项目文档

谈瑞

项目一：考生管理系统

|  |  |  |
| --- | --- | --- |
|  |  |  |
| 专业班级：软件工程4班  电话：18936361545 | 学号：1452775  电子邮件：tanrui106@163.com | 课目：数据结构课程设计  个人网站：http://guitoubing.top |
|  |  |  |

目录

[项目简介 3](#_Toc503091699)

[项目概要 3](#_Toc503091700)

[项目功能及要求 3](#_Toc503091701)

[项目结构 4](#_Toc503091702)

[项目类的实现 5](#_Toc503091703)

[ExamineeList类 5](#_Toc503091704)

[Examinee类 6](#_Toc503091705)

[运行测试 7](#_Toc503091706)

[正常运行 7](#_Toc503091707)

[错误检测 8](#_Toc503091708)

# 项目简介

## 项目概要

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。

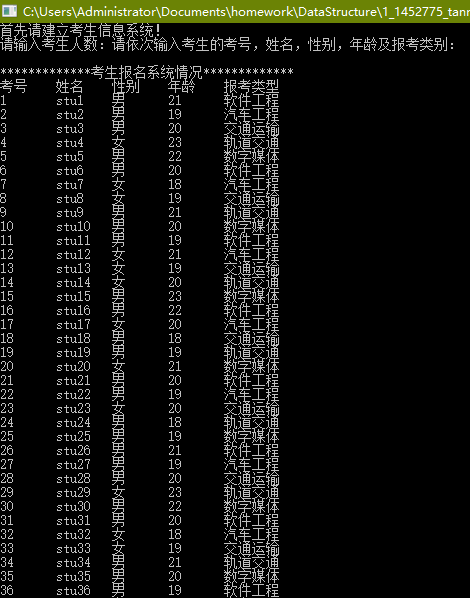
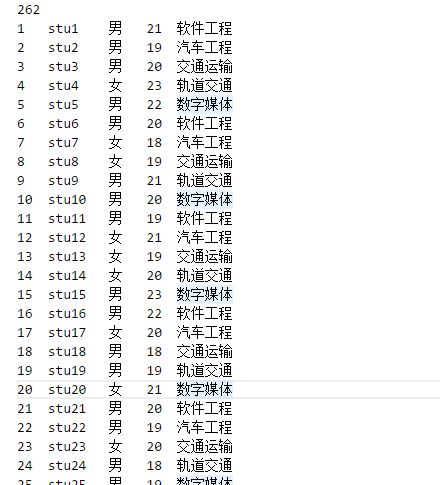
## 项目功能及要求

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。

# 项目结构

本项目采用单链表实现，基本上就是考察链表的插入、删除等基本操作，应该说写起来很顺畅，按照题设思路一步一步往下做就能写成功。

输入采用标准输入台输入以及文件输入两种方式，文件输入格式如下：

第一行注明总共有多少学生，以后的分别写出每名学生的基本信息，从而建立考生信息系统。

# 项目类的实现

## ExamineeList类

|  |  |
| --- | --- |
| 类成员 | 作用 |
| ExamineeList(istream &in = cin); | 构造函数，从给定的输入流输入考生数据并建立考生系统 |
| ~ExamineeList(); | 析构函数 |
| bool ifNumExist(int); | 考号是否存在 |
| void printList(); | 输出系统列表 |
| void operateList(int); | 选择操作 |
| void insertExaminee(); | 添加考生 |
| void deleteExaminee(int); | 删除考生 |
| Examinee\* searchExaminee(int); | 查找考生 |
| void modificationExaminee(int); | 修改考生信息 |
| void countExaminee(int); | 统计考生信息 |
| Examinee\* examinee; | 当前考生指针 |
| Examinee\* head\_examinee; | 考生列表头指针 |
| Examinee\* tail\_examinee; | 考生列表尾指针 |
| vector<int> num\_vec; | 存储考号的vector，便于查询确定考号的考生是否存在 |
|  |  |

## Examinee类

|  |  |
| --- | --- |
| 类成员 | 作用 |
| Examinee(); | 默认构造函数 |
| Examinee(int \_num, string  \_name, bool \_sex,  int \_age,string \_type); | 给定参数的构造函数 |
| ~Examinee(); | 析构函数 |
| void setExam\_num(int); | 设置考生考号 |
| void setName(string); | 设置考生姓名 |
| void setSex(bool); | 设置考生性别 |
| void setAge(int); | 设置考生年龄 |
| void setType(string); | 设置考生报考类型 |
| int getExam\_num(); | 获取考生考号 |
| string getName(); | 获取考生姓名 |
| string getSex(); | 获取考生性别 |
| int getAge(); | 获取考生年龄 |
| string getType(); | 获取考生报考类型 |
| Examinee\* getNextExaminee(); | 获取指向其后一个考生指针 |
| Examinee\* getPrevExaminee(); | 获取指向其前一个考生指针 |
| void setNextExaminee(Examinee&); | 将参数考生设为当前考生的后一个考生 |
| void setPrevExaminee(Examinee&); | 将参数考生设为当前考生的前一个考生 |
| void noPrevExaminee(); | 将当前考生前考生指针置空 |
| void noNextExaminee(); | 将当前考生后考生指针置空 |
| friend ostream& operator<<  (ostream&, Examinee&); | 输出重载函数 |
| int exam\_num; | 考生考号 |
| string name; | 考生姓名 |
| bool sex; | 考生性别（true表示男，false表示女） |
| int age; | 考生年龄 |
| string type; | 考生报考类型 |
| Examinee\* next\_examinee; | 指向后一个考生的指针 |
| Examinee\* prev\_examinee; | 指向前一个考生的指针 |
|  |  |

# 运行测试

## 正常运行

## 错误检测