项目1说明文档

数据结构课程设计

——考试报名系统

作 者 姓 名： 谭欢秘

学 号： 1954185

指 导 教 师： 张颖

学院、 专业： 软件学院 软件工程

同济大学

Tongji University

目 录

[1 分析 1](#_Toc60123021)

[1.1 背景分析 1](#_Toc60123022)

[1.2 功能分析 1](#_Toc60123023)

[2 设计 2](#_Toc60123024)

[2.1 数据结构设计 2](#_Toc60123025)

[2.2 类结构设计 2](#_Toc60123026)

[2.3 成员与操作设计 2](#_Toc60123027)

[2.4 主函数设计 3](#_Toc60123028)

[3 实现 4](#_Toc60123029)

[3.1 创建表单功能的实现 4](#_Toc60123030)

[3.1.1 创建表单功能流程图 4](#_Toc60123031)

[3.1.2 创建表单功能核心代码 4](#_Toc60123032)

[3.1.3 创建表单功能演示 5](#_Toc60123033)

[3.2 打印表单功能的实现 5](#_Toc60123034)

[3.2.1 打印表单功能流程图 5](#_Toc60123035)

[3.2.2 打印表单功能核心代码 6](#_Toc60123036)

[3.2.3 打印表单功能演示 6](#_Toc60123037)

[3.1 插入功能的实现 7](#_Toc60123038)

[3.1.1 插入功能流程图 7](#_Toc60123039)

[3.1.2 插入功能核心代码 8](#_Toc60123040)

[3.1.3 插入功能截屏示例 8](#_Toc60123041)

[3.2 删除功能的实现 9](#_Toc60123042)

[3.2.1 删除功能流程图 9](#_Toc60123043)

[3.2.2 删除功能核心代码 10](#_Toc60123044)

[3.2.3 删除功能截屏示例 10](#_Toc60123045)

[3.3 查找功能的实现 10](#_Toc60123046)

[3.3.1 查找功能流程图 11](#_Toc60123047)

[3.3.2 查找功能核心代码 12](#_Toc60123048)

[3.3.3 查找功能截图示例 12](#_Toc60123049)

[3.4 修改功能的实现 13](#_Toc60123050)

[3.4.1 修改功能流程图 13](#_Toc60123051)

[3.4.2 修改功能核心代码 14](#_Toc60123052)

[3.4.3 修改功能截屏示例 14](#_Toc60123053)

[3.5 主函数的实现 15](#_Toc60123054)

[3.5.1 主函数流程图 15](#_Toc60123055)

[3.6.2 总体系统核心代码 16](#_Toc60123056)

[3.6.3 总体系统截屏示例 16](#_Toc60123057)

[4 测试 17](#_Toc60123058)

[4.1 功能测试 17](#_Toc60123059)

[4.1.1 插入功能测试 17](#_Toc60123060)

[4.1.2 删除功能测试 17](#_Toc60123061)

[4.1.3 查找功能测试 18](#_Toc60123062)

[4.1.4 修改功能测试 19](#_Toc60123063)

[4.2 边界测试 20](#_Toc60123064)

[4.2.1 初始化无输入数据 20](#_Toc60123065)

[4.2.2 删除头结点 20](#_Toc60123066)

[4.2.3 删除后链表为空 20](#_Toc60123067)

[4.3 出错测试 21](#_Toc60123068)

[4.3.1 考生人数错误 21](#_Toc60123069)

[4.3.2 操作码错误 21](#_Toc60123070)

[4.3.3 插入位置不存在 22](#_Toc60123071)

[4.3.4 删除考号不存在 22](#_Toc60123072)

[4.3.5 查找考号不存在 23](#_Toc60123073)

[4.3.6 修改考号不存在 24](#_Toc60123074)

[5 总结 24](#_Toc60123075)

[5.1 遇到的错误 24](#_Toc60123076)

[5.1.1 xx函数中 24](#_Toc60123077)

[5.2 改进与优化 24](#_Toc60123078)

[5.3 项目心得 24](#_Toc60123079)

# 1 分析

## 1.1 背景分析

考试报名系统是一个学校不可缺少的部分，它对于学校的管理者和学生来说都至关重要，所以一个良好的考试报名系统应该能够为用户提供充足的信息和功能。考试报名系统对于学校加强考试管理有极其重要的作用。随着学生数量和考试数量的日益庞大，如何管理如此庞大的数据显得极为复杂，传统的手工管理工作量大且容易出错。

随着计算机科学技术的不断成熟，使用计算机对考试报名系统进行管理，具有手工管理所无法比拟的优势。这些优点能够极大地提高学校和学生的效率，也是学校走向信息化、科学化、国际化的重要条件。因此，开发一套考试报名系统具有十分重要的意义。

## 1.2 功能分析

作为一个最简易的考试报名系统，首先应该有的功能就是输入同学们的考试报名情况并且可以予以显示。其次，考试报名系统还应该具有插入、删除、修改功能，以保证同学可以随时更改自己的考试报名情况。最后，考试报名系统软件还应该确保软件可以正常关闭。

综上所述，一个考试报名系统至少应该具有输入、输出、插入、删除、修改、退出的功能。

# 2 设计

## 2.1 数据结构设计

如上功能分析所述，该系统要求大量的增加、删除、修改操作，而链表进行增加、删除等操作十分简便，因此考虑使用链表数据结构。同时，为了实现简易，在第一个结点之前附加一个头结点，这样就使得增加或者删除头结点与处理其他结点方法相同，使得程序简洁。

## 2.2 类结构设计

经典的链表一般包括两个抽象数据类型（ADT）——链表结点类（LNode）与链表类（LinkList），而两个类之间的耦合关系可以采用嵌套、继承等多种关系。为方便处理，本系统采用描述链表结点类（LNode），这样使得链表结点类（LinkList）可以访问链表结点。

## 2.3 成员与操作设计

**链表结点类（StuNode）**

**公有成员：**

Int number; //表示学生考号

string name;//表示学生姓名

string gender;//表示学生性别

int age;//表示学生年龄

string post;//表示学生报考类型

StuNode \*link;//链表结点的指针域

**公有操作：**

StuNode(**int** num=0,string name\_="stu0",string gender\_="男",**int** age\_=0,string post\_="无",StuNode \*p=**NULL**)

{

number=num;

name=name\_;

gender=gender\_;

age=age\_;

post=post\_;

p=**NULL**;

}//具有默认参数的构造函数

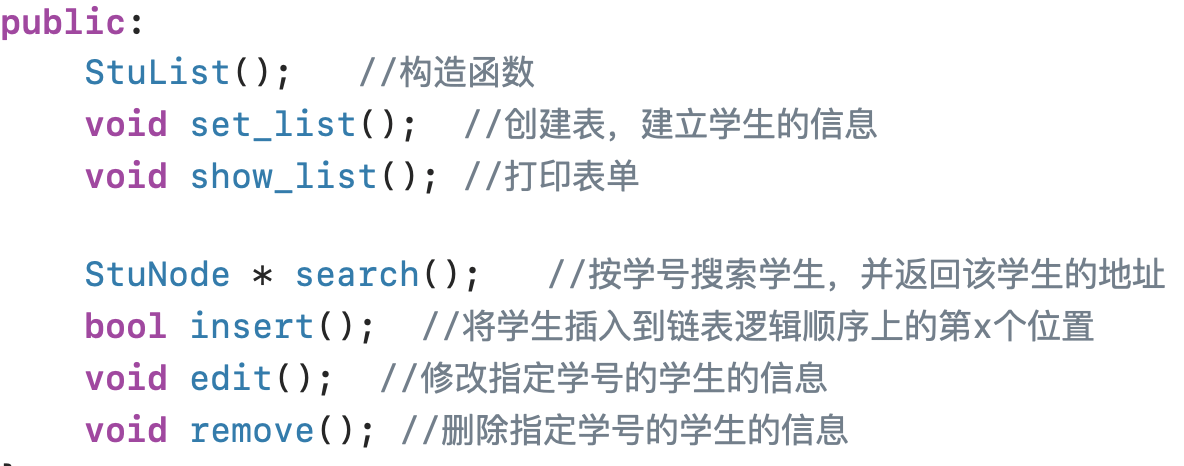
**链表类（StuList）**

**私有成员：**

StuNode \*first,\*last;//链表的头指针、用于后插法建立链表用的尾指诊

Int length;//链表的长度

**公有操作：**

****

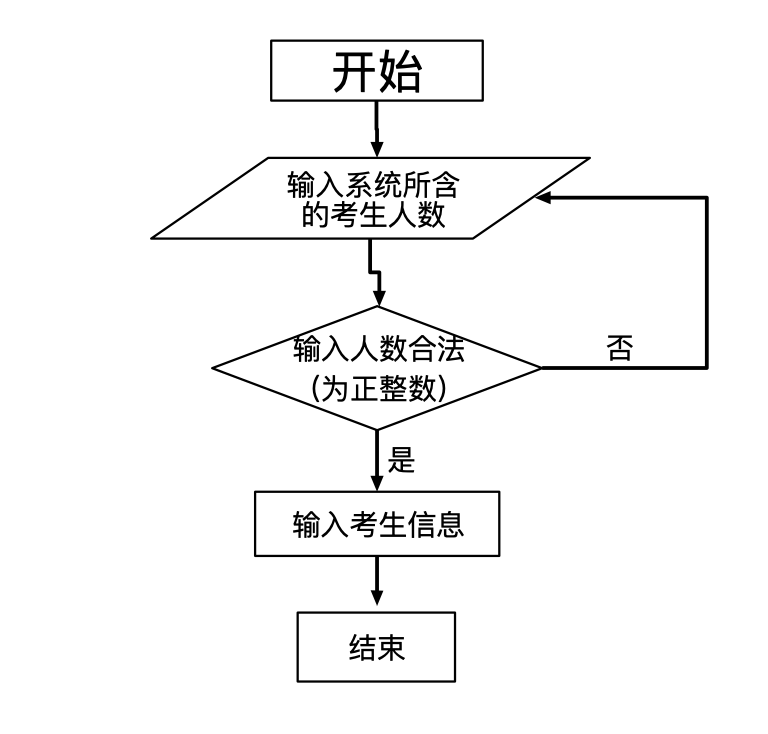
## 2.4 主函数设计

声明一个StuList类的对象slist，slist是一个有头节点的链表。首先调用slitst.set\_list()函数实现表单的初始化，完成对链表slist的创建和输入数据工作后打印出表单，然后根据用户所选择的操作（choice变量）执行链表slist对应的成员函数。

# 3 实现

## 3.1 创建表单功能的实现

### 3.1.1 创建表单功能流程图



### 3.1.2 创建表单功能核心代码

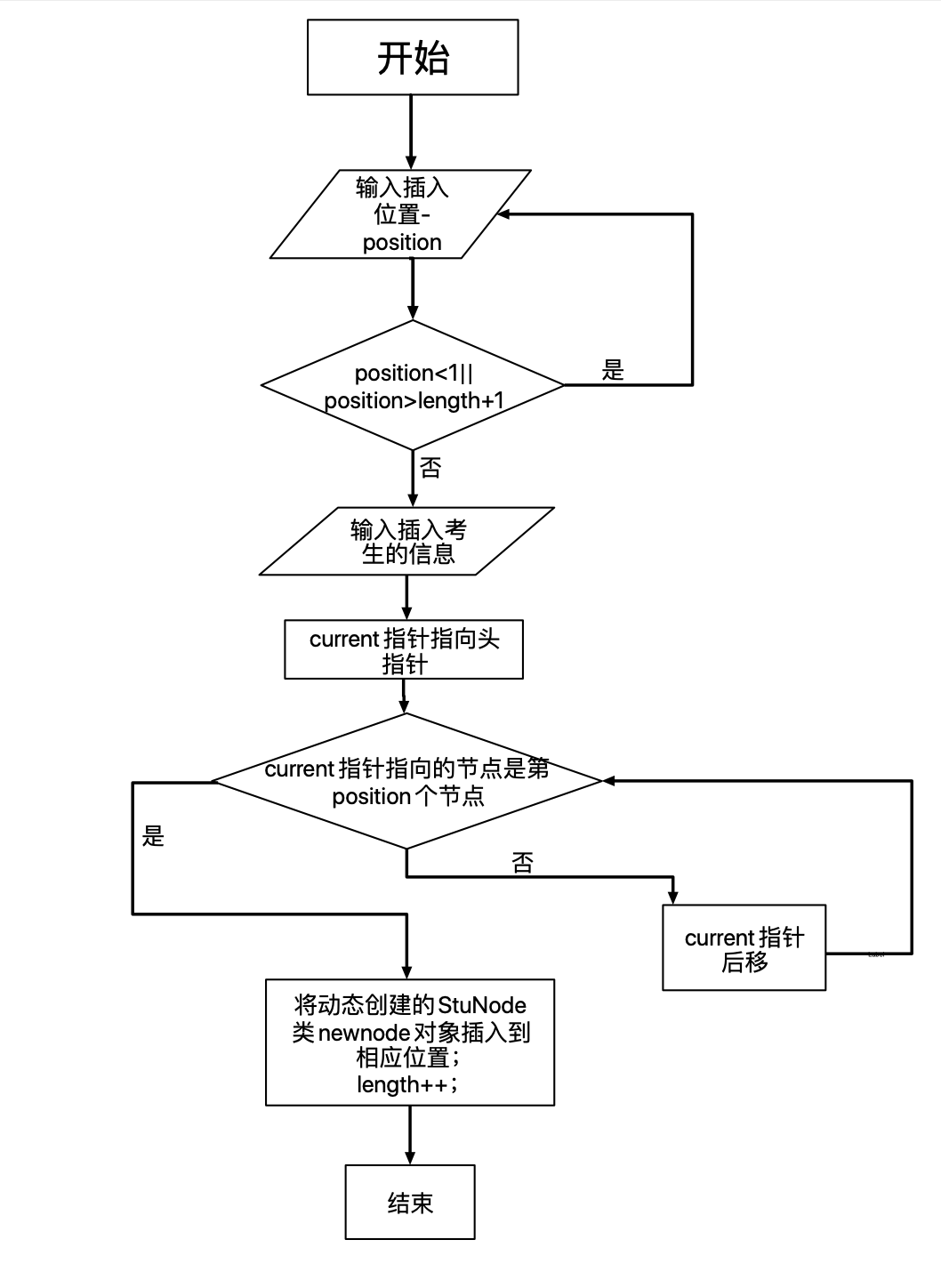
## 3.2 打印表单功能的实现

### 3.2.1 打印表单功能流程图

### 3.2.2 打印表单功能核心代码

## 3.1 插入功能的实现

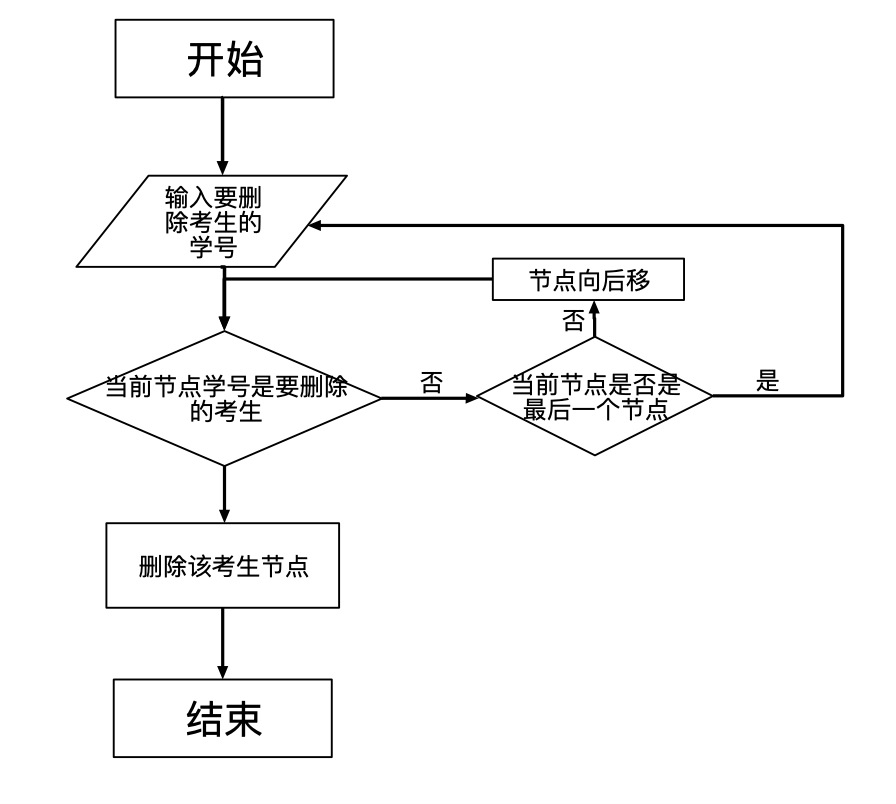
### 3.1.1 插入功能流程图



### 3.1.2 插入功能核心代码

## 3.2 删除功能的实现

### 3.2.1 删除功能流程图



### 3.2.2 删除功能核心代码

## 3.3 查找功能的实现

### 3.3.1 查找功能流程图

### 3.3.2 查找功能核心代码

## 3.4 修改功能的实现

### 3.4.1 修改功能流程图

### 3.4.2 修改功能核心代码

## 3.5 主函数的实现

### 3.5.1 主函数流程图

### 3.6.2 总体系统核心代码

### 3.6.3 总体系统截屏示例

# 4 测试

## 4.1 功能测试

### 4.1.1 插入功能测试

**测试用例**：4 stu4 女 22 移动开发员

**预期结果**：

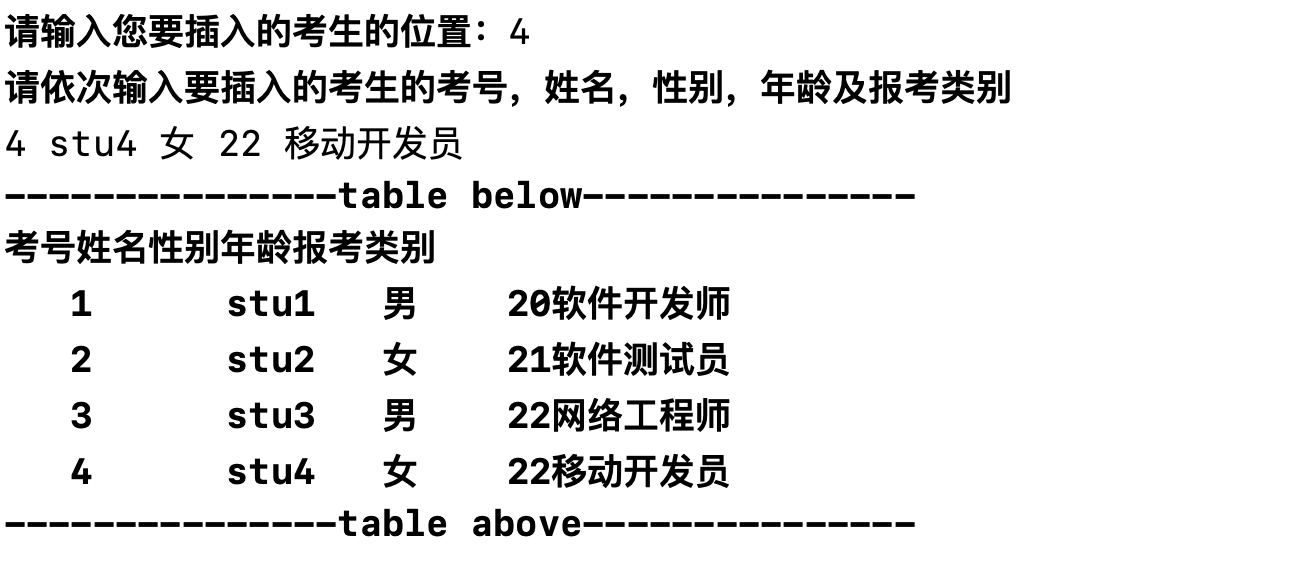
1 stu1 男 20 软件开发师

2 stu2 女 21 软件测试员

3 stu3 男 22 网络工程师

4 stu4 女 22 移动开发员

**实验结果：**



### 4.1.2 删除功能测试

**测试用例：**删除考号为3的考生

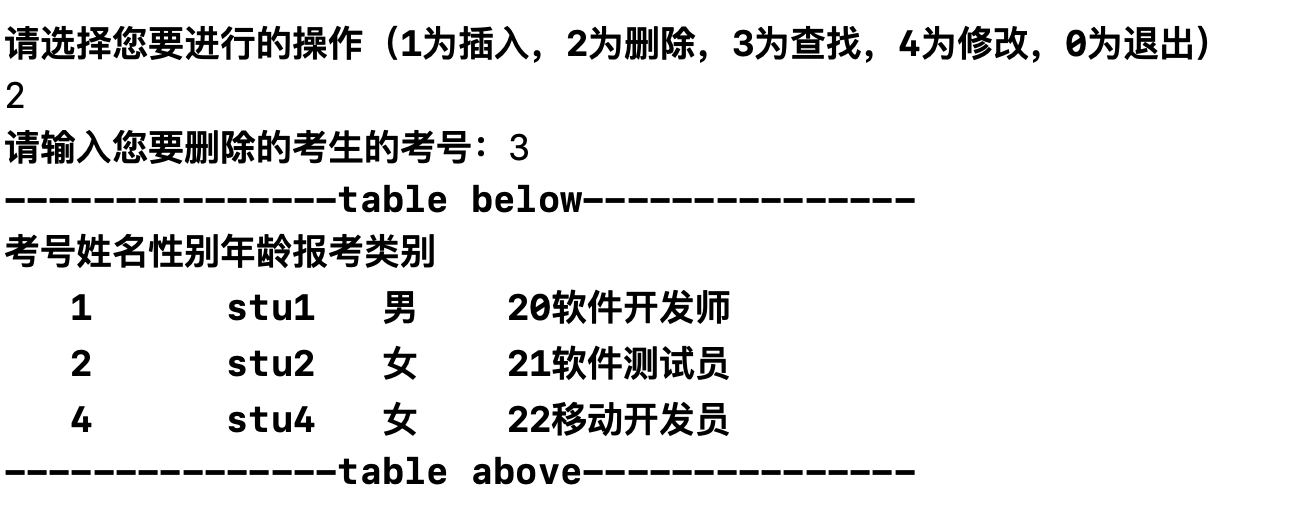
**预期结果：**

1 stu1 男 20 软件开发师

2 stu2 女 21 软件测试员

4 stu4 女 22 移动开发员

**实验结果：**

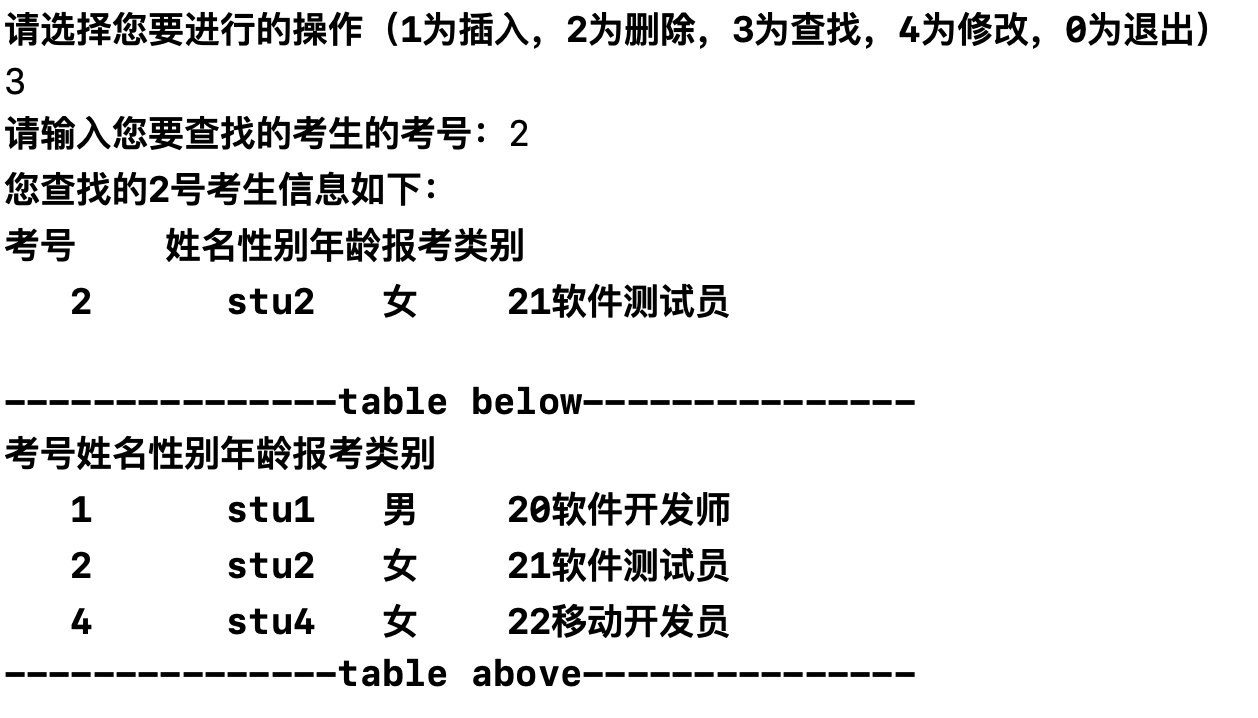


### 4.1.3 查找功能测试

**测试用例：**查找考号为2的考生

**预期结果：**

2 stu2 女 21 软件测试员

**实验结果：**

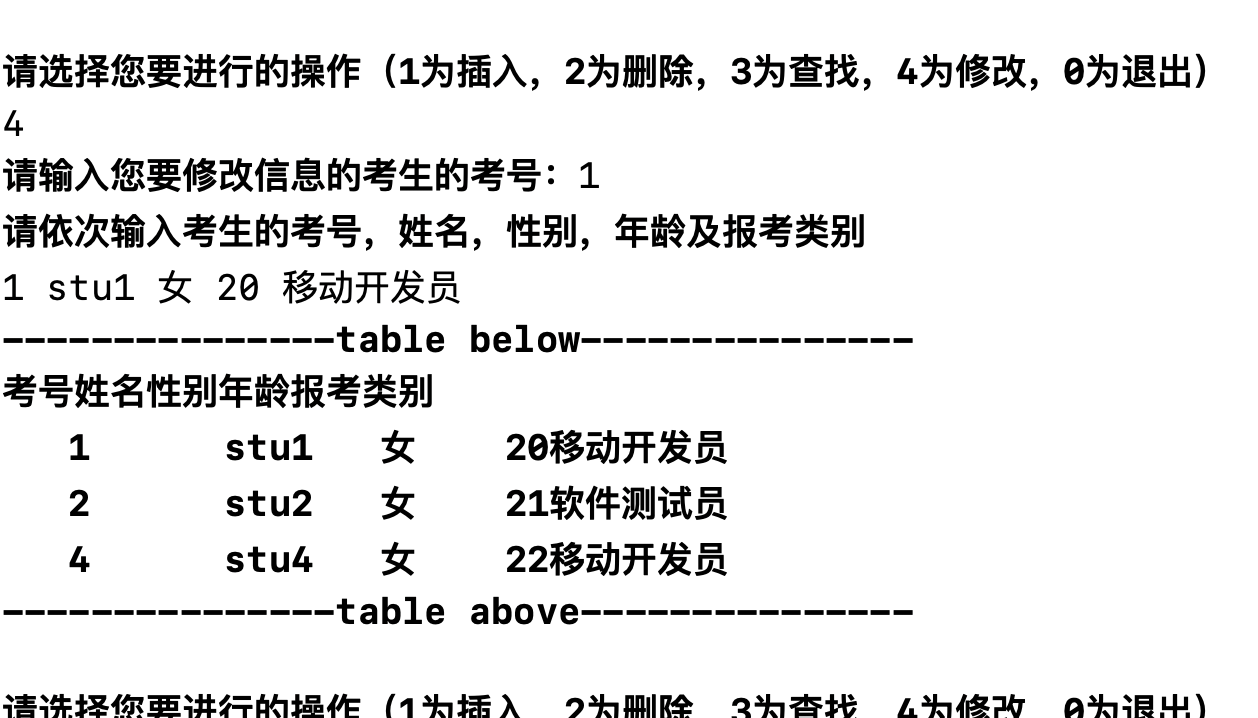
### 4.1.4 修改功能测试

**测试用例：**将考号1修改为性别女，年龄20，报考种类移动开发员。

**预期结果：**

1 stu1 女 20 移动开发员

**实验结果：**

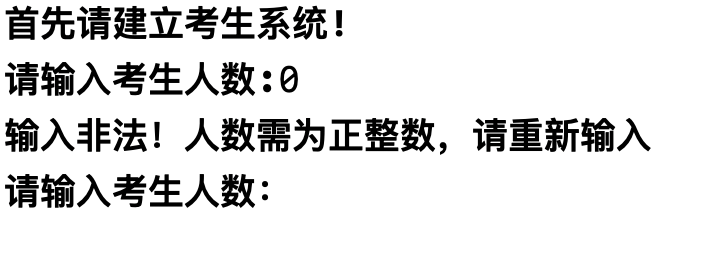


## 4.2 边界测试

### 4.2.1 初始化数据非法

**测试用例：**初始输入数据非法

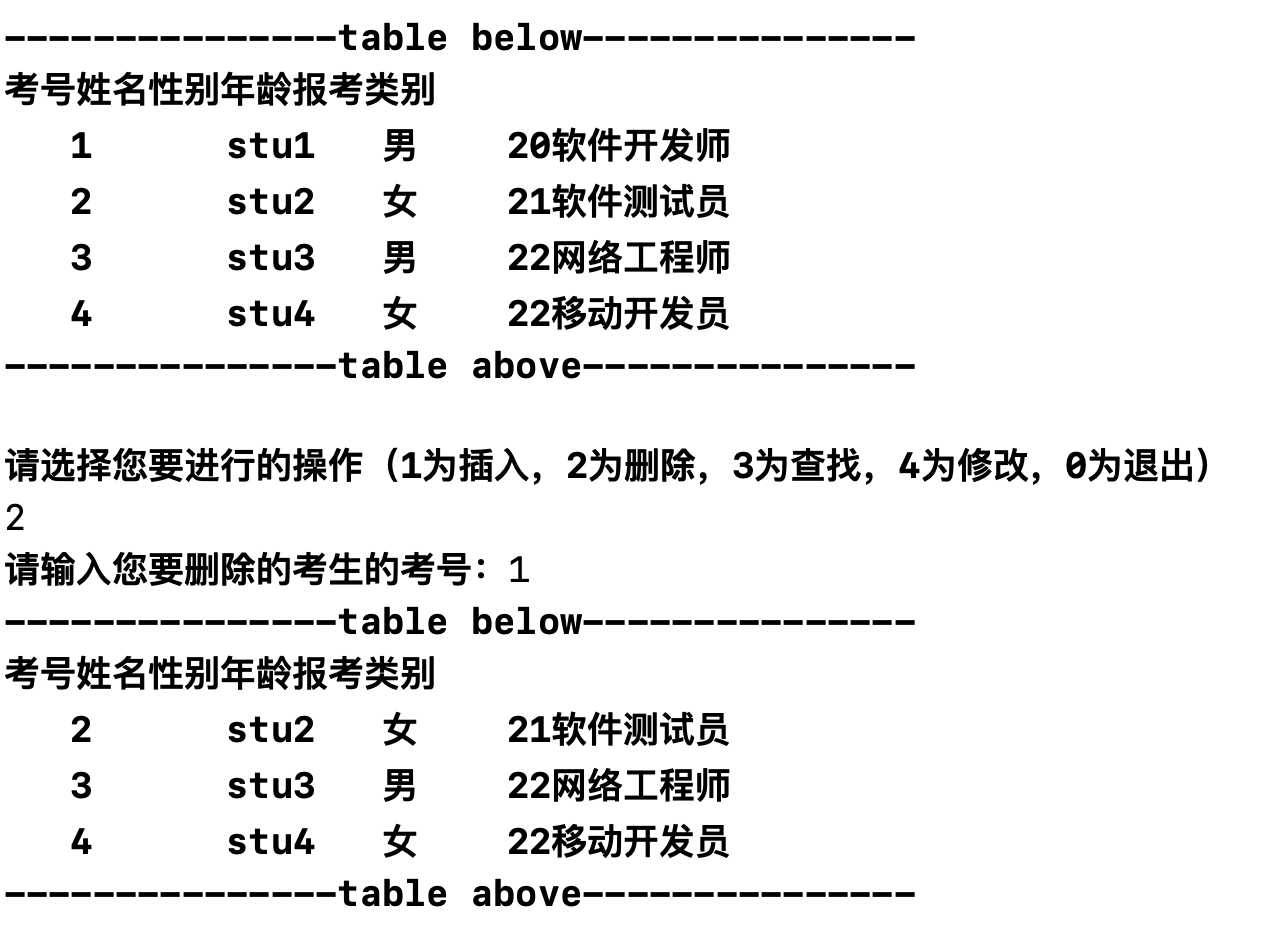
**预期结果：**若输入非正整数的人数，立即给出错误提示，程序运行正常不崩溃。

**实验结果：**

### 4.2.2 删除头结点

**测试用例：**删除头结点

**预期结果：**程序正常运行，不崩溃。

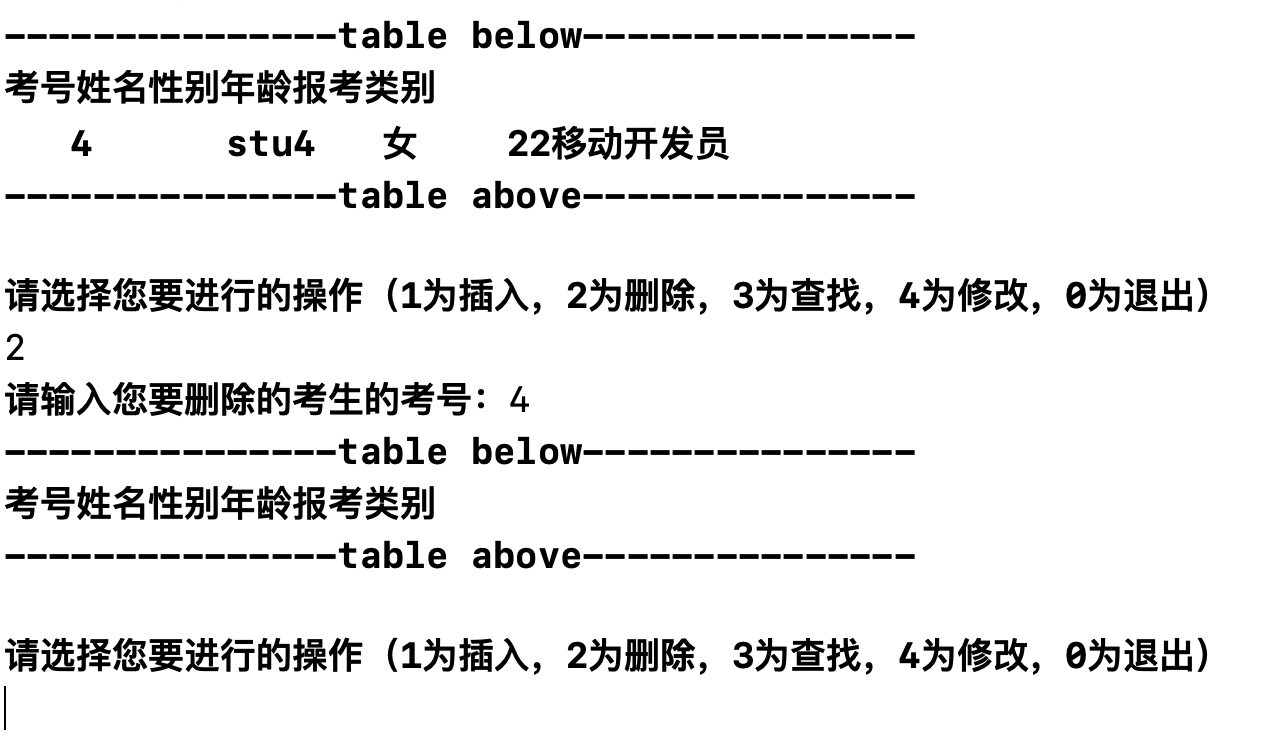
**实验结果：**

### 4.2.3 删除后链表为空

**测试用例：**删除前链表只有一个结点，删除后链表为空

**预期结果：**程序正常运行，不崩溃。

**实验结果：**

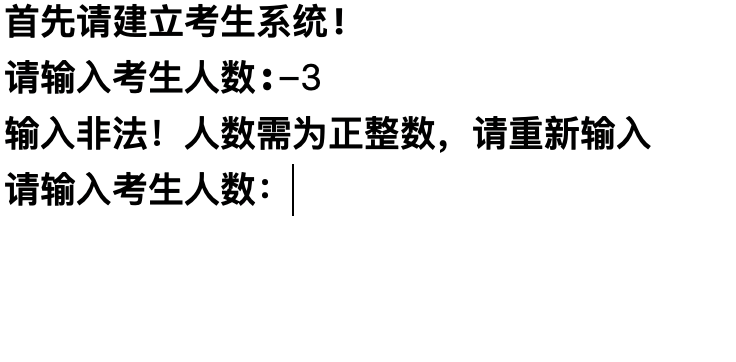


## 4.3 出错测试

### 4.3.1 考生人数错误

**测试用例：**输入考生人数为负数

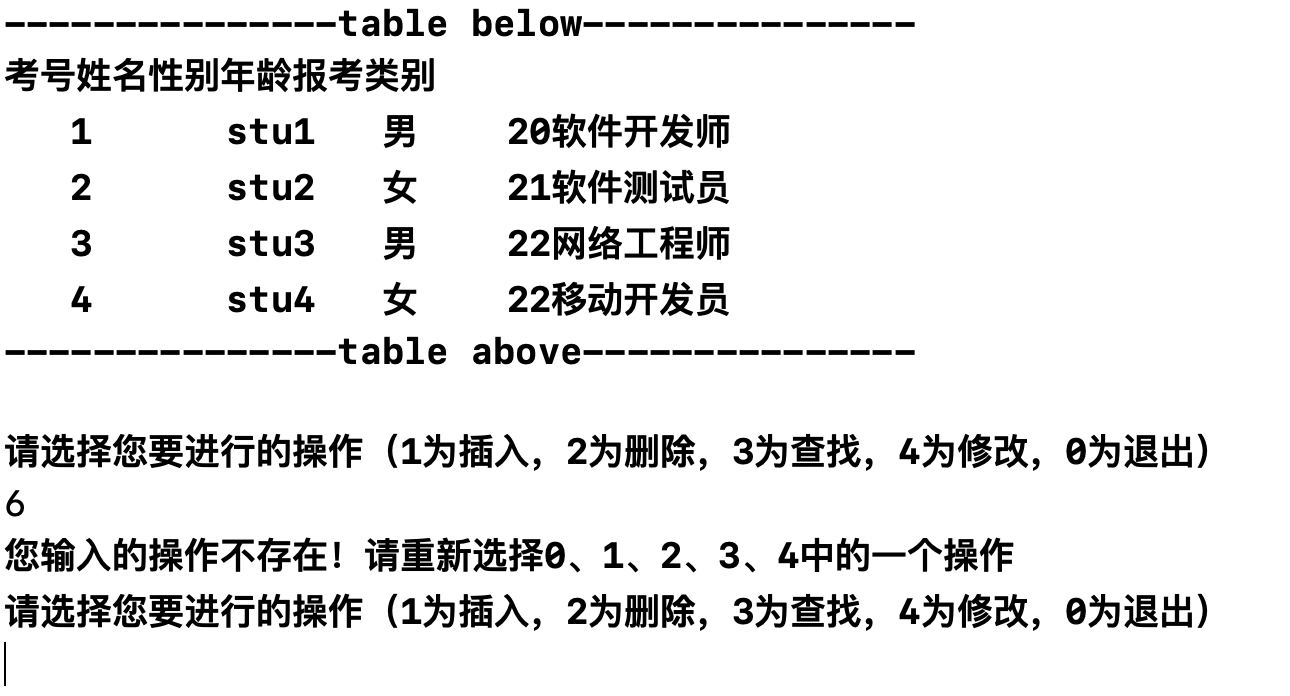
**预期结果：**程序给出提示信息，程序正常运行不崩溃。

**实验结果：**

### 4.3.2 操作码错误

**测试用例：**输入操作码错误

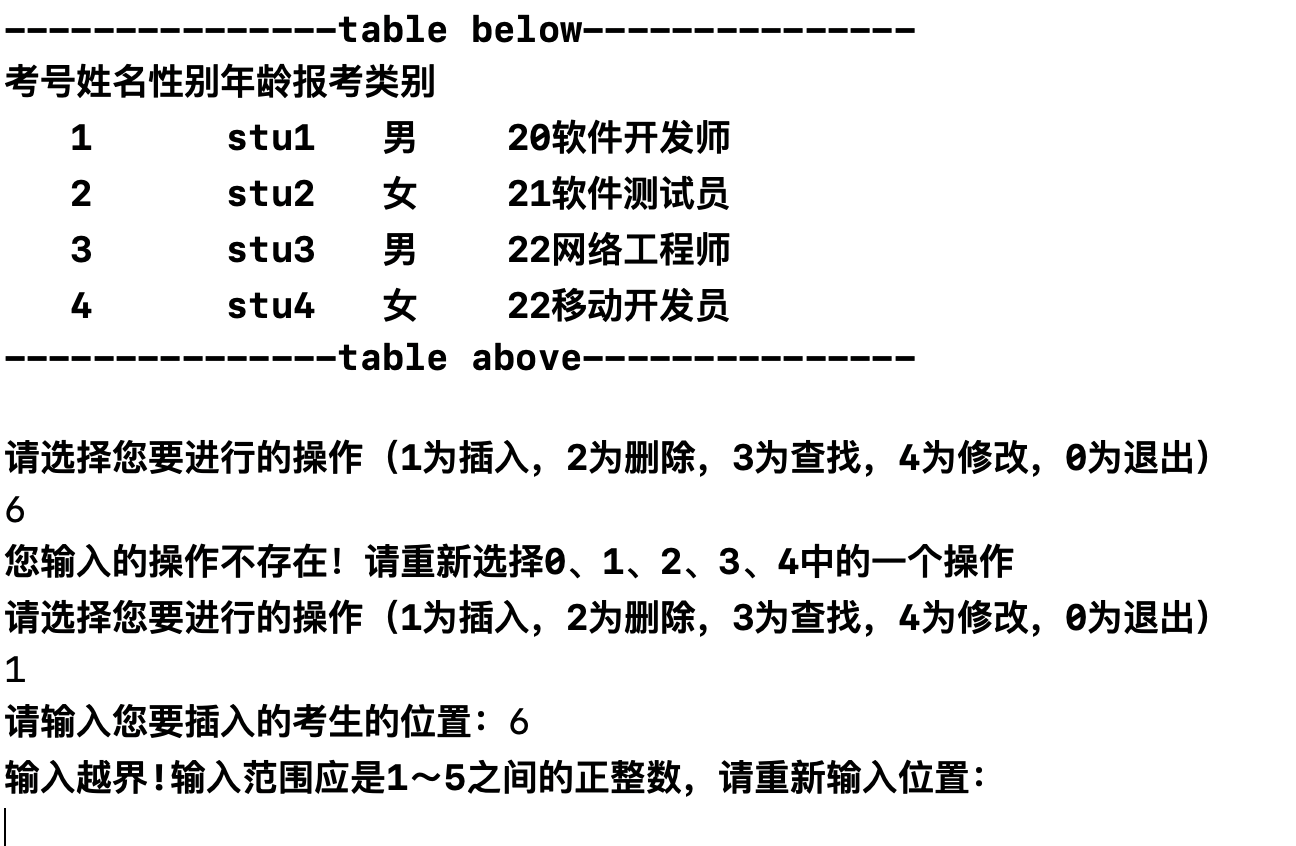
**预期结果：**程序给出提示信息，程序正常运行不崩溃。

**实验结果：**

### 4.3.3 插入位置不存在

**测试用例：**链表里有两条记录，向链表的第四个位置插入结点

**预期结果：**程序给出错误信息，程序正常运行不崩溃。

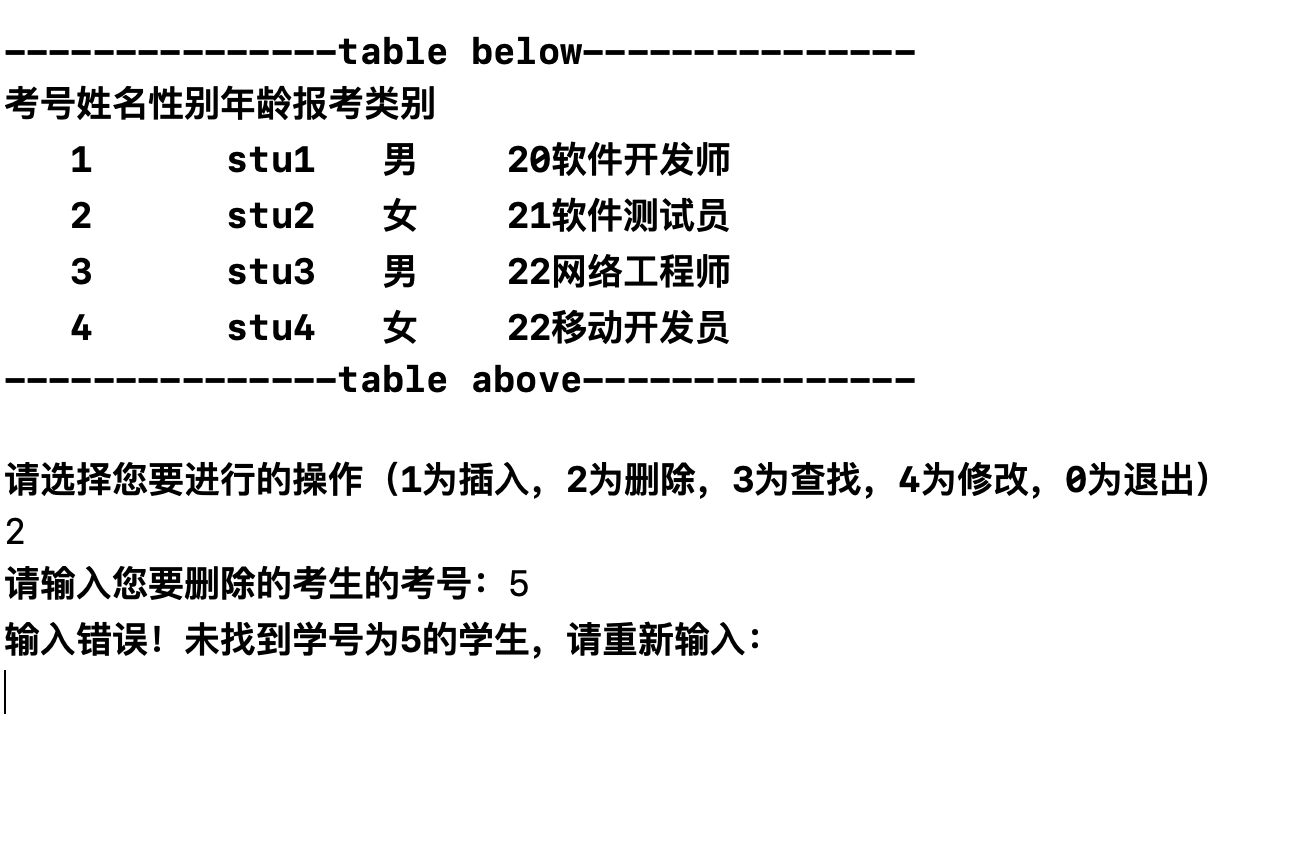
**实验结果：**

### 4.3.4 删除考号不存在

**测试用例：**要删除的考号不存在

**预期结果：**程序给出错误信息，程序正常运行不崩溃。

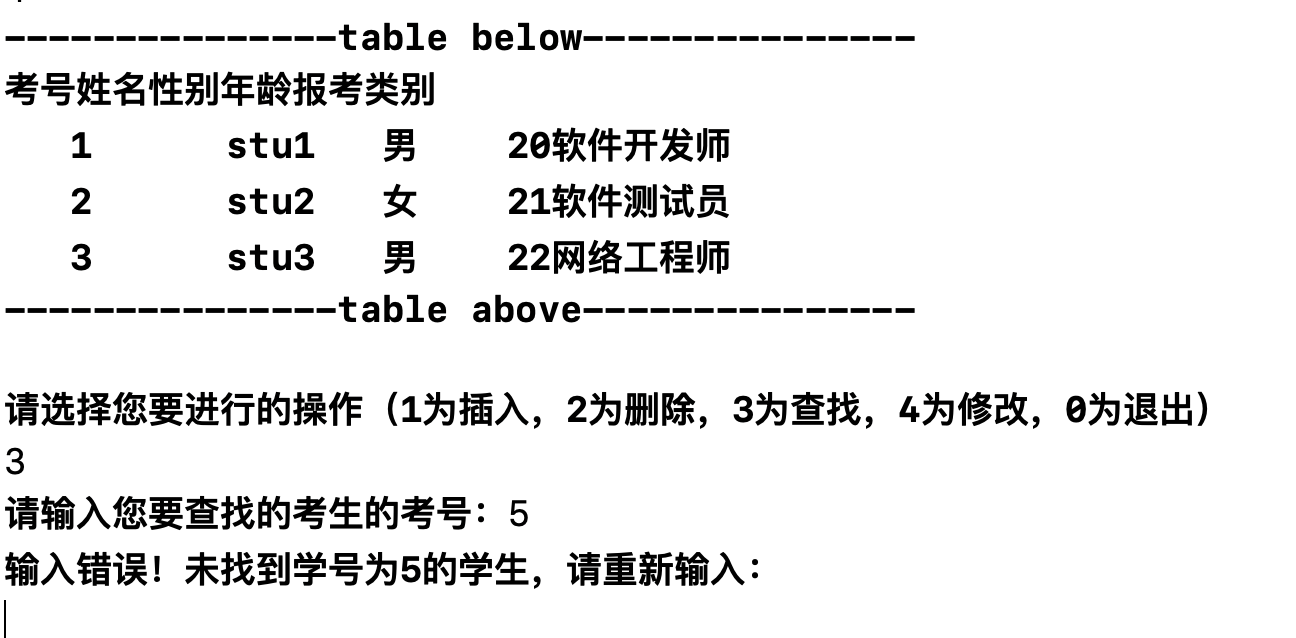
**实验结果：**

****

### 4.3.5 查找考号不存在

**测试用例：**要查找的考号不存在

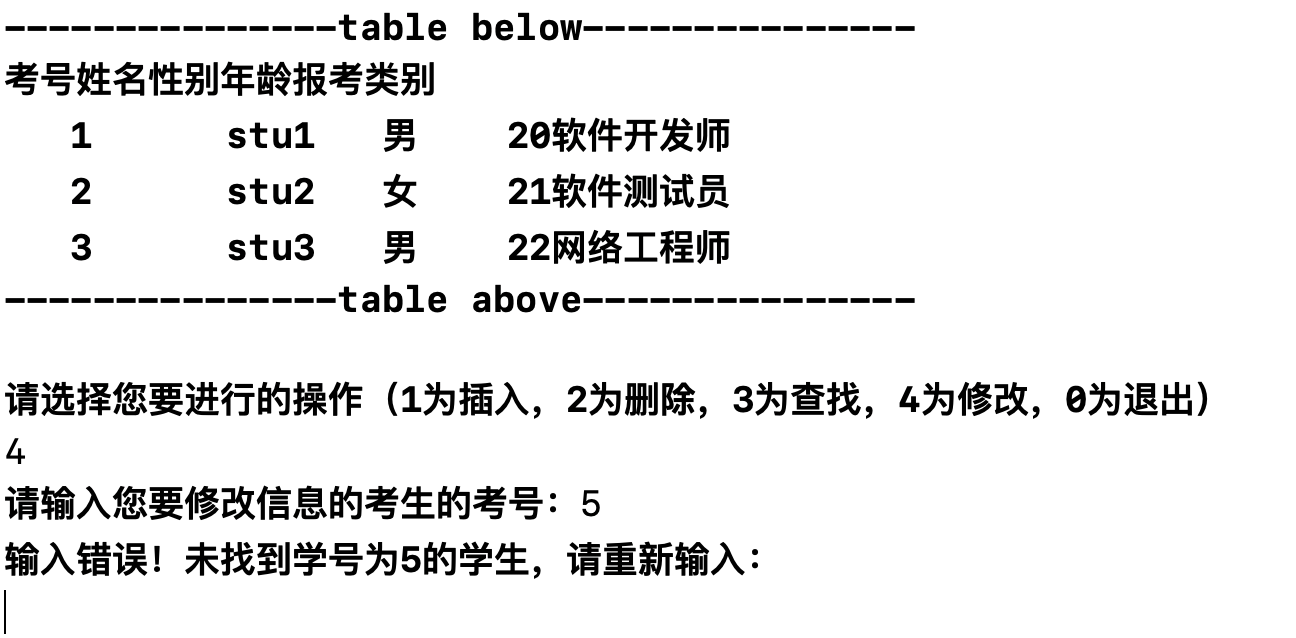
**预期结果：**程序给出错误信息，程序正常运行不崩溃。

**实验结果：**

### 4.3.6 修改考号不存在

**测试用例：**要修改的考号不存在

**预期结果：**程序给出错误信息，程序正常运行不崩溃。

**实验结果：**

# 5 总结

## 5.1 遇到的错误

### 5.1.1 xx函数中

## 5.2 改进与优化

## 5.3 项目心得