

大类基础课程大型实验

**2014/2015(2)**



实验题目 教职工信息管理系统

学生姓名 沈小双

学生学号 201426810718

学生班级 1407

任课教师 陈志扬

提交日期 未定

**计算机科学与技术学院**

**教职工信息管理系统 实验报告**

1. **大型实验的内容**

教职工信息管理系统用于管理教职工信息. 能够根据工号、姓名、科室精确查询职工信息.能分系部进行职称统计，计算各职称的人数,根据职工的职称排序输出

根据工号修改或删除职工信息.

1. **运行环境**

教职工信息管理系统在 visual studio 2015 平台下开发,操作系统:windows 8.1

硬件环境:

处理器：Intel(R) Core(TM) i5-4200U CPU @ 1.60GHz 2.30GHz

内存:4.00GB

系统类型：64位操作系统

1. **实验课题分析（主要的模块功能、流程图）**

**3.1教职工管理系统主要功能**

设计菜单实现功能选择；

输入功能：输入职工信息，并保存到文件中；

3、查询功能：

1）能够根据工号精确查询职工信息；

2）能够根据姓名、科室查询职工信息

3）分系部进行职称统计，计算各职称的人数

4、根据职工的职称排序输出

5、根据工号修改职工信息

6、根据工号删除职工信息

**3.2程序菜单项**

**四、实验主要模块的功能、流程**

**(1)类的编写**

教职工管理系统 包括两个类 employee类与empNode类.

Employee包含教职工的所有基本信息,以及更改和查看基本信息的所有方法

empNode类为链表,节点值是一个employee对象,以及指针指向下一个节点

data.txt 用于储存教职工所有基本信息

开始

显示菜单

输入选择

否 否

-1?

合理?

是 是

执行所选项

查询

输入查询的值

遍历链表

是

是否找到

否

对不起没找到

显示结果

删除

输入工号

否

找到?

显示没找到并退出

是

显示结果

是否删除

**否**

返回上级

**是**

删除

**·**employee

class employee

{

public:

employee();

//默认构造函数

employee(string na,bool sex,long long jobNu,long long phoneNu,string depart,string title);

~employee();

//显示教职工信息

void display();

//返回工号

long long getJobNumber();

//返回科室

string getDepartment();

//返回姓名

string getName();

//返回职称

string getTitle();

//返回性别

bool getSex();

//返回电话号码

long long getPhoneNumber();

//返回职称等级

int getlevelOfTitle();

//用于修改姓名

void setName(const string a);

//修改性别

void setSex(const bool a);

//修改工号

void setJobNumber(const long long a);

//修改电话号码

void setPhoneNumber(const long long a);

//修改所属部门

void setDepartment(const string a);

//修改职称

void setTitle(const string a);

//输入所有信息

void setAllFromeCin();

private:

string name;//姓名

bool sex;//性别 1---男 0----女

long long jobNumber;//工号

long long phoneNumber;//电话号码

string department;//所在部门科室

string title;//职称

int levelOftitle;//职称等级

};

(2)链表的使用

链表用于程序运行中储存每一个教职工的信息.考虑到信息量比较大所以使用链表.对于增删比较容易

·empNode

class empNode

{

public:

empNode();

empNode(const employee &item, empNode \*nextNode = NULL) :emp(item), next(nextNode) {}

//find\*\*\*按照相应的变量查找

void findName(string name);

void findJobNumber(long long jobNumber);

void findDepartment(string department);

void findTitle(string title);

//统计概览 各个职称人数与各个部门人数

void overView();

//将链表中的值写入文件

void write();

//修改某个工号的个人信息

void change(long long);

//删除某个工号

void erease(long long);

//按照职称排序输出

void paixushuchu();

~empNode();

employee emp;//节点值

empNode \*next;

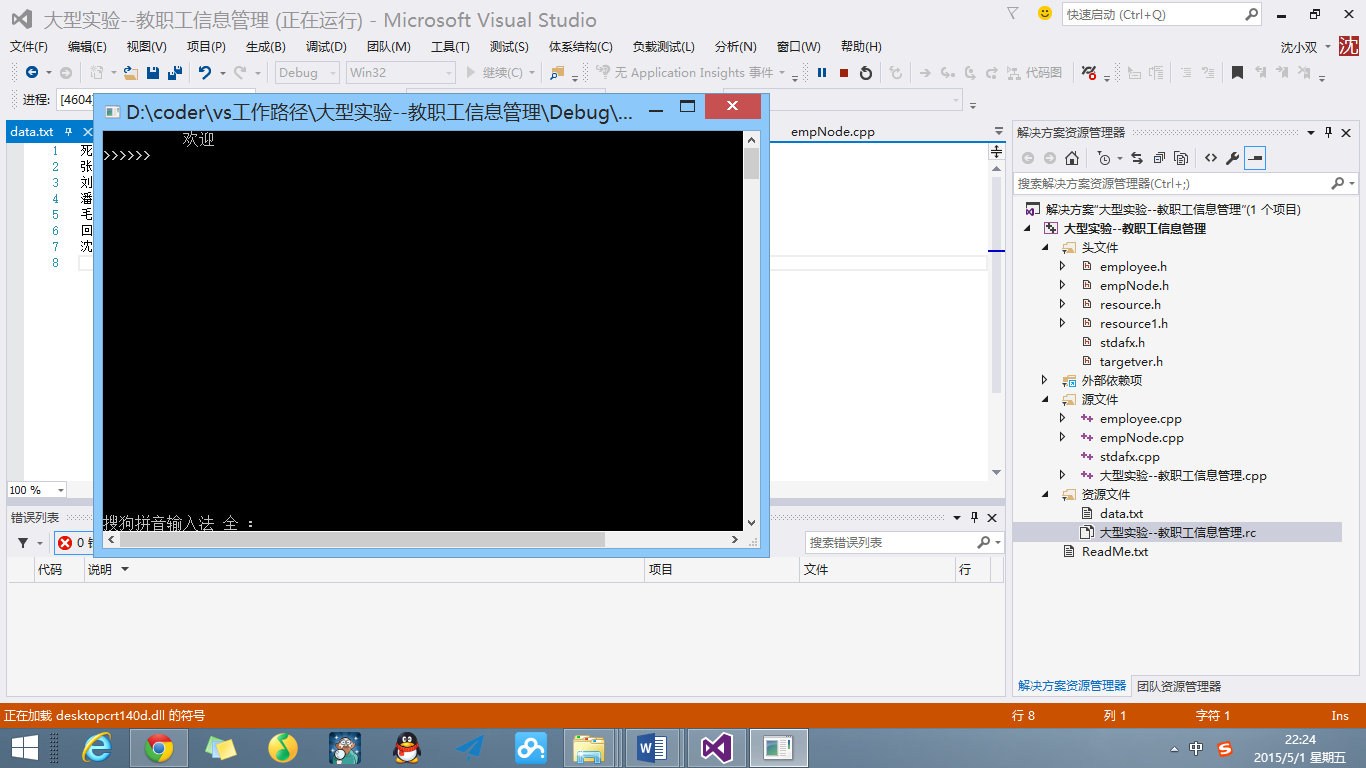
private:

};

**(3)程序流程**

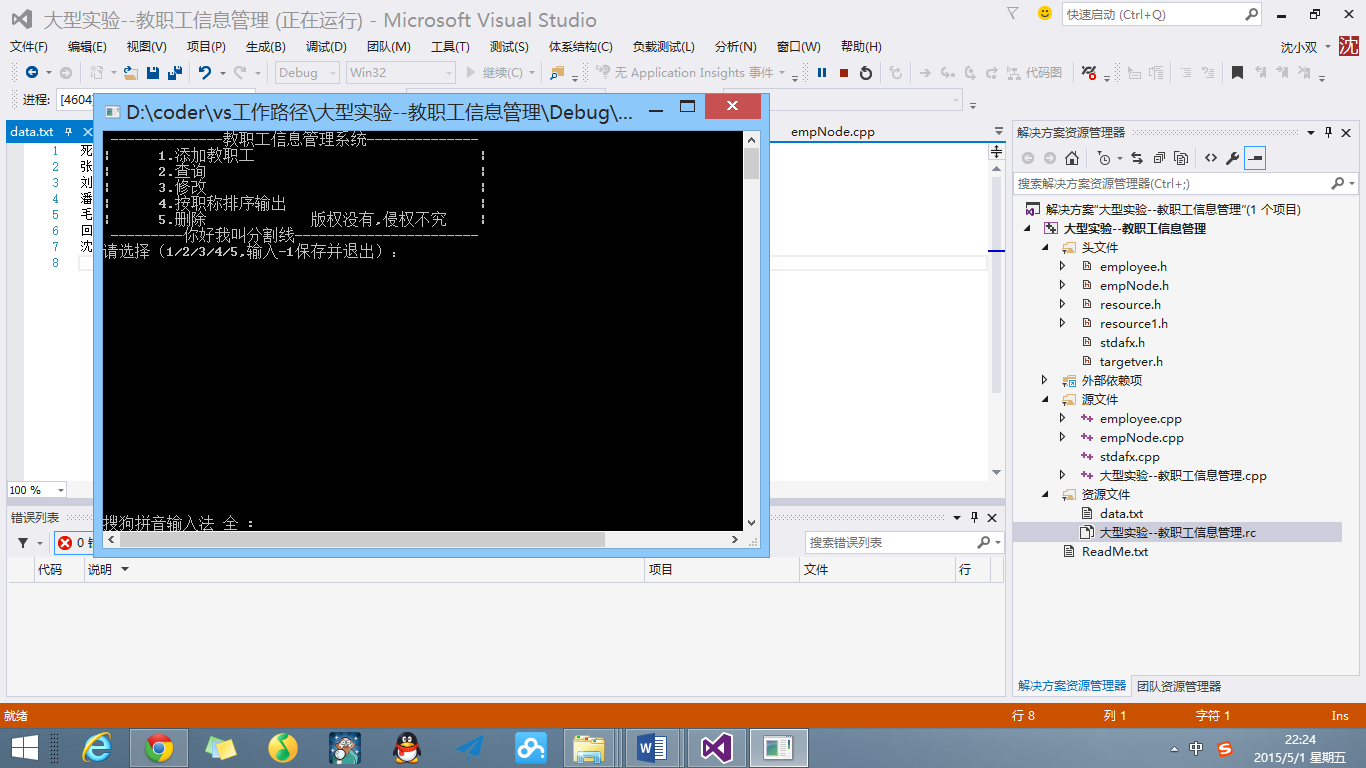
**1**.读入文件(将data.txt文件中的数据读入链表)

2.开始小过场(简单的文字效果)



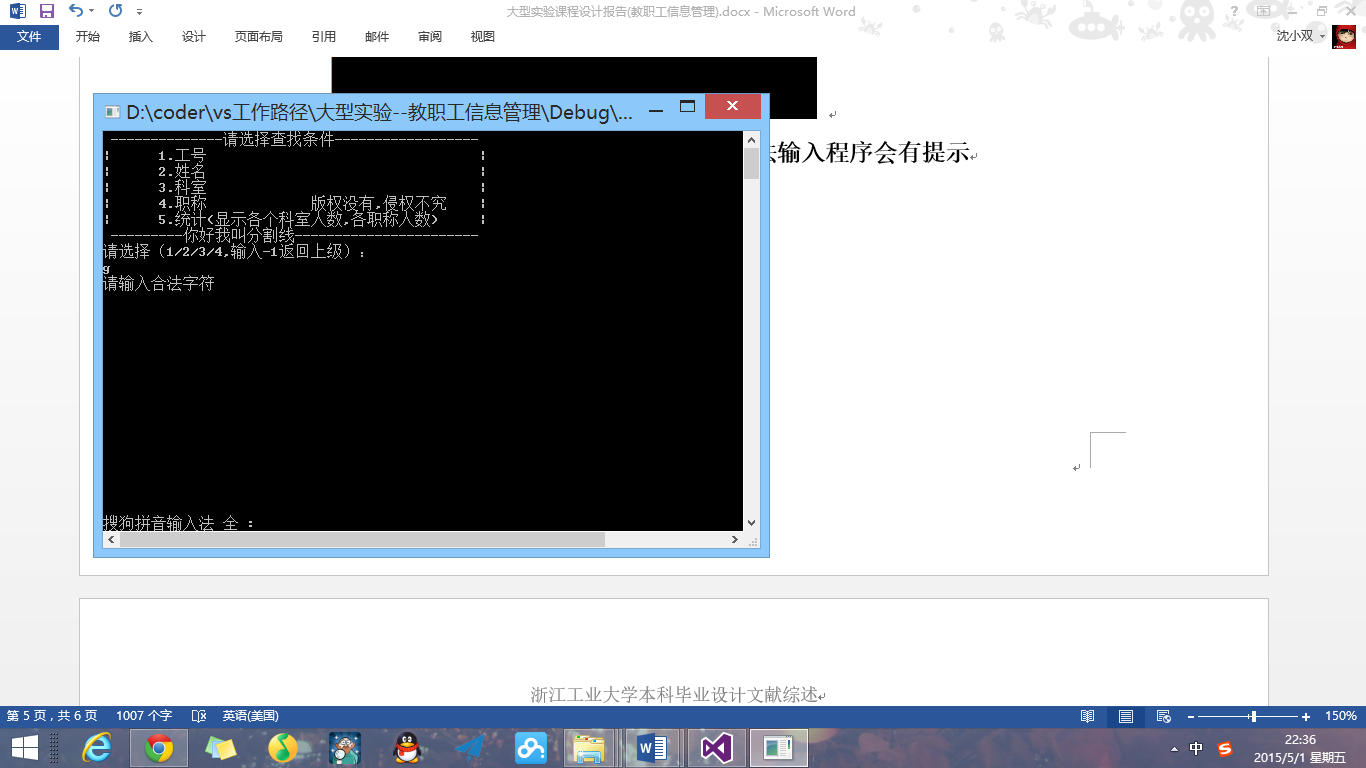
3.显示菜单(signal1作为一级菜单的选择信号)

(1)

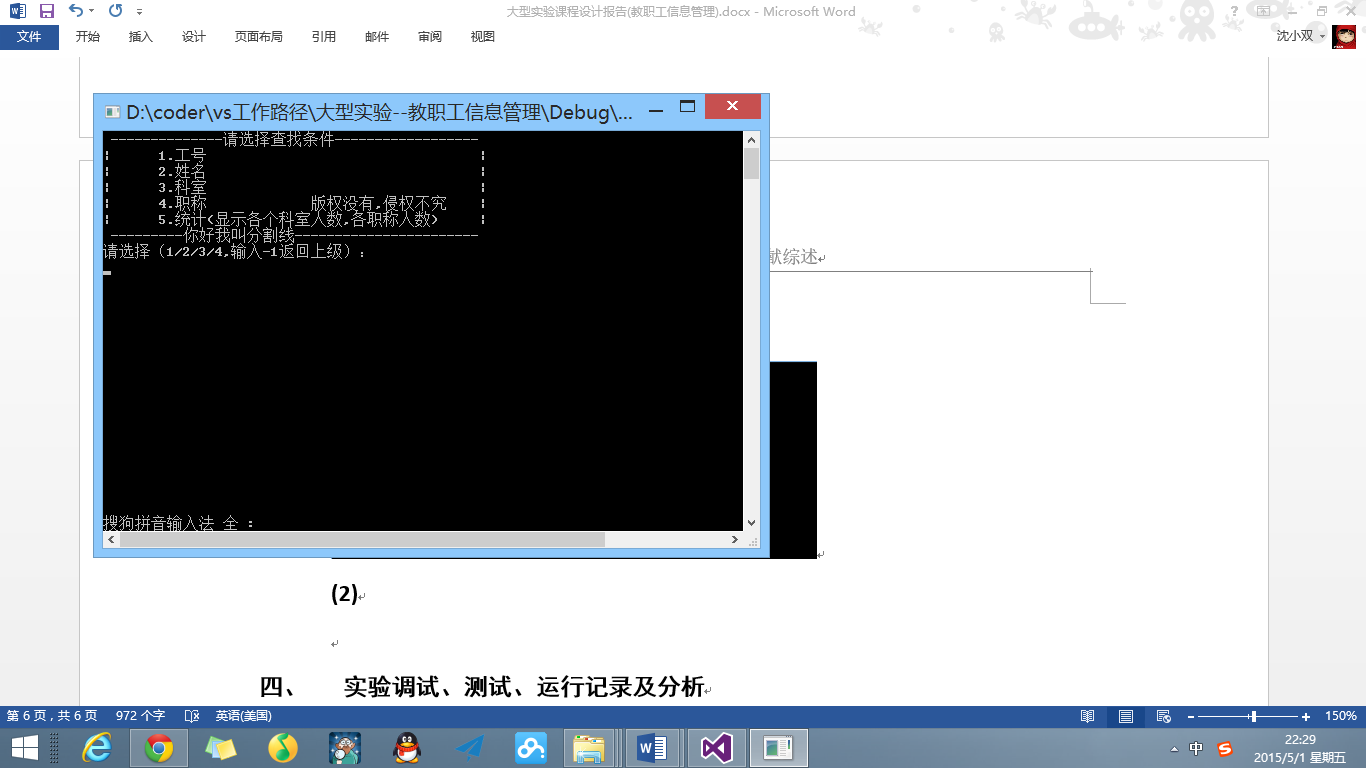
****

可以从一级菜单选择相应个功能,若为非法输入程序会有提示

例如:



(2)二级查询菜单



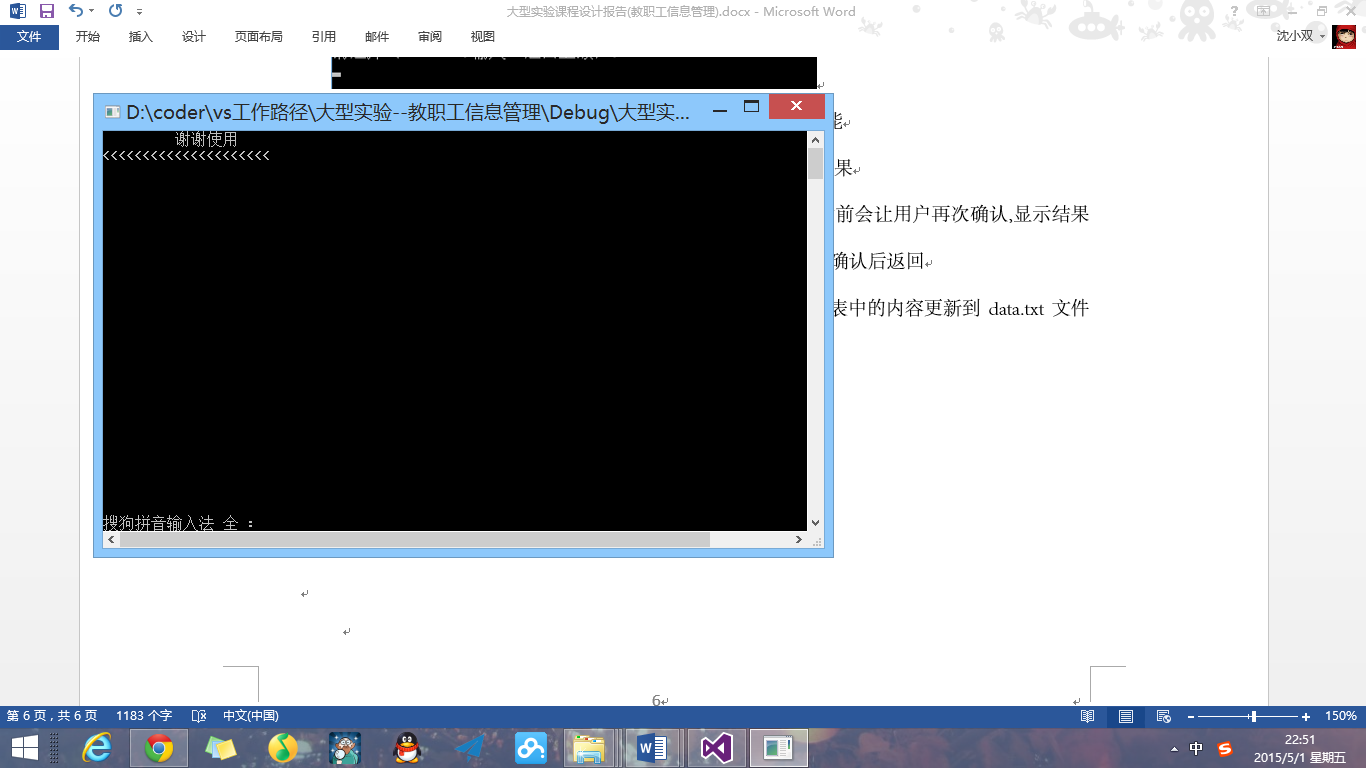
接下来的流程取决于用户的选择,用户可以自由选择各项功能

程序从用户得到输入,查询或对链表进行增删改,再显示处理结果

所有的输入都进行过滤防止非法字符导致的程序崩溃,在删除前会让用户再次确认,显示结果有些会在显示一到两秒内自动跳转到上级菜单,有些会让用户确认后返回

直到用户在一级菜单输入-1退出一级菜单,退出菜单后将链表中的内容更新到data.txt文件中

结束的文字效果



**五. 实验调试、测试、运行记录及分析,程序的key point;**

while (!(cin >> signal2))

{

cout << "请输入合法字符" << endl;

cin.clear();

cin.ignore();

}

用于过滤非法输入

菜单:提供与用户互交的接口,读取用户需求并调用链表的方法实现这些需求

链表:具体的数据结构,并提供具体的实现方法(是这个程序设计的难点与重点)

Employee:结构化的储存程序所处理的对象.提供写入及查看基本参数的方法

Data.txt:以文本形式储存数据

string TITLE[4] = { "教授","副教授","讲师","助教" };//职称列表

在程序中我将职称这分为以上四项.因为教职工的职称是相对固定的.不会无缘无故多出副讲师之类的职称.而且这么做易于进行排序.在employee类的成员变量中有一项是levelOfTitle就是通过对比字符串表得到的,在进行排序的时候就可以简单的依据levelOfTitle排序.

**六.实验总结（优点、不足、收获及体会）**

优点:改进了用户体验(有开场的类似动画的文字效果),与程序的稳定性(解决非法输入问题)

缺点:对于大量的长短不一样的文本显示效果没有对齐,看上去比较凌乱.

收获:实践是检验真理的唯一途径,实际操作中会遇到很多课本上没有提到的问题,这时候要学会自己解决或者学会使用Baidu与Google. 通过这次C++的大型实验，我深刻的明白到：课本知识与实践能力相结合的重要性。会读程序的人并不一定会编程序。要想把一门专业课程学好，必须增强自己的动手实践能力，而不是一天到晚只知道看书，那种行为只不过是“纸上谈兵”。看再多的书都不如自己亲手试一试。俗话说的好：会打仗的士兵才是好士兵。再者，课本上的知识不一定是完全准确的，只有自己动手进行试验过才知道。实践是检验真理的唯一标准，这话不假。