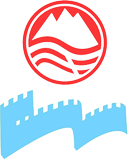
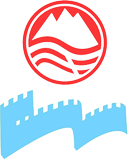
**东北大学秦皇岛分校**



**计算机与通信工程学院**

**C++课程设计报告**

**题目：小型公司工资管理系统系统**

|  |  |
| --- | --- |
| 专业名称 | 计算机类 |
| 学号 | 20168325 |
| 班级序号 | 160314 |
| 学生姓名 | 刘永辉 |
| 指导教师 | 曲荣欣 |
| 设计时间 | 2017.1.3-2017.1.13 |

**东北大学秦皇岛分校**

【课程设计要求】

题目：小型公司工资管理系统设计

**1、问题描述**

**某公司需要存储雇员的编号、姓名、性别、所在部门，级别，并进行工资的计算。其中，雇员分为经理、技术人员、销售人员和销售经理。四类人员的月薪计算方法如下：经理拿固定月薪；技术人员按小时领取月薪；销售人员按其当月销售额的提成领取工资；销售经理既拿固定月薪也领取销售提成。**

**设计一程序能够对公司人员进行管理，应用到继承、抽象类、虚函数、虚基类、多态和文件的输入/输出等内容。**

**2、功能要求**

**（1）添加功能：程序能够任意添加上述四类人员的记录，可提供选择界面供用户选择所要添加的 人员类别，要求员工的编号要唯一，如果添加了重复编号的记录时，则提示数据添加重复并取消添加。**

**（2）查询功能：可根据编号、姓名等信息对已添加的记录进行查询，如果未找到，给出相应的提 示信息，如果找到，则显示相应的记录信息；**

**（3）显示功能：可显示当前系统中所有记录，每条记录占据一行。**

**（4）编辑功能：可根据查询结果对相应的记录进行修改，修改时注意编号的唯一性。**

**（5）删除功能：主要实现对已添加的人员记录进行删除。如果当前系统中没有相应的人员记录， 则提示“记录为空！”并返回操作；否则，输入要删除的人员的编号或姓名，根据所输入的信息删除该人员记录，如果没有找到该人员信息，则提示相应的记录不存。**

**（6）统计功能：能根据多种参数进行人员的统计。例如，统计四类人员数量以及总数， 或者统计男、女员工的数量，或者统计平均工资、最高工资、最低工资等信息。**

**（7）保存功能：可将当前系统中各类人员记录存入文件中，存入方式任意。**

**（8）读取功能：可将保存在文件中的人员信息读入到当前系统中，供用户进行使用。**

**3、问题的解决方案**

**根据系统功能要求，可以将问题解决分为以下步骤：**

**（1）应用系统分析，建立该系统的功能模块框图以及界面的组织和设计；**

**（2）分析系统中的各个实体及它们之间的关系；**

**（3）根据问题描述，设计系统的类层次；**

**（4）完成类层次中各个类的描述；**

**（5）完成类中各个成员函数的定义；**

**（6）完成系统的应用模块；**

**（7）功能调试；**

**（8）完成系统总结报告。**

【题目分析】

首先需要用到面向对象编程的知识，题目中明确可以看出需要定义职员基类，包含编号、姓名、性别、所在部门，级别的基本属性，派生出经理和销售人员以及技术人员，销售经理兼有销售人员和经理的性质，即销售经理继承于销售人员和技术人员，派生类除基本属性还有其他工资计算所需要的数据，数据的存储可以采取链表的形式，方便查找和储存，其他功能的实现都是基于链表节点的查找、修改、删除。最后的保存和读取采取文件流。

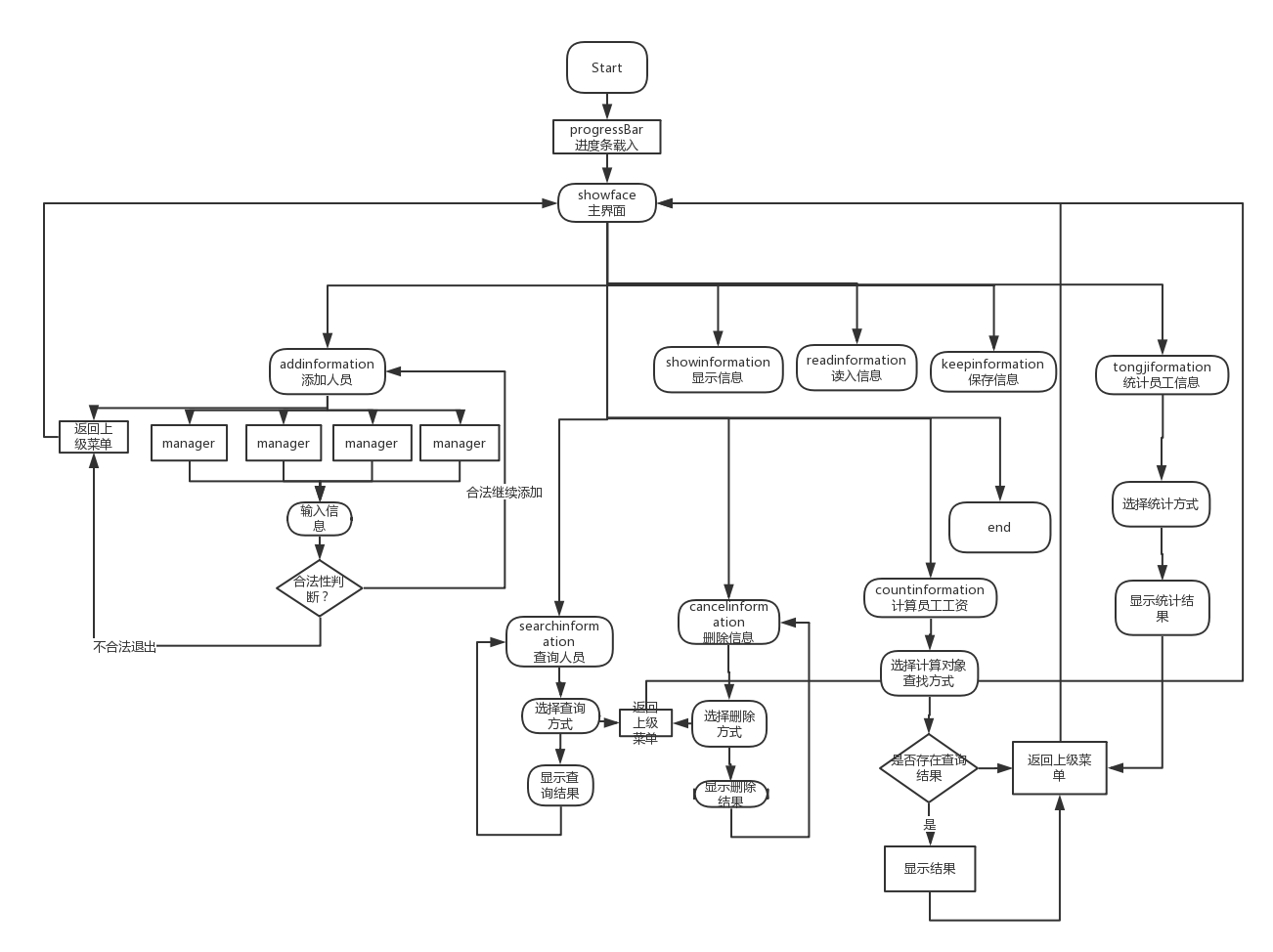


图1程序流程图

【类结构设计】

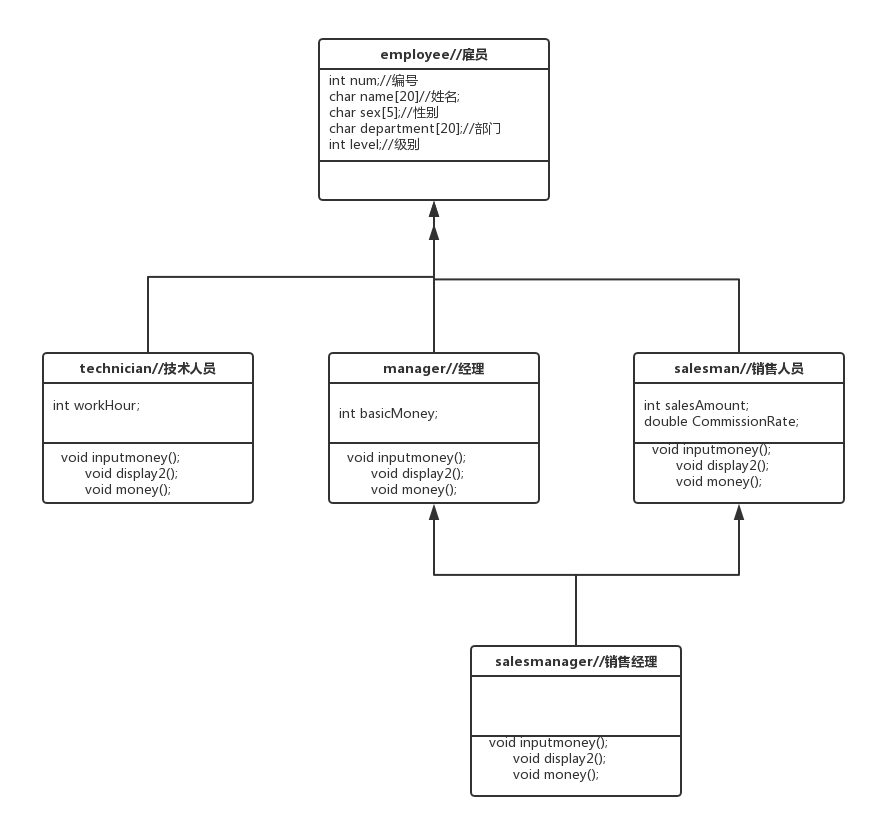


图2 类之间的继承关系

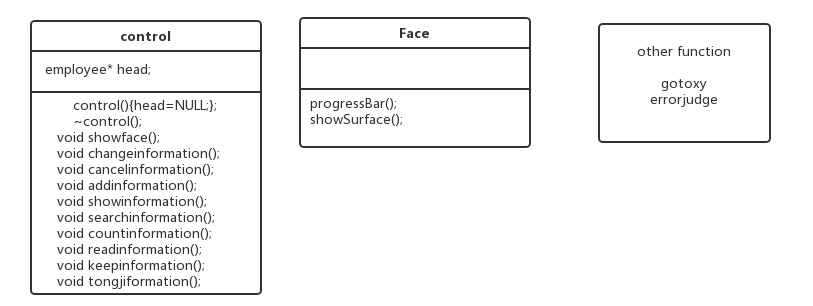


图3 类之间的继承关系

其中，employee类为基类，定义了的编号、姓名、性别、所在部门，级别提供了基类指针，基本信息输入输出的接口。

technician类采用公有继承方式继承自employee类，增加了工作时间和每小时工资数，基本工资计算，以及工作时间的输入。

manager采取虚继承方式继承自employee类，增加了基本工资数，基本工资计算，以及基本工资的输入。

salesman采取虚继承方式继承自employee类，增加了销售额数和销售额提成，基本工资计算，以及销售额的输入。

Salesmanager多重继承方式继承自manager和salesman，公有继承manager和salesman的输入，增加基本工资计算。

【工程结构组织】

本项目使用Dev-CPP实现，工程文件结构图如图2所示。

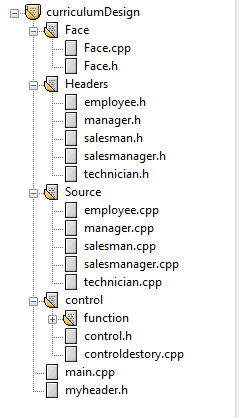
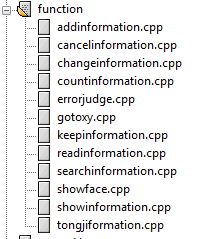


图4 工程结构图



本项目一共由28个文件组成，各文件的功能如下表（表1）所示。

表1项目文件功能列表

|  |  |  |
| --- | --- | --- |
| 序号 | 文件名 | 说明 |
| 1 | Face.h | Face类定义头文件，其他界面函数声明，界面声明 |
| 2 | Face.cpp | Face类实现文件，标题，进度条及界面实现 |
| 3 | employee.h | 职员类定义头文件 |
| 4 | employee.cpp | 职员类实现文件 |
| 5 | manager.h | 经理类定义头文件 |
| 6 | manager.cpp | 经理类实现文件 |
| 7 | salesman.h | 销售人员类定义头文件 |
| 8 | salesman.cpp | 销售人员实现文件 |
| 9 | technician.h | 技术人员类定义头文件 |
| 10 | technician.cpp | 技术人员类实现文件 |
| 11 | salesmanager.h | 销售经理类定义头文件 |
| 12 | salesmanager.cpp | 销售经理类实现文件 |
| 13 | control.h | 控制类定义头文件 |
| 14 | controldestory.cpp | 控制类析构函数实现文件 |
| 15 | addinformation.cpp | 控制类添加信息函数实现文件 |
| 16 | cancelinformation.cpp | 控制类删除信息函数实现文件 |
| 17 | changeinformation.cpp | 控制类修改信息函数实现文件 |
| 18 | countinformation.cpp | 控制类统计信息函数实现文件 |
| 19 | errorjudge.cpp | 输入整数时其他字符处理函数实现文件 |
| 20 | gotoxy.cpp | 光标跳转到xy函数实现文件 |
| 21 | keepinformation.cpp | 控制类保存信息函数实现文件 |
| 22 | readinformation.cpp | 控制类读取信息函数实现文件 |
| 23 | searchinformation.cpp | 控制类查询信息函数实现文件 |
| 24 | showface.cpp | 控制类界面函数实现文件 |
| 25 | showinformation.cpp | 控制类显示信息函数实现文件 |
| 26 | tongjiformation.cpp | 控制类统计信息函数实现文件 |
| 27 | main.cpp | 主函数文件 |
| 28 | myheader.h | 头文件文件 |

【核心代码解释】

1、控制类提供所有接口//control.h

#ifndef CONTROL\_H

#define CONTROL\_H

#include "myheader.h"

class control

{

public:

control(){

head=NULL;};

~control();

void showface();

void changeinformation();//修改

void cancelinformation();//删除

void addinformation();//添加信

void showinformation();// 显示

void searchinformation();//查找

void countinformation();//计算

void readinformation();// 读取

void keepinformation();//保存

void tongjiformation();//统计

private:

employee\* head;//定义头指针

};

2、控制析构函数实现最后链表删除

#include "control.h"

control::~control()

{

employee\* p=NULL;

p=head;

while (p!=NULL)

{

p=p->next;

delete head;

head=p;

}

head=NULL;

}

#endif

3、#include "control.h"//主界面函数

void control::showface() //

{

control control1;

Face face1;

int i;

do

{

system("cls");

face1.progressBar();

int loacatx=25,loacat=5;

/\*string temp\_str;

temp\_str="===================================================================";

face1.showSurface(loacatx,loacat++,temp\_str);\*/

gotoxy(loacatx,loacat++);cout<<"==================================================================="<<endl;

gotoxy(loacatx,loacat++);cout<<"| |"<<endl;

gotoxy(loacatx,loacat++);cout<<"| 欢迎使用本职工信息管理系统 |"<<endl;

gotoxy(loacatx,loacat++);cout<<"| |"<<endl;

gotoxy(loacatx,loacat++);cout<<"| 1、添加职工信息 2、查询职工信息 |"<<endl;

gotoxy(loacatx,loacat++);cout<<"| 3、删除职工信息 4、修改职工信息 |"<<endl;

gotoxy(loacatx,loacat++);cout<<"| 5、显示所有职工信息 6、计算员工工资 |"<<endl;

gotoxy(loacatx,loacat++);cout<<"| 7、读入员工信息 8、保存员工信息 |"<<endl;

gotoxy(loacatx,loacat++);cout<<"| 9、统计员工信息 0、退出系统 |"<<endl;

gotoxy(loacatx,loacat++);cout<<"| |"<<endl;

gotoxy(loacatx,loacat++);cout<<"| 请选择相应编号: |"<<endl;

gotoxy(loacatx,loacat++);cout<<"==================================================================="<<endl;

while(1)

{

char temp[100];

gotoxy(66,15);

cin>>temp;

i=temp[0]-48;

if(i>9||i<0)

{

gotoxy(50,18);

cout<<"输入错误请重新输入"<<endl;

Sleep(2000);

gotoxy(50,18);

cout<<" "<<endl;

}

else break;

}

switch(i)

{

case 1:

control1.addinformation();

break;

case 2:

control1.searchinformation();

break;

case 3:

control1.cancelinformation();

break;

case 4:

control1.changeinformation();

break;

case 5:

control1.showinformation();

break;

case 6:

control1.countinformation();

break;

case 7:

control1.readinformation();

break;

case 8:

control1.keepinformation();

break;

case 9:

control1.tongjiformation();

break;

case 0:

gotoxy(5,17);

break;

}

}

while(i!=0);

}

4、输入错误处理函数

#include<iostream>

using namespace std;

#include <windows.h>

void gotoxy(int x,int y);

int first=0;

extern int cinErrorJudge()

{

int temp;

if(first)

cout<<"\t\t\t";

//else

//cout<<"\t";

cin>>temp;

try{

if(cin.fail())

throw 1;

}

catch(int){

cin.clear();

cin.sync();

cerr<<"\t\t\t======================请输入一个整数:=========================="<<endl;

first=1;

return cinErrorJudge();

}

first=0;

return temp;

}

5、文件读入函数

#include "control.h"

void control::readinformation() //读入员工信息

{

int level; //类标记

employee \*p=NULL;

ifstream infile("employee1.txt",ios::in);//读取一条记录

gotoxy(25,18);

if(!infile)

{

cerr<<"Open error!"<<endl;

abort();

}

else

{

do

{

infile>>level;

switch(level)

{

case 1:

{

p=new manager(1);

p->level=1;

infile>>p->num>>p->name>>p->sex>>p->department>>p->job>>p->basicMoney;

}/\*>>p->jibie\*/

break;

case 2:

{

p=new salesman(2);

p->level=2;

infile>>p->num>>p->name>>p->sex>>p->department>>p->job>>p->salesAmount;

}

break;

case 3:

{

p=new technician(3);

p->level=3;

infile>>p->num>>p->name>>p->sex>>p->department>>p->job>>p->workHour;

}

break;

case 4:

{

p=new salesmanager(4);

p->level=4;

infile>>p->num>>p->name>>p->sex>>p->department>>p->job>>p->basicMoney>>p->salesAmount;

}/\*p->jibie>>\*/

break;

}

p->next=NULL;

if(level!=0)

{

if (head!=NULL) //链表已经存在

{

employee\* p1=NULL; //指针初始化

p1=head;

while (p1->next!=NULL) //查找尾结点

{

p1=p1->next;

}

p1->next=p; //连接结点

cout<<"\t\t\t信息读入成功!"<<endl;

}

else //链表不存在

{

head=p; //连接结点

cout<<"\t\t\t信息读入成功!"<<endl;

}

}

}while(level!=0);

infile.close();

cout<<"\t\t\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*信息读入成功\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

Sleep(2000);

}

}

6、文件保存函数

#include "control.h"

void control::keepinformation() //保存员工信息

{

employee\* p=head;

if(p==NULL)

{

gotoxy(38,20);

cout<<" 系统暂无信息,无法保存,请录入信息!"<<endl;

Sleep(1500);

return;

}

ofstream outfile("employee1.txt",ios::out);

if(!outfile)

{

cerr<<"Open error!"<<endl;

abort();//

Sleep(1500);

}

while(p!=NULL)

{

outfile<<p->level<<" "<<p->num<<" "<<p->name<<" "<<p->sex<<" "<<p->department<<" "<<p->job<<" ";

if(p->level==1)

{

outfile<<p->basicMoney/\*<<" "<<p->getjibie()\*/;

}

else if(p->level==2)

{

outfile<<p->/\*Get\*/salesAmount/\*()\*/;

}

else if(p->level==3)

{

outfile<<p->/\*get\*/workHour/\*()\*/;

}

else if(p->level==4)

{

outfile<<p->basicMoney<</\*" "<<p->getjibie()<<\*/" "<<p->/\*Get\*/salesAmount/\*()\*/;

}

outfile<<endl;

p=p->next;

}

outfile<<"0"<<endl;

outfile.close();

cout<<"\n\t\t\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*保存信息成功\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

Sleep(1500);

}

7、#include "control.h"

void control::searchinformation() //查询职工信息

{

employee \*p1=NULL,\*p2=NULL;

p1=head;

if(p1==NULL)

{

gotoxy(38,20);

cout<<"系统暂无信息,无法查询,请录入信息!"<<endl;

Sleep(3000);

return;

}

int i; //选择变量

do

{

system("cls");

gotoxy(25,5);

cout<<"==================================================================="<<endl;

cout<<"\t\t\t| |"<<endl;

cout<<"\t\t\t| 请选择查询方式 |"<<endl;

cout<<"\t\t\t| 1、按照编号 2、按照姓名 |"<<endl;

cout<<"\t\t\t| 3、按照部门 4、按照职务 |"<<endl;

cout<<"\t\t\t| 5、返回上级菜单 |"<<endl;

cout<<"\t\t\t| |"<<endl;

cout<<"\t\t\t| 请选择相应编号: |"<<endl;

cout<<"\t\t\t==================================================================="<<endl;

while(1)

{

char temp[100];

gotoxy(70,12);

cin>>temp;

i=temp[0]-48;

if(i>5||i==0)

{

gotoxy(25,15);

cout<<"========================输入错误请重新输入========================="<<endl;

Sleep(2000);

gotoxy(25,15);

cout<<" "<<endl;

}

else break;

}

gotoxy(30,15);

switch(i)

{

case 1:

{

int num;

cout<<"\t\t编号:";

cin>>num;

p1=head;

while(p1!=NULL)

{

if(p1->num==num)

break;

else

{

p2=p1;

p1=p1->next;

}

}

if(p1!=NULL)

{ //若找到结点

cout<<"\t\t\t职工信息为:"<<endl<<endl<<(\*p1)<<endl;

getch();

}

else

{

cout<<"\t\t\t该职工信息不存在!"<<endl;

Sleep(1500);

}

}

break;

case 2:

{

char name[20];

cout<<"姓名:";

cin>>name;

p1=head;

while (p1!=NULL)

{

if (strcmp(p1->name,name)==0) break;

else

{

p2=p1;

p1=p1->next;

}

}

if (p1!=NULL)//若找到结点

{

cout<<(\*p1)<<endl;

getch();

}

else

{

cout<<"\t\t\t信息查询失败!"<<endl;

Sleep(1500);

}

}

break;

case 3:

{

char department[20];

int m=0;

cout<<"部门:";

cin>>department;

p1=head;

while (p1!=NULL)

{

if (strcmp(p1->department,department)==0)

{

cout<<(\*p1)<<endl;

p2=p1;

p1=p1->next;

m++;

}

else

{

p2=p1;

p1=p1->next;

}

}

if (m==0)

{

cout<<"\t\t\t信息查询失败!"<<endl;

Sleep(1500);

}

getch();

}

break;

case 4:

{

int job;

int n=0;

cout<<"\t\t\t职务:";

cin>>job;

p1=head;

while (p1!=NULL)

{

if (p1->job==job)

{

cout<<(\*p1)<<endl;

p2=p1;

p1=p1->next;

n++;

}

else

{

p2=p1;

p1=p1->next;

}

}

if (n==0)

{

cout<<"\t\t\t查询失败!"<<endl;

Sleep(1500);

}

getch();

}

break;

case 5:

return;

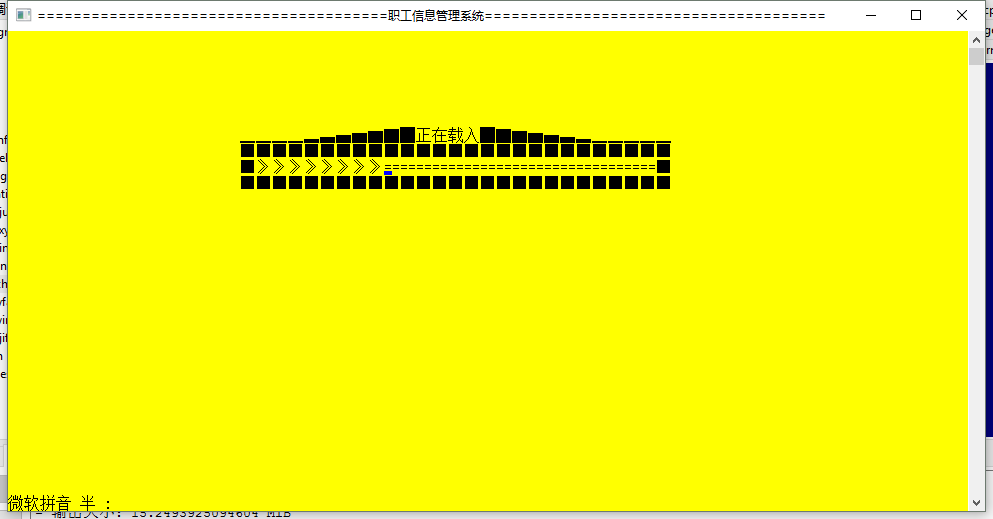
}

}

while (i!=5);

}

【运行主要界面】

图1 载入程序进度条

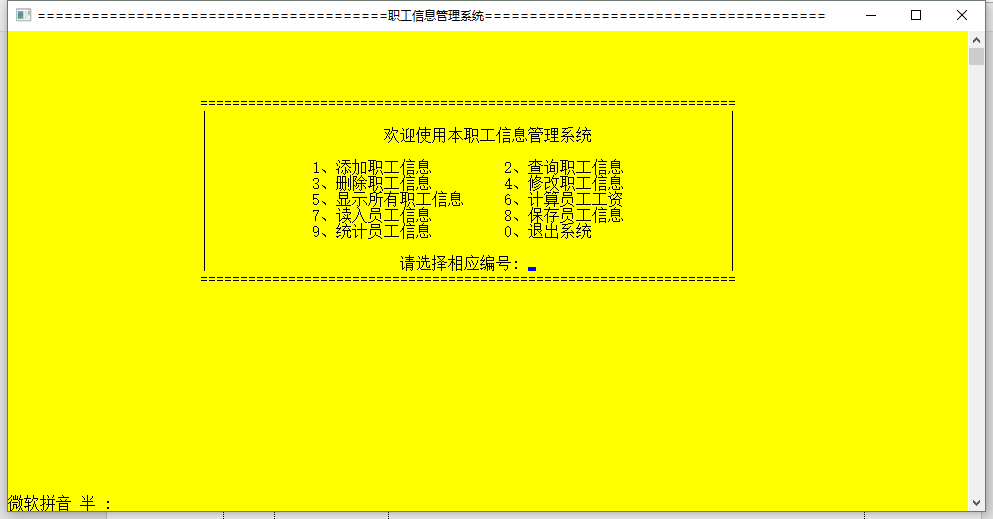


图2 程序主界面

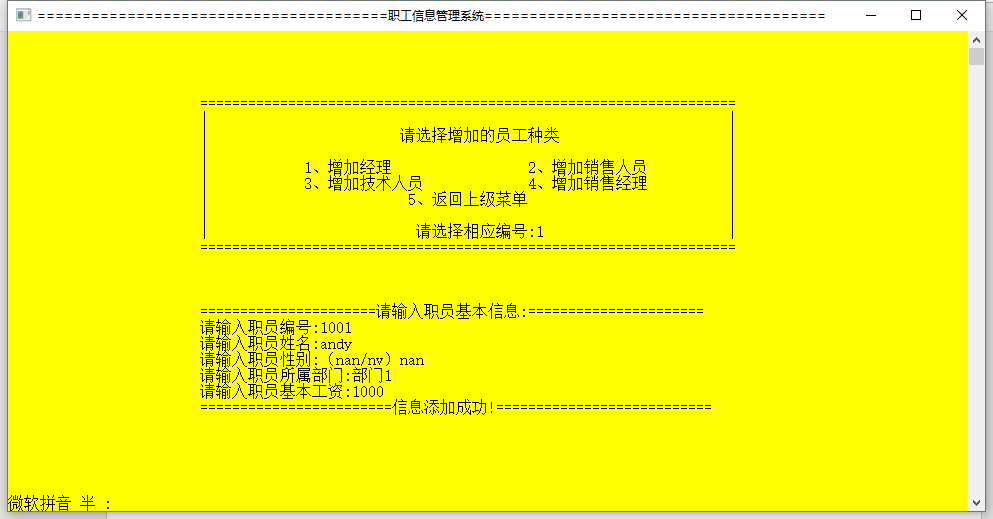


图3 添加信息界面

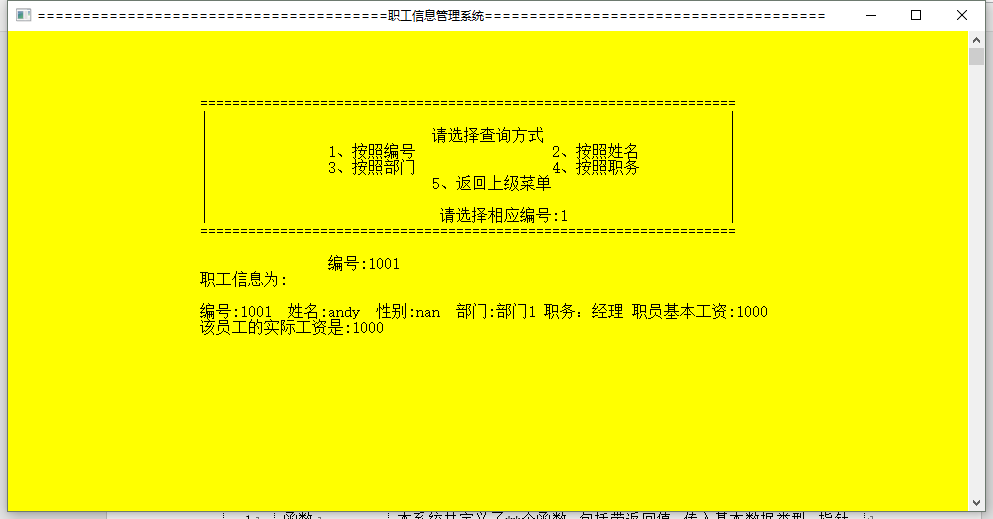


图4 查询信息界面



图5 删除信息界面

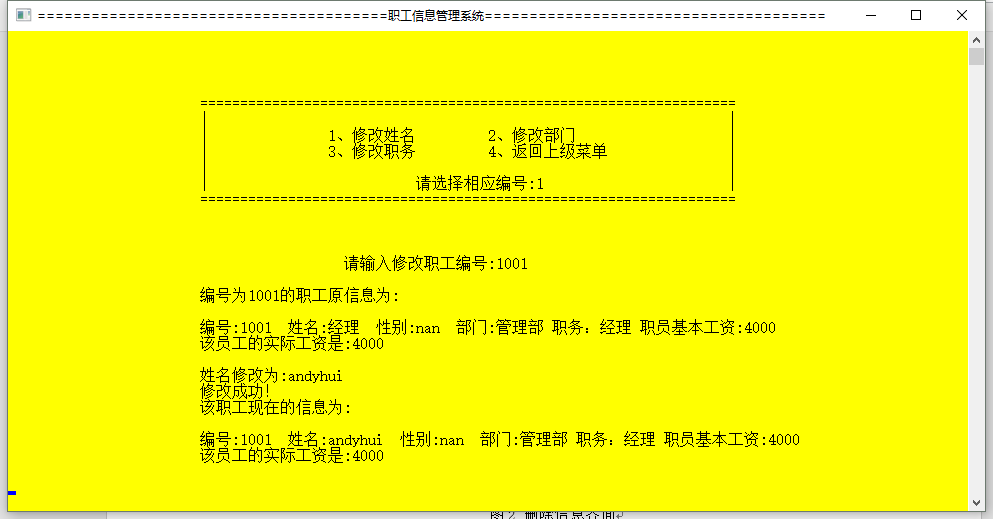


图6修改信息界面

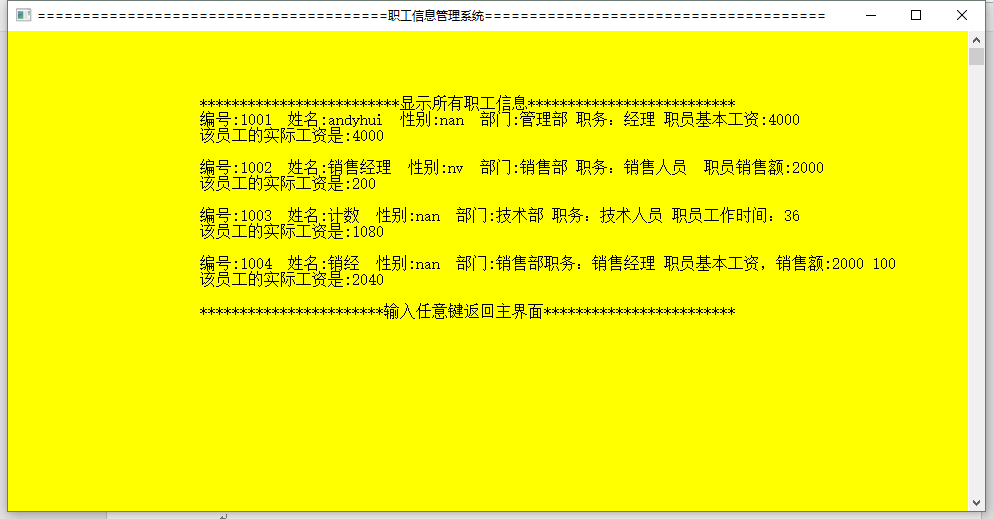


图7显示信息界面

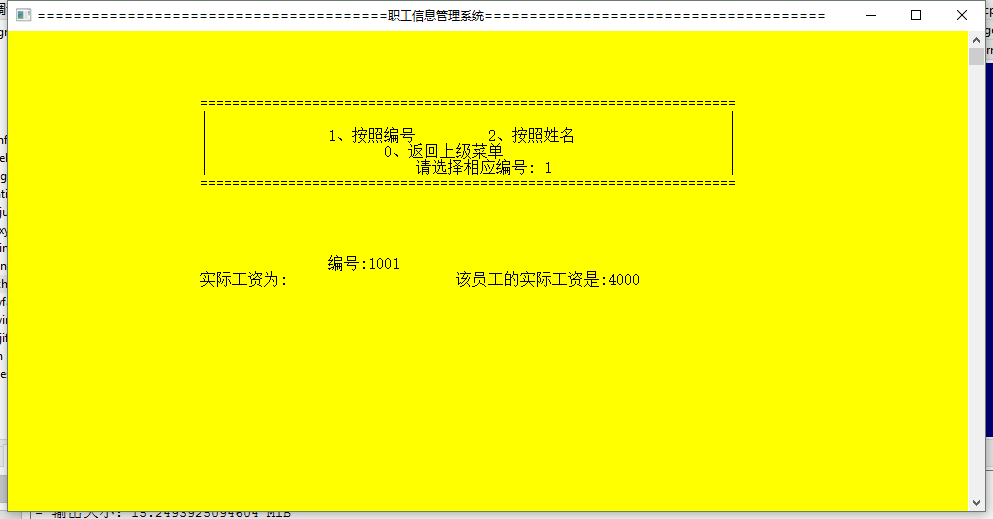


图8计算工资界面



图9统计员工信息界面

【课程设计所用知识点总结】

表2. 项目所涉及知识点总结

|  |  |  |
| --- | --- | --- |
| 序号 | 知识点 | 说明 |
| 1 | 函数 | 本系统共定义了52个函数，包括带返回值、传入基本数据类型、指针、操作符重载、虚函数纯、虚函数、函数重载。 |
| 2 | 数组 | 本项目定义了6数组，用于存储姓名、性别、部门等 |
| 3 | 文件 | 本项目使用txt格式文件做数据持久化存储 |
| 4 | 类与对象 | 定义了7个类 |
| 5 | 安全头文件包含 | 基本全部实现头文件安全包含 |
| 6 | 继承 | 使用单继承、虚继承、多重继承。 |
| 7 | 文件流 | 使用文件流输入输出 |
| 8 | 存储方式 | 使用数据结构链表实现存储 |
| 9 | 异常处理 | Try catch throw语句进行整数错误检测 |
| 10 | 操作符重载 | 重载插入提取运算符 |

【项目总结】

这次课设首先做好了程序基本分析，设计类的继承派生关系，想到用链表进行存储从最基本的显示功能开始采取瀑式开发逐步添加新的功能，直到实现所有功能。

本次课设较为满意的地方是整体界面实现以及功能的完整性，我认为是难点的是文件流的读入读出方式，以及链表的一系列功能。

本次最大的收获是对面向对象编程有了较好的理解，充分将学到的知识转化为了实体。

【参考文献】

[1]谭浩强. C++语言程序设计（第3版)[M]. 北京：清华大学出版社.2015.

[2]C++primer

【参考文献（资料）】

[1] [美]克莱茨 著，李慧军 译 VHDL数字电子学

[2] 袁静波，丁顺利，宋欣，王和兴 计算机组成与结构 机械工业出版社，2011

[3] 陈耀和.VHDL语言设计技术[M]. 北京：电子工业出版社，2004.

[4] 李云松，宋锐．Xilinx FPGA 数据基础（VHDL）版[M] ．陕西：西安电子科技大学出版社，2008.