的排序，本系统采用冒泡排序，针对单链表结构的特点，也可采用插入排序（只需要比较插入数和前一位数字，时间复杂度较低些）的方法（但本程序中暂未采用，可在后续进行优化）。进行完任何操作后，都会对操作进行保存，显示更新。保存信息后，修改后的员工、事务信息被写入数据文件。

3.总体设计

3.1功能模块设计

根据系统功能描述和问题分析，可将系统功能划分为若干模块，如图3-1所示。

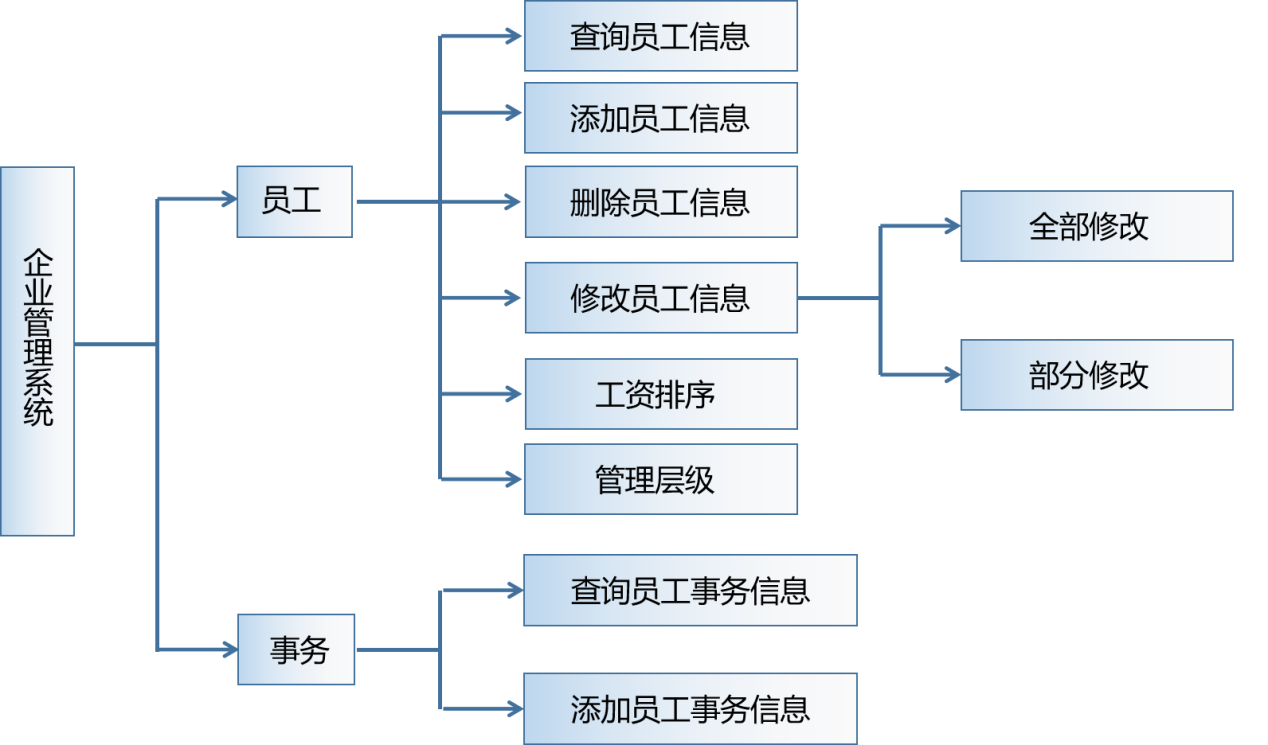


图3-1 系统功能模块图

3.2系统界面设计

进入企业管理系统前，首先会提示本系统是否为新系统：若是，则给出提示信息，要求先录入员工及事务信息后使用；若不是，则给出“员工、事务信息已存在的提示”后进入选择页面，选择需要操作的对象：“员工”或“事务”。

进入系统后，若选择的是“员工”：则显示员工页面主菜单，列出7个选项：查询信息，添加信息，删除信息，修改信息，工资排序，管理层级和退出系统。

若选择的是“事务”：则显示事务页面主菜单，列出3个选项：查询、添加和退出。

用户根据主菜单中显示的功能模块及其相应编号，选择相应编号，执行相应的功能。所选模块执行完毕后，用户可再次选择操作对象并返回系统主菜单。

3.3数据结构设计

系统内容涉及到多条记录的处理，每条记录含有不同类型的数据项，使用单链表来存储较为方便，链表结点设计如下：

**typedef** **struct** node**{** *//创建员工节点*

**int** num**;**

**char** name**[**20**];**

**char** sex**[**10**];**

**int** age**;**

**char** department**[**20**];**

**int** salary**;**

**int** leader**;**

**char** variable**[**20**];** *//用于排序交换变量信息*

**struct** node **\***next**;**

**}**emp**;**

单链表结构中，每个节点的next作为指针域，用于存储其后续节点的地址，其他部分为数据域，存储结点中的数据。

3.4函数设计

函数设计的主要内容包括函数的名称，函数的功能与函数中参数类型的说明。

|  |  |  |
| --- | --- | --- |
| **函数名称** | **函数功能** | **参数说明** |
| **int** amount**(**emp **\***head**);** | 计算员工节点个数 | head表示员工链表的头指针 |
| **int** amount2**(**thing **\***head**);** | 计算事务节点个数 | head表示事务链表的头指针 |
| **void** bound**(char** ch**,** **int** n**);** | 分割线：显示美化 | ch表示分隔符-，n表示长度 |
| emp **\***checkfirst**(char** **\***filename1**);** | 进入系统前查看“员工”文件是否为空，即确认系统是否第一次使用 | Filename1表示需要打开的员工信息文件 |
| thing **\***checkfirst2**(char** **\***filename1**);** | 进入系统前查看“事务”文件是否为空，即确认系统是否第一次使用 | Filename1表示需要打开的事务信息文件 |
| emp **\***createList**(int** n**,** **char** **\***filename1**);** | 建立员工链表 | n表示需要录入的员工个数Filename1表示需要打开的员工信息文件 |
| emp **\***createListX**(char** **\***filename1**);** | 将“员工”文件中的信息导入员工链表 | Filename1表示需要打开的员工信息文件 |
| thing **\***createList2**(int** n**,** **char** **\***filename1**);** | 建立事务链表 | n表示需要录入的事务个数Filename1表示需要打开的事务信息文件 |
| thing **\***createListX2**(char** **\***filename1**);** | 将“事务”文件中的信息导入事务链表 | Filename1表示需要打开的事务信息文件 |
| **void** display **(**emp **\***head**);** | 显示“员工”全部信息 | head表示员工链表的头指针 |
| **void** display2 **(**thing **\***head**);** | 显示“事务”全部信息 | head表示事务链表的头指针 |
| **void** displayperson **(**emp **\***head**,** **int** index**);** | 显示“员工”个人信息 | head表示员工链表的头指针 |
| **void** displayperson2 **(**thing **\***head**,** **int** index**);** | 显示员工“事务”信息 | head表示事务链表的头指针 |
| emp **\***insert**(**emp **\***head**,** **int** index**,** **char** **\***filename1**);** | 插入“员工”信息 | head表示员工链表的头指针  Index表示插入员工编号  Filename1表示需要打开的员工信息文件 |
| thing **\***insert2**(**thing **\***head2**,** emp**\***head**,** **int** index**,** **char** **\***filename1**);** | 插入“事务”信息 | head表示事务链表的头指针  Index表示插入员工编号  Filename1表示需要打开的事务信息文件 |
| **void** leader**(**emp **\***head**,** **int** index**);** | 查找领导并显示 | Filename1表示需要打开的员工信息文件  Index表示员工编号 |
| **void** paixv**(**emp **\***head**);** | 根据工资从高到低排序 | Filename1表示需要打开的员工信息文件 |
| **void** rank**(**emp **\***head**);** | 按工资排序并显示 | Filename1表示需要打开的员工信息文件 |
| emp **\***remove**(**emp **\***head**,** thing **\***head2**,** **int** index**,** **char** **\***filename1**,char** **\***filename2**);** | 删除某个员工的全部信息（包括事务） | head表示员工链表头指针  head2表示事务链表头指针  Index表示插入员工编号  Filename1表示需要打开的员工信息文件  Filename2表示需要打开的事务信息文件 |
| thing **\***remove2**(**thing **\***head**,** **int** index**,** **char** **\***filename1**);** | 删除事务信息（仅在删除员工时调用，不作为单个功能使用） | head表示事务链表头指针  Index表示插入员工编号  Filename1表示需要打开的事务信息文件 |
| emp **\***revise**(**emp **\***head**,** **int** index**,** **char** **\***filename1**);** | 修改员工信息 | head表示员工链表头指针  Index表示插入员工编号  Filename1表示需要打开的事务信息文件 |
| **void** save**(char** **\***filename1**,** emp **\***head**);** | 保存员工链表信息于“员工”文件中 | head表示员工链表头指针  Filename1表示需要打开的员工信息文件 |
| **void** save2**(char** **\***filename1**,** thing **\***head**);** | 保存事务链表信息于“事务”文件中 | head表示事务链表头指针  Filename1表示需要打开的事务信息文件 |
| **void** s2b**(**emp **\***head**);** | small to big 员工信息按编号从小到大排序 | head表示员工链表头指针 |
| **void** s2bt**(**thing **\***head**);** | small to big things 事务信息按编号从小到大排序 | head表示事务链表头指针 |
| **void** welcome\_menu**(int** select**);** | 进入系统欢迎菜单 | Select表示选择的操作对象 |

4.详细设计

详细设计包括的源代码的框架及其功能描述。

4.1各个函数的调用关系图

如图4-1所示。

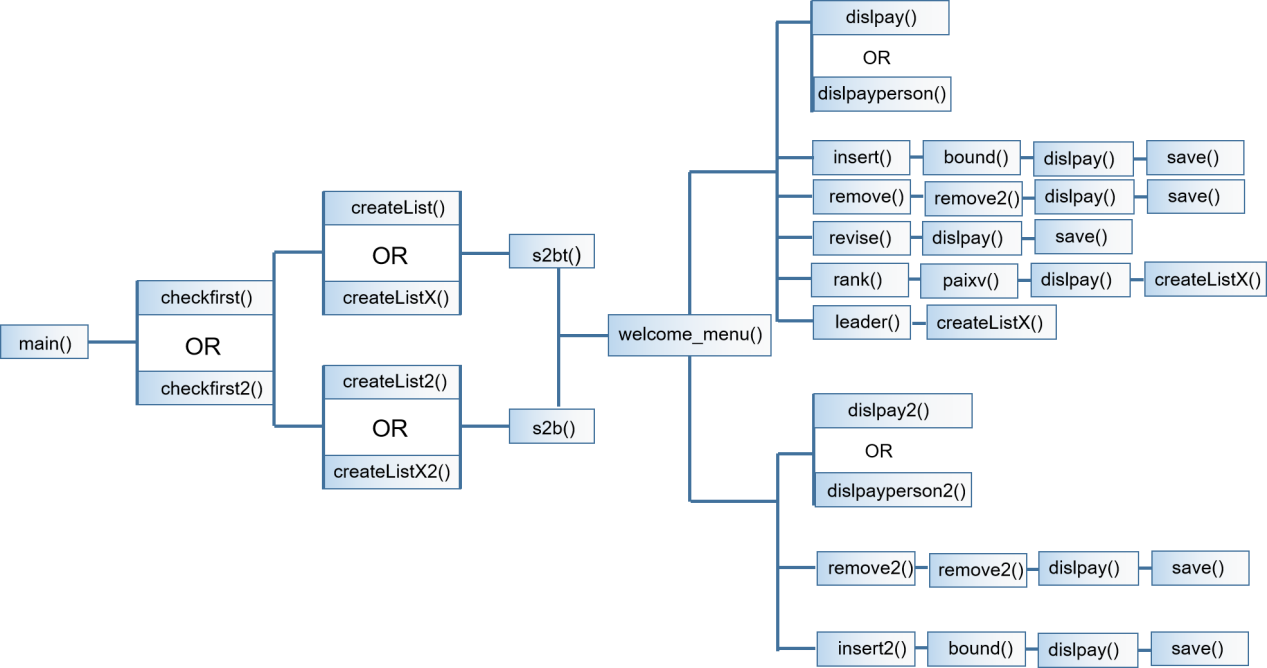


图4-1

4.2各个函数的功能描述

4.2.1初始化系统界面

**void** welcome\_menu**(int** select**)** *//欢迎菜单*

**{**

cout **<<** **" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* "<<**endl**;**

cout **<<** **" \* 欢迎使用企业管理系统 \* "<<**endl**;**

cout **<<** **" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* "<<**endl**;**

cout **<<** **" \* 1.查询信息 \* "<<**endl**;**

cout **<<** **" \* 2.添加信息 \* "<<**endl**;**

**if** **(** select **!=** 2**)**

**{**

cout **<<** **" \* 3.删除信息 \* "<<**endl**;**

cout **<<** **" \* 4.修改信息 \* "<<**endl**;**

cout **<<** **" \* 5.员工排序 \* "<<**endl**;**

cout **<<** **" \* 6.管理层级 \* "<<**endl**;**

**}**

cout **<<** **" \* 0.退出系统 \* "<<**endl**;**

cout **<<** **" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* "<<**endl**<<**endl**;**

**if** **(** select **!=** 2**)**

**{**

cout **<<** **"请输入您需要的操作（0 - 6）：";**

**}else**

**{**

cout **<<** **"请输入您需要的操作（0 - 3）：";**

**}**

**}**

4.2.2 具体功能的实现

**/\*\***

**查询函数，进入查询功能子菜单**

1. **查询全部**
2. **查询个人**

**\*/**

**//查询全部如下：**

**void** display **(**emp **\***head**)** *//展示员工信息*

**{**

emp **\***p **=** head**->**next**;**

**while** **(** p **!=** NULL**)**

**{**

cout **<<**setiosflags**(**ios**::**left**)<<**setw**(**8**)<<** p**->**num**<<**setw**(**8**)<<**p**->**name**<<**setw**(**4**)**

**<<**p**->**sex**<<**setw**(**6**)<<** p**->**age**<<**setw**(**25**)**

**<<**p**->**department**<<**setw**(**10**)<<**p**->**salary**<<**setw**(**2**)<<**p**->**leader**<<**endl**;**

p **=** p**->**next**;**

**}**

**}**

**//根据编号查询个人，如下：**

**void** displayperson **(**emp **\***head**,** **int** index**)** *//显示个人信息*

**{**

**int** mark **=** 0**;** *//定义标记符*

**char** name**[**20**];**

cout **<<"请输入姓名：";**

cin **>>** name**;**

emp **\***p **=** head**->**next**;** *//head节点没东西*

**while** **(** p**->**next**!=**NULL**)**

**{**

**if** **(** p**->**num **==** index**)**

**{**

*// cout <<"name is = " <<name<<endl; //测试姓名是否相等*

*// cout << "p->name is = "<<p->name<<endl;*

**if** **(** **(**string**)**p**->**name **!=** **(**string**)**name**)** *//再次输入名字加以确认*

**{**

cout **<<"信息不匹配";**

**return** **;**

**}**

**else**

**{**

mark **=** 1**;** *//员工存在时，标记符为真*

**break;**

**}**

**}**

p **=** p**->**next**;**

**}**

**if** **(** mark **==** 0**)**

**{**

cout **<<** **"不存在该员工"<<**endl**;**

**return;**

**}**

cout **<<**setiosflags**(**ios**::**left**)<<**setw**(**8**)<<** p**->**num**<<**setw**(**8**)<<**p**->**name**<<**setw**(**4**)**

**<<**p**->**sex**<<**setw**(**6**)<<** p**->**age**<<**setw**(**25**)**

**<<**p**->**department**<<**setw**(**10**)<<**p**->**salary**<<**setw**(**2**)<<**p**->**leader**<<**endl**;**

**}**

**void** displayperson2 **(**thing **\***head**,** **int** index**)** *//显示个人信息 （注释信息同员工函数）*

**{**

**int** mark **=** 0**;**

thing **\***p **=** head**->**next**;** *//head节点没东西*

**while** **(** p**->**next**!=**NULL**)**

**{**

**if** **(** p**->**num **==** index**)**

**{**

mark **=** 1**;**

cout **<<**setiosflags**(**ios**::**left**)<<**setw**(**8**)<<** p**->**num**<<**setw**(**10**)<<**p**->**duty**<<**endl**;**

**}**

p **=** p**->**next**;**

**}**

**if** **(** mark **==** 0**)**

**{**

cout **<<** **"不存在该员工事务信息"<<**endl**;**

**return;**

**}**

**}**

**/\*\***

**添加员工信息**

**变量定义；**

**提示用户输入所要添加的员工的编号**

**如果存在，提示“已存在”；**

**如果不存在，添加，输出更新信息**

**\*\*/**

emp **\***insert**(**emp **\***head**,** **int** index**,** **char** **\***filename1**)** *//插入一个员工*

**{**

cout **<<** **"现在共有 "<<** amount**(**head**)<<" 个员工"<<**endl**;**

printf**("您要插入编号为%d的员工\n",**index**);**

emp **\***p **=** head**;**

emp **\***newnode **=** **new** emp**;**

**while** **(** p **!=** NULL**)**

**{**

**if** **(** p**->**num **==** index**)** *//遍历，若需要添加的员工编号已存在，给出提示信息*

**{**

bound**(**'\_'**,**100**);**

cout **<<** endl**;**

cout **<<** **"抱歉！编号已存在！"<<**endl**;**

**return** head**;**

**}**

p **=** p**->**next**;**

**}**

p **=** head**;**

**if** **(**index **==** 1**)** *//分类：插入为首位、末尾、中间*

**{**

cout **<<** **"Loading ... ..."<<**endl**;**

bound**(**'\_'**,**100**);**

cout **<<** endl**;**

cout **<<** **"请输入插入员工的信息(编号 姓名 性别 年龄 部门 薪资 直系领导编号): "<<**endl**;**

cin **>>** newnode**->**num**>>**newnode**->**name**>>**newnode**->**sex

**>>**newnode**->**age**>>**newnode**->**department**>>**newnode**->**salary**>>**newnode**->**leader**;**

system**("cls");**

newnode**->**next **=** head**->**next**;**

head**->**next **=** newnode**;**

**}**

**else**

**{**

**while(**p**->**next**!=**NULL**)**

**{**

p **=** p**->**next**;**

**}**

**if** **(** index **>** p**->**num**)** *//判断输入数值和最后一个emp的编号大小，若大，则末尾*

**{**

cout **<<** **"Loading ... ..."<<**endl**;**

bound**(**'\_'**,**100**);**

cout **<<** endl**;**

cout **<<** **"请输入插入员工的信息(编号 姓名 性别 年龄 部门 薪资 直系领导编号): "<<**endl**;**

cin **>>** newnode**->**num**>>**newnode**->**name**>>**newnode**->**sex

**>>**newnode**->**age**>>**newnode**->**department**>>**newnode**->**salary**>>**newnode**->**leader**;**

p **=** head**;**

**while(**p**->**next **!=**NULL**)**

**{**

p **=** p**->**next**;**

**}**

p**->**next **=** newnode**;**

newnode**->**next **=** NULL**;**

**}**

**else** *//若小，则中间*

**{**

cout **<<** **"Loading ... ..."<<**endl**;**

bound**(**'\_'**,**100**);**

cout **<<** endl**;**

cout **<<** **"请输入插入员工的信息(编号 姓名 性别 年龄 部门 薪资 直系领导编号): "<<**endl**;**

cin **>>** newnode**->**num**>>**newnode**->**name**>>**newnode**->**sex

**>>**newnode**->**age**>>**newnode**->**department**>>**newnode**->**salary**>>**newnode**->**leader**;**

p **=** head**->**next**;**

emp **\***pre **=** head**;**

**while(**p**)** *//循环保证上一个节点指向要插入的上一个位置*

**{**

**if(**pre**->**num **<** index **&&** index **<** p**->**num**)**

**{**

**break;**

**}**

p **=** p**->**next**;**

pre **=** pre**->**next**;**

**}**

**if** **(** p**->**next **==** NULL**)**

**{**

p**->**next **=** newnode**;**

newnode**->**next **=** NULL**;**

**}else** **if** **(**pre**->**num **<** index **&&** index **<** p**->**num**)**

**{**

pre**->**next **=** newnode**;**

newnode**->**next **=** p**;**

**}**

**}**

**}**

system**("cls");**

save**(**filename1**,** head**);**

cout **<<** **"更新完成！"<<**endl**;**

cout **<<** **"更新信息如下："<<**endl**;**

display**(**head**);**

**return** head**;**

**}**

**/\*\***

**删除员工信息**

**变量定义；**

**提示用户输入所要删除的员工的编号**

**如果存在，删除（包括该员工事务信息），输出更新信息**

**如果不存在，输出“没有此员工”**

**\*/**

emp **\***remove**(**emp **\***head**,** thing **\***head2**,** **int** index**,** **char** **\***filename1**,char** **\***filename2**)** *//删除员工信息*

**{**

**int** mark **=** 0**;** *//定义标识符*

**char** name**[**20**];**

cout **<<"请输入姓名：";**

cin **>>** name**;**

emp **\***p1 **=** head**->**next**;** *//head节点没东西*

**while** **(** p1**!=**NULL**)**

**{**

**if** **(** p1**->**num **==** index**)**

**{**

*// cout <<"name is = " <<name<<endl;*

*// cout << "p->name is = "<<p1->name<<endl;*

**if** **(** **(**string**)**p1**->**name **!=** **(**string**)**name**)** *//char型时为什么这里是== 而不是 ！= 2020.11.16修改完成，改为string型*

**{**

cout **<<"信息不匹配";**

**return** head**;**

**}**

**else**

**{**

mark **=** 1**;**

**}**

**}**

p1 **=** p1**->**next**;**

**}**

**if** **(** mark **==** 0**)**

**{**

cout **<<** **"不存在该员工"<<**endl**;**

**return** head**;**

**}**

cout **<<** **"现在共有 "<<** amount**(**head**)<<" 个员工"<<**endl**;**

printf**("您要删除的是编号为%d的员工\n",**index**);**

cout **<<** **"Loading ... ..."<<**endl**;**

system**("cls");**

emp **\***pre **=** head**;**

emp **\***p **=** head**->**next**;**

**while(**p **!=** NULL**)**

**{**

**if** **(** p**->**num **==** index**)**

**{**

**if(**p**->**next **==** NULL**)** *//p为末尾时*

**{**

pre**->**next **=** NULL**;**

**}**

**else** **if(**index **==** 1**)** *//p为首位时*

**{**

head **=** pre**->**next**;**

**}**

**else** *//中间情况*

**{**

pre**->**next **=** p**->**next**;**

**}**

**delete** p**;**

**}**

p **=** p**->**next**;**

pre **=** pre**->**next**;**

**}**

remove2**(**head2**,** index**,** filename2**);** *//删除员工全部信息，包括事务信息，调用子函数*

save**(**filename1**,** head**);** *//保存至文件*

cout **<<** **"更新完成！"<<**endl**;**

cout **<<** **"更新信息如下："<<**endl**;**

display**(**head**);** *//显示更新信息*

**return** head**;**

**}**

//删除事务信息子函数

thing **\***remove2**(**thing **\***head**,** **int** index**,** **char** **\***filename1**)***//删除事务信息（由于题目要求，这个函数并为列为单一功能使用，而是作为子函数使用）*

**{**

thing **\***p **=** head**->**next**;**

thing **\***pre **=** head**;**

**while(**p **!=** NULL**)**

**{**

**if** **(** p**->**num **==** index**)**

**{**

**if(**p**->**next **==** NULL**)** *//末尾*

**{**

pre**->**next **=** NULL**;**

**}**

**else** **if(**index **==** 1**)** *//首位*

**{**

head **=** pre**->**next**;**

**}**

**else** *//中间情况*

**{**

pre**->**next **=** p**->**next**;**

**}**

**delete** p**;**

**}**

p **=** p**->**next**;**

pre **=** pre**->**next**;**

**}**

save2**(**filename1**,** head**);** *//保存至文件*

**return** head**;**

**}**

**/\*\***

**修改员工信息**

**变量定义；**

**提示用户输入所要修改的内容，0-全部修改，1-编号，2-姓名，3-性别，4-年龄，5-部门，6-薪资，7-直系领导编号；**

**如果存在，先输出这个员工的原始信息，再修改这个员工指定信息，输出更新信息；**

**如果不存在，输出“没有此员工”**

**\*/**

emp **\***revise**(**emp **\***head**,** **int** index**,** **char** **\***filename1**)** *//修改一个员工信息*

**{**

**int** mark **=** 0**;**

emp **\***p **=** head**->**next**;** *//head节点没东西*

**while** **(** p**!=**NULL**)**

**{**

**if** **(** p**->**num **==** index**)** *//检测到后标识符为1*

**{**

mark **=** 1**;**

**break;**

**}**

p **=** p**->**next**;**

**}**

**if** **(** mark **==** 0**)**

**{**

cout **<<** **"不存在该员工"<<**endl**;**

**return** head**;**

**}**

cout **<<** **"现在共有 "<<** amount**(**head**)<<" 个员工"<<**endl**;**

printf**("您要修改编号为%d的员工原信息如下：\n",**index**);**

cout **<<** **"Loading ... ..."<<**endl**<<**endl**;**

bound**(**'\_'**,**100**);**

cout **<<**endl**;**

p **=** head**->**next**;** *//重置p头指针*

emp **\***pre **=** head**;** *//定义一个p前一项的指针*

**while(**p **!=** NULL**)**

**{**

**if** **(** p**->**num **==** index**)**

**{**

printf**("编号为%d的员工原信息如下：\n",**index**);**

cout **<<**setiosflags**(**ios**::**left**)<<**setw**(**10**)<<** p**->**num**<<**setw**(**8**)<<**p**->**name**<<**setw**(**4**)**

**<<**p**->**sex**<<**setw**(**6**)<<** p**->**age**<<**setw**(**15**)**

**<<**p**->**department**<<**setw**(**10**)<<**p**->**salary**<<**setw**(**2**)<<**p**->**leader**<<**endl**;**

emp **\***newnode **=** **new** emp**;**

newnode **=** p**;**

cout **<<** **"\n您想要修改的是 0.全部修改 1.编号 2.姓名 3.性别 4.年龄 5.部门 6.薪资 7.直系领导编号"<<**endl**;**

cout **<<** **"请输入：";**

**int** option**[**8**],** cnt **=** 0**;** *//定义选项数 数组*

**do{**

cin **>>** option**[**cnt**];**

**if(**option**[**cnt**]** **==** 0**)**

**{**

**break;**

**}**

cnt**++;**

**}while(**getchar**()** **!=** '\n'**);** *//以防用户输入多个需要修改的数据*

**if** **(** option**[**0**]** **==** 0**)**

**{**

cout **<<** **"请输入修改员工的信息(编号 姓名 性别 年龄 部门 薪资 直系领导编号): "<<**endl**;**

cin **>>** newnode**->**num**>>**newnode**->**name**>>**newnode**->**sex

**>>**newnode**->**age**>>**newnode**->**department**>>**newnode**->**salary**>>**newnode**->**leader**;**

**}**

**for** **(** **int** i **=** 0**;** i **<** cnt**;**i**++)**

**{**

**if** **(** option**[**i**]** **==** 1**)**

**{**

cout **<<** **"请输入修改编号：";**

cin **>>** newnode**->**num**;**

**}**

**if** **(** option**[**i**]** **==** 2**)**

**{**

cout **<<** **"请输入修改姓名：";**

cin **>>** newnode**->**name**;**

**}**

**if** **(** option**[**i**]** **==** 3**)**

**{**

cout **<<** **"请输入修改性别：";**

cin **>>** newnode**->**sex**;**

**}**

**if** **(** option**[**i**]** **==** 4**)**

**{**

cout **<<** **"请输入修改年龄：";**

cin **>>** newnode**->**age**;**

**}**

**if** **(** option**[**i**]** **==** 5**)**

**{**

cout **<<** **"请输入修改部门：";**

cin **>>** newnode**->**department**;**

**}**

**if** **(** option**[**i**]** **==** 6**)**

**{**

cout **<<** **"请输入修改薪资：";**

cin **>>** newnode**->**salary**;**

**}**

**if** **(** option**[**i**]** **==** 7**)**

**{**

cout **<<** **"请输入修改直系领导编号：";**

cin **>>** newnode**->**leader**;**

**}**

**}**

**}**

p **=** p**->**next**;**

pre **=** pre**->**next**;**

**}**

system**("cls");**

save**(**filename1**,** head**);**

cout **<<** **"更新完成！"<<**endl**;**

cout **<<** **"更新信息如下："<<**endl**;**

display**(**head**);**

**return** head**;**

**}**

**/\*\***

**排序函数，进入排序功能子菜单**

**输出选择排序方式**

**按照工资**

**\*/**

**void** rank**(**emp **\***head**)** *//根据工资排序*

**{**

paixv**(**head**);** *//调用排序函数*

cout **<<**endl**;**

display**(**head**);** *//显示*

**}**

**//排序子函数**

**void** paixv**(**emp **\***head**)** *//对所给员工的工资从高到低排序（采用冒泡排序法）*

**{**

emp **\***p **=** head**->**next**->**next**;**

emp **\***pre **=** head**->**next**;**

**int** tmp **=** 0**;** *//定义中间变量*

**for** **(** pre **=** head**->**next**;** pre **!=** NULL **;**pre **=** pre**->**next**)**

**{**

**for(** p **=** pre**->**next**;**p **!=** NULL**;**p **=** p**->**next**)**

**{**

**if** **(**pre**->**salary **<** p**->**salary**)** *//判断薪资大小*

**{**

tmp **=** pre**->**salary**;**

pre**->**salary **=** p**->**salary**;**

p**->**salary **=** tmp**;**

tmp **=** pre**->**num**;**

pre**->**num**=** p**->**num**;**

p**->**num **=** tmp**;**

tmp **=** pre**->**age**;**

pre**->**age**=** p**->**age**;**

p**->**age **=** tmp**;**

tmp **=** pre**->**leader**;**

pre**->**leader **=** p**->**leader**;**

p**->**leader **=** tmp**;**

**for(** **int** x **=** 0**;**x **<** **sizeof(**pre**->**name**);**x**++)** *//由于调整顺序，而name是char型，直接用string有一定问题*

**{**

**char** a**[**20**];**

a**[**x**]** **=** pre**->**name**[**x**];**

pre**->**name**[**x**]** **=** p**->**name**[**x**];**

p**->**name**[**x**]** **=** a**[**x**];**

**}**

**for(** **int** x **=** 0**;**x **<** **sizeof(**pre**->**department**);**x**++)**

**{**

**char** a**[**20**];**

a**[**x**]** **=** pre**->**department**[**x**];**

pre**->**department**[**x**]** **=** p**->**department**[**x**];**

p**->**department**[**x**]** **=** a**[**x**];**

**}**

**for(** **int** x **=** 0**;**x **<** **sizeof(**pre**->**sex**);**x**++)**

**{**

**char** a**[**10**];**

a**[**x**]** **=** pre**->**sex**[**x**];**

pre**->**sex**[**x**]** **=** p**->**sex**[**x**];**

p**->**sex**[**x**]** **=** a**[**x**];**

**}**

**}**

**}**

**}**

**}**

**/\*\***

**领导层级：输出指定编号的员工直系领导信息**

**若有：输出信息；**

**若没有：输出“不存在领导”**

**\*/**

**void** leader**(**emp **\***head**,** **int** index**)** *//直系领导 .debug 建议，对指针p,q 的循环判断关键在于看：出循环后p，q是否需要使用，要p->next!=NULL,不要p!=NUll*

**{**

**int** mark **=** 0**,** lab **=** 0**;** *//定义两个标记符*

emp **\***p **=** head**->**next**;** *//head节点没东西*

**while** **(** p**!=**NULL**)** *//不需要对p进行后续操作，因此遍历到p*

**{**

**if** **(** p**->**num **==** index**)**

**{**

mark **=** 1**;** *//标记已经记录到*

lab **=** p**->**leader**;** *//记录leader的编号*

**break;** *//记录到就跳出循环*

**}**

p **=** p**->**next**;**

**}**

**if** **(** mark **==** 0**)**

**{**

cout **<<** **"不存在该员工"<<**endl**;**

**return;**

**}**

p **=** head**->**next**;**

emp **\***q **=** head**->**next**;**

*// cout << " lab = "<<lab<<endl; //调试用*

**while(** p**->**num **!=** lab **)** *//循环条件为判断这个人的序号是否为所要求领导的序号，若不是，继续下一个*

**{**

p **=** p**->**next**;**

**if(p->next == NULL && p->num != lab)**

**{**

**cout << "不存在领导信息"<<endl;**

**return;**

**}**

**}**

mark **=** p**->**num**;**

**int** label **=** 0**;**

*// paixv(head);*

*// cout << "mark = "<<mark<<endl;*

printf**("编号为%d的直系领导如下：\n",**index**);**

**while** **(**q **!=** NULL**)**

**{**

**if** **(** q**->**num **==** mark **)**

**{**

cout **<<"领导"<<": "<<**setiosflags**(**ios**::**left**)<<**setw**(**10**)<<** q**->**num**<<**setw**(**8**)<<**q**->**name**<<**setw**(**4**)**

**<<**q**->**sex**<<**setw**(**6**)<<** q**->**age**<<**setw**(**15**)**

**<<**q**->**department**<<**setw**(**10**)<<**q**->**salary**<<**setw**(**2**)<<**q**->**leader**<<**endl**;**

label **=** 1**;** *//相当于上述的mark 标记已经检测到*

**}**

q **=** q**->**next**;**

**}**

**}**

**/\*\***

**查询函数，进入查询功能子菜单**

1. **查询全部事务**
2. **查询个人事务**

**\*/**

**//查询全部事务如下：**

**void** display2 **(**thing **\***head**)** *//展示全部事务信息*

**{**

thing **\***p **=** head**->**next**;** *//head节点没东西*

**while** **(** p **!=** NULL**)**

**{**

cout **<<**setiosflags**(**ios**::**left**)<<**setw**(**8**)<<** p**->**num**<<**setw**(**8**)<<**p**->**duty**<<**endl**;**

p **=** p**->**next**;**

**}**

**}**

**//查询个人事务如下：**

**void** displayperson2 **(**thing **\***head**,** **int** index**)** *//显示个人信息 （注释信息同员工函数）*

**{**

**int** mark **=** 0**;**

thing **\***p **=** head**->**next**;** *//head节点没东西*

**while** **(** p**->**next**!=**NULL**)**

**{**

**if** **(** p**->**num **==** index**)**

**{**

mark **=** 1**;**

cout **<<**setiosflags**(**ios**::**left**)<<**setw**(**8**)<<** p**->**num**<<**setw**(**10**)<<**p**->**duty**<<**endl**;**

**}**

p **=** p**->**next**;**

**}**

**if** **(** mark **==** 0**)**

**{**

cout **<<** **"不存在该员工事务信息"<<**endl**;**

**return;**

**}**

**}**

**/\*\***

**添加事务信息**

**变量定义；**

**提示用户输入所要添加的被分配事务员工的编号和职责**

**如果不存在，添加，输出更新信息**

**\*\*/**

thing **\***insert2**(**thing **\***head2**,** emp **\***head**,** **int** index**,** **char** **\***filename1**)** *//插入一个事务 （注释信息同员工函数）*

**{**

**int** mark **=** 0**;**

emp **\***p1 **=** head**->**next**;** *//head节点没东西*

**while** **(** p1**!=**NULL**)**

**{**

**if** **(** p1**->**num **==** index**)**

**{**

mark **=** 1**;**

**}**

p1 **=** p1**->**next**;**

**}**

**if** **(** mark **==** 0**)**

**{**

cout **<<** **"不存在该员工"<<**endl**;**

**}**

**if** **(** mark **!=** 0**)**

**{**

printf**("您要插入编号为%d的员工事务\n",**index**);**

cout **<<** **"Loading ... ..."<<**endl**;**

bound**(**'\_'**,**100**);**

cout **<<** endl**;**

thing **\***p **=** head2**;**

thing **\***newnode **=** **new** thing**;**

**while** **(** p **!=** NULL**)**

**{**

**if** **(** p**->**num **==** index**)**

**{**

cout **<<** **"请输入插入事务的信息(编号 职务): "<<**endl**;**

cin **>>** newnode**->**num**>>**newnode**->**duty**;**

system**("cls");**

newnode**->**next **=** p**->**next**;**

p**->**next **=** newnode**;**

**save2(filename1, head2);**

**cout << "更新完成！"<<endl;**

**cout << "更新信息如下："<<endl;**

**display2(head2);**

**return** head2**;**

**}**

p **=** p**->**next**;**

**}**

p **=** head2**;**

**if** **(**index **==** 1**)**

**{**

cout **<<** **"请输入插入事务的信息(编号 职务): "<<**endl**;**

cin **>>** newnode**->**num**>>**newnode**->**duty**;**

system**("cls");**

newnode**->**next **=** head2**->**next**;**

head2**->**next **=** newnode**;**

**}**

**else**

**{**

cout **<<** **"请输入插入事务的信息(编号 职务): "<<**endl**;**

cin **>>** newnode**->**num**>>**newnode**->**duty**;**

p **=** head2**->**next**;**

thing **\***pre **=** head2**;**

**while(**p**)** *//循环保证上一个节点指向要插入的上一个位置*

**{**

**if(**p**->**next **==** NULL **||** **(**pre**->**num **<** index **&&** index **<** p**->**num**))**

**{**

**break;**

**}**

p **=** p**->**next**;**

pre **=** pre**->**next**;**

**}**

**if** **(** p**->**next **==** NULL**)**

**{**

p**->**next **=** newnode**;**

newnode**->**next **=** NULL**;**

**}else** **if** **(**pre**->**num **<** index **&&** index **<** p**->**num**)**

**{**

pre**->**next **=** newnode**;**

newnode**->**next **=** p**;**

**}**

**}**

system**("cls");**

save2**(**filename1**,** head2**);**

cout **<<** **"更新完成！"<<**endl**;**

cout **<<** **"更新信息如下："<<**endl**;**

display2**(**head2**);**

**}**

**return** head2**;**

**}**

**4.2.3 文件读写等工具函数**

**/\*\***

**从文件中读取员工信息**

**\*/**

emp **\***createListX**(char** **\***filename1**)** *//员工文件存在时*

**{**

emp **\***head **=** **new** emp**;** *//头节点 一般不存储数据*

emp **\***pre **=** head**;** *//声明上一个节点*

*// emp head; //这样写head不是动态的，而是存在于栈上*

ifstream infile**;**

infile**.**open**(**filename1**,**ios**::**in**);** *//输入方式打开文件*

string line**;** *//定义一个字符串型line*

**while(** getline**(**infile**,**line**))** *//当还在读入时进入循环*

**{**

emp **\***p **=** **new** emp**;**

stringstream ss**(**line**);** *//对line进行ss操作，将文件中信息存入链表中*

ss **>>**p**->**num**>>**p**->**name**>>**p**->**sex**>>**p**->**age**>>**p**->**department**>>**p**->**salary**>>**p**->**leader**;**

**if** **(** head **==** NULL**)**

**{**

head **=** p**;**

**}**

**else**

**{**

pre**->**next **=** p**;**

**}**

pre **=** p**;**

p**->**next **=** NULL**;**

**}**

infile**.**close**();**

**return** head**;**

**}**

**/\*\***

**从文件中读取事务信息**

**\*/**

thing **\***createListX2**(char** **\***filename1**)** *//事务文件存在时 （注释信息同员工函数）*

**{**

thing **\***head **=** **new** thing**;** *//头节点 一般不存储数据*

thing **\***pre **=** head**;** *//声明上一个节点*

ifstream infile**;**

infile**.**open**(**filename1**,**ios**::**in**);**

string line**;**

**while(** getline**(**infile**,**line**))**

**{**

thing **\***p **=** **new** thing**;**

stringstream ss**(**line**);**

ss **>>**p**->**num**>>**p**->**duty**;**

**if** **(** head **==** NULL**)**

**{**

head **=** p**;**

**}**

**else**

**{**

pre**->**next **=** p**;**

**}**

pre **=** p**;**

p**->**next **=** NULL**;**

**}**

infile**.**close**();**

**return** head**;**

**}**

**/\*\***

**输出录入的员工链表信息**

**\*/**

emp **\***createList**(int** n**,** **char** **\***filename1**)** *//员工文件不存在时*

**{**

emp **\***head **=** **new** emp**;** *//头节点 一般不存储数据*

emp **\***pre **=** head**;** *//声明上一个节点*

*// emp head; //这样写head不是动态的，而是存在于栈上*

ofstream outfile**;** *//定义outfile*

outfile**.**open**(**filename1**,**ios**::**out**);** *//以输出方式打开文件*

**for** **(** **int** i **=** 0**;**i **<** n**;**i**++)**

**{**

emp **\***p **=** **new** emp**;** *//定义一个emp型指针*

printf**("请输入第%d个职员的信息(编号 姓名 性别 年龄 部门 薪资 直系领导编号)\n",**i**+**1**);**

cin **>>**p**->**num**>>**p**->**name**>>**p**->**sex**>>**p**->**age**>>**p**->**department**>>**p**->**salary**>>**p**->**leader**;**

cout **<<** endl**;**

outfile **<<** p**->**num**<<**'\t'**<<**p**->**name**<<**'\t' *//输出信息至文件中，使用：outfile<<*

**<<**p**->**sex**<<**'\t' **<<** p**->**age**<<**'\t'

**<<**p**->**department**<<**'\t'**<<**p**->**salary**<<**'\t'**<<**p**->**leader**<<**endl**;**

**if** **(** head **==** NULL**)**

**{**

head **=** p**;**

**}**

**else**

**{**

pre**->**next **=** p**;**

**}**

pre **=** p**;**

p**->**next **=** NULL**;**

**}**

outfile**.**close**();**

system**("cls");**

**return** head**;**

**}**

**/\*\***

**输出录入的事务链表信息**

**\*/**

thing **\***createList2**(int** n**,** **char** **\***filename1**)** *//事务文件不存在时 （注释信息同员工函数）*

**{**

thing **\***head **=** **new** thing**;** *//头节点 一般不存储数据*

thing **\***pre **=** head**;** *//声明上一个节点*

ofstream outfile**;**

outfile**.**open**(**filename1**,**ios**::**out**);**

**for** **(** **int** i **=** 0**;**i **<** n**;**i**++)**

**{**

thing **\***p **=** **new** thing**;**

printf**("请输入员工的事务信息(员工编号 职务)\n",**i**+**1**);**

cin **>>**p**->**num**>>**p**->**duty**;**

cout **<<** endl**;**

outfile **<<** p**->**num**<<**'\t'**<<**p**->**duty**<<**endl**;**

**if** **(** head **==** NULL**)**

**{**

head **=** p**;**

**}**

**else**

**{**

pre**->**next **=** p**;**

**}**

pre **=** p**;**

p**->**next **=** NULL**;**

**}**

outfile**.**close**();**

system**("cls");**

**return** head**;**

**}**

**/\*\***

**保存修改的全部员工链表信息**

**\*/**

**void** save**(char** **\***filename1**,** emp **\***head**)** *//保存员工*

**{**

emp **\***p **=** head**->**next**;**

ofstream outfile1**(**filename1**);** *//开文件*

**while** **(**p **!=** NULL**)** *//遍历*

**{**

outfile1 **<<** p**->**num**<<**'\t'**<<**p**->**name**<<**'\t' *//存入*

**<<**p**->**sex**<<**'\t' **<<** p**->**age**<<**'\t'

**<<**p**->**department**<<**'\t'**<<**p**->**salary**<<**'\t'**<<**p**->**leader**<<**endl**;**

p **=** p**->**next**;**

**}**

outfile1**.**close**();** *//关文件*

**}**

**/\*\***

**保存修改的全部事务链表信息**

**\*/**

**void** save2**(char** **\***filename1**,** thing **\***head**)** *//保存事务*

**{**

thing **\***p **=** head**->**next**;**

ofstream outfile1**(**filename1**);** *//开文件*

**while** **(**p **!=** NULL**)** *//遍历*

**{**

outfile1 **<<** p**->**num**<<**'\t'**<<**p**->**duty**<<**endl**;** *//存入*

p **=** p**->**next**;**

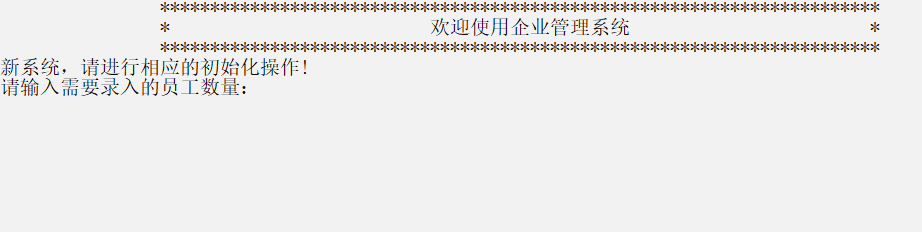
**}**

outfile1**.**close**();** *//关文件*

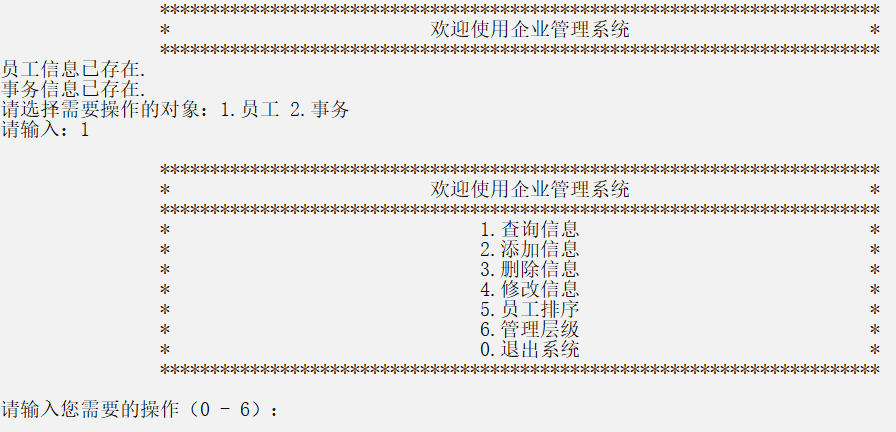
**}**

5.功能测试

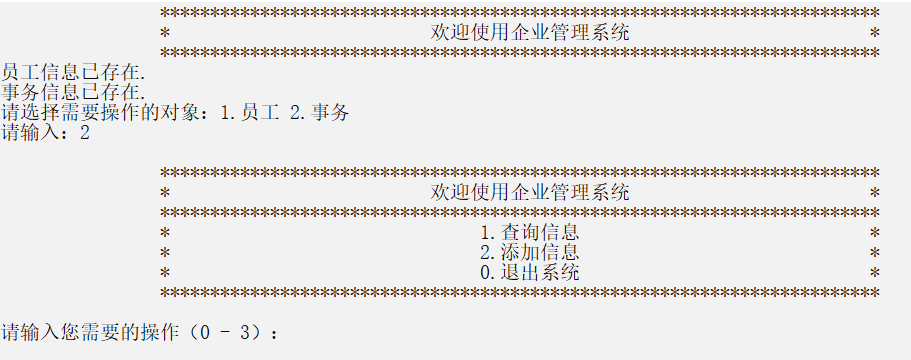
5.1 初始化界面



新系统界面



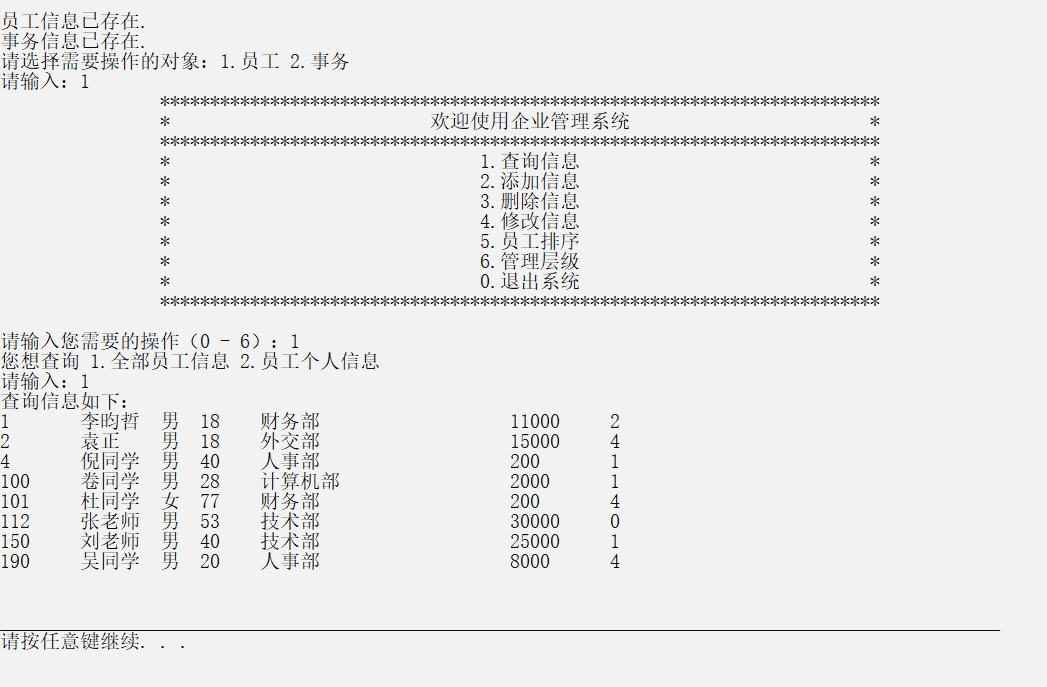
欢迎界面（员工版）



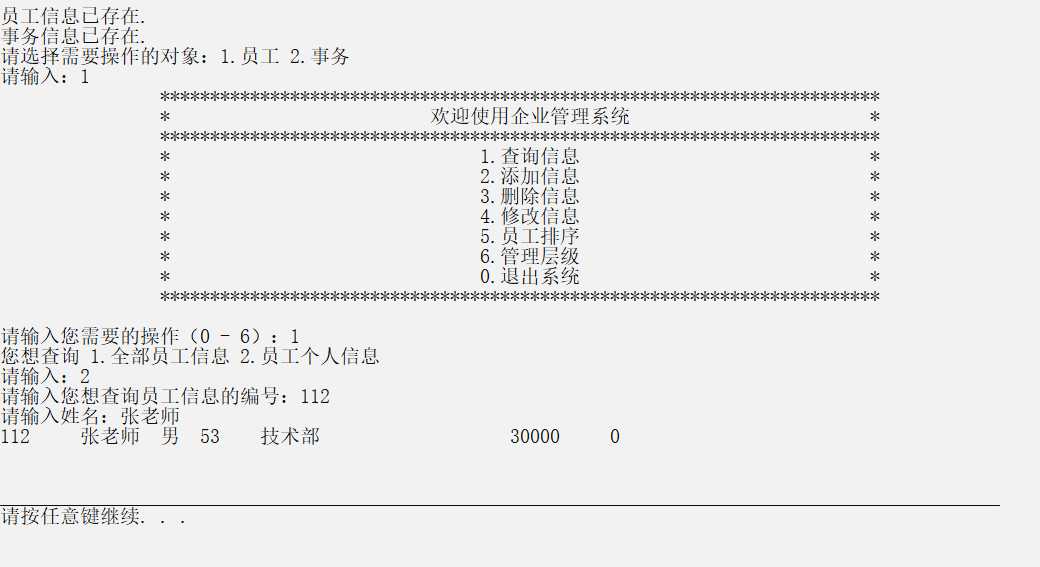
欢迎界面（事务版）

5.2查询员工信息

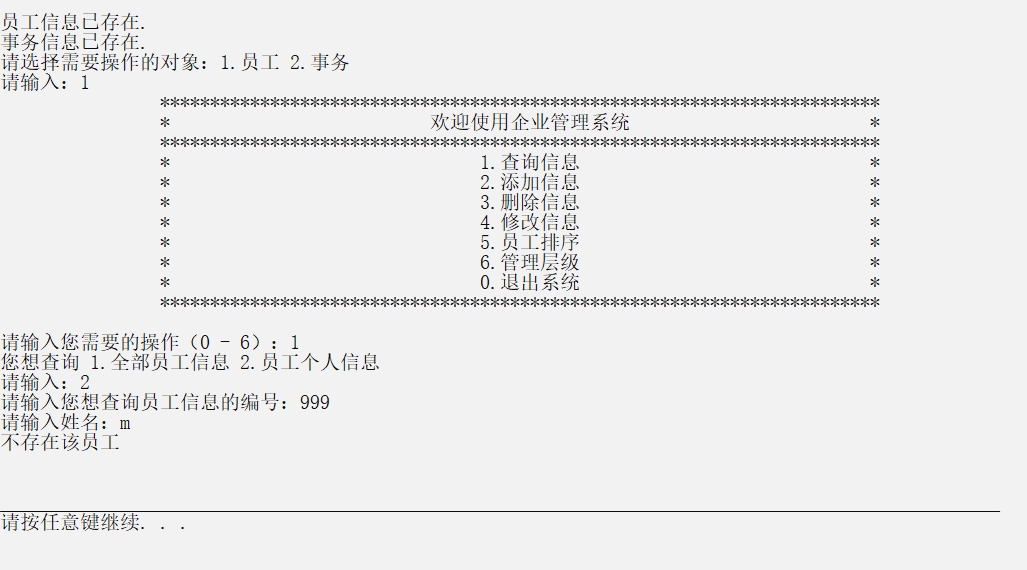
5.2.1全部员工信息



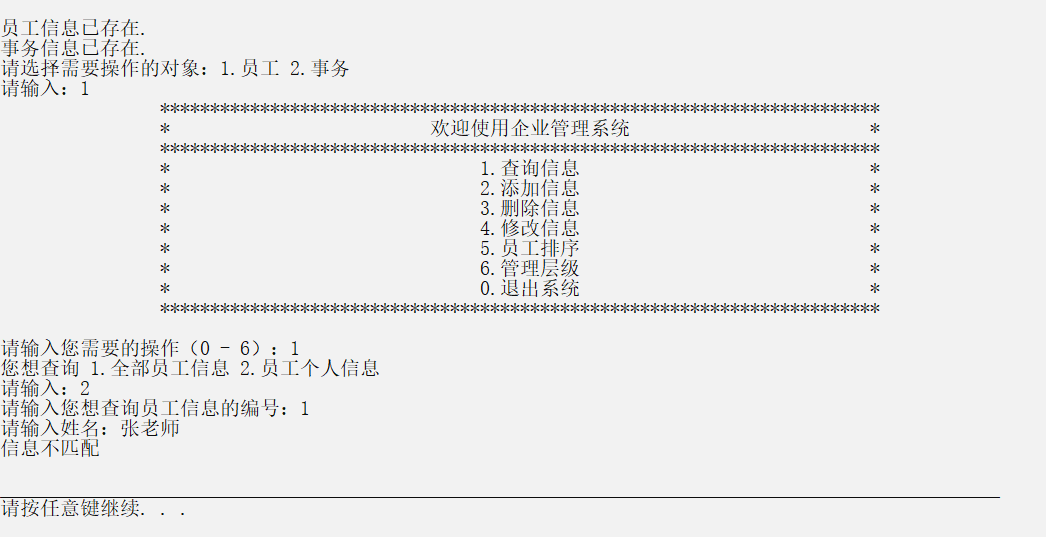
5.2.2某个员工信息



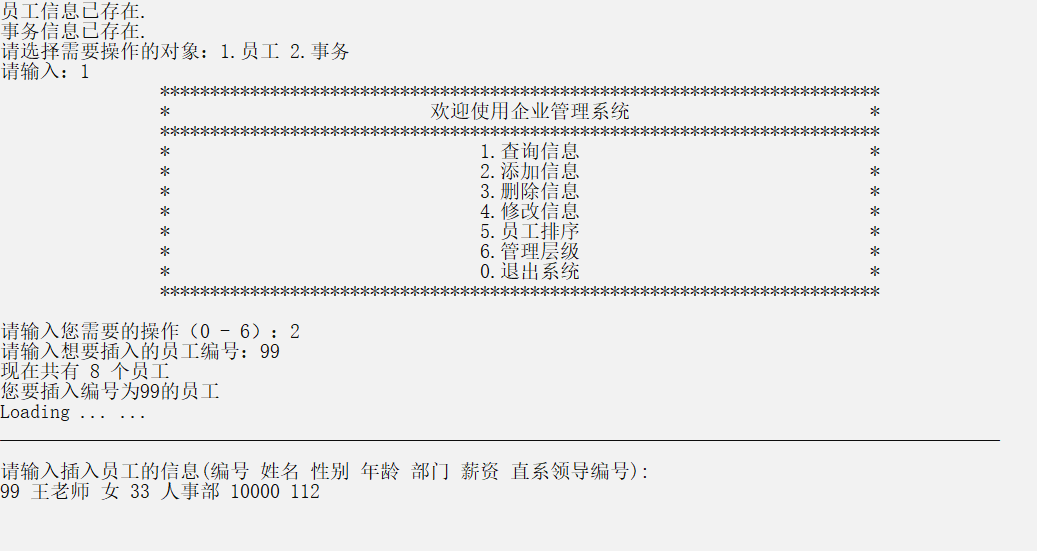
5.2.3员工不存在，输出提示信息



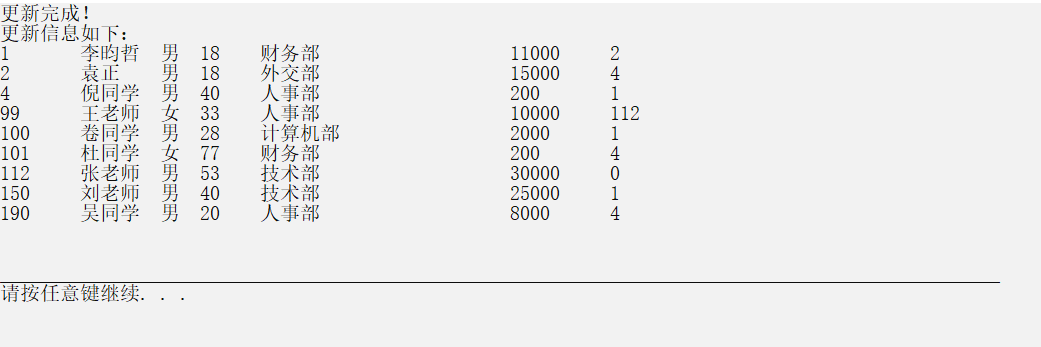
5.2.4 编号与姓名不匹配

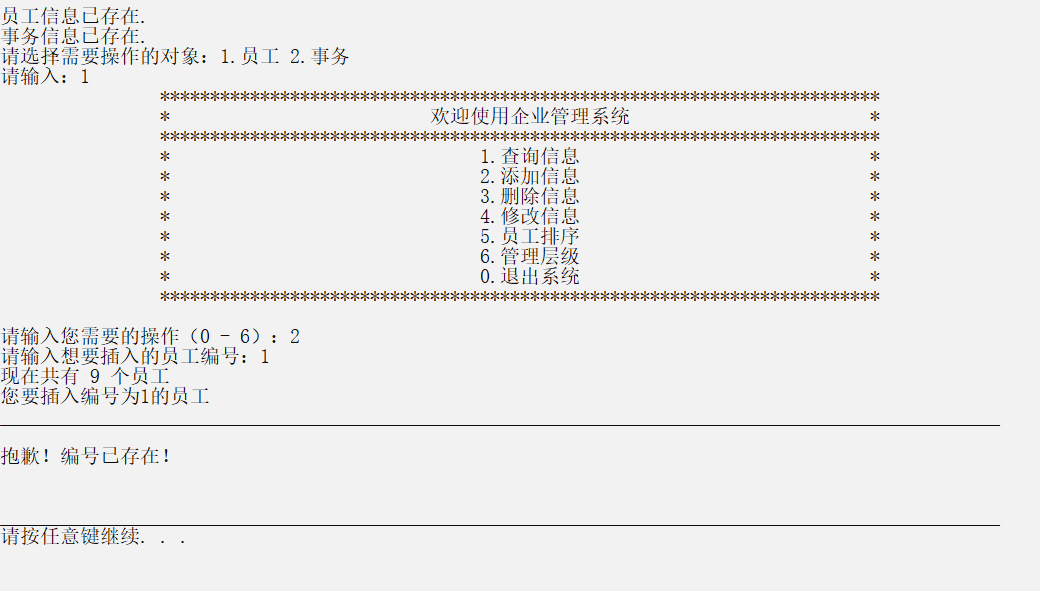


5.3添加员工信息



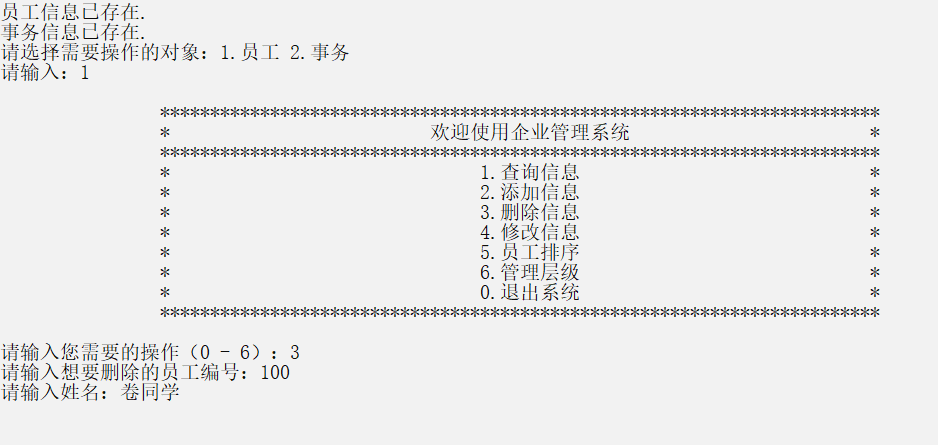
5.3.1添加完成：



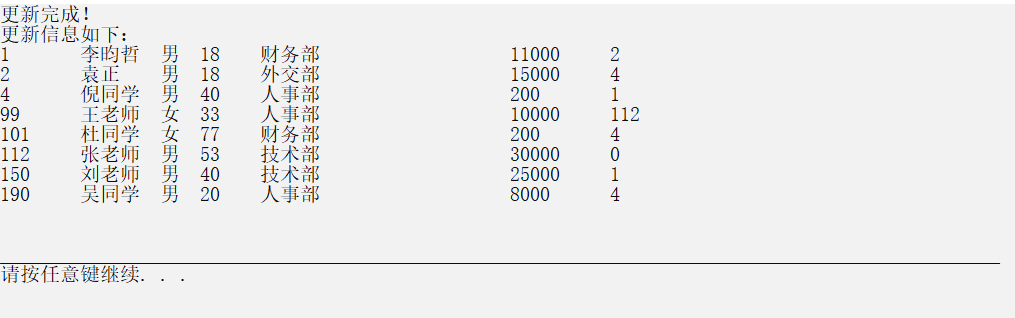
5.3.2添加已存在：  


5.4删除员工和事务信息

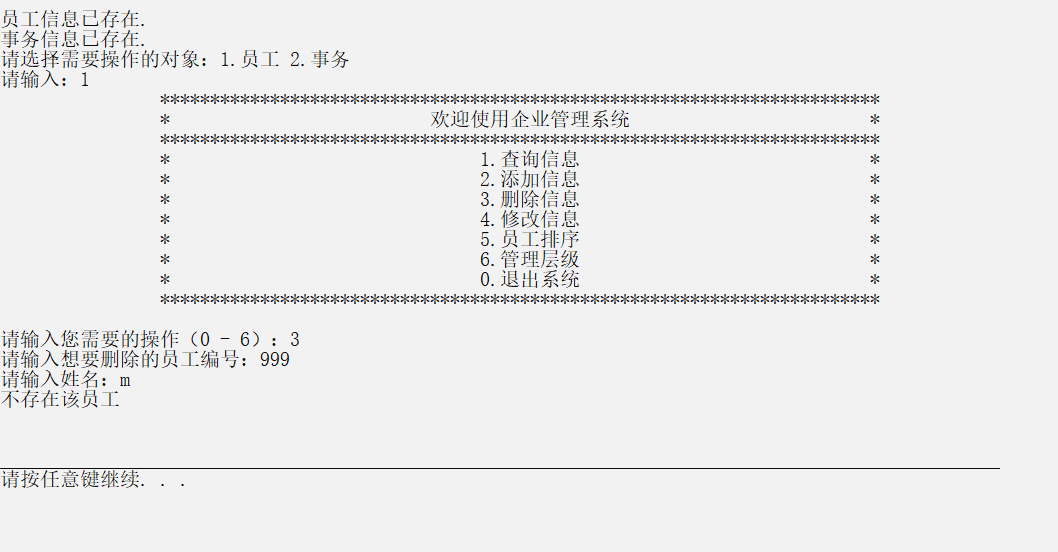
5.4.1员工存在，删除信息。



5.4.2 删除成功 更新信息

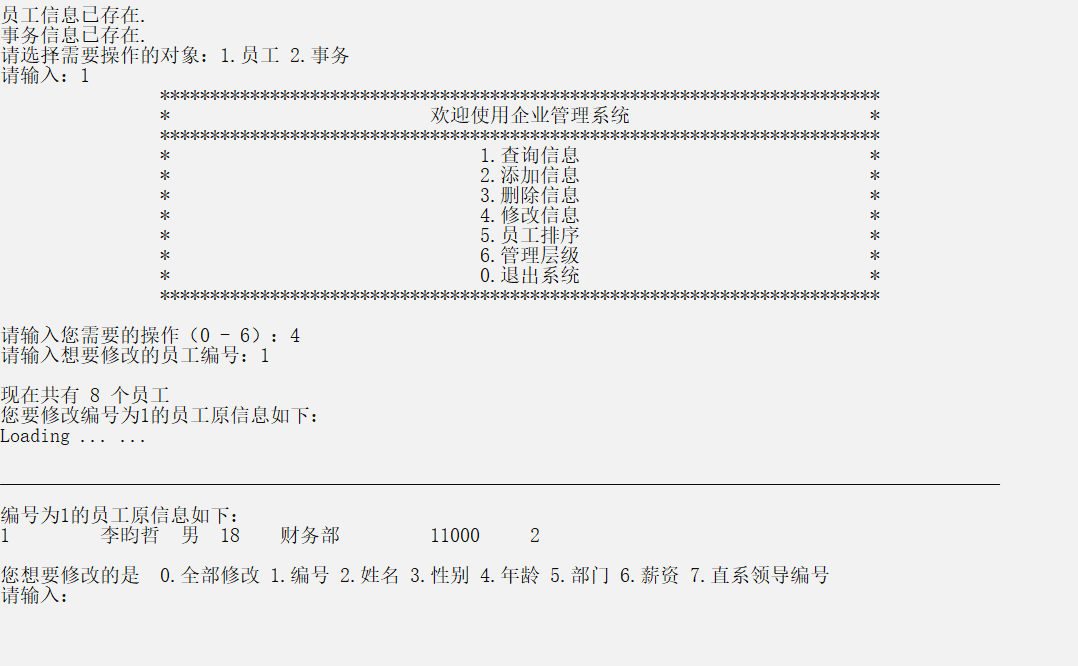


5.4.3员工不存在，输出提示信息。

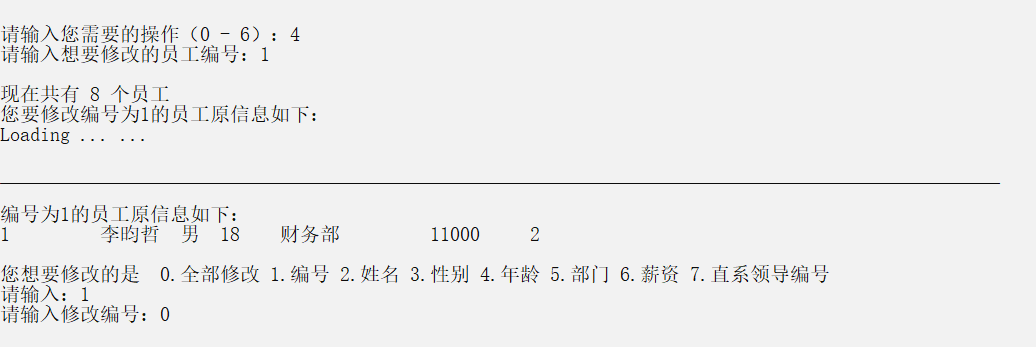


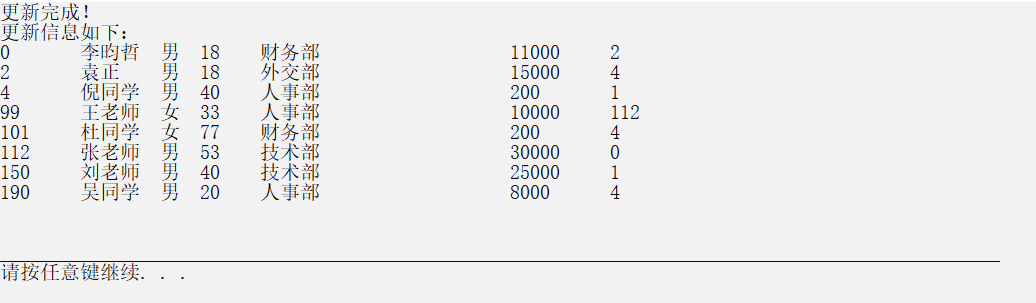
5.5 修改员工信息

5.5.1 修改界面（显示原信息，提示选择修改项目）

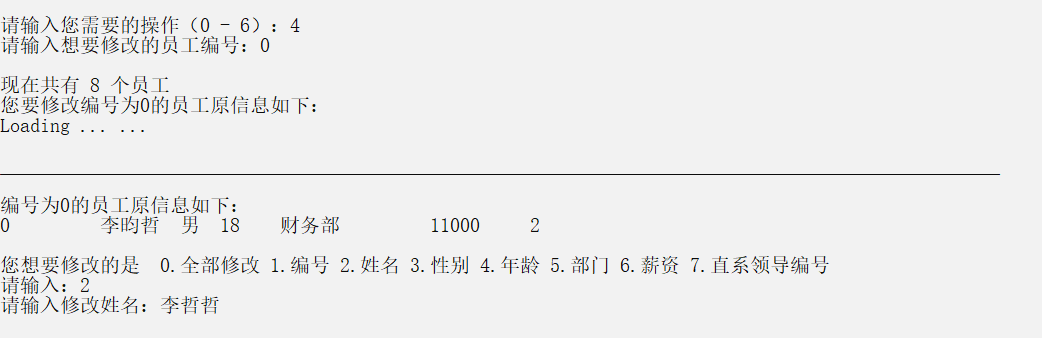


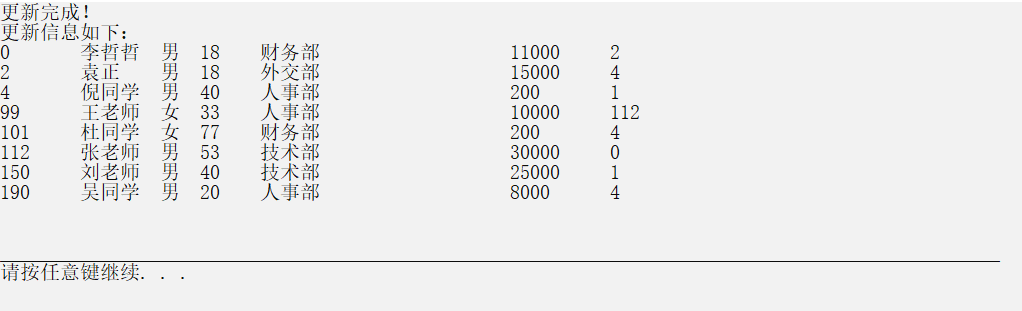
5.5.2 修改编号



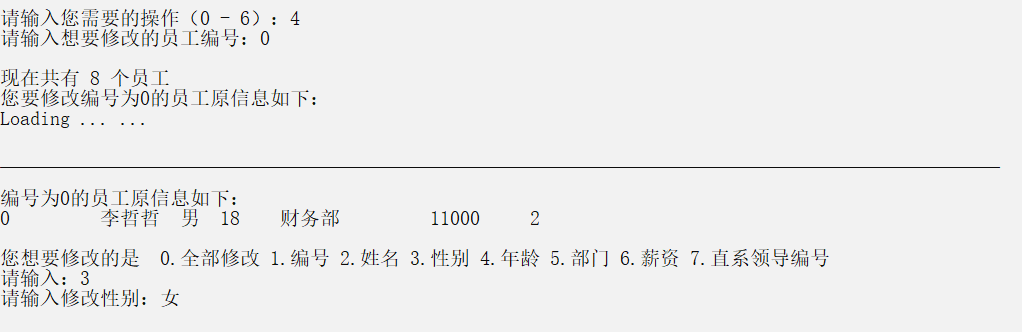


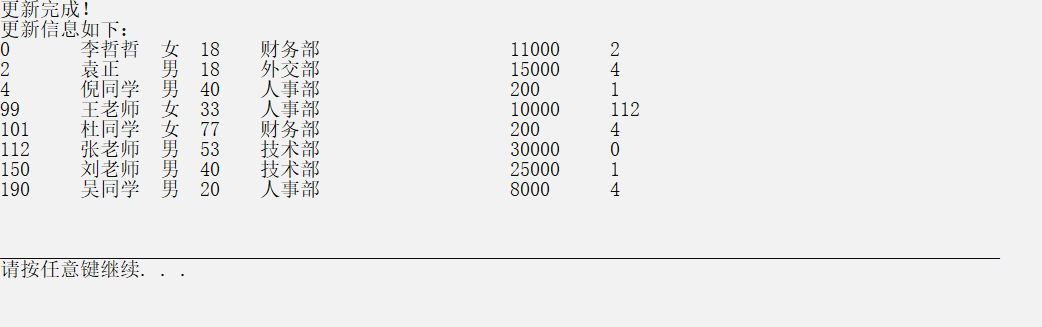
5.5.3 修改姓名



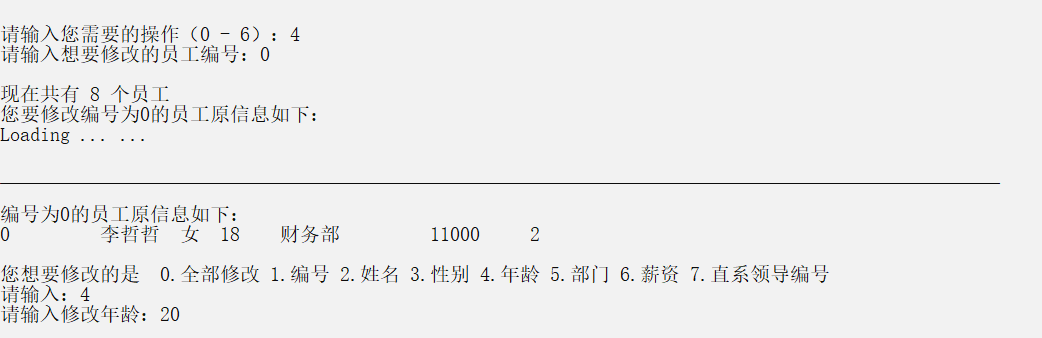


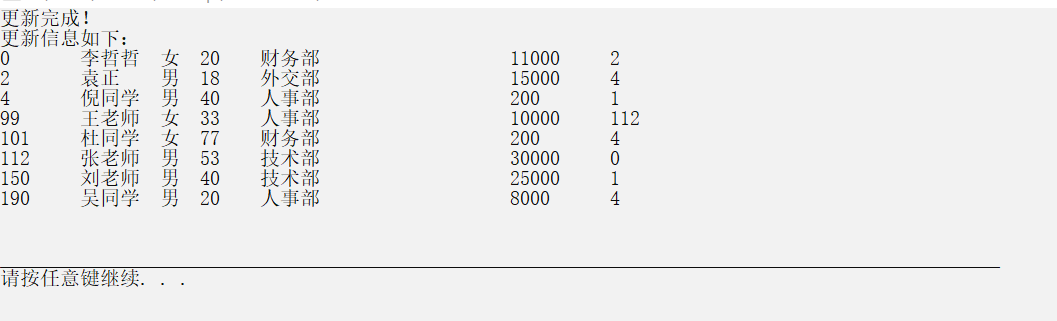
5.5.4 修改性别



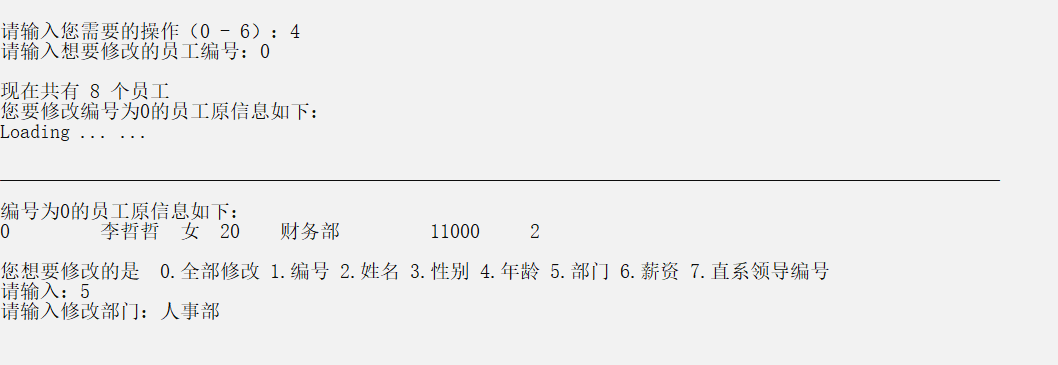


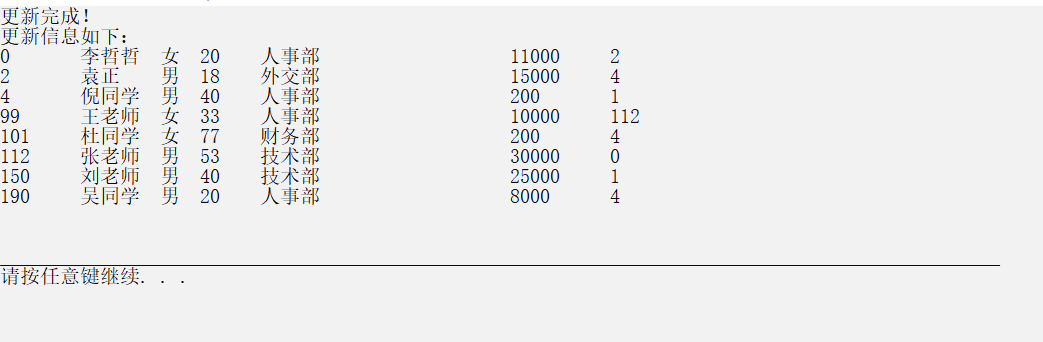
5.5.5 修改年龄



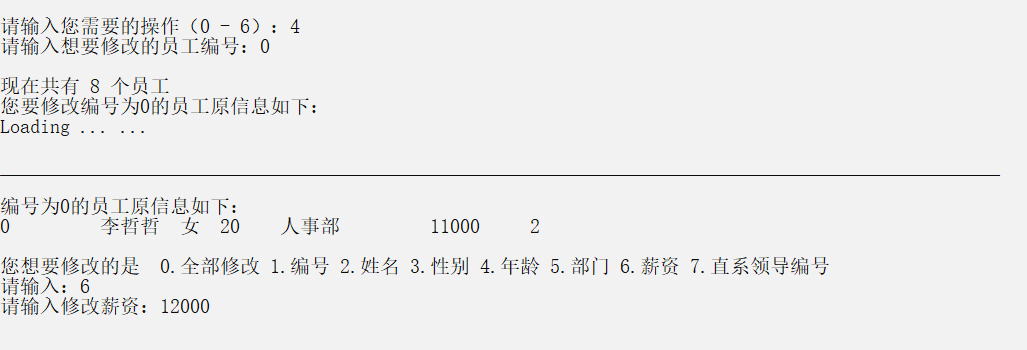


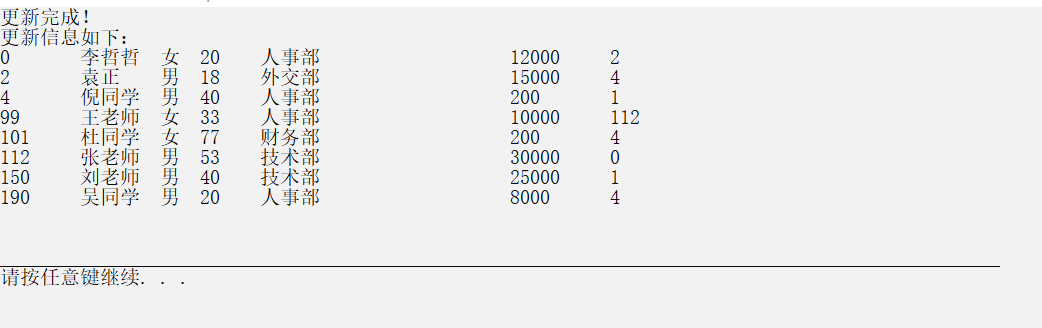
5.5.6 修改部门



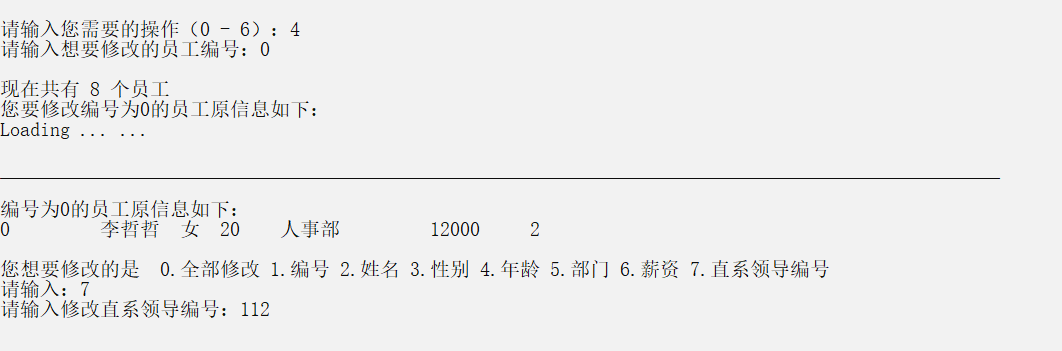


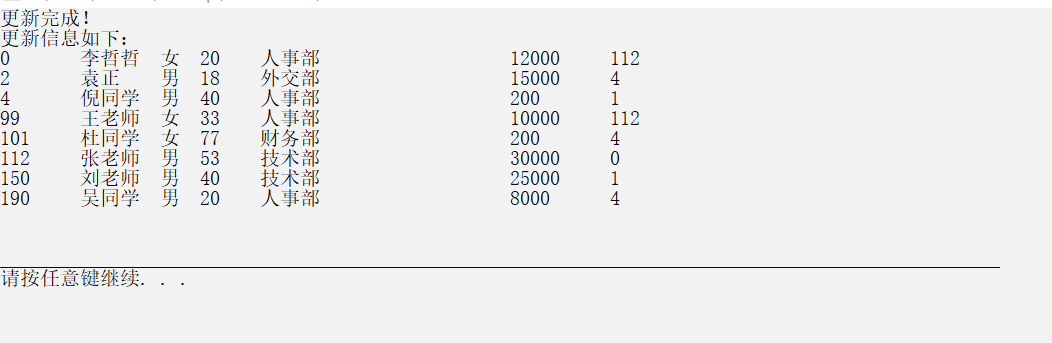
5.5.7 修改薪资



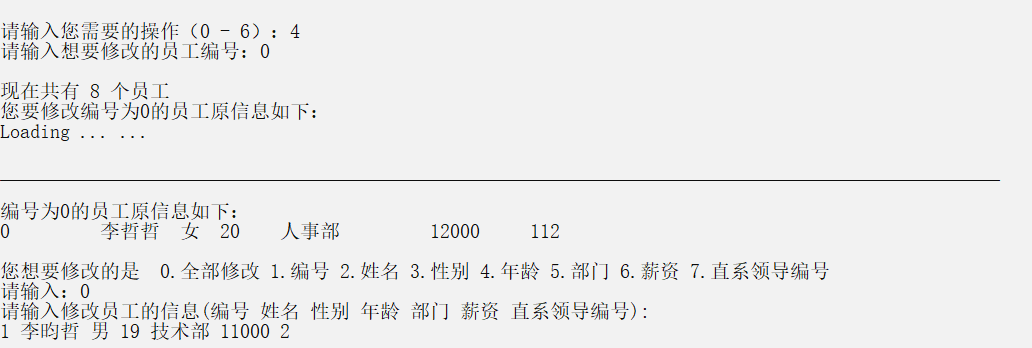


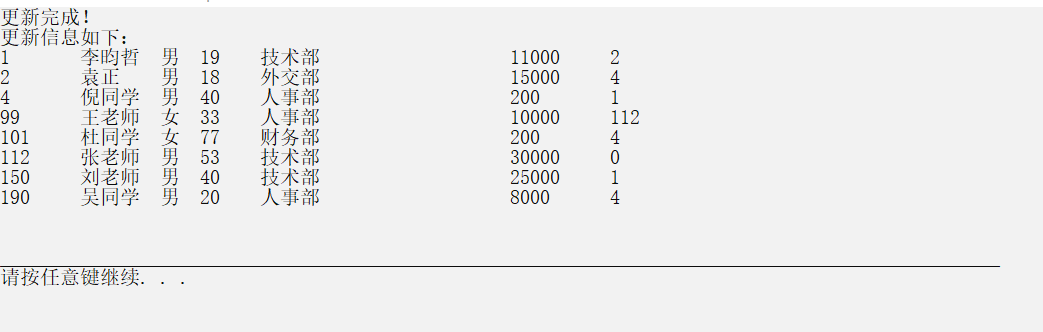
5.5.8 修改直系领导编号



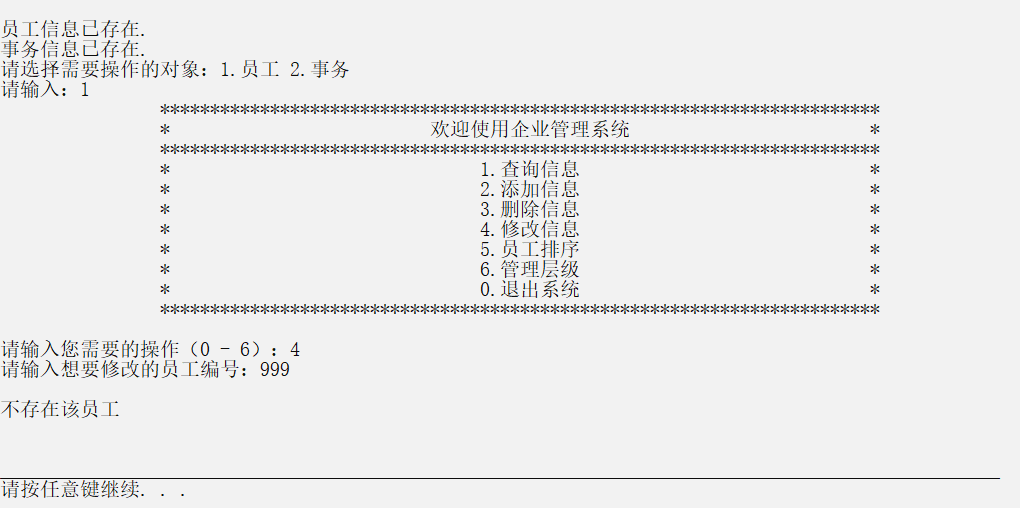


5.5.9 全部更新



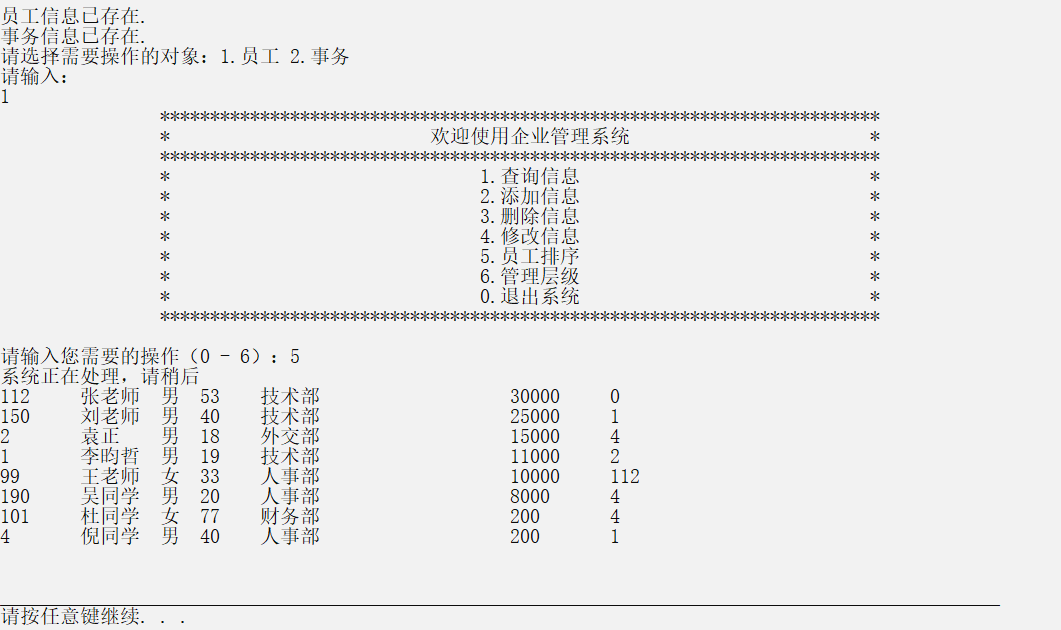


5.5.10员工不存在，输出提示信息。



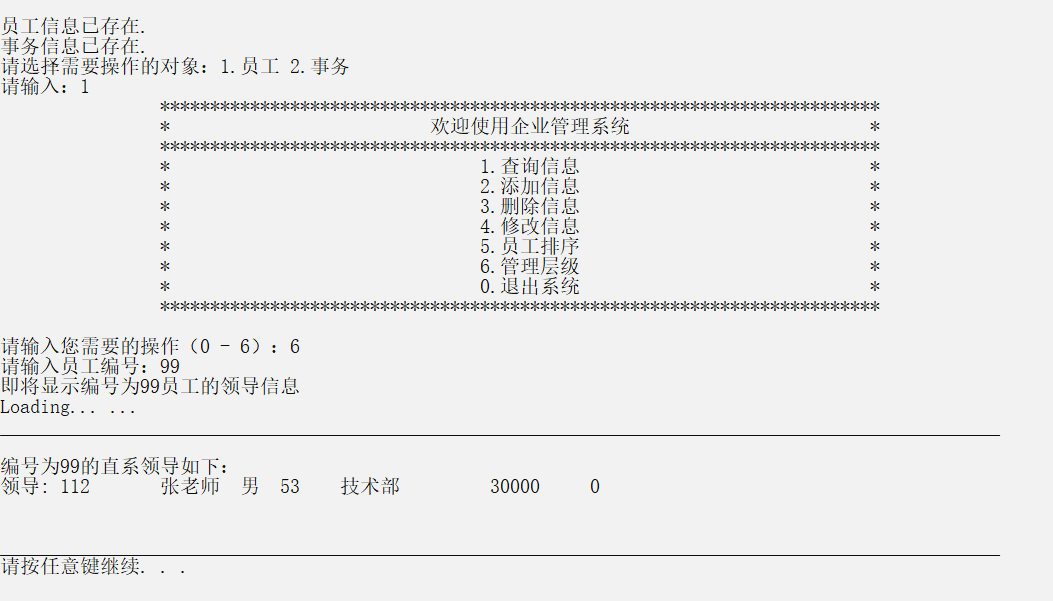
5.6员工排序

5.6.1根据薪资排序

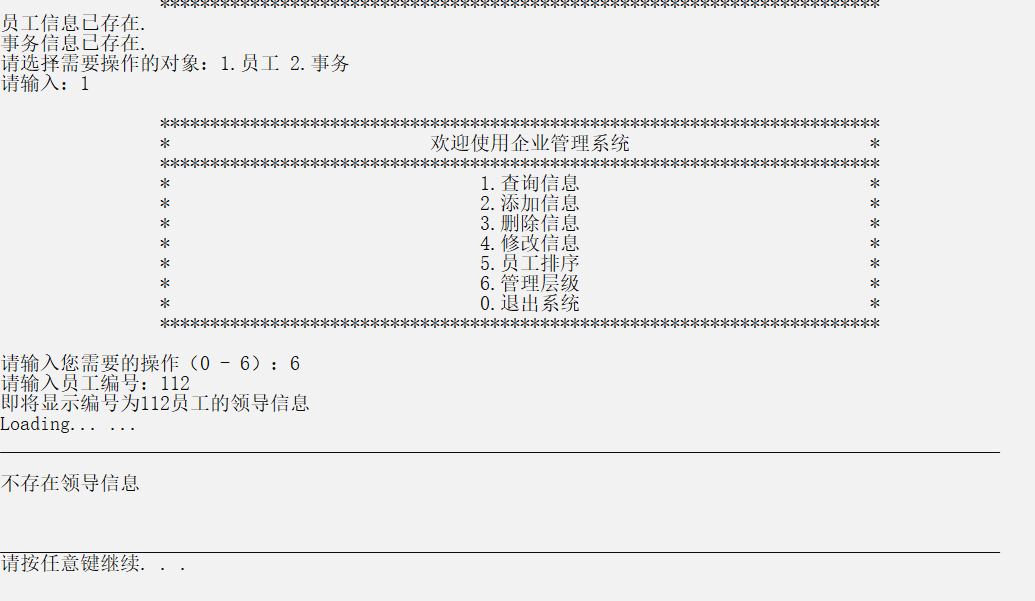


# 5.7 领导层级

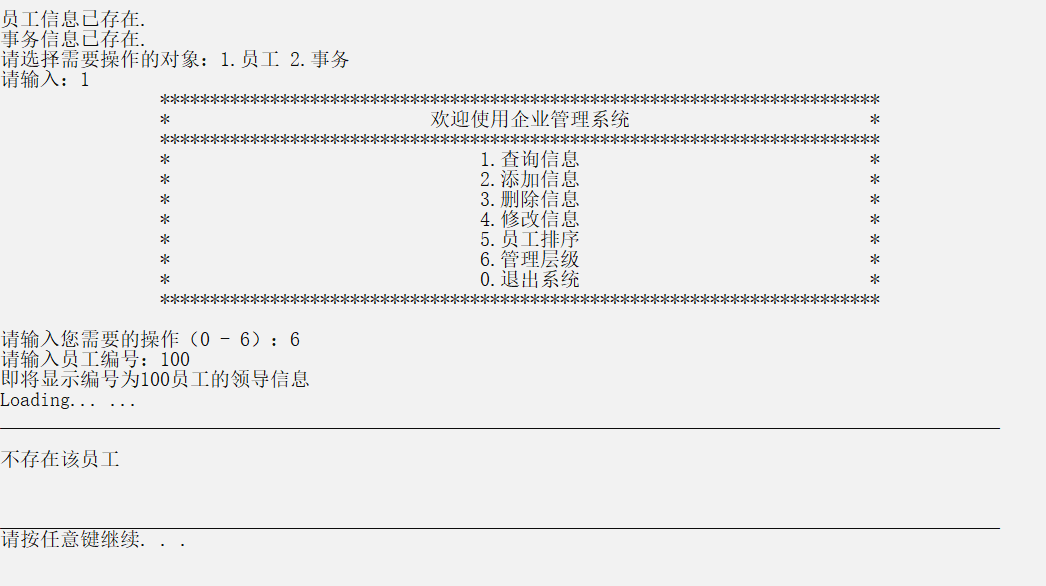
5.7.1 领导存在



5.7.2 领导不存在

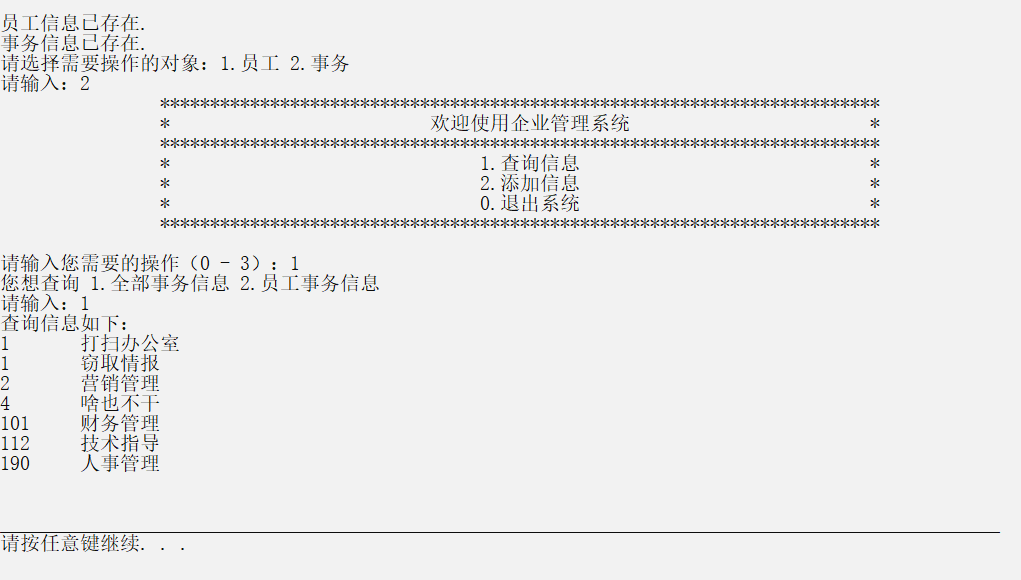


5.7.3员工不存在

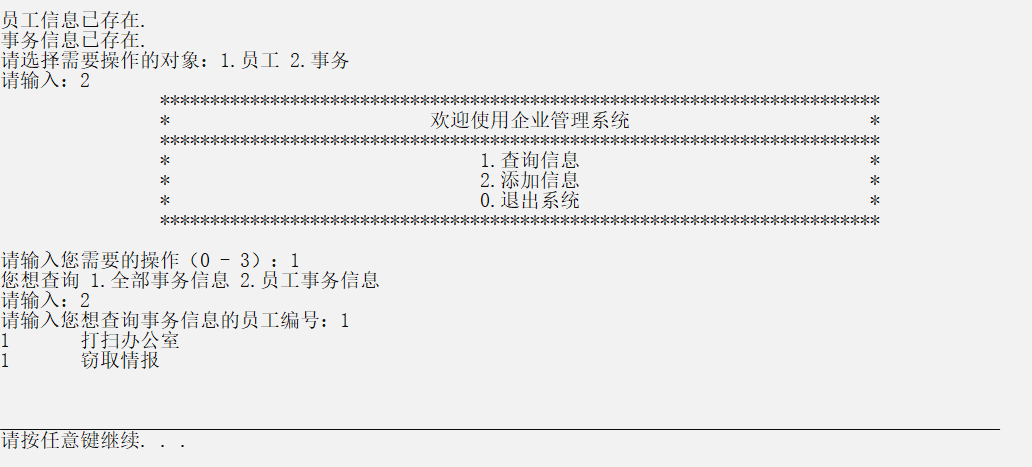


# 5.8 查询事务信息

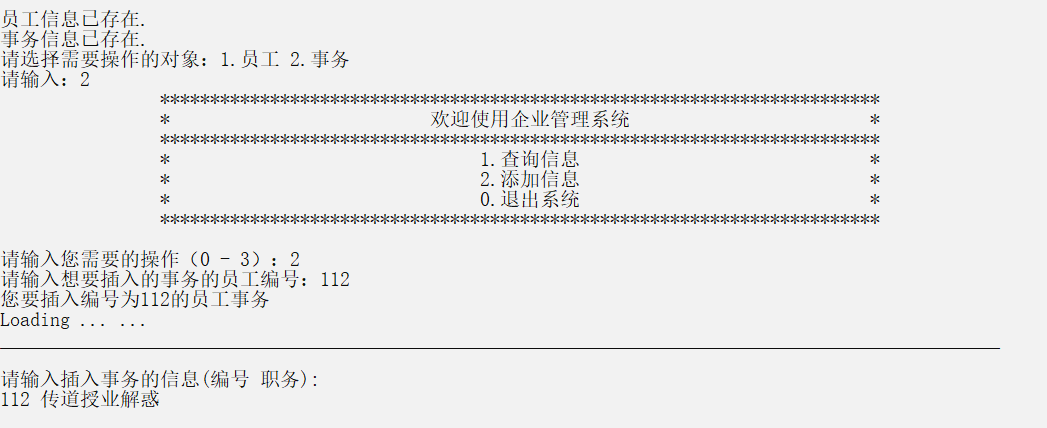
5.8.1 查询全部



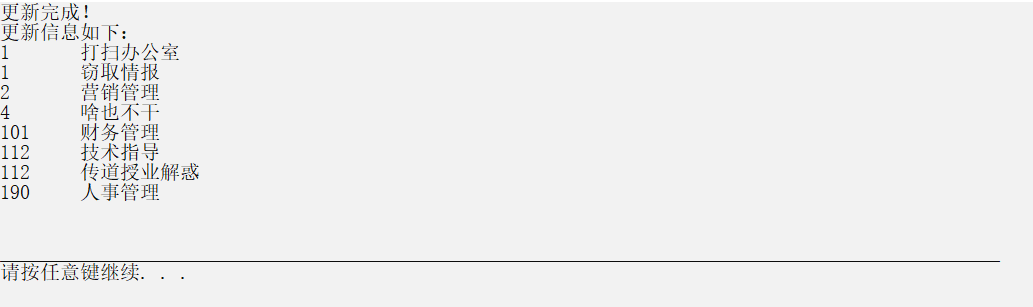
5.8.2 查询个人



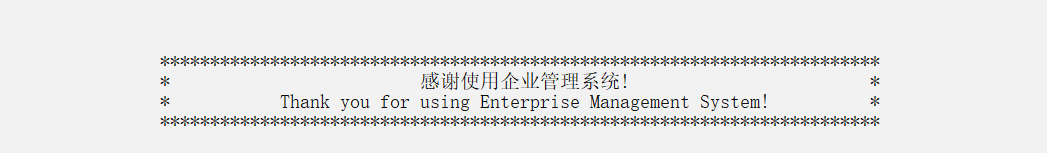
5.8.3添加事务



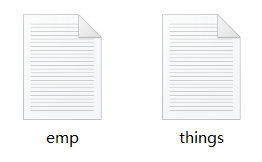
5.8.4 添加完成



# 5.9 退出系统



# 6.0 保存的文件





1. 总 结

首先，对于程序优化角度，由于题目要求设置，对于“事务”的删除只能依附于“员工删除”，而不能在单独的事务菜单删除，之后的优化中，可将“事务”也加上序号，由此能在事务菜单单独删除。

其次，对于程序问题方面如下：

1. save函数，也就是对于文件的写入，是否每次都是从文件开头开始输入？

2.写入记事本的文字能否换行？

A:写入时就对格式加以调整

1. 是否用fwrite就只能给结构体的每个成员都定义为char类型？

A: 运用iostream流文件成功。

1. 对文件的运用仍需加强

最后，就个人编写系统角度而言，从最开始的对链表了解不多到最后的熟练运用，这个系统设计不仅培养了我们的代码能力，更是培养了我们的自主学习能力。

链表不会，怎么办？文件不会，怎么办？最开始是焦虑且畏惧的，但ddl的催促让我不得不从工作日中抽时间学习相关内容。经过两三天的挑灯夜战，对链表、文件、结构体、指针有了更深的理解。

11.14日周六下午两点开始着手大作业，仍是先通过视频再次了解链表，通过测试链表可行性逐步开始编写，构建功能子函数思路，再逐个子函数分开调试，写一个子函数进行一次debug，每个子函数中遇到的问题即刻解决：或是通过视频，或是通过网上资料，或是通过书本。看着功能一个一个的丰富，心情也放松了些。同时，我认为这个系统编写也培养了我的时间管理能力，坐在电脑前太久可能会导致思维停滞，因此执行了45分钟coding，15分钟休息的模式，让自己在coding时间内保证效率，一个小时接着一个小时过去了，0点过后的时间尤为迅速，虽然知道夜已深，但看着自己的系统逐步完善，逐步赏心悦目，我就停不下敲打键盘的手，即使过程中遇到了数次因为看不出哪里出错的崩溃，但这个时候，站起来，走一走，看看远方，再回来时，总能解决，有时候，困难并不可怕，可怕的是自己觉得它可怕。11.15周日7：02分完成了完整版的系统demo，满足的合上电脑躺下了。即使还存在很多的不完善，但这一夜，我战胜了自己！

虽说大学前也有过几次通宵做项目的经历，但这貌似是几个项目、比赛中最成功的一个，因此，虽然和同学们说着那天多累多累，但心里总是喜悦的。这次的系统编写是突破自我的一课，作为上海大学计算机学院的学生，有顽强的意志和不断创新的意识是尤为重要的，少年强则国强，只有我们的强大，未来的祖国才能强大。

感谢张博锋老师给我们这样一次独立思考的机会，相信这一课将会是我人生中难以忘怀且影响颇深的一课！