**数据结构课程设计**

**题 目: 学生成绩管理**

**班 级：**

**学 生 姓 名：**

**学 生 学 号：**

**2022 年 2月 25日**

目录

[1. 开发环境 3](#_Toc477208932)

[2. 程序的功能 3](#_Toc477208933)

[3. 代码清单 3](#_Toc477208934)

[4. 运行界面 16](#_Toc477208935)

开发环境

IDE：Visual Studio 2022，开发包：Visual C++ 143，Windows SDK 10.0.22000

本程序使用了第三方库easyx（[EasyX Graphics Library for C++](https://easyx.cn/)），在此鸣谢原作者

## 功能描述

本程序用于管理学员道路驾驶技能考试成绩，可以添加学员，录入修改成绩（科目一至科目三），学员毕业（删除学员），适用于驾校教练或交管部门管理学生模拟考试或正式考试成绩。

1. 输入功能：可在用户界面输入数据保存到数据集合（一个带头节点的单链表）中
2. 显示功能：完成全部记录的显示。
3. 查找功能：完成按姓名查找学员信息功能，并显示。
4. 编辑功能：对指定的学员的各科目考试成绩进行编辑，更改数据源中的值。
5. 删除功能：按行号选中学生，将其从数据源中删除。

## 代码清单

本程序采用结构化设计，分为数据实体，dao层，服务层，表示层几大部分。

数据实体：

|  |
| --- |
| Student.h |
| /\*\*  \* 作业1：考试成绩管理软件 - 学生实体定义  \* @author Steve  \* @version 1.0.0  \* @targetSDK 10.0.19041-22000  \* wirtten in std C99  \*  \* NOTE: Only for X64 systems, please compile this into X64 arch!  \*/  #pragma once  #define \_CRT\_SECURE\_NO\_WARNINGS  typedef struct Student {  char name[16];  int grade[3];  }Student;//for compatibility with compiliers lower than C99, here we use typedef. all following structs are same  Student createStudent();  Student createStudent(const char\* name, int grade\_1, int grade\_2, int grade\_3);  char\* getName(Student\* s);  int getGrade(Student\* s,int index);  void setName(Student\* s, const char\* name);  void setGrade(Student\* s, int index, int grade);  char\* toString(Student\* s); |

|  |
| --- |
| Student.cpp |
| /\*\*  \* 作业1：考试成绩管理软件 - 学生实体实现  \* @author Steve  \* @version 1.0.0  \* @targetSDK 10.0.19041-22000  \* wirtten in std C99  \*  \* NOTE: Only for X64 systems, please compile this into X64 arch!  \*/  #define \_CRT\_SECURE\_NO\_WARNINGS  #include "Student.h"  #include <string.h>  #include <stdlib.h>  #include <stdio.h>  Student createStudent()  {  Student s;  strcpy(s.name,"");  for (int i = 0; i < 3; i++)  {  s.grade[i] = 0;  }  return s;  }  Student createStudent(const char\* name, int grade\_1, int grade\_2, int grade\_3)  {  Student s;  strcpy(s.name, name);  s.grade[0] = grade\_1;  s.grade[1] = grade\_2;  s.grade[2] = grade\_3;  return s;  }  char\* getName(Student\* s)  {  char\* name = (char\*)calloc(16, sizeof(char));  strcpy(name, s->name);  return name;  }  int getGrade(Student\* s, int index)  {  return (s->grade)[index];  }  void setName(Student\* s, const char\* name)  {  strcpy(s->name, name);  }  void setGrade(Student\* s, int index, int grade)  {  (s->grade)[index] = grade;  }  char\* toString(Student\* s)  {  char\* st = (char\*)calloc(128, sizeof(char));  sprintf(st, "姓名: %s, 科目一: %d, 科目二: %d, 科目三: %d",  s->name,  (s->grade)[0],  (s->grade)[1],  (s->grade)[2]);  return st;  } |

Dao层：

|  |
| --- |
| StudentList.h |
| /\*\*  \* 作业1：考试成绩管理软件 - 学生列表类定义  \* @author Steve  \* @version 1.0.0  \* @targetSDK 10.0.19041-22000  \* wirtten in std C99  \*  \* NOTE: Only for X64 systems, please compile this into X64 arch!  \*/  #pragma once  #define \_CRT\_SECURE\_NO\_WARNINGS  #include "student.h"  #include <stddef.h> //VC++ needs this  typedef struct StudentListNode {  Student student;  struct StudentListNode\* next;  }StudentListNode;  typedef StudentListNode\* SLPtr;  typedef struct StudentList {  StudentListNode\* head = NULL;  int count = 0;  }StudentList;//a linkedlist with head node(will be created in its constructor)  StudentList createStudentList();  bool addStudent(StudentList\* list, Student student, int index);  bool removeStudent(StudentList\* list, int index);  Student\* getStudent(StudentList\* list, int index);  int length(StudentList\* list); |

|  |
| --- |
| StudentList.cpp |
| /\*\*  \* 作业1：考试成绩管理软件 - 学生列表类实现  \* @author Steve  \* @version 1.0.0  \* @targetSDK 10.0.19041-22000  \* wirtten in std C99  \*  \* NOTE: Only for X64 systems, please compile this into X64 arch!  \*/  #define \_CRT\_SECURE\_NO\_WARNINGS  #include "StudentList.h"  #include <stdlib.h>  #include <stddef.h>  StudentList createStudentList()  {  StudentList list;  list.count = 0;  list.head = (StudentListNode\*)malloc(sizeof(StudentListNode));  return list;  }  bool addStudent(StudentList\* list, Student student, int index)  {  if (!list || index<0 || index>list->count) { return false; }  SLPtr p = list->head;  for (int i = 0; i < index; i++)  {  p = p->next;  }  SLPtr nwnode = (SLPtr)malloc(sizeof(StudentListNode));  if (!nwnode) { return false; }  nwnode->student = student;  nwnode->next = p->next;  p->next = nwnode;  list->count++;  return true;  }  bool removeStudent(StudentList\* list, int index)  {  if (!list || index < 0 || index >= list->count) { return false; }  SLPtr p = list->head;  for (int i = 0; i < index; i++)  {  p = p->next;  }  SLPtr q = p->next;  p->next = q->next;  free(q);  list->count--;  return true;  }  Student\* getStudent(StudentList\* list, int index)  {  if (!list || list->count == 0 || index < 0 || index >= list->count) { return NULL; }  SLPtr p = list->head->next;  for (int i = 0; i < index; i++)  {  p = p->next;  }  return &(p->student);  }  int length(StudentList\* list)  {  return list->count;  } |

服务层：

|  |
| --- |
| Modules.h |
| /\*\*  \* 作业1：考试成绩管理软件 - 操作模块定义  \* @author Steve  \* @version 1.0.0  \* @targetSDK 10.0.19041-22000  \* wirtten in std C99  \*  \* NOTE: Only for X64 systems, please compile this into X64 arch!  \*/  #pragma once  #define \_CRT\_SECURE\_NO\_WARNINGS  #include <stddef.h>  #include "StudentList.h"  bool initModule();  bool insertStudent(int index, const char\* name, int grade\_1, int grade\_2, int grade\_3);  bool deleteStudent(int index);  Student\* queryForName(const char\* name);  Student\* queryForIndex(int index);  void queryAllAndPrint(); |

|  |
| --- |
| Modules.cpp |
| /\*\*  \* 作业1：考试成绩管理软件 - 操作模块实现  \* @author Steve  \* @version 1.0.0  \* @targetSDK 10.0.19041-22000  \* wirtten in std C99  \*  \* NOTE: Only for X64 systems, please compile this into X64 arch!  \*/  #define \_CRT\_SECURE\_NO\_WARNINGS  #include <stdio.h>  #include <stdlib.h>  #include <string.h>  #include "Modules.h"  StudentList dataList;  bool initModule()  {  dataList = createStudentList();  if (dataList.head == NULL) { return false; }  return true;  }  bool insertStudent(int index, const char\* name, int grade\_1, int grade\_2, int grade\_3)  {  Student s = createStudent(name, grade\_1, grade\_2, grade\_3);  return addStudent(&dataList, s, index);  }  bool deleteStudent(int index)  {  return removeStudent(&dataList,index);  }  Student\* queryForName(const char\* name)  {  if (length(&dataList) == 0) { return NULL; }  for (int i = 0; i < length(&dataList); i++)  {  if (strcmp(name, getStudent(&dataList, i)->name) == 0)  {  return getStudent(&dataList, i);  }  }  return NULL;  }  Student\* queryForIndex(int index)  {  return getStudent(&dataList, index);  }  void queryAllAndPrint()  {  printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*【学员成绩报表】\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n\n");  for (int i = 0; i < length(&dataList); i++)  {  printf("%02d\t%s\n", i, toString(getStudent(&dataList, i)));  }  printf("\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");  } |

表示层：

|  |
| --- |
| Menus.h |
| /\*\*  \* 作业1：考试成绩管理软件 - 菜单定义  \* @author Steve  \* @version 1.0.0  \* @targetSDK 10.0.19041-22000  \* wirtten in std C99  \*  \* NOTE: Only for X64 systems, please compile this into X64 arch!  \*/  #pragma once  #define \_CRT\_SECURE\_NO\_WARNINGS  #include "Modules.h"  void showStudentByName();  void addStudentAt();  void modifyStudentAt();  void deleteStudentAt();  void printStudents(); |

|  |
| --- |
| Menus.cpp |
| /\*\*  \* 作业1：考试成绩管理软件 - 菜单实现  \* @author Steve  \* @version 1.0.0  \* @targetSDK 10.0.19041-22000  \* wirtten in std C99  \*  \* NOTE: Only for X64 systems, please compile this into X64 arch!  \*/  #define \_CRT\_SECURE\_NO\_WARNINGS  #include <stdio.h>  #include <stdlib.h>  #include "Menus.h"  void showStudentByName()  {  rewind(stdin);  printf("请输入学生姓名：");  char a[16];  scanf("%s", a);  rewind(stdin);  Student\* s=queryForName(a);  printf("=================================\n");  printf("%s\n", toString(s));  printf("=================================\n");  system("pause");  rewind(stdin);  }  void addStudentAt()  {  rewind(stdin);  printf("请输入姓名，科目一成绩，科目二成绩，科目三成绩，插入位置，每输入完一项请按一下回车换行\n");  char name[16];  int grade\_1;  int grade\_2;  int grade\_3;  int index;  scanf("%s", name);  rewind(stdin);  scanf("%d", &grade\_1);  rewind(stdin);  scanf("%d", &grade\_2);  rewind(stdin);  scanf("%d", &grade\_3);  rewind(stdin);  scanf("%d", &index);  rewind(stdin);  if (insertStudent(index, name, grade\_1, grade\_2, grade\_3))  {  printf("插入成功！\n");  }  else  {  printf("插入失败，原因是成员下标超过阈值或内存不足\n");  }  system("pause");  }  void modifyStudentAt()  {  rewind(stdin);  printf("请输入学员编号（以当前屏幕显示为准），科目名称（1=科目一，2=科目二，3=科目三），修改后成绩，每输入完一项请按一下回车换行");  int index, kemu, grade;  scanf("%d", &index);  rewind(stdin);  scanf("%d", &kemu);  rewind(stdin);  scanf("%d", &grade);  rewind(stdin);  Student\* s=queryForIndex(index);  if (s)  {  setGrade(s, kemu - 1, grade);  printf("修改成功！");  }  else  {  printf("查无此人\n");  }  system("pause");  }  void deleteStudentAt()  {  rewind(stdin);  printf("请输入学员编号（以当前屏幕显示为准）\n");  int index;  scanf("%d", &index);  rewind(stdin);  if (deleteStudent(index))  {  printf("删除成功\n");  }  else  {  printf("删除失败，查无此人\n");  }  system("pause");  }  void printStudents()  {  printf("|当前学员详情（学员编号实时变化，请以此表为准） |\n");  printf("============================================\n");  queryAllAndPrint();  printf("============================================\n");  } |

|  |
| --- |
| Ad.h |
| /\*\*  \* 作业1：考试成绩管理软件 - 图形模块定义  \* @author Steve  \* @version 1.0.0  \* @targetSDK 10.0.19041-22000  \* wirtten in std C99  \*  \* NOTE: Only for X64 systems, please compile this into X64 arch!  \*/  #pragma once  #define \_CRT\_SECURE\_NO\_WARNINGS  void startUp(); |

|  |
| --- |
| Ad.cpp |
| /\*\*  \* 作业1：考试成绩管理软件 - 图像模块实现  \* @author Steve  \* @version 1.0.0  \* @targetSDK 10.0.19041-22000  \* wirtten in std C99  \*  \* NOTE: Only for X64 systems, please compile this into X64 arch!  \*/  #define \_CRT\_SECURE\_NO\_WARNINGS  #include "Ad.h"  #include <stdio.h>  #include <windows.h>  #include "graphics.h"  #include "easyx.h"  char str[100] = ""; //这里必须定义全局变量  char\* NumToChar(int num)  {  sprintf(str, "广告剩余%d秒", num); //格式转换，将整数打印在字符串里；  return str;  }  void startUp()//从以前代码山里面扒出来的。。。  {  IMAGE img;  int i = 5;  loadimage(&img, "./data/AD.jpg", 800, 400);  initgraph(800, 400, SHOWCONSOLE);  for (i = 5; i >= 0; i--)  {  putimage(0, 0, &img);  setbkcolor(RGB(131, 139, 139));  settextstyle(20, 10, "黑体");  outtextxy(500, 10, NumToChar(i));  Sleep(1000);  }  closegraph();  return;  } |

程序启动器：

|  |
| --- |
| App.cpp |
| // 作业1-考试成绩管理系统.cpp : 此文件包含 "main" 函数。程序执行将在此处开始并结束。  //  /\*\*  \* 作业1：考试成绩管理软件 - 应用程序启动器  \* @author Steve  \* @version 1.0.0  \* @targetSDK 10.0.19041-22000  \* wirtten in std C99  \*  \* NOTE: Only for X64 systems, please compile this into X64 arch!  \*/  #define \_CRT\_SECURE\_NO\_WARNINGS  #include <stdio.h>  #include <stdlib.h>  #include "Menus.h"  #include "Ad.h"  int main()  {  system("title 智能教练 - 驾考学员成绩管理系统");  initModule();  startUp();  while (true)  {  system("cls");  printStudents();  printf("\n");  printf("请输入操作类型（1=按名查看学员信息，2=添加学员，3=修改学员成绩，4=学员毕业（删除相关记录））：");  int optype;  scanf("%d", &optype);  rewind(stdin);  switch (optype)  {  case 1: {  showStudentByName();  break;  }  case 2: {  addStudentAt();  break;  }  case 3: {  modifyStudentAt();  break;  }  case 4: {  deleteStudentAt();  break;  }  default: {  return 0;  }  }  }  }  // 运行程序: Ctrl + F5 或调试 >“开始执行(不调试)”菜单  // 调试程序: F5 或调试 >“开始调试”菜单  // 入门使用技巧:  // 1. 使用解决方案资源管理器窗口添加/管理文件  // 2. 使用团队资源管理器窗口连接到源代码管理  // 3. 使用输出窗口查看生成输出和其他消息  // 4. 使用错误列表窗口查看错误  // 5. 转到“项目”>“添加新项”以创建新的代码文件，或转到“项目”>“添加现有项”以将现有代码文件添加到项目  // 6. 将来，若要再次打开此项目，请转到“文件”>“打开”>“项目”并选择 .sln 文件 |

## 运行界面

本程序无需安装，双击打开App.exe后，屏幕显示启动画面，5秒后进入程序

图形用户界面, 文本, 应用程序

描述已自动生成

屏幕实时呈现当前学员列表，下方通过输入对应编号进入相应操作

图形用户界面

描述已自动生成

添加学员：

图形用户界面

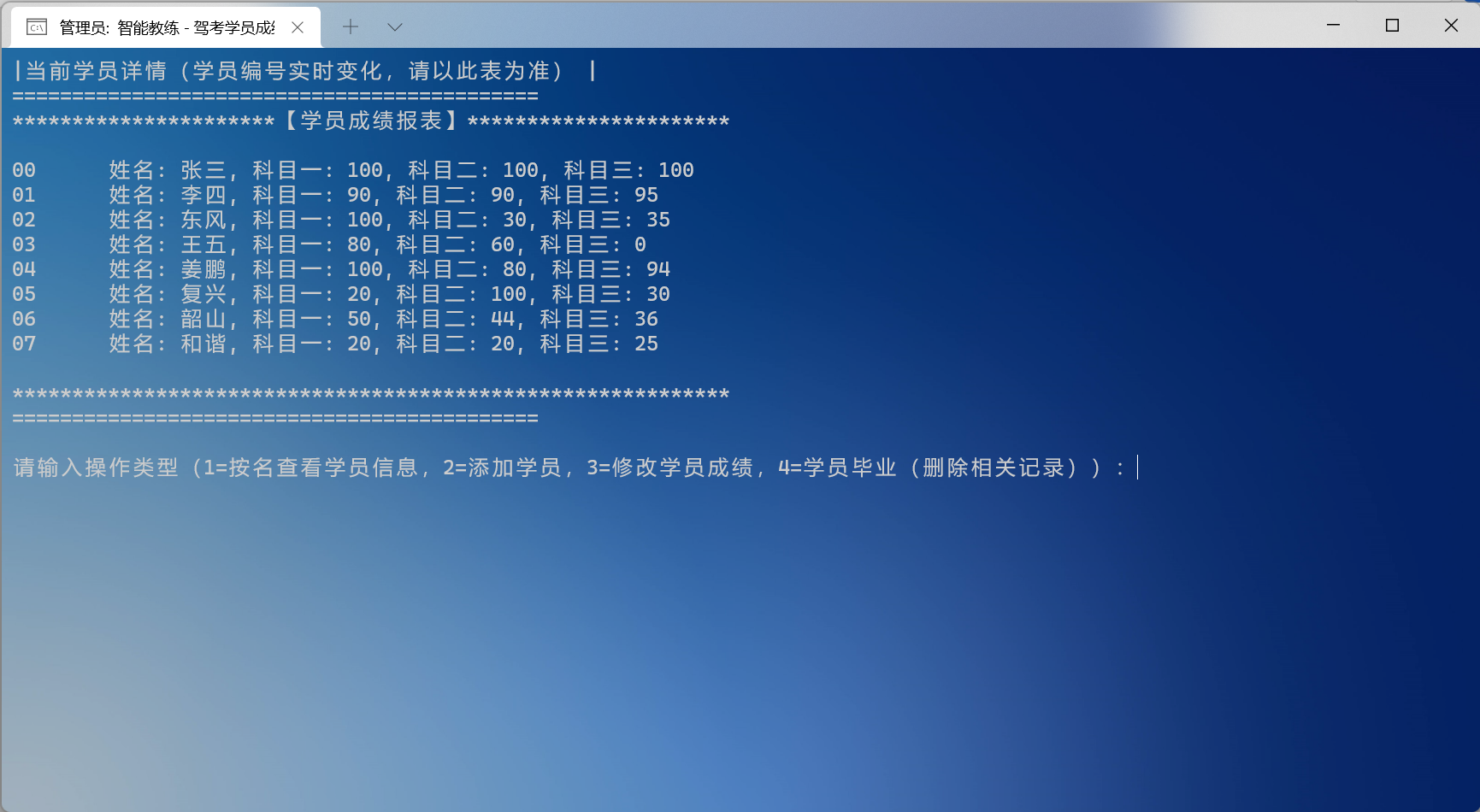
描述已自动生成

添加后屏幕上已经显示新学员了

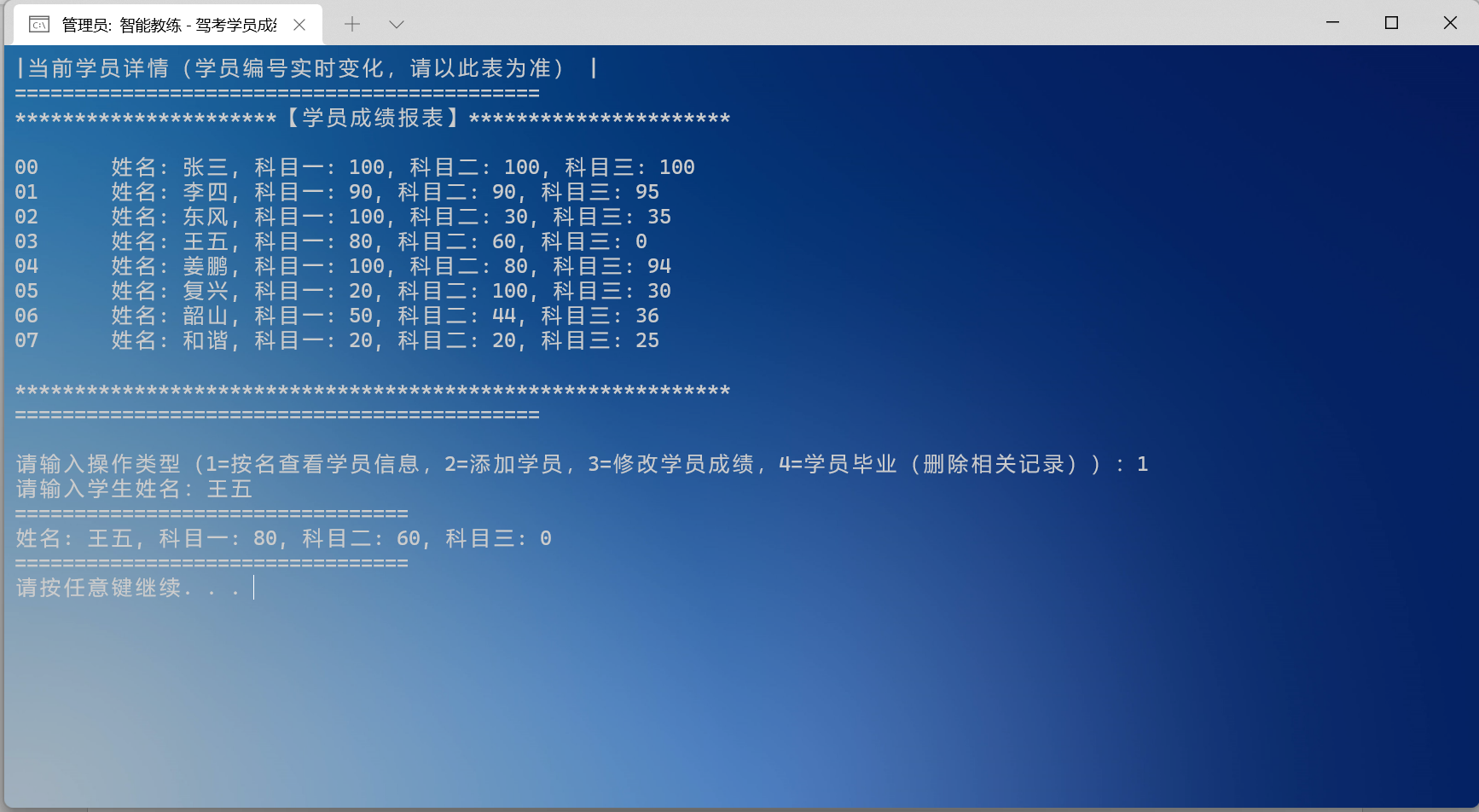
图形用户界面

描述已自动生成

多添加几条



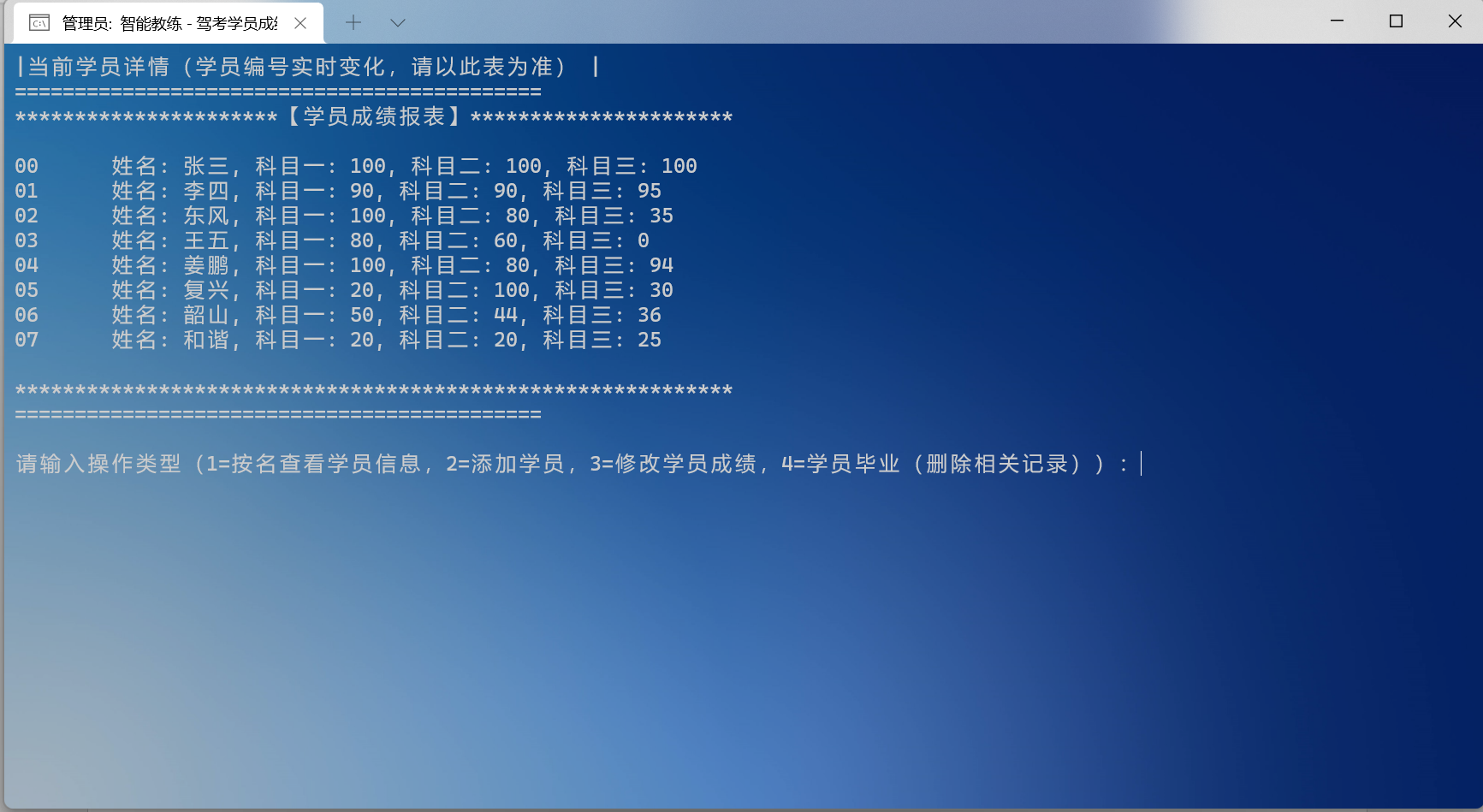
查看“王五“的详细信息：

****

将“东风“的科目二成绩改合格

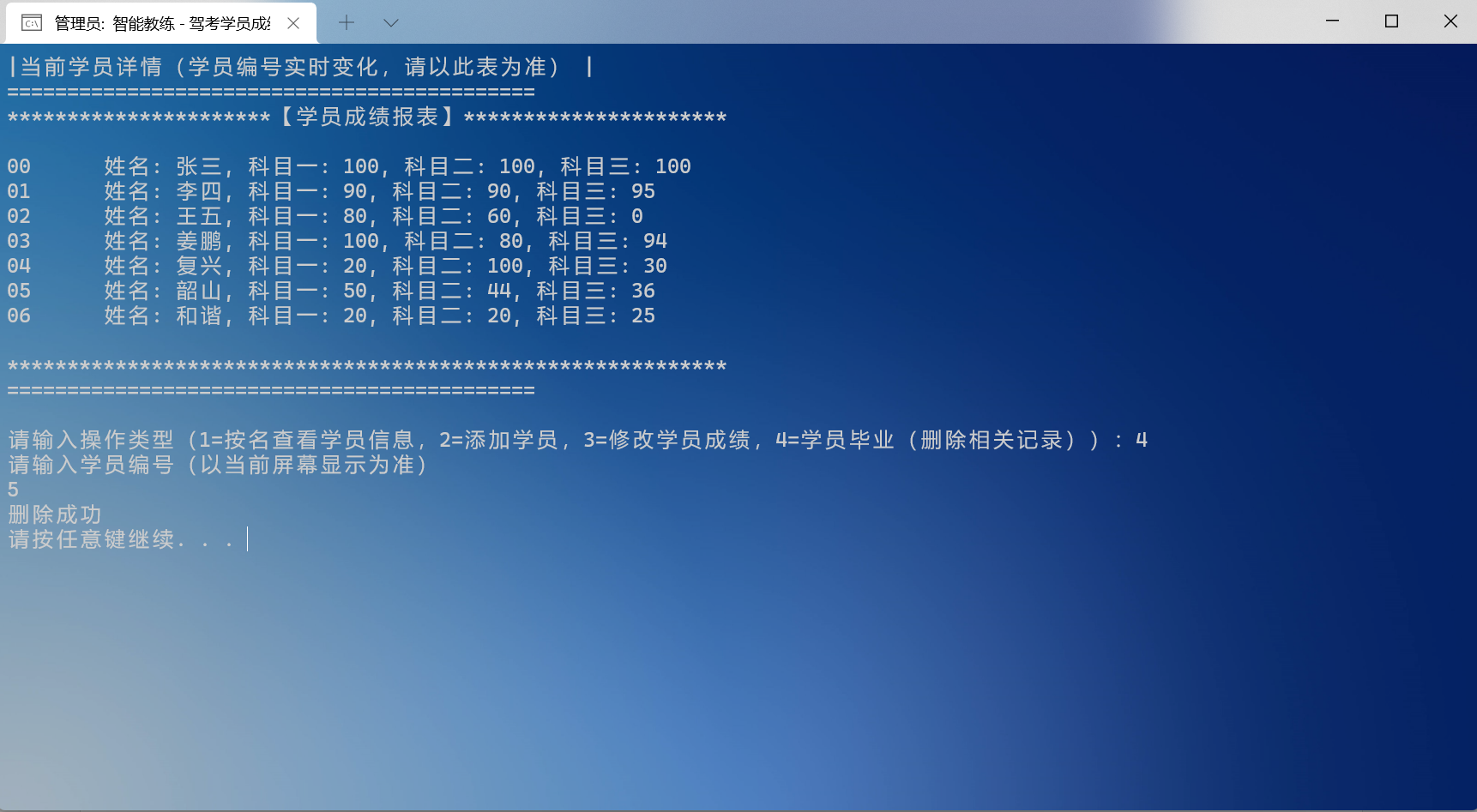
****

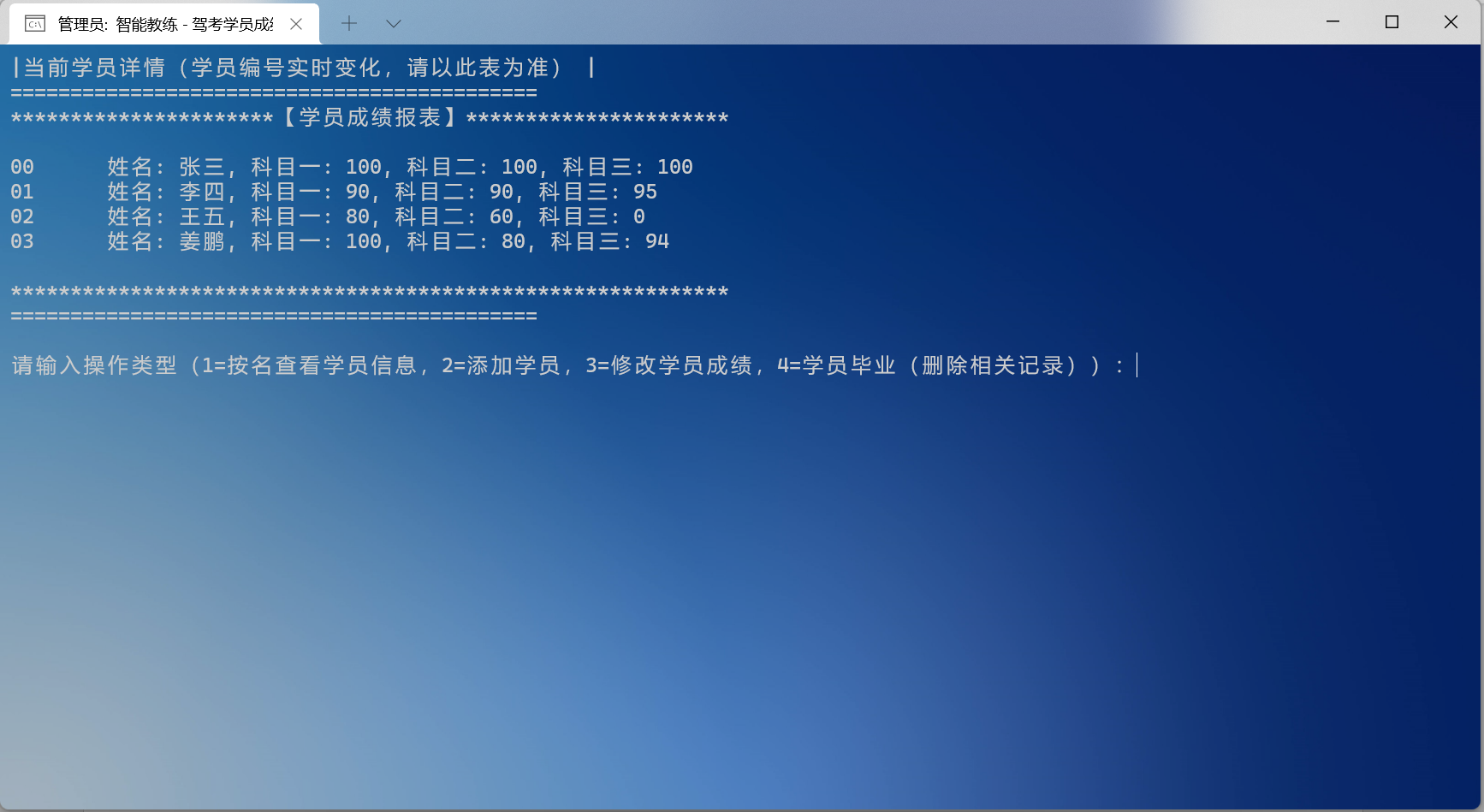
看你练车挺不容易的，算了，80分给你过了。。。

****

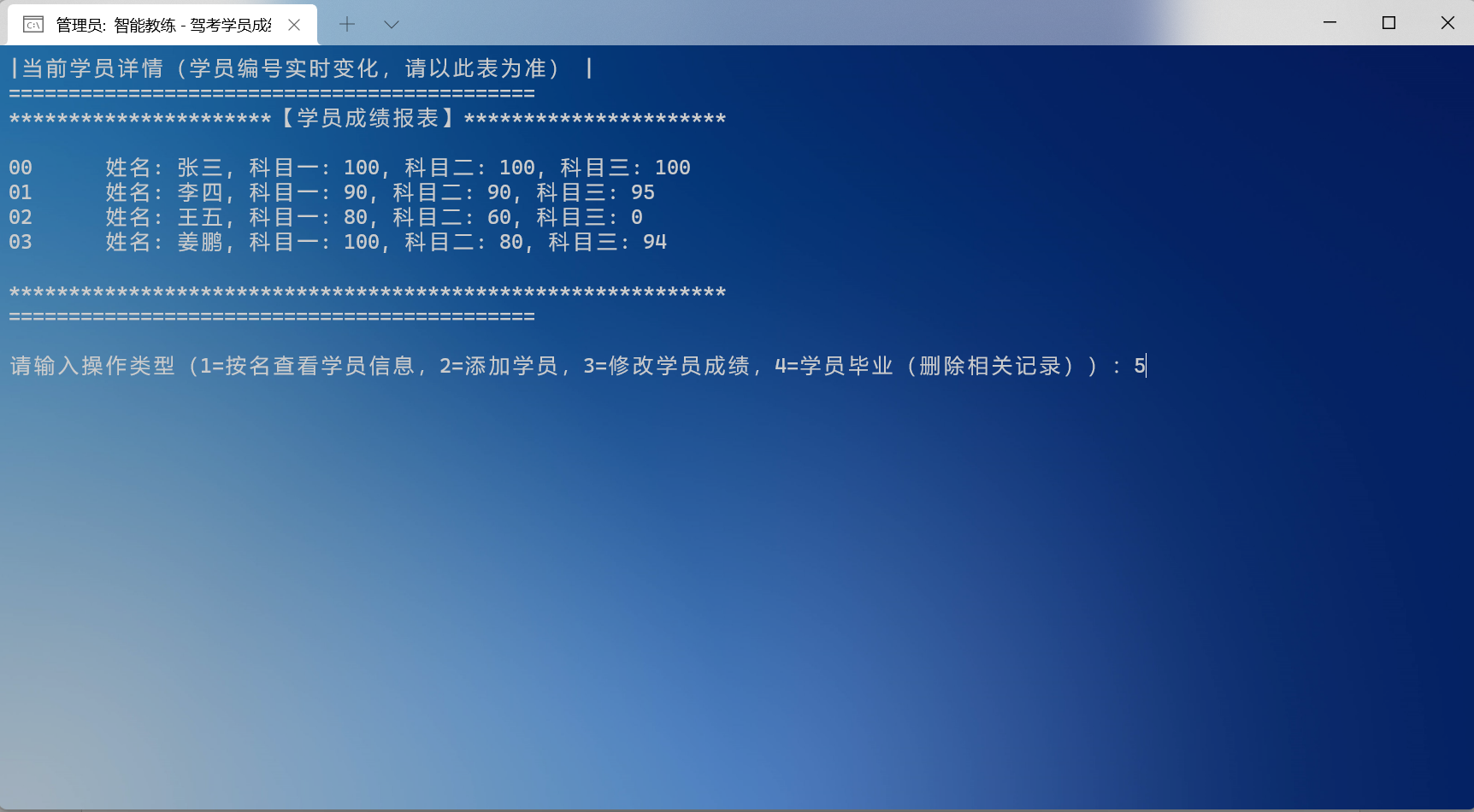
注意到“东风“的科目二已经是80分了

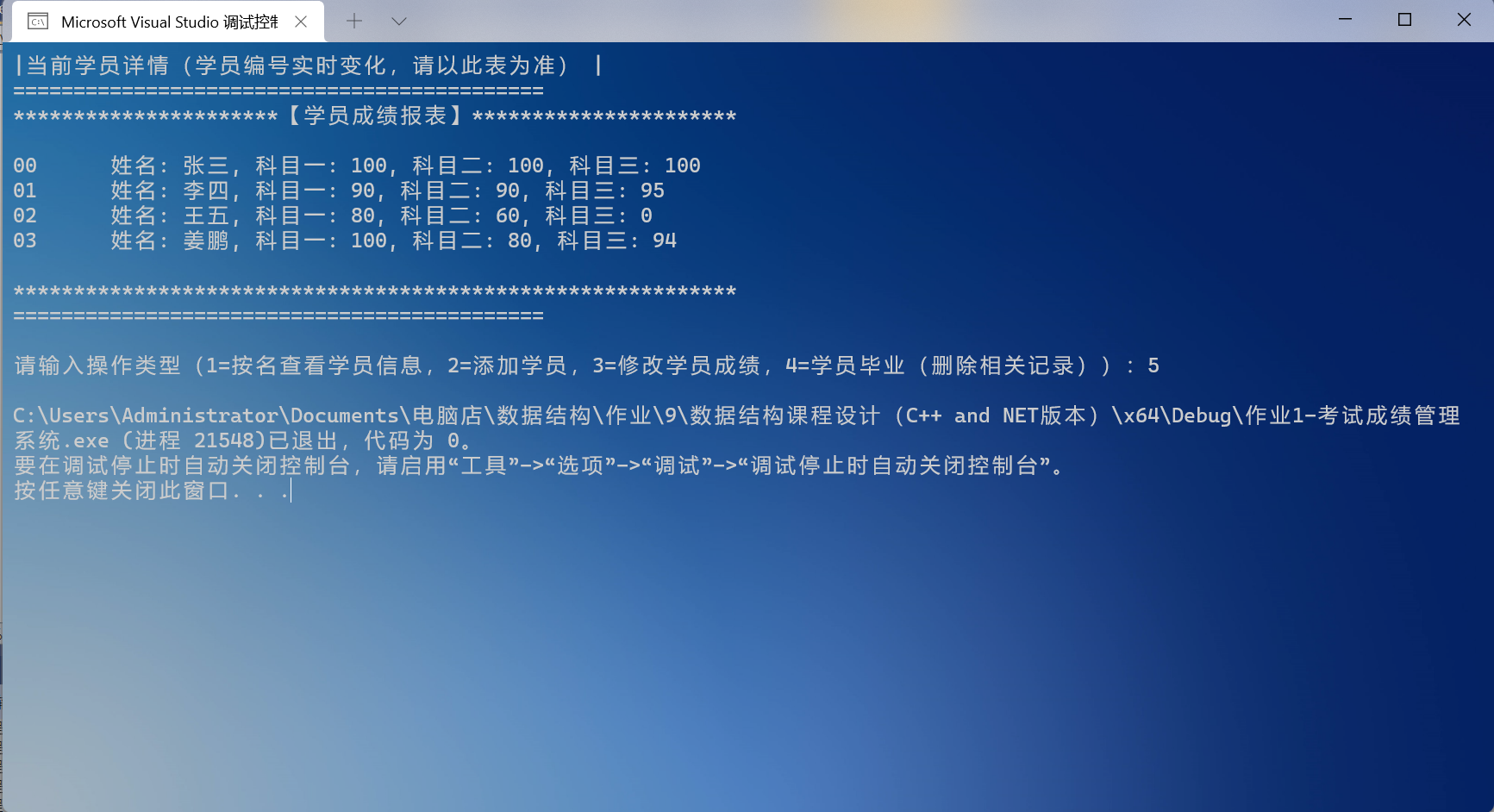
让“东风”，“韶山”，“和谐”，“复兴”四名学员毕业（删除记录）

****

****

主界面输入其他数字则退出程序

****

****