**数据结构课程设计**

**-考试报名系统**

# 题目描述

1. 项目简介

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. 项目功能要求

本项目的实质是完成对考生信息的建立，查找，插入，修改，删除等功能。其中考生信息包括准考证号，姓名，性别，年龄和报考类别等信息。项目在设计时应首先确定系统的数据结构，定义类的成员变量和成员函数；然后实现各成员函数以完成对数据操作的相应功能；最后完成主函数以验证各个成员函数的功能并得到运行结果。

# 总体思路

根据项目需求可知，类似数据库的增删改查操作，使用链表来存储每个考生的信息成为最佳选择，因为链表的结构为以结构体作为存储单元，每个单元包含了指向下一节点的指针，使得增加和删除操作无需对其他元素进行移动，只需对需要进行操作的元素周围的节点中的指针修改，实现空间的自由扩充和缩减。由于此项目中对向前寻找没有过多要求，因此采用单链表，即只有指向下一节点的指针。为了方便增删代码的统一，我们增加一个附加头节点，用极低的代价降低了代码的复杂性。最终选择带附加头结点的单链表作为数据存储结构，每个单元用结构体来包含一个考生的信息。

# 代码类设计

源文件中包含了对考生报名系统菜单的搭建。

头文件mylist.h包含了对考生结构体student、链表节点node类和链表list类的定义。以下是数据类型和链表相关类的定义：

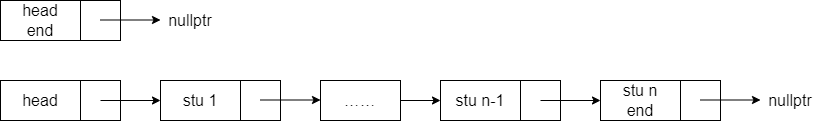
# 具体实现

## 输入处理

对于学生姓名、性别和报考类型存储的类型为char数组，设定长度为50，未考虑输入溢出的情况。由于菜单类界面的操作数及数据输入等多半复杂，因此将每种功能都写一个函数。如第一次输入就调用input函数，输入和错误处理均在此函数完成，并返回是否输入成功。其他功能类似实现。

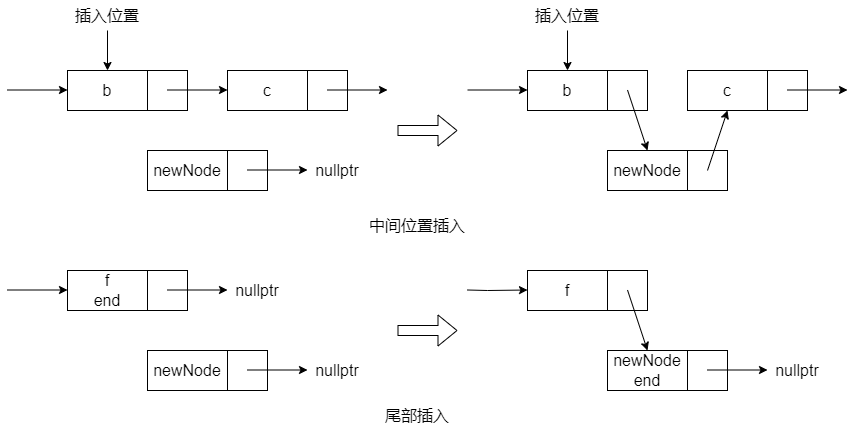
## 链表实现

链表形式为带有附加头结点和尾结点的单链表。头结点不代表任何值，仅作为开头，尾结点是链表的最后一个元素，结构如下：



### 添加新元素

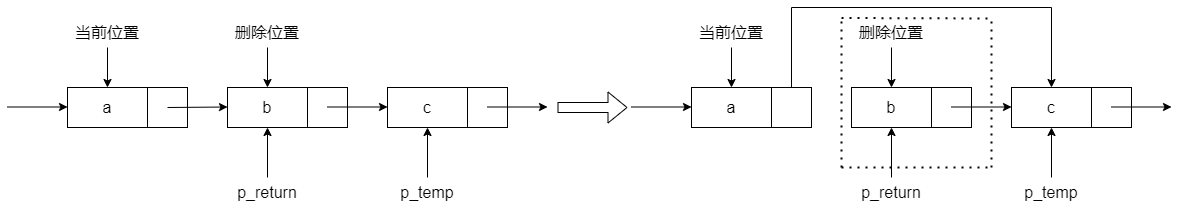
如果在尾部添加新元素，则需要将尾结点后继节点指向新元素，再将尾结点移至新节点。如果在链表中间添加元素，我们选择后插法则需要，先找到插入的位置，令新节点指向插入位置后继的位置，再将插入位置的后继指针指向新节点，如下图：



添加新元素的插入函数代码如下：

### 删除元素

仍然需要先找到删除的位置，但需要找的是要删除的节点的前一位。先暂存当前节点的后继节点为。再令当前节点的后继指针赋值为的后继指针，再释放。如果删除的是最后一个节点，在上述操作后再令尾结点指向当前位置即可。上述操作图示如下：



删除函数的代码如下：

### 查找元素和修改

由于链表为无序链表，因此选择顺序查找，即从头结点开始逐步访问后一个节点，如果找到与查找考号相同的就返回该节点位于链表的序号。修改则是在查找的基础上修改节点元素信息，但需要注意的是，修改后的考生信息的考号不能与已有考生信息重复，这就需要在赋值前查找是否存在考号为修改值的节点。

以下是修改信息的代码：

## 基于链表的考生系统

由于菜单系统的输入提示的复杂程度，所有操作的实现全挤在一个if-else语句中代码会过于冗长，因此对每个功能封装一个函数，里面包括对输入的提示和错误处理：

在初始录入时或添加新信息或修改信息前，要用find函数检查系统中无该考号的考生信息。

添加新元素可能会遇到系统数据为空，这需要判断可插入范围，正常的范围是1到链表当前长度加一，对为空进行特判，我们规定为空时插入位置为1才可插入数据。

# 项目小结

1. 性能分析

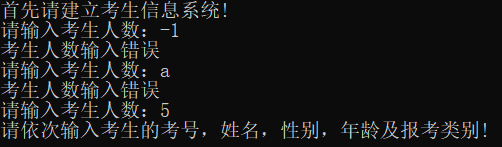
由于链表的特殊结构，增加和删除元素的时间复杂度都是常数级。而查找和修改由于使用的顺序搜索，时间复杂度平均为。空间复杂度与节点数据结构有关，忽略系数链表空间复杂度为。

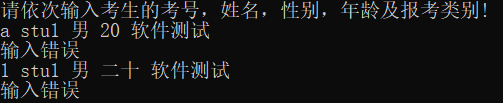
1. 优化和扩展

由于项目中对插入操作的限制，因为插入有插入位置的需求而不是按照考生的考号信息自动排序，因此链表的关键字是无序的。如果我们规定插入与位置无关，考生信息会自动选择按照考号为关键字排序的位置，如此链表就是按考号为关键字的有序链表。对于有序链表我们可以使用跳表对链表做索引表，或使用B+树优化，这些方法在查找和修改元素时可以大大减少查找的时间开销。如使用跳表可以将查找的时间降为。为了输入信息方便，使用了char数组作为字符串输入，可以单独写个string类对输入长度不加限制。初始录入为手动输入，可以添加对包含考生信息的txt文件或csv文件读入进行扩展。

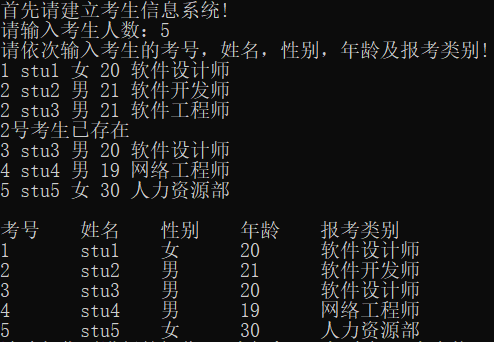
# 程序功能测试

1. 初始输入
2. 错误提示

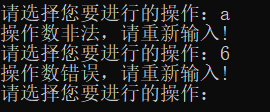




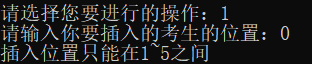
1. 输入有重复考号



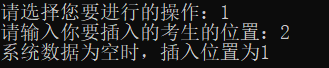
1. 操作数输入



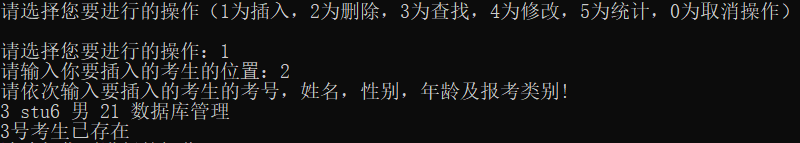
1. 插入
2. 插入位置错误



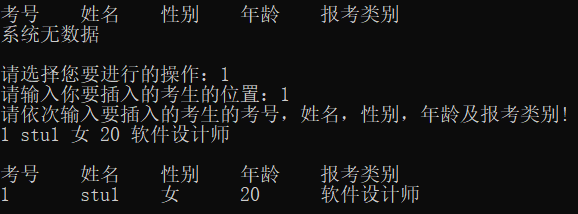
1. 对空链表进行插入



1. 插入的考生考号重复



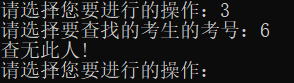
1. 插入到空链表



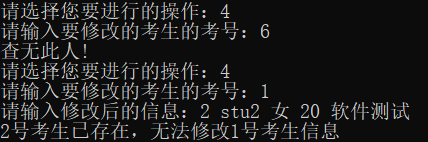
1. 删除



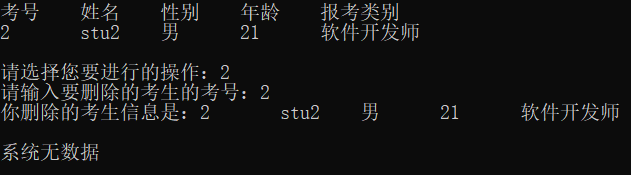
1. 查找



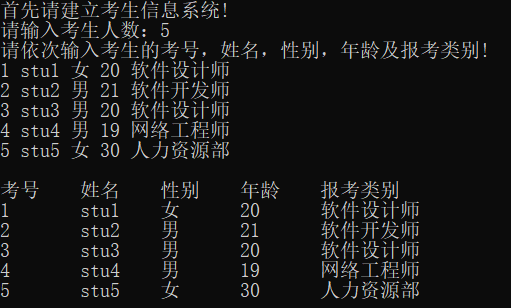
1. 修改后与已有考号重复



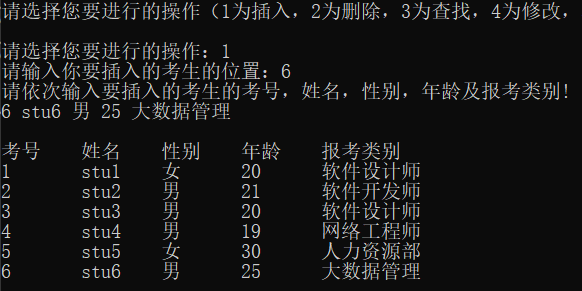
1. 打印空链表



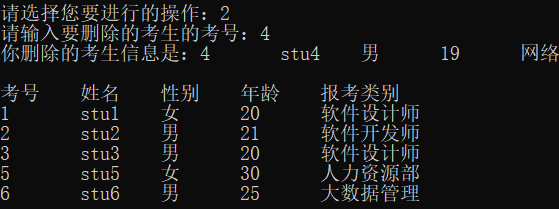
1. 正常测试
2. 初始输入



1. 插入



1. 删除



1. 查找、删除、统计

