|  |  |  |  |
| --- | --- | --- | --- |
| **学号** |  | **成绩** |  |

**课程设计说明书**

**设计名称 C++课程设计**

**设计题目 学生证管理系统**

**设计时间**

**学 院 计算机科学与工程学院**

**专 业 计算机科学与技术**

**班 级**

**姓 名**

**指导教师**

**年 月 日**

C++课程设计说明书

**目录**

1. 设计任务说明

2. 开发工具

3. 设计思路

4. 系统效果图

5. 程序清单

6. 设计体会

7. 参考文献

8. 教师评语

1. **设计任务说明**

完成一个**学生证管理系统**，系统应该具有以下功能：

1. 录入某学生的学位证信息；
2. 给定学号，显示某位学生的学生证信息；
3. 给定某个班级的班号，显示该班级所有学生的学生证信息；
4. 给定某位学生的学号，修改该学生的学生证信息；
5. 给定某位学生的学号，删除该学生的学生证信息；
6. 给定某个班级的班号，显示该班的学生人数。

【提示】

（1）使用类表示学生证信息（学号、姓名、性别、班级号、专业）；

（2）用文件存储学生证信息；

（3）分别定义函数实现上述各个功能；

（4）在 main 函数中调用上述函数进行演示。

1. **开发工具**

我使用的开发工具是DevC++

1. **设计思路**

先创建一个Student类，存储学生证系统的成员变量和成员函数，变量满足题干的基本要求：学号、姓名、性别、班级号、专业。

然后再实现成员函数来满足各项功能，包括：录入学生信息的函数：add()；按学号查询信息函数和按班级查询信息函数：find()；修改函数：modify(); 删除函数：del()；统计函数：statistics(); 我又自己设计添加了一个显示所有数据记录的显示函数：showAll();

我又想实现对记录的排序功能，所有实现了一个小于号重载，然后就可以通过sort()函数进行排序了。

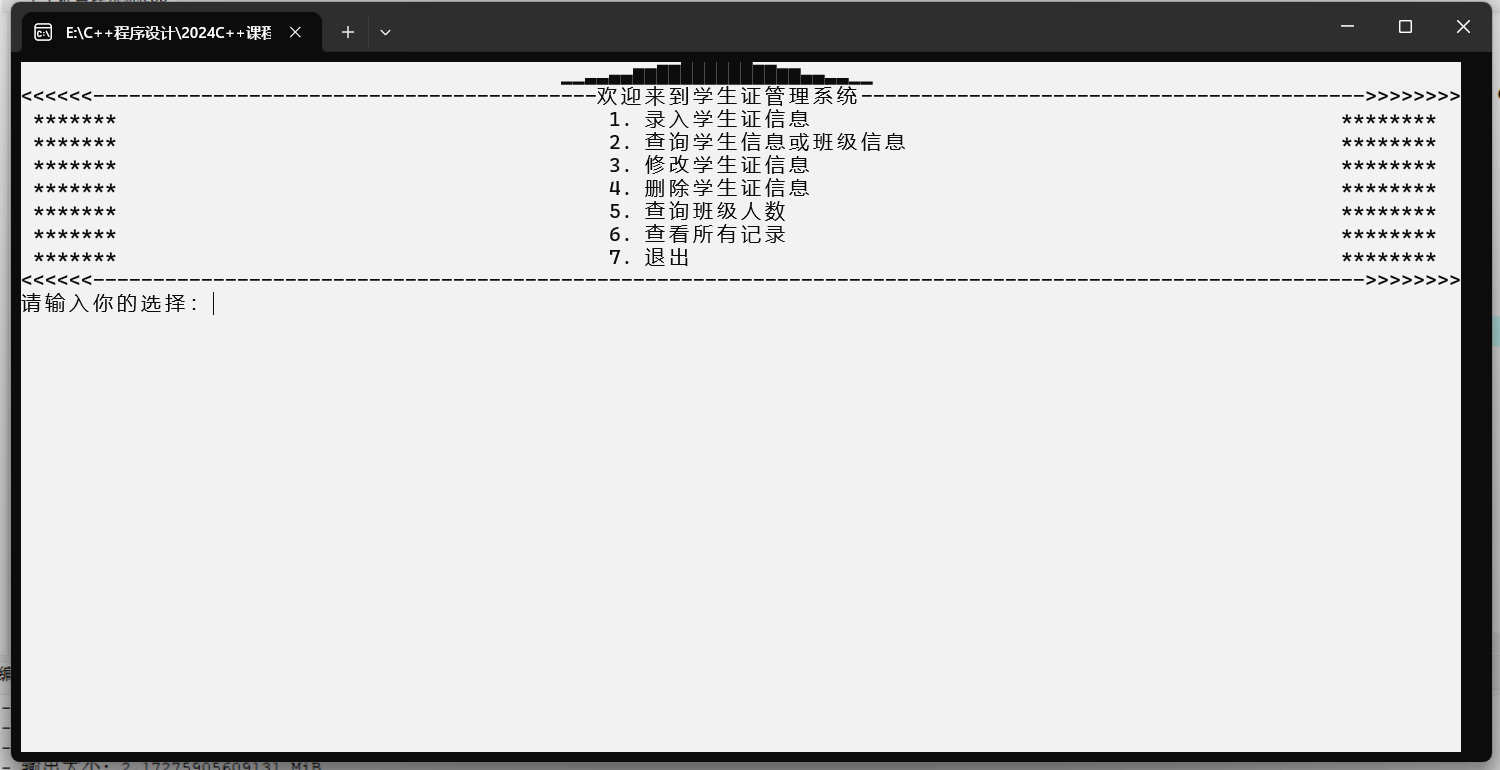
接着是界面的相关函数，我首先设置了界面为白底黑字，然后编写了欢迎界面，接着就会进行到switch-case函数，来进行每个功能的实现和跳转。

使用图示表示可以如下图所示：

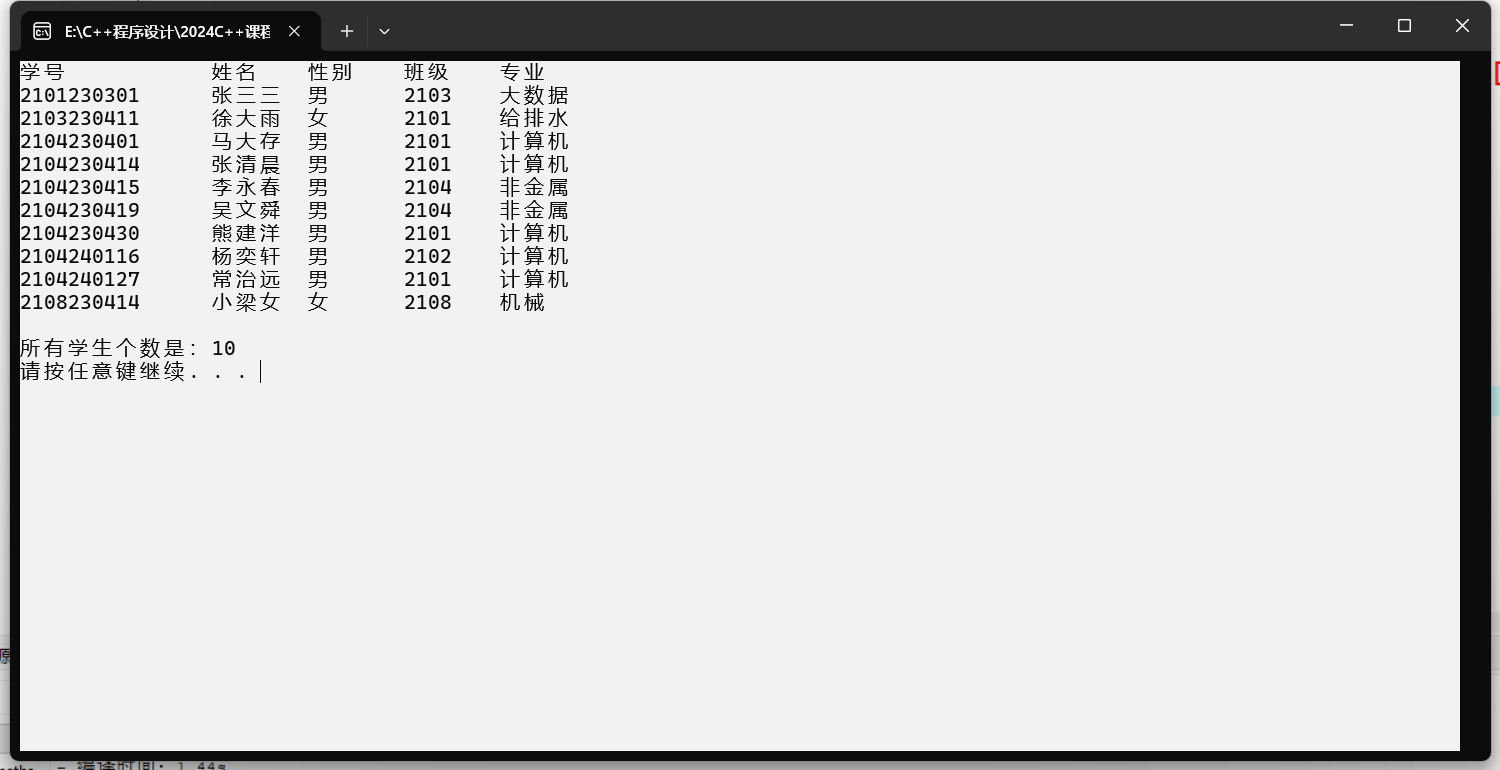
|  |
| --- |
|  |

1. **系统效果图**

首先运行系统，会显示如下界面：



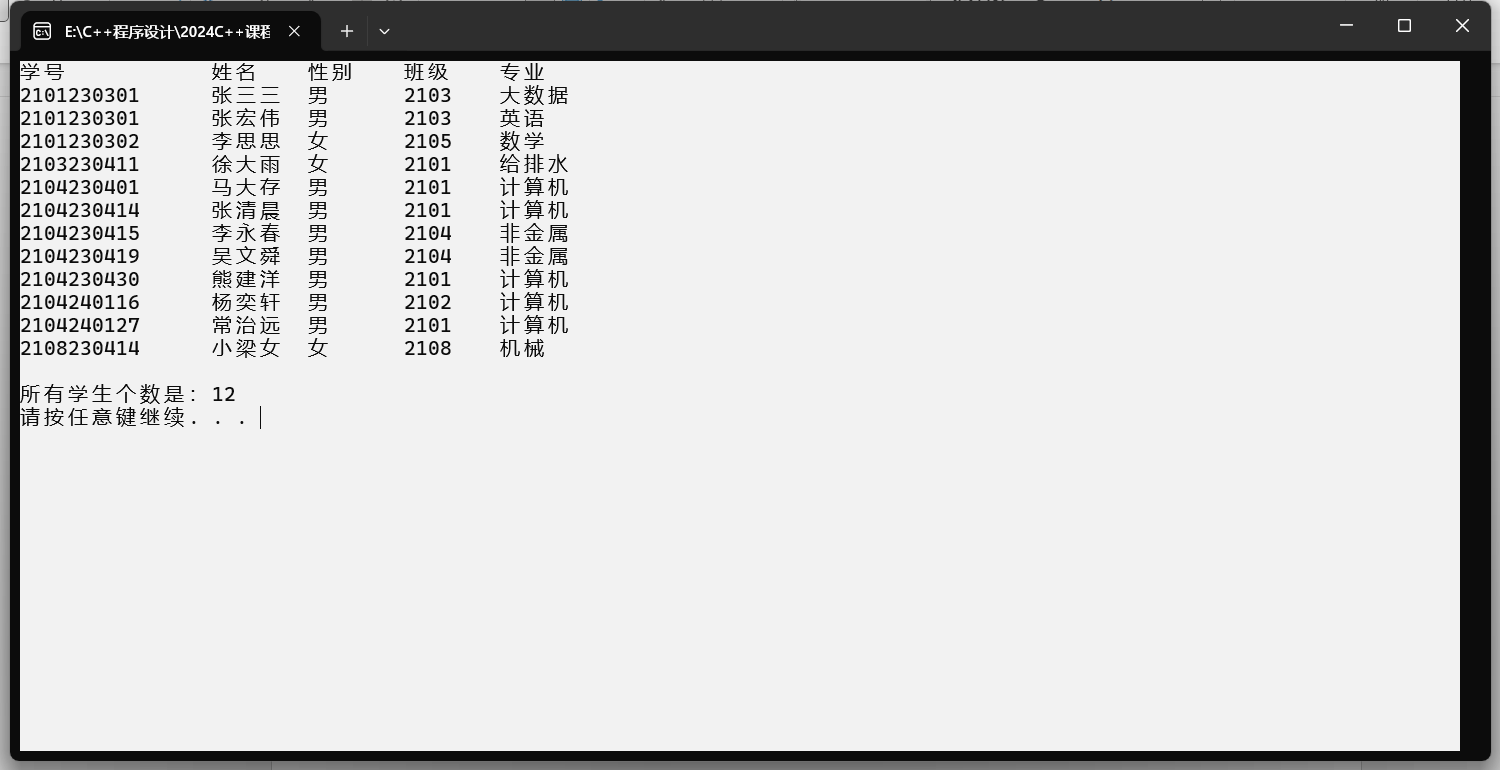
接着选择功能6，查看已经存在txt文件中的所有记录，如下图所示：



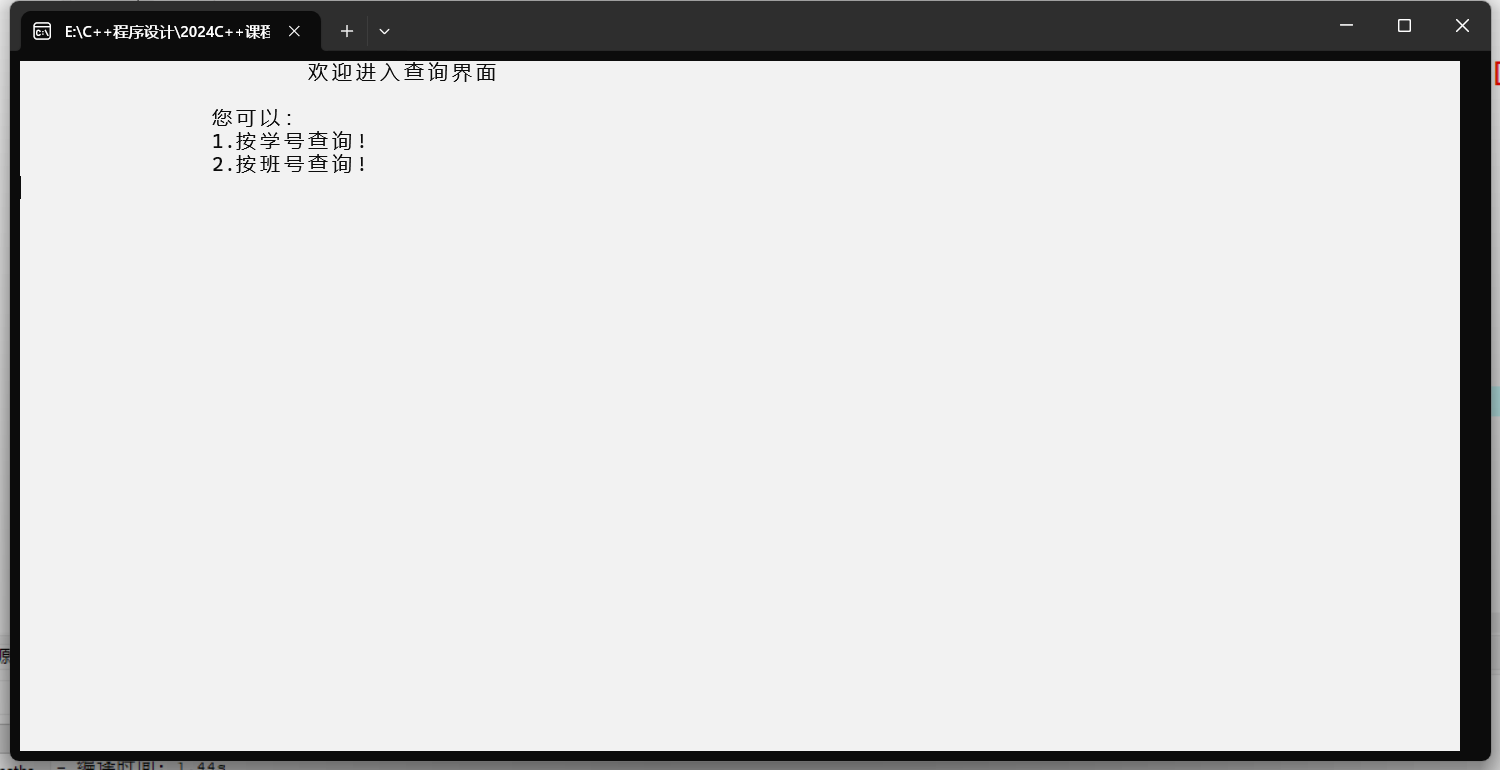
返回到主界面后，选择功能1，进行学生证信息录入，先输入想要录入的学生个数，然后按Y开始录入，一次输入学号、姓名、性别、班级、专业：



输入完之后可以选择时候查看所有记录，输入的两个记录已经添加进去了：



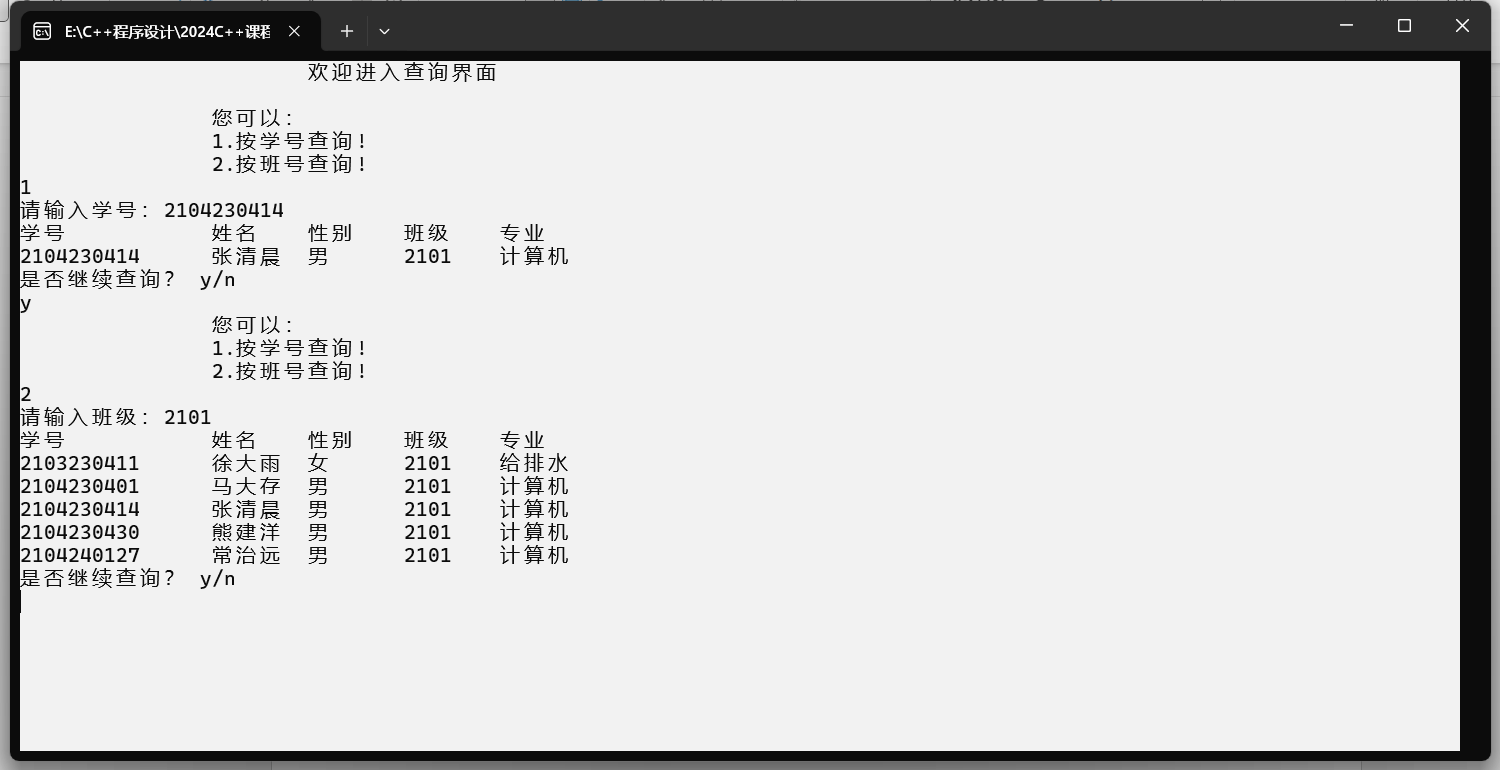
然后是功能2的展示，可以按学号查询信息和按班级查询信息：



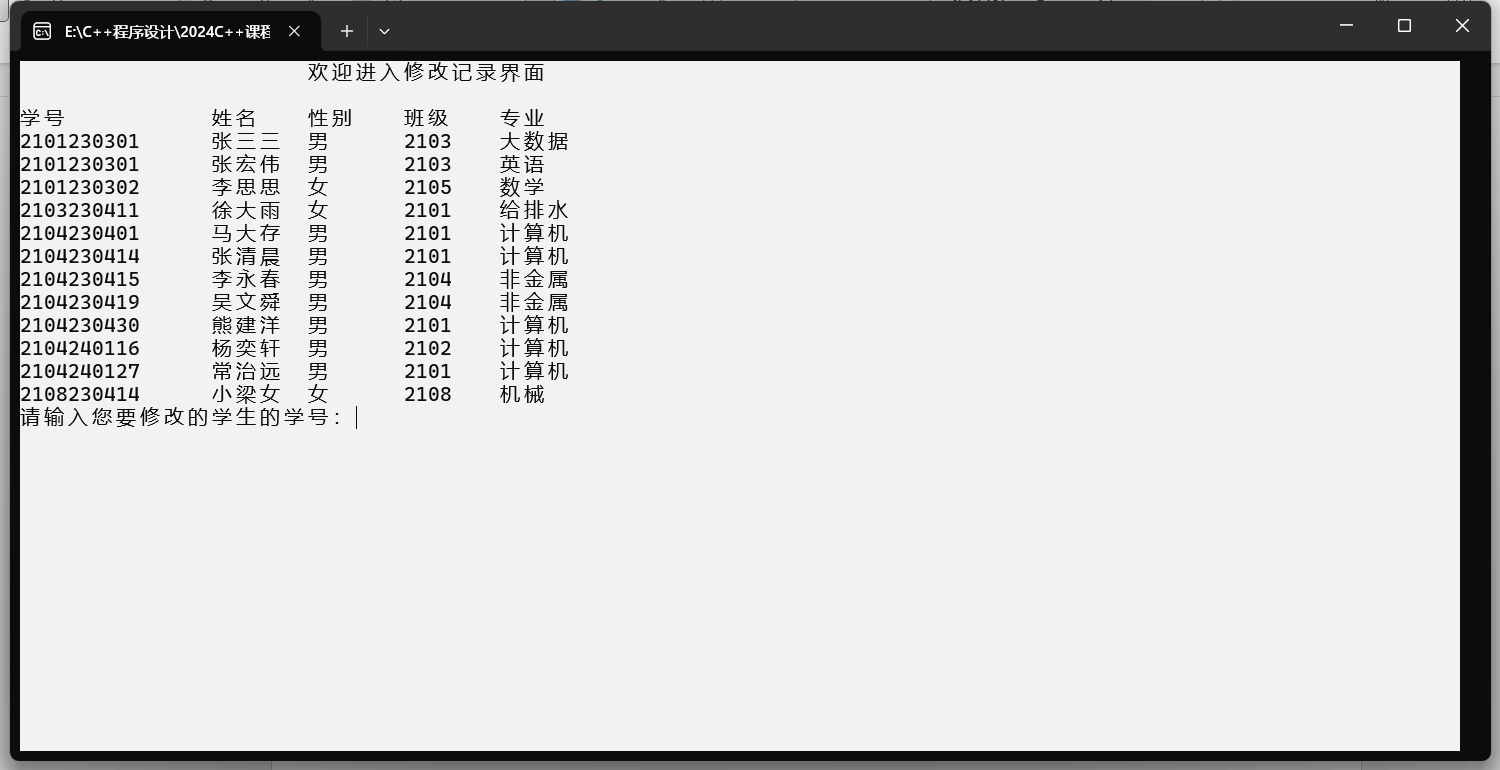
选择1，按学号查询，输入一个学号后，会显示这个记录：



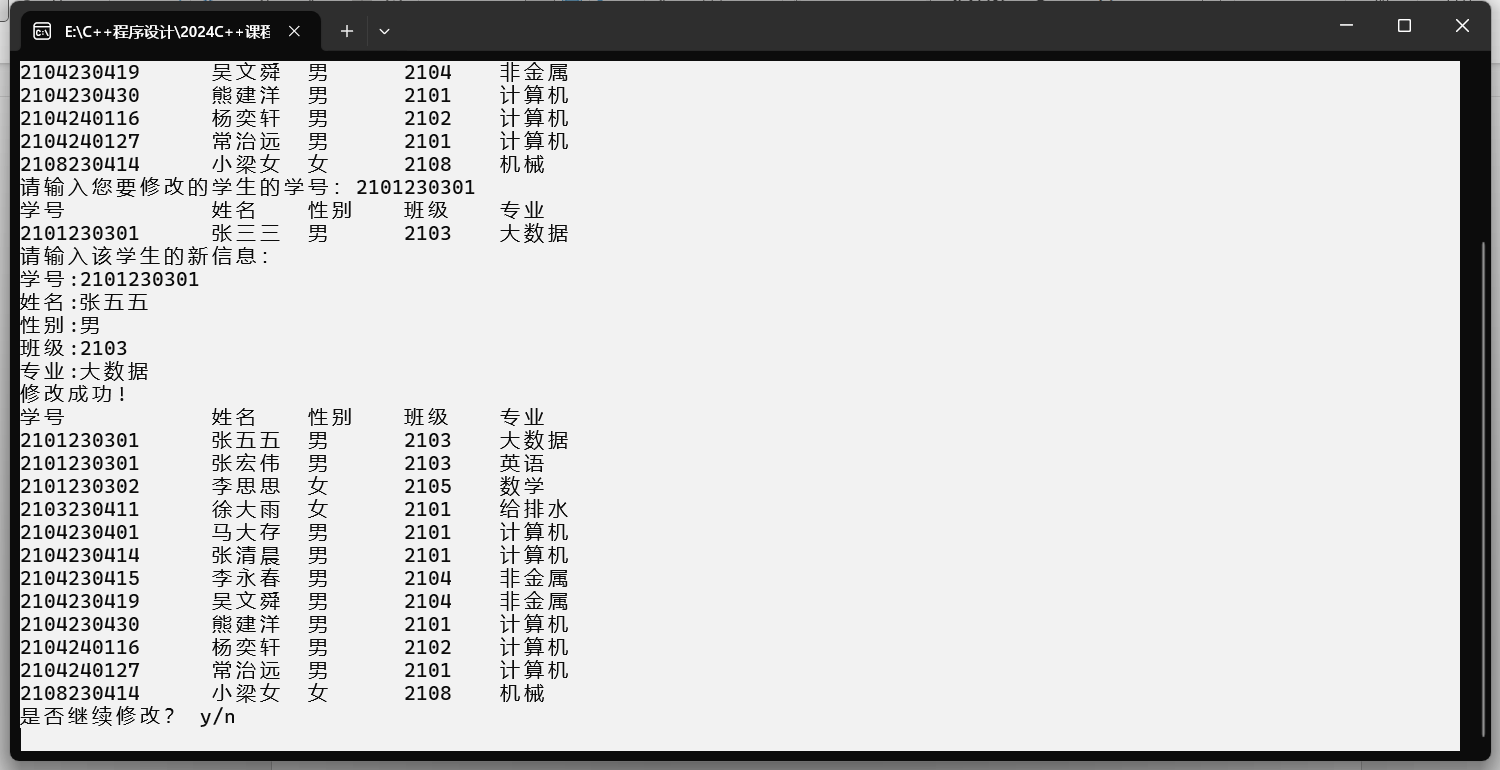
然后点击Y，继续输入，输入2，按班级号查询，输入班级号后回车，会显示这个班级的所有同学：



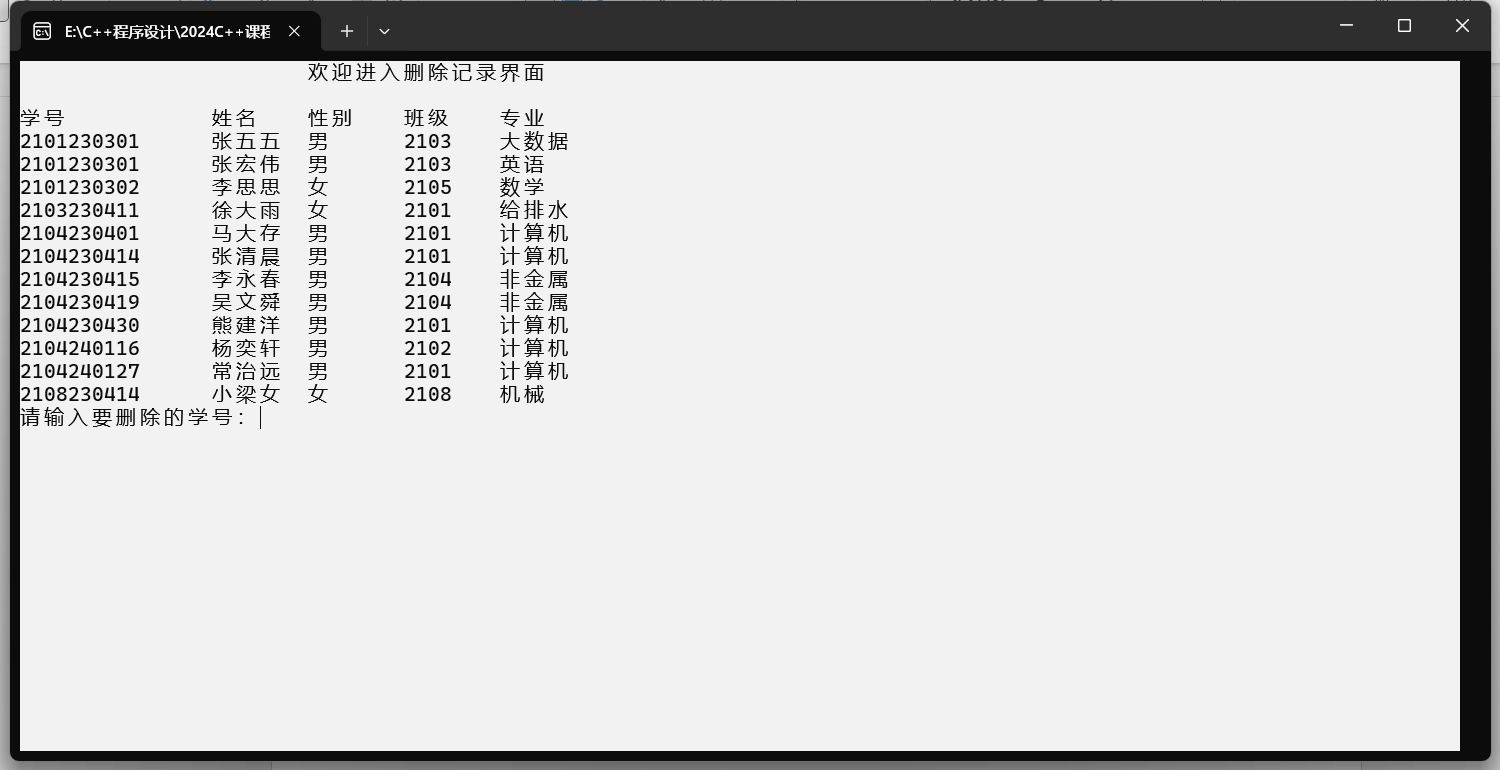
然后是修改功能，进入修改界面后，先会显示所有的学生信息，然后可以输入想要修改的学生学号：



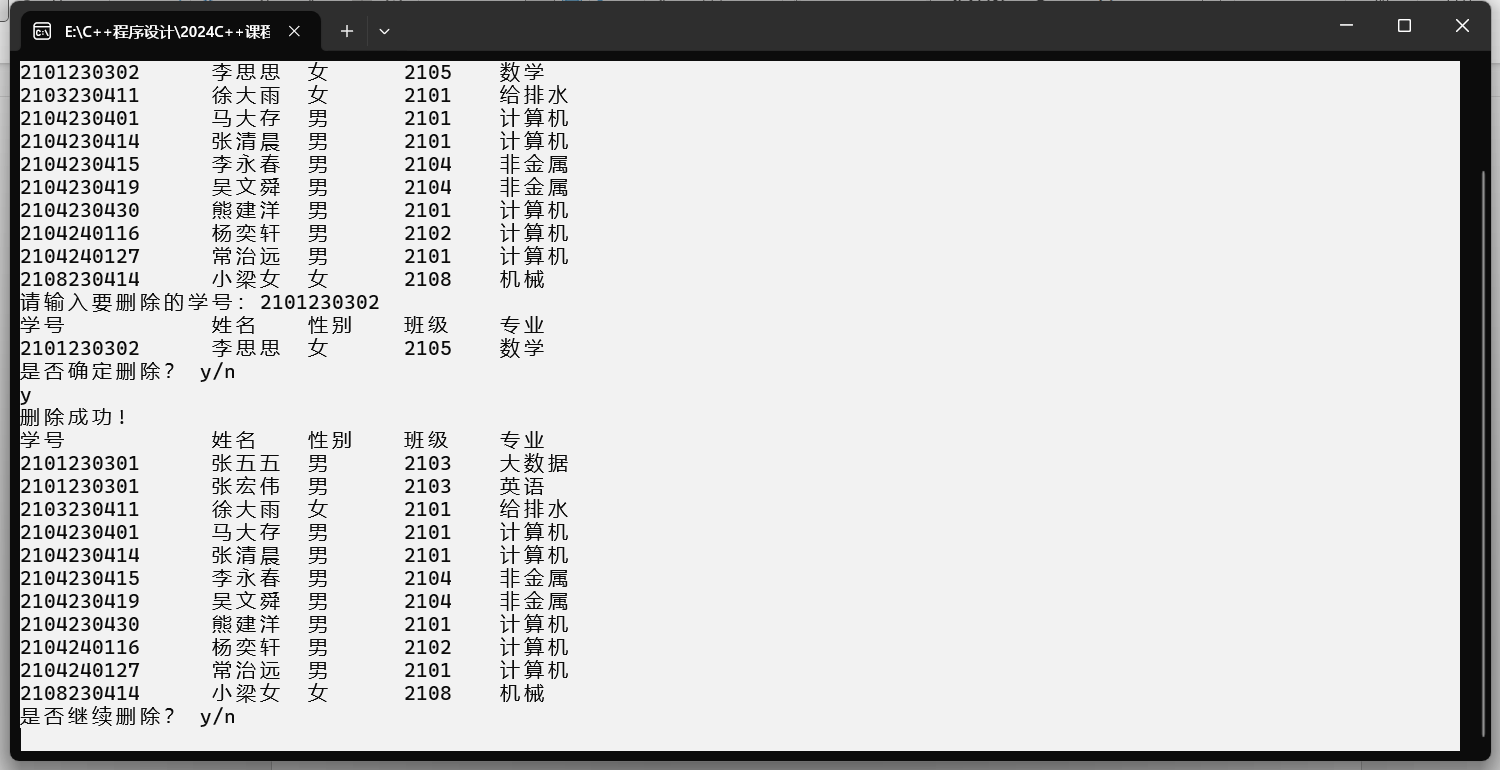
输入学号后，会显示这个学生的具体信息，然后就再输入新的更改后的信息：



接下来是删除功能，进入删除界面后，也是会显示所有的记录，然后输入想要删除记录的学号：



输入后，会显示这个学生的具体信息，然后询问你是否确认删除，输入Y，就删除成功，删除完成后，会显示删除之后，剩下的学生证信息：



接着是统计记录界面，这个可以输入想要统计的班级号，会显示这个班级的学生个数：



1. **程序清单**

|  |
| --- |
| #include <iostream>  #include <cstring>  #include <fstream>  #include <vector>  #include <iomanip> //使用stew函数  #include <algorithm>  using namespace std;  class Student {  public:  int id; // 学号  string name; // 姓名  string sex; // 性别  int classNum; // 班级号  string major; // 专业  // 构造函数  Student(){  id = 0;  name = "";  sex = "";  classNum = 0;  major = "";  }    Student(int id, string name, string sex, int classNum, string major)  : id(id), name(name), sex(sex), classNum(classNum), major(major) {}    bool operator < (const Student& tmp) const  {  return id < tmp.id;  }    void add(); //定义录入学生信息的函数；  void find(); //定义按学号查询信息函数和按班级查询信息函数；  void modify(); //定义修改函数；  void del(); //定义删除函数;  void statistics(); //定义一个统计函数；statistics 统计；  void showAll(); //定义一个显示函数，显示所有记录；  };  void welcome();  void entrance();  void add\_from\_file();  void outfile();  Student stu;  vector<Student> vec;  int sum = 0;  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*欢迎\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  void welcome()  {  system("cls");  cout<<" ▁▁▂▂▃▃▅▅▆▆▇▇▇▇▇▇▆▆▅▅▃▃▂▂▁▁"<<endl;  cout << "<<<<<<------------------------------------------欢迎来到学生证管理系统------------------------------------------>>>>>>>>" << endl;  cout << " \*\*\*\*\*\*\* 1. 录入学生证信息 \*\*\*\*\*\*\*\*" << endl;  cout << " \*\*\*\*\*\*\* 2. 查询学生信息或班级信息 \*\*\*\*\*\*\*\*" << endl;  cout << " \*\*\*\*\*\*\* 3. 修改学生证信息 \*\*\*\*\*\*\*\*" << endl;  cout << " \*\*\*\*\*\*\* 4. 删除学生证信息 \*\*\*\*\*\*\*\*" << endl;  cout << " \*\*\*\*\*\*\* 5. 查询班级人数 \*\*\*\*\*\*\*\*" << endl;  cout << " \*\*\*\*\*\*\* 6. 查看所有记录 \*\*\*\*\*\*\*\*" << endl;  cout << " \*\*\*\*\*\*\* 7. 退出 \*\*\*\*\*\*\*\*" << endl;  cout << "<<<<<<---------------------------------------------------------------------------------------------------------->>>>>>>>" << endl;  cout << "请输入你的选择：";  entrance();  }  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*进入\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  void entrance()  {  int a;  cin >> a;  system("cls");  switch(a)  {    case 1: //录入信息  cout<<"\t\t\t欢迎进入录入信息界面\n\n";  stu.add();  break;  case 2: //给定学号，显示某学生信息和给定班号，显示全班学生  cout<<"\t\t\t欢迎进入查询界面\n\n";  stu.find();  break;  case 3: //进入修改界面  cout<<"\t\t\t欢迎进入修改记录界面\n\n";  stu.modify();  break;  case 4: //进入删除界面  cout<<"\t\t\t欢迎进入删除记录界面\n\n";  stu.del();  break;  case 5: //统计班级人数  cout<<"\t\t\t欢迎进入统计记录界面\n\n";  stu.statistics();  break;  case 6:  stu.showAll();  break;  case 7: //退出  return ;  default:  cout << "输入错误!" << endl;  system("pause");  welcome();  break;  }  }  /\*\*\*\*\*\*\*\*\*\*\*\*\*添加学生信息\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  void Student :: add()  {  int figure;//定义要输入信息的学生的个数  char h,m;  cout<<"你要输入多少个学生的信息? " << endl;  cin >> figure;  cout<<"您确定要输入" << figure << "个学生的信息么？\n";  fflush(stdin);  cout<<"<按Y开始 N返回>" << endl;  cin >> h;  if(h=='Y'||h=='y')  {  for(int j=0;j<figure;j++)  {  Student s;  cout<<"请输入第" << j + 1 << "个学生的信息：\n";  cout<<"学号:";  cin >> s.id;  cout<<"姓名:";  cin >> s.name;  cout << "性别:";  cin >> s.sex;  cout<<"班级:";  cin >> s.classNum;  cout << "专业:";  cin >> s.major;    sum++;  vec.push\_back(s);  }    sort(vec.begin(), vec.end()); // 按学号排序  outfile(); //更新文件    cout<<"添加成功!\n";    cout<<"是否现在查看所有记录？ y/n" << endl;  fflush(stdin);  cin >> m;  if(m=='y'||m=='Y')  {  system("cls");  stu.showAll();  }  else  {  system("cls");  welcome();  }  }  else  {  system("cls");  welcome();  }  }  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*给定学号，显示某位学生学生证信息\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*给定班级，显示全班学生信息\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  void Student :: find()  {  int i, j;  int sno;  int classes;  char t = ' ';    cout<<"\t\t您可以：\n";  cout<<"\t\t1.按学号查询!\n";  cout<<"\t\t2.按班号查询!\n";    int a;  cin >> a;  switch(a)  {  case 1:  cout<<"请输入学号：";  cin >> sno;    for(i=0;i<sum;i++)  {  if(sno == vec[i].id)  {  cout<<"学号\t\t姓名\t性别\t班级\t专业\n";  cout << vec[i].id << "\t" << vec[i].name << "\t" << vec[i].sex << "\t" << vec[i].classNum << "\t" << vec[i].major;  cout << endl;  break;  }  }    if(i == sum)  {  cout<<"没有找到!\n";  }    cout<<"是否继续查询？ y/n\n";  cin >> t;  if (t == 'Y' || t == 'y')  {  stu.find();  }  else  {  welcome();  }  break;  case 2:  cout<<"请输入班级：";  cin >> classes;    for(j = 0; j < sum; j++)  {  if(classes == vec[j].classNum)  {  break;  }  }    if(j == sum)  {  cout<<"没有找到!\n";  }  else  {  cout<<"学号\t\t姓名\t性别\t班级\t专业\n";  for(i=0;i<sum;i++)  {  if(classes == vec[i].classNum)  {  cout << vec[i].id << "\t" << vec[i].name << "\t" << vec[i].sex << "\t" << vec[i].classNum << "\t" << vec[i].major;  cout<<"\n";  }  }  }    cout<<"是否继续查询？ y/n\n";  cin >> t;  if (t == 'Y' || t == 'y')  {  stu.find();  }  else  {  welcome();  }  break;  default:  stu.find();  break;  }  }  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*修改信息\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  void Student :: modify()  {  int i;  int numb;  char h;    cout<<"学号\t\t姓名\t性别\t班级\t专业\n";  for(i=0;i<sum;i++)  {  cout << vec[i].id << "\t" << vec[i].name << "\t" << vec[i].sex << "\t" << vec[i].classNum << "\t" << vec[i].major;  cout<<"\n";  }  printf("请输入您要修改的学生的学号：");    cin >> numb;  for(i=0;i<sum;i++)  {  if(numb == vec[i].id)  {  cout<<"学号\t\t姓名\t性别\t班级\t专业\n";  cout << vec[i].id << "\t" << vec[i].name << "\t" << vec[i].sex << "\t" << vec[i].classNum << "\t" << vec[i].major;  cout << endl;  break;  }  }    if(i == sum)  {  cout<<"没有找到!\n";  cout<<"是否继续修改？ y/n\n";  cin >> h;  if (h == 'Y' || h == 'y')  {  stu.modify();  }  else  {  welcome();  }  }  else  {  cout<<"请输入该学生的新信息：\n";    cout<<"学号:";  cin >> vec[i].id;  cout<<"姓名:";  cin >> vec[i].name;  cout << "性别:";  cin >> vec[i].sex;  cout<<"班级:";  cin >> vec[i].classNum;  cout << "专业:";  cin >> vec[i].major;    sort(vec.begin(), vec.end()); // 按学号排序  outfile(); //更新文件    cout<<"修改成功!\n";  cout<<"学号\t\t姓名\t性别\t班级\t专业\n";  for(i=0;i<sum;i++)  {  if(vec[i].id != 0)  {  cout << vec[i].id << "\t" << vec[i].name << "\t" << vec[i].sex << "\t" << vec[i].classNum << "\t" << vec[i].major;  cout<<"\n";  }  }    cout<<"是否继续修改？ y/n\n";  cin >> h;  if (h == 'Y' || h == 'y')  {  stu.modify();  }  else  {  welcome();  }  }  }  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*删除记录\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  void Student :: del()  {  int i;  int numb;  char h,k;    cout<<"学号\t\t姓名\t性别\t班级\t专业\n";  for(i=0;i<sum;i++)  {  if(vec[i].id != 0)  {  cout << vec[i].id << "\t" << vec[i].name << "\t" << vec[i].sex << "\t" << vec[i].classNum << "\t" << vec[i].major;  cout<<"\n";  }  }    cout<<"请输入要删除的学号：";  fflush(stdin);  cin >> numb;    for(i=0;i<sum;i++)  {  if(numb == vec[i].id)  {  cout<<"学号\t\t姓名\t性别\t班级\t专业\n";  cout << vec[i].id << "\t" << vec[i].name << "\t" << vec[i].sex << "\t" << vec[i].classNum << "\t" << vec[i].major;  cout<<"\n";  break;  }  }    if(i == sum)  {  cout<<"没有找到!\n";  cout<<"是否继续删除？ y/n\n";  cin >> h;  if (h == 'Y' || h == 'y')  {  stu.del();  }  else  {  welcome();  }  }  else  {  cout<<"是否确定删除？ y/n\n";  fflush(stdin);  cin >> h;    if(h =='y'||h =='Y')  {  for (auto it = vec.begin(); it != vec.end(); it++)  {  if (it->id == vec[i].id)  {  it = vec.erase(it);  break;  }  }  sum--;    cout<<"删除成功!\n";    sort(vec.begin(), vec.end()); // 按学号排序  outfile(); //更新文件    cout<<"学号\t\t姓名\t性别\t班级\t专业\n";  for(i=0;i<sum;i++)  {  if(vec[i].id != 0)  {  cout << vec[i].id << "\t" << vec[i].name << "\t" << vec[i].sex << "\t" << vec[i].classNum << "\t" << vec[i].major;  cout<<"\n";  }  }    cout<<"是否继续删除？ y/n\n";  cin >> k;  if (k == 'Y' || k == 'y')  {  stu.del();  }  else  {  welcome();  }  }  else  {  del();  }  }  }  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*统计某个班的学生人数\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  void Student :: statistics()  {  int count = 0;  int s;  cout << "请输入要统计的班级号：";  cin >> s;    for (int i = 0; i < sum; i++)  {  if (s == vec[i].classNum) count++;  }    cout << s << "班的学生个数是：" << count << endl;    cout<<"是否继续统计？ y/n\n";  char k;  cin >> k;  if (k == 'Y' || k == 'y')  {  stu.statistics();  }  else  {  welcome();  }  }  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*显示所有的学生证信息\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  void Student :: showAll()  {  if (sum)  {  cout<<"学号\t\t姓名\t性别\t班级\t专业\n";  for(int i=0;i<sum;i++)  {  cout << vec[i].id << "\t" << vec[i].name << "\t" << vec[i].sex << "\t" << vec[i].classNum << "\t" << vec[i].major;  cout<<"\n";  }  cout << endl;  }  cout << "所有学生个数是：" << sum << endl;  system("pause");  welcome();  }  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*将文件中所有的学生证信息加入到vector对象数组中\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  void add\_from\_file()  {  int id, classNum;  string name, sex, major;  vec.clear();    ifstream in("学生证管理系统.txt");  if (!in.is\_open())  {  cerr << "无法打开文件！不能正常写入数组" << endl;  return ;  }    string str;  getline(in, str); //只读取第一行数据（汉字），不进行操作    while (in >> id >> name >> sex >> classNum >> major)  {  // in >> id >> name >> sex >> classNum >> major;  Student s(id, name, sex, classNum, major);  vec.push\_back(s);  sum++;  }    in.close();    sort(vec.begin(), vec.end()); // 按学号排序  }  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*将所有的学生证信息重新写入到txt文件中\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  void outfile()  {  ofstream out("学生证管理系统.txt");  if (!out.is\_open())  {  cerr << "无法打开文件！不能正常写入文件" << endl;  return ;  }    out << "学号" << setw(21) << "姓名" << setw(15) << "性别" << setw(9) << "班级号" << setw(5) << "专业" << endl;  for (int i = 0; i < vec.size(); i++)  {  out << vec[i].id << setw(10) << vec[i].name << setw(10) << vec[i].sex << setw(10) << vec[i].classNum << setw(10) << vec[i].major << endl;  }    out.close();  }  int main()  {  add\_from\_file();  welcome(); //进入欢迎界面；    return 0;  } |

**源代码共计：546行**

1. **设计体会**

通过这次的C++课程设计，让我巩固了所学习的书本知识，也更有信心能够完成之后的学习，这门课程我是补修完成的，但是我也认真努力学习，了解了C++面向对象的思想，并且最后设计出了这个程序，这个程序的最初原型是C语言写的，然后我在它的基础上进行修改，将其改为C++类型，完善了类，构造函数，创建对象等。在完成老师题目的要求基础上，我也增添了一些功能，我认为这次课程设计对我提升还是很大的。

1. **参考文献**

1、《C++大学教程》，Harvey M.Deitel等著，邱仲潘等译，电子工业出版社 2001

2、《C++编程思想》，Bruce Eckel著，刘宗田等译，机械工业出版社 2002

3、《C++精粹》，Ira Pohl著，王树武等译，机械工业出版社 2003

4、《数据结构》，严蔚敏编，清华大学出版社 2004

1. **教师评语**

|  |
| --- |
|  |