考试报名系统

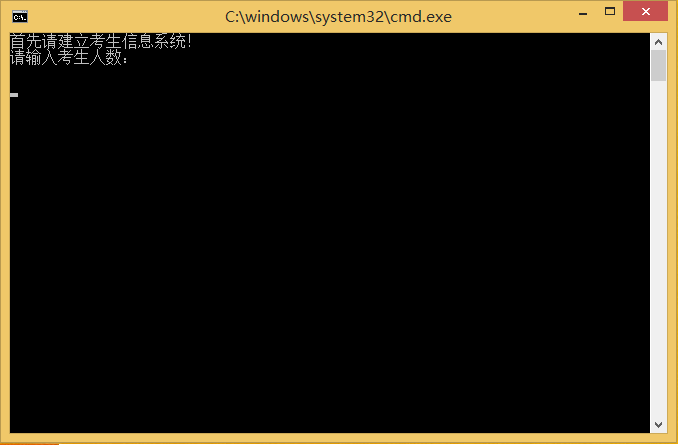
1452834 钱煜

1. 项目背景

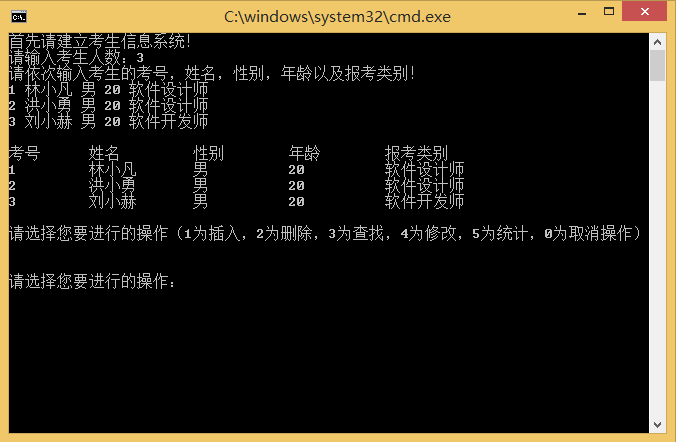
考试报名工作给各高校报名工作带来了新的挑战，给教务管理部门增加了很大的工作量。本项目是对考试报名管理的简单模拟，用控制台选项的选择方式完成下列功能：输入考生信息；输出考生信息；查询考生信息；添加考生信息；修改考生信息；删除考生信息。

1. 项目功能
2. 建立系统

运行考生信息系统程序，如下：



按照提示输入考生信息，建立考生信息系统。



1. 使用系统

按照提示输入操作编号，来执行操作。

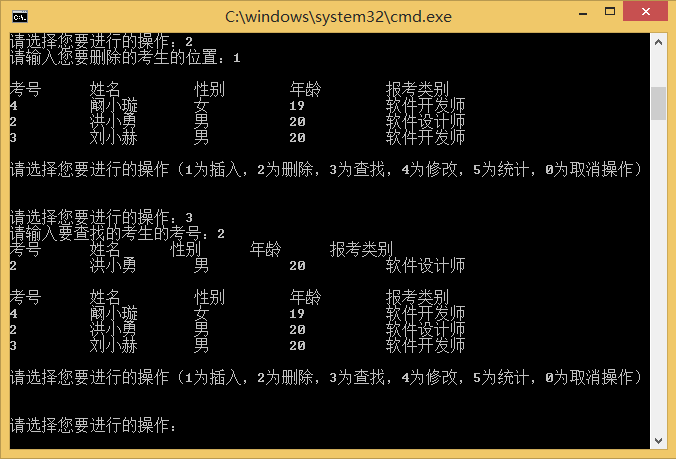
1. 插入



1. 删除



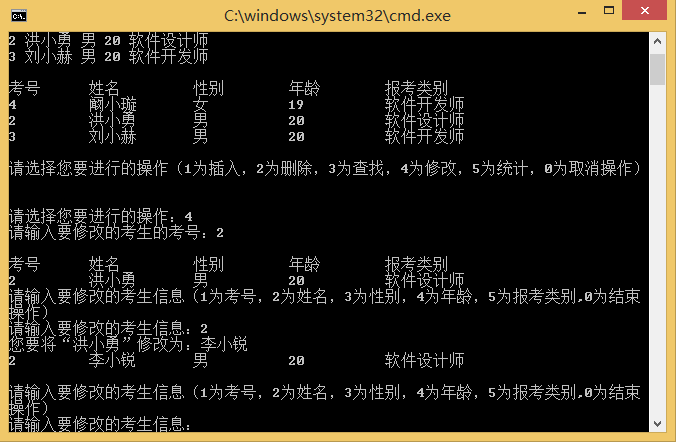
1. 查找



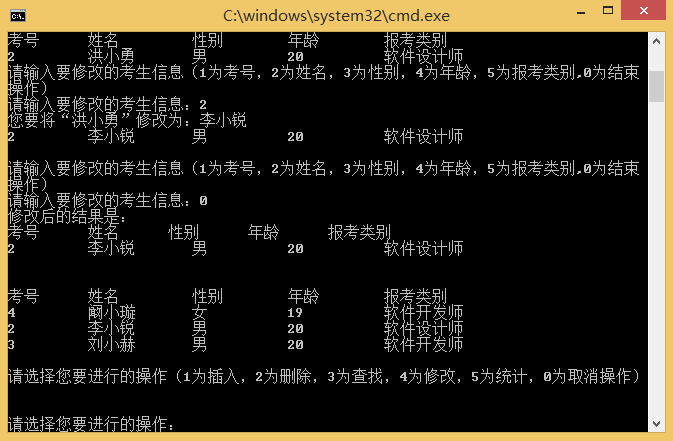
1. 修改



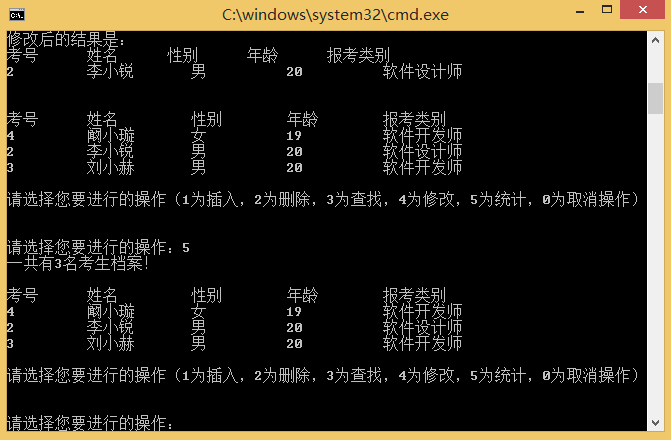
按提示输入要修改信息对应的操作编码即可进行修改。（例为修改考试姓名信息）



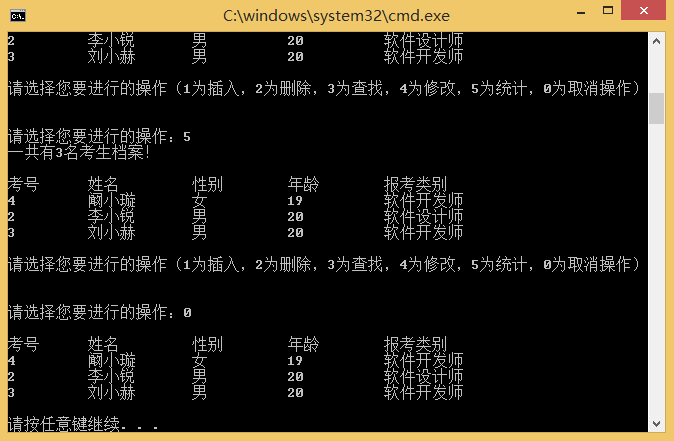
输入0结束修改时会提示显示后的信息。



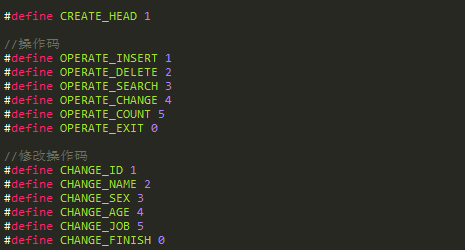
1. 统计

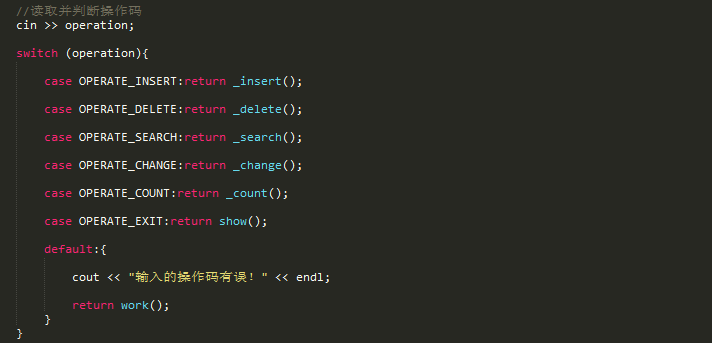


1. 退出



1. 程序优点
2. 对操作码进行宏定义，以增加程序的易读性以及便于后续维护修改





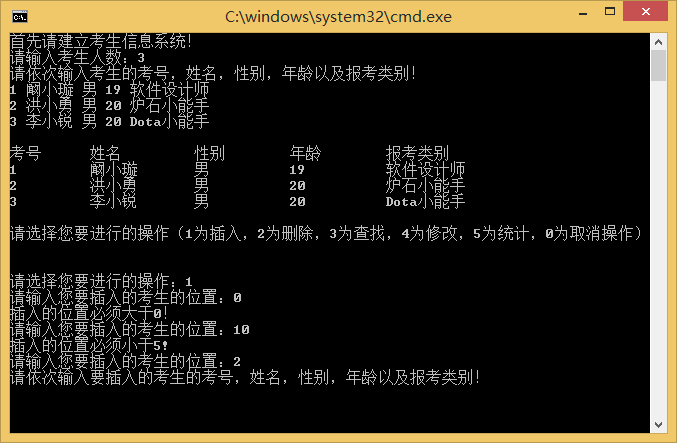
1. 采用string类型维护程序稳定

程序以string类型从输入流中读取数据，使得程序可以出色的应对各种用户错误输入，以维护程序的稳定。

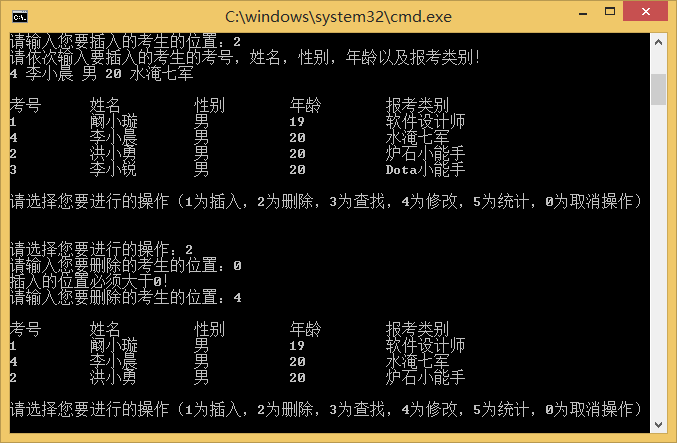
1. 错误信息检测

由于该程序采用链表实现考生信息系统的存储，因为错误输入识别尤为重要，程序对各项输入均进行了检测，在确保正确的情况下才会执行操作，以维护程序稳定。

(1)插入



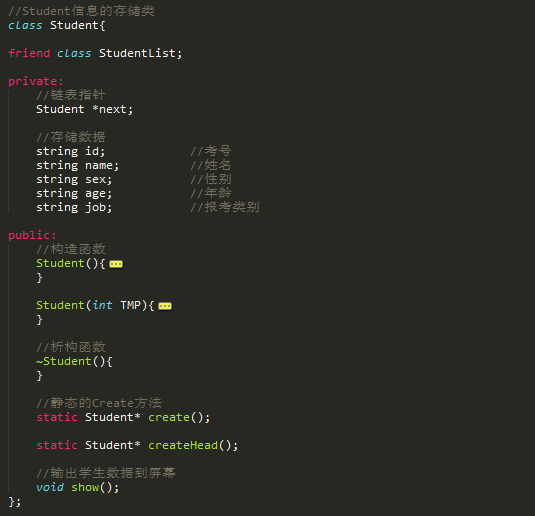
(2)删除



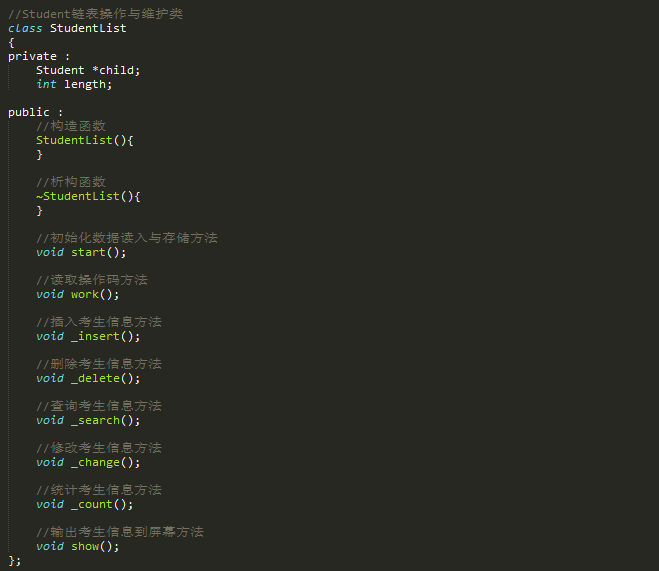
1. 程序描述

程序由两个类构成，分别为Student类与StudentList类。

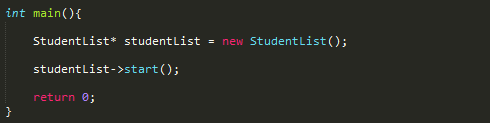
Student类：对考生信息进行存储。



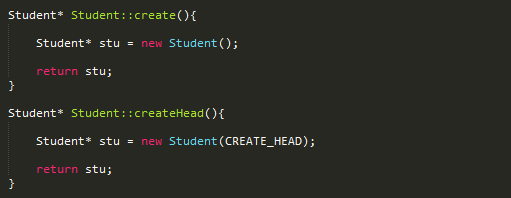
StudentList类：Student类的链表结构，并实现插入、删除、修改等操作。

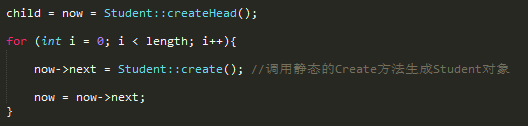


Main函数：首先建立StudentList类的一个事例，然后调用其Start方法，开始考生信息系统的建立。



在设计程序时，在Student类里提供了静态的Create函数作为接口，从而使建立Student类的实例操作从StudentList类中转移到了Student类中，使得StudentList可以调用该接口来生成Student类的实例，而并非直接调用Student类的构造器，增强了类的封装性与程序易读性。





1. 程序接口

**Class Stident:**

**Public Constructors:**

Student()

Student类的构造函数，从输入流中读取信息并存储。

Student(int TMP)

Student类的构造函数，创建空实例。

**Public Methods：**

static Student\* create()

静态的Create函数，从输入流中读取信息，创建并返回Student类实例的指针。

static Student\* createHead();

静态的Create函数，创建并返回空的Student类实例的指针。

void show();

打印Student实例的信息。

**Class StidentList:**

**Public Constructors:**

StudentList()

StudentList类的默认构造函数。

**Public Methods:**

void start()

建立考生报名系统。

void work()

读取操作码，并执行相应操作。

void \_insert()

插入考生信息操作。

void \_delete()

删除考生信息操作。

void \_search()

查询考生信息操作。

void \_change()

修改考生信息操作。

void \_count()

统计考生信息操作。

void show()

打印考生信息操作。