C语言-学生成绩管理系统【含完整代码】

目录

1.系统功能分析

1.1系统信息要求

1）学生基本信息如下：

学号、姓名、宿舍号、性别、年龄；

学号为标准格式1234567890十位，其中前两位代表学生入学年份，3-4位代表学生所在学院，5-6位代表学生所学专业、7-8位代表学生所在班级，9-10位代表学生在班级中的序号。

姓名最多为4个汉字；

性别为“男”或“女”；

年龄为2位正整数

2）课程信息如下：

课程号、课程类别、课程所在学期、课程名称、学分；

课程号为标准课号。例如B08010100；

课程类别为：选修/必修

所在学期用阿拉伯数字1-8代表

课程名称为专业为2014版人才培养计划中的课程名称

3）学生成绩信息如下：

学号、课程号、课程成绩、是否重修

学号为学生信息中的主关键字，可以唯一识别学生。课程号为课程信息的主关键字，可以唯一识别课程。是否重修用于判断课程成绩是否是第一次考试取得。

4）综合信息

学号、姓名、获奖类别、获奖时间、惩处类别、惩处时间、所获学分、奖励分值、惩罚分值。奖惩分起始值均为0分。

1.2功能分析

1.2.1录入部分

1）能实现学生信息的录入、修改并保存；

2）能实现课程信息的录入、修改并保存；

3）能分学期录入品行表现成绩（辅导员、班主任、班级评议）、修改并保存

4）能实现课程成绩的录入，并且在实现某课程成绩录入时，能够自动按学号排好顺序，并提示“某学号、某同学 某门功课成级”，例如“1508100201 丁兆元 C语言程序设计A成绩”。

5）能录入学生的各种奖惩信息

1.2.2修改部分

1）能对录入的课程成绩进行修改，例如成绩录错、重考、重修原因引起的成绩更改等。

2）能对个人信息进行修改

3）能对课程信息进行修改

4）能对学生奖惩信息进行修改

1.2.3统计分析部分

1）能对某门功课各分数段成绩进行统计

2）能分学期对学生业务课程平均分按分数段进行统计

3）能统计任意一名同学每门功课的班级排名以及业务课成绩总体排名

4）能以宿舍为单位进行成绩统计分析

5）能以挂科次数为依据分学期对比分析

6）能以业务课班级排名为依据分学期对比分析（前进或退步情况）

1.2.4排序部分

1）分学期按业务课程成绩对学生由高到低排序，并显示业务成绩平均分。

2）分学期按不及格门次对学生由高到低排序，并显示不及格门次。

3）分学期按不及格学生数对课程进行由高到低排序，并显示课程名及不及格学生数。

4）能分学期以宿舍为单位按成绩由高到低进行排序，并显示宿舍平均成绩。

5）能随时根据奖励对学生进行由高到低排序并输出信息。

6）能随时根据惩罚情况对学生由低到高排序并输出信息。

1.2.5奖学金自动评定

能根据学校奖学金评选办法，分学期进行奖学金评定并显示，并能够将评选结果自动追加到学生的奖惩信息库。

1.2.6数据的导入导出

基础数据一次录入永久存放，在需要时导入内存变量，如有修改重新导入文件，使永久保存的数据与临时使用的数据保持一致性。

1.3奖学金评选标准

1.3.1奖学金评选资格。有下列情形之一者，不能获得奖学金：

1）品行表现测评名次在班级70%以后者；

2）必修课或选修课有不及格者；

3）未达到《大学生体育锻炼合格标准》者；

4）受警告以上处分者，半年或察看期内不得参加奖学金评选；

5）品行表现具有不宜获得奖学金的其他情形者。

1.3.2综合奖学金的评选

该奖项用于奖励品学兼优、素质全面发展的优秀学生。

（1）奖励等级、金额、比例为：

特等奖2000元／人·年 3‰

一等奖1200元／人·年 3%

二等奖700元／人・年7%

三等奖400元/人・年15%

（2） 一、二、三等奖学金

以班级为单位，根据综合测评名次排列，列前35%名次者可参评奖学金。一、二、三等奖学金不全部评选的班级，可按学生数的3%、7%、15%的比例，只评选其中的一个等级。

1.4综合成绩计算标准

奖励分值计算办法，起始分值为0分。

（1）奖学金计分：

获得单项奖学金+1，三等奖学金+2，二等奖学金+3，一等奖学金+4，特等奖学金+5，校长奖学金+6。

（2）荣誉积分：

校级各种优秀个人+3，省级各种个人优秀+6，国家级各种优秀+12；

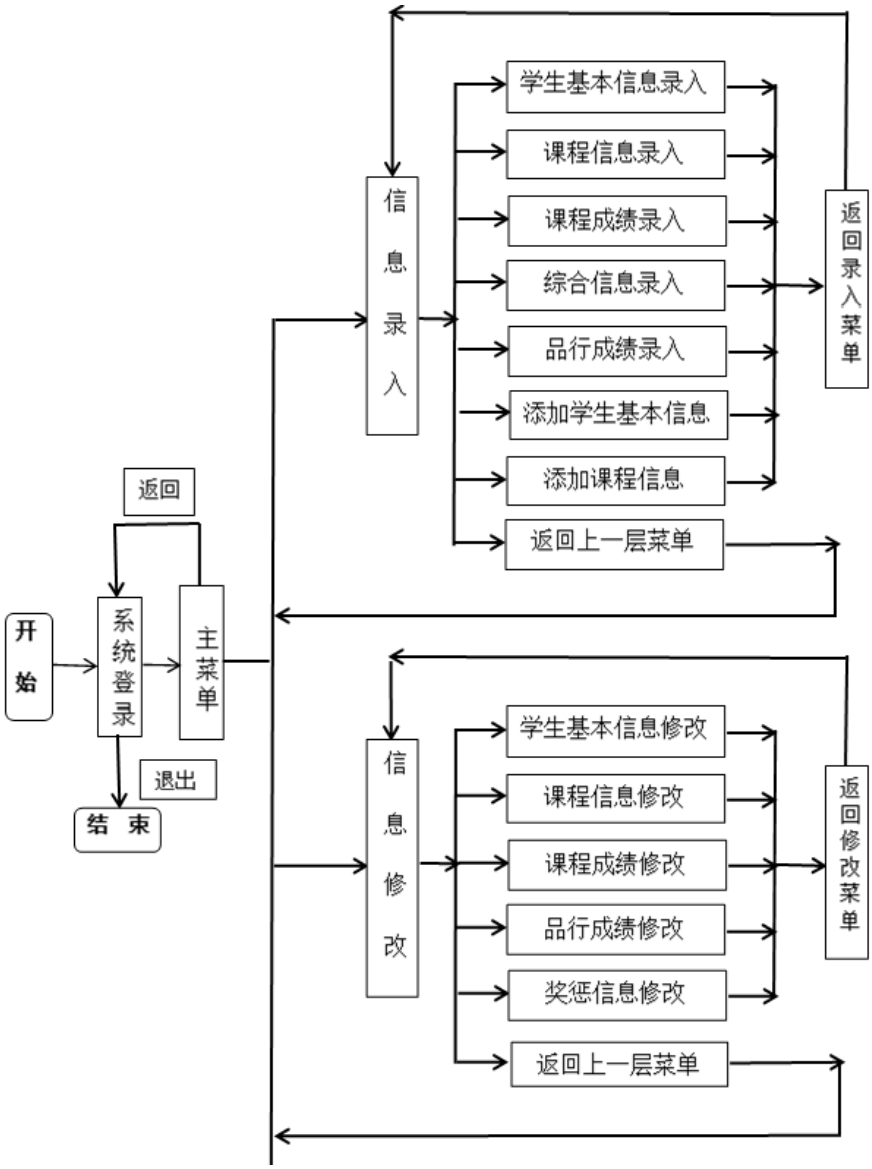
（3）学科竞赛：

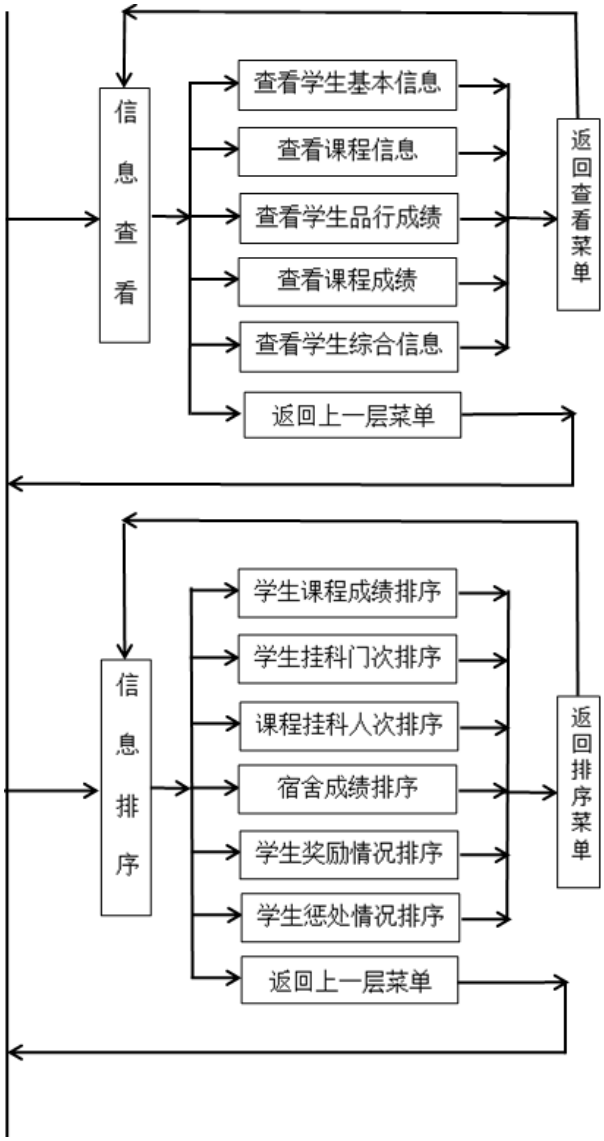
省级以上学科竞赛成功参赛奖+1，省级三等奖+4、省二等奖+5、生一等奖+6，国家级三等奖+6、国家级二等奖+9、国家级一等奖+12，校级三等奖+1.校级二等奖+2，校级一等奖+3分。

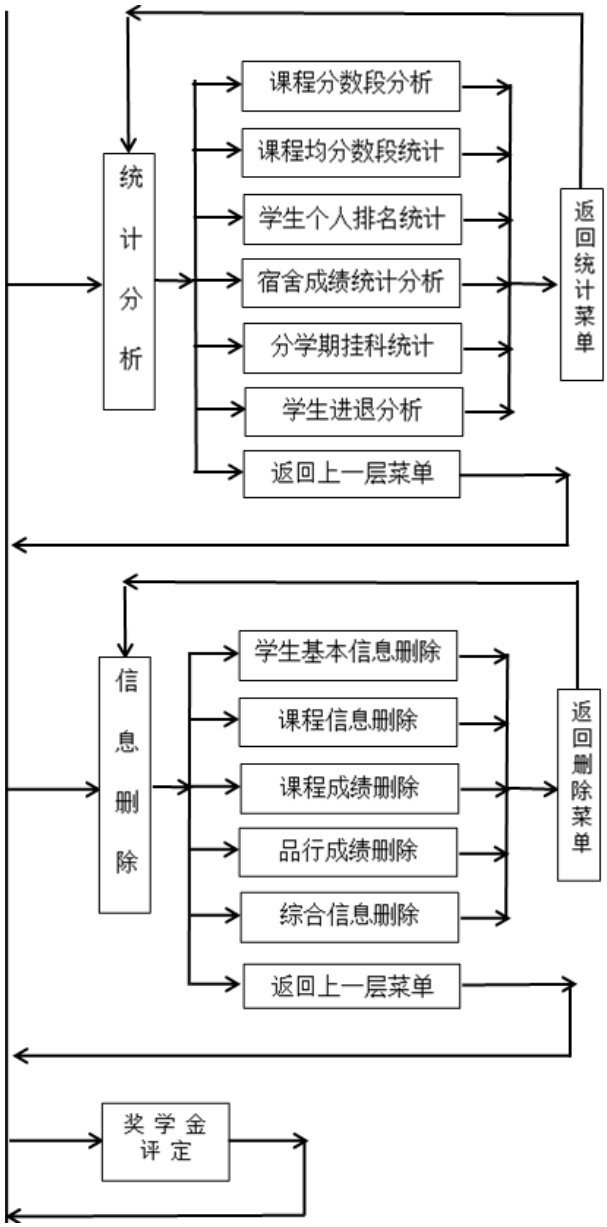
（4）惩罚分值计算办法，起始分值为0分。

学院通报批评-1，校级警告-2，严重警告-3，记过-4，记大过-5，开除学籍留校察看-6。

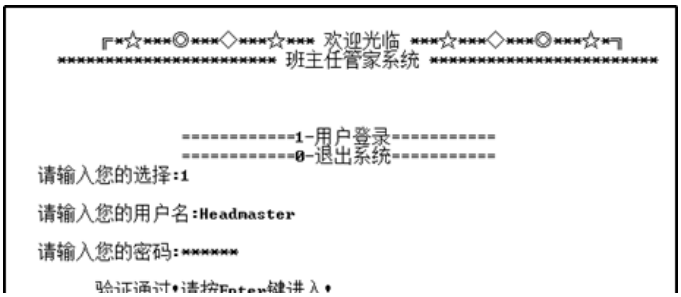
2.系统设计

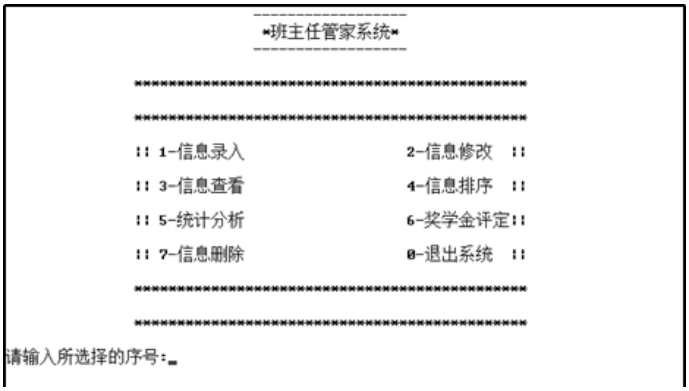

开始
系统登录
返回
退出
结束
主菜单
信息录入
学生基本信息录入
课程信息录入
课程成绩录入
综合信息录入
品行成绩录入
添加学生基本信息
添加课程信息
返回上一层菜单
返回录入菜单
信息修改
学生基本信息修改
课程信息修改
课程成绩修改
品行成绩修改
奖惩信息修改
返回上一层菜单
返回修改菜单


信息查看
查看学生基本信息
查看课程信息
查看学生品行成绩
查看课程成绩
查看学生综合信息
返回上一层菜单
返回查看菜单
信息排序
学生课程成绩排序
学生挂科门次排序
课程挂科人次排序
宿舍成绩排序
学生奖励情况排序
学生惩处情况排序
返回上一层菜单
返回排序菜单



3.运行效果图【界面太多，文章中只放两个示例】





4.测试数据:

4.1 账号、密码

账号：Headmaster

密码：123456

4.2 输入测试数据

（1）学生基本信息

存于“student.txt”文件。

（2）课程信息

存于“subject.txt”文件。

（3）课程成绩信息

C语言程序设计：存于“C语言程序设计.txt”文件。

高等数学1：存于“高等数学_1.txt”文件。

大学英语1：存于“大学英语_1.txt”文件。

计算机导论：存于“计算机导论_1.txt”文件。

大学英语2：存于“大学英语_2.txt”文件。

离散数学：存于“离散数学_2.txt”文件。

（4）学生品行成绩信息

存于“stCondSc.txt”文件。

（5）学生综合信息

存于“cominfo.txt”文件。

5.文件说明:

bin			文件夹
obj			文件夹
副本文件			文件夹
副本文件2			文件夹
cominfo.txt	1 KB	1 KB	文本文档
cominfo343.txt	1 KB	1 KB	文本文档
C语言程序设计_1.txt	1 KB	1 KB	文本文档
main.cpp	93.1 KB	12.7 KB	CPP 文件
main.exe	1.0 MB	300.3 KB	应用程序
main.o	100.0 KB	33.2 KB	O 文件
stCondSc.txt	1 KB	1 KB	文本文档
student.txt	1 KB	1 KB	文本文档
subject.txt	1 KB	1 KB	文本文档
班主任管家系统.cbp	1.1 KB	1 KB	project file
班主任管家系统.cscope_file_list	1 KB	1 KB	CSCOPE_FILE_LIS...
班主任管家系统.depend	1 KB	1 KB	DEPEND 文件
班主任管家系统.layout	1 KB	1 KB	LAYOUT 文件
大学英语_1.txt	1 KB	1 KB	文本文档
大学英语_2.txt	1 KB	1 KB	文本文档
高等数学_1.txt	1 KB	1 KB	文本文档
计算机导论_1.txt	1 KB	1 KB	文本文档

6.下面附上源代码

【运行前需要自行新建所需文件，必要时自己加上测试数据】

```
#include <iostream>
#include <cstdio>
#include <cstdlib>
#include <cstring>
#include <conio.h>

using namespace std;

//学生基本信息
struct student
{
    int stage; //年龄
    char stnum[12]; //学号
    char stname[20]; //姓名
    char stsex[6]; //性别
    char dornum[10]; //宿舍号
    struct student *next;
};

//课程信息
struct subject
{
    char subNum[20]; //课程编号
    char subName[30]; //课程名
    char cate[20]; //课程类别(选修、必修)
    int term; //课程所在学期
    float credit; //学分
    struct subject *next;
};

//学生成绩信息
struct stscore
{
    char stnum[12]; //学号
    char subNum[20]; //课程编号
    int mark; //课程成绩
    char repair[4]; //是否重修
    int cnt = 0; //不及格门次
    struct stscore *next;
};

//学生品行分信息
struct stCondSc
{
    char stnum[12]; //学号
    int coumark; //品行成绩——辅导员
    int headmark; //品行成绩——班主任
    int classsmark; //品行成绩——班级
    int conductmark; //品行分总分
    struct stCondSc *next;
};

//学生综合信息
struct cominfo
{
    char stnum[12]; //学号
    char stname[20]; //姓名
    char awdlev[10]; //获奖类别
    int ayear, amon, aday; //获奖时间
    int amark; //奖励分值
    char puncate[10]; //惩处类别
    int pyear, pmon, pday; //惩处时间
    int pmark; //惩罚分值
    int getcredit; //所获学分
```

```
    struct cominfo *next;
};

struct student *st;//学生基本信息
struct subject *sub;//课程信息
struct stscore *stsc;//学生成绩信息
struct cominfo *coin;//学生综合信息
struct stCondSc *stcosc; //学生品行成绩信息

/* ***** 模块函数 ********* */
void creat(); //信息录入
void modify(); //信息修改
void ddelete(); //信息删除
void countStatics(); //统计分析
void ssort(); //信息排序

/* ****** 信息录入、修改、删除 ********* */
struct student* creatStudent(); //录入学生基本信息
struct subject* creatSubject(); //录入课程信息
struct stscore* creatSubScore(); //录入课程成绩
struct cominfo* creatAwdPunScore(); //录入综合信息
struct stCondSc* creatConductScore(); //录入学生品行成绩信息
struct student* insertStudent(struct student *st); //添加学生基本信息
struct subject* insertSubject(struct subject *head); //添加课程基本信息

struct student * modifyStudent(); //修改学生基本信息
struct subject * modifySubject(); //修改课程信息
struct stscore * modifySubScore(); //修改课程成绩信息
struct cominfo * modifyAwdPunScore(); //修改奖惩信息
struct stCondSc * modifyConductScore(); //修改品行成绩信息

void deleteStudent(); //删除学生基本信息
void deleteSubject(); //删除课程信息
void deleteStScore(); //删除课程成绩信息
void deleteStCondSc(); //删除品行成绩信息
void deleteCominfo(); //删除综合信息

/* ******* 输出查看 ********** */
void output(); //输出信息
void outputStudent(); //输出学生基本信息
void outputSubject(); //输出课程信息
void outputStScore(); //输出课程成绩
void outputStCondSc(); //输出学生品行分
void outputCominfo(); //输出学生综合信息

/* ***** 统计分析 ********* */
void countSubScore(); //某门课程按分数段成绩统计
void countAveScore(); //业务课程平均分按分数段统计
void countStScore(); //统计任意一名同学每门功课的班级排名以及业务课成绩总体排名
void countDorRank(); //以宿舍为单位进行成绩统计分析
void countFail(); //分学期统计挂科
void countChange(); //以业务课班级排名为依据分学期对比分
void schoshipAsse(); //奖学金评定

/* ****** 排序 ********* */
void subStScoreSort(); //按课程成绩（高->低）对学生排序
void failStcntSort(); //按挂科门次（高->低）对学生排序
void failSubSort(); //按不及格学生数对课程进行由高到低排序
void dorScoreSort(); //以宿舍为单位按成绩由高到低进行排序
void awdStSort(); //根据奖励对学生进行由高到低排序
void punStScore(); //根据惩罚情况对学生由低到高排序

void fprintfStudent(struct student *head);//将学生信息写入文件
struct student* loadStudent();//从文件中读取学生信息
void fprintfsubject(struct subject *head);//将课程信息写入文件
struct subject* loadSubject(); //从文件中读取课程信息
```

```
struct subject* loadSubject();//从文件中读取课程信息

//主函数
int main()
{
    int choice, n, y = 1;
    char admname[20]="1", password[16]="1"; //用户名，密码
    char an[20], pw[16];
    char c;
    int i;

    while (y)
    {
        system("cls");
        printf("\n\n\n\n\n\n\n");
        printf("             ╔*☆***◎***◇***☆*** 欢迎光临 ***☆***◇***◎***☆*╗\n");
        printf("         ********************** 班主任管家系统 ************************\n\n");
        printf("\n\n");
        printf("                      ============1-用户登录===========\n");
        printf("                      ============0-退出系统===========\n");
        printf("       请输入您的选择:");
        scanf("%d", &n);
        printf("\n");
        getchar();
        switch (n)
        {
            case 0: y = 0; break;
            case 1:{
                printf("       请输入您的用户名:");
                scanf("%s", an);
                printf("\n");
                printf("       请输入您的密码:");
                i = 0;
                do
                {
                    c = getch();
                    if (c == '\b')
                    {
                        printf("\b");
                        putchar('\0');
                        i--;
                        printf("\b");
                    }
                    else if (c != '\r')
                    {
                        printf("*");
                        pw[i] = c;
                        i++;
                    }
                }while (c != '\r');
                pw[i] = '\0';
                printf("\n\n");
                if (!(strcmp(admname,an)==0 && strcmp(password,pw)==0))
                {
                    printf("           验证失败,请重新输入!\n");
                    getchar();
                    getchar();
                }
                else
                {
                    printf("             验证通过!请按Enter键进入!\n");
                    getchar();
                    getchar();
                    int x = 1;
                    while (x)
                    {
```

```
            {
                system("cls");
                printf("                          ------------------\n");
                printf("                           *班主任管家系统*\n");
                printf("                          ------------------\n\n");
                printf("            ************************************************\n\n");
                printf("            ************************************************\n\n");
                printf("            || 1-信息录入              2-信息修改  ||\n\n");
                printf("            || 3-信息查看              4-信息排序  ||\n\n");
                printf("            || 5-统计分析              6-奖学金评定||\n\n");
                printf("            || 7-信息删除              0-退出系统  ||\n\n");
                printf("            ************************************************\n\n");
                printf("            ************************************************\n\n");
                printf("请输入所选择的序号:");
                scanf("%d",&choice);
                system("cls");
                switch (choice)
                {
                    case 0: x = 0; break;
                    case 1: creat(); break;
                    case 2: modify(); break;
                    case 3: output(); break;
                    case 4: ssort(); break;
                    case 5: countStatics(); break;
                    case 6: schoshipAsse(); getchar(); getchar(); break;
                    case 7: ddelete(); break;
                    default :
                        printf("            输入错误，请重新输入！\n");
                        getchar();
                        break;
                }//switch
            }//while(x)
        }//else
    }//case 1
    break;
    default :
        printf("            输入错误，请重新输入！\n");
        getchar();
        break;
  }//switch(n)
}//while(y)
return 0;
}//main

//信息录入
void creat()
{
  st = (struct student *)malloc(sizeof(struct student));
  sub = (struct subject *)malloc(sizeof(struct subject));
  stsc = (struct stscore *)malloc(sizeof(struct stscore));
  coin = (struct cominfo *)malloc(sizeof(struct cominfo));
  stcosc = (struct stCondSc *)malloc(sizeof(struct stCondSc));

  int choice, y = 1;

  while (y)
  {
    system("cls");
    printf("                        ------------------\n");
    printf("                         *班主任管家系统*\n");
    printf("                           *信息录入*\n");
    printf("                        ------------------\n\n");
    printf("            ************************************************\n\n");
    printf("            ************************************************\n\n");
    printf("            || 1-学生基本信息录入     2-课程信息录入    ||\n\n");
    printf("            || 3-课程成绩录入         4-综合信息录入    ||\n\n");
```

```
        printf("              || 3-课程成绩录入          4-综合信息录入    ||\n\n");
        printf("              || 5-品行成绩录入          6-添加学生基本信息||\n\n");
        printf("              || 7-添加课程信息          0-退出系统        ||\n\n");
        printf("              ************************************************\n\n");
        printf("              ************************************************\n\n");
        printf("请输入所选择的序号:");
        scanf("%d",&choice);
        system("cls");
        switch (choice)
        {
            case 0: y = 0; break;
            case 1: st = creatStudent();  getchar(); break;
            case 2: sub = creatSubject(); getchar(); break;
            case 3: stsc = creatSubScore(); getchar(); break;
            case 4: coin =  creatAwdPunScore(); getchar(); break;
            case 5: stcosc = creatConductScore(); getchar(); break;
            case 6:
                st = loadStudent();
                st = insertStudent(st);
                getchar();
                break;
            case 7:
                sub = loadSubject();
                sub = insertSubject(sub);
                getchar();
                break;
            default:
                printf("              输入错误，请重新输入！\n");
                getchar();
                break;
        }//switch(choice)
        getchar();
    }//while(y)
}//creat

//信息修改
void modify()
{
    int choice, y = 1;

    while (y)
    {
        system("cls");
        printf("                          ------------------\n");
        printf("                           *班主任管家系统*\n");
        printf("                             *信息修改*\n");
        printf("                          ------------------\n\n");
        printf("              ************************************************\n\n");
        printf("              ************************************************\n\n");
        printf("              || 1-学生基本信息修改        2-课程信息修改 ||\n\n");
        printf("              || 3-课程成绩修改            4-品行成绩修改 ||\n\n");
        printf("              || 5-奖惩信息修改            0-退出系统     ||\n\n");
        printf("              ************************************************\n\n");
        printf("              ************************************************\n\n");
        printf("请输入所选择的序号:");
        scanf("%d",&choice);
        system("cls");
        switch (choice)
        {
            case 0: y = 0; break;
            case 1: st = modifyStudent(); getchar(); break;
            case 2: sub = modifySubject(); getchar(); break;
            case 3: stsc = modifySubScore(); getchar(); break;
            case 4: stcosc = modifyConductScore(); getchar(); break;
            case 5: coin = modifyAwdPunScore(); getchar(); break;
            default:
```

```
        default:
            printf("            输入错误，请重新输入！\n");
            getchar();
            break;
    }//switch(choice)
    getchar();
  }//while(y)
}//modify

//信息输出
void output()
{
  int choice, y = 1;

  while (y)
  {
    system("cls");
    printf("                        ------------------\n");
    printf("                         *班主任管家系统*\n");
    printf("                           *信息查看*\n");
    printf("                        ------------------\n\n");
    printf("            ************************************************\n\n");
    printf("            ************************************************\n\n");
    printf("            || 1-查看学生基本信息      2-查看课程信息 ||\n\n");
    printf("            || 3-查看学生品行成绩      4-查看课程成绩 ||\n\n");
    printf("            || 5-查看学生综合信息      0-退出系统     ||\n\n");
    printf("            ************************************************\n\n");
    printf("            ************************************************\n\n");
    printf("请输入所选择的序号:");
    scanf("%d",&choice);
    system("cls");
    switch (choice)
    {
      case 0: y = 0; break;
      case 1: outputStudent(); getchar(); break;
      case 2: outputSubject(); getchar(); break;
      case 3: outputStCondSc(); getchar(); break;
      case 4: outputStScore(); getchar(); break;
      case 5: outputCominfo(); getchar(); break;
      default:
        printf("            输入错误，请重新输入！\n");
        getchar();
        break;
    }//switch(choice)
    getchar();
  }//while(y))
}//output

//统计分析
void countStatics()
{
  int choice, y = 1;

  while (y)
  {
    system("cls");
    printf("                        ------------------\n");
    printf("                         *班主任管家系统*\n");
    printf("                           *统计分析*\n");
    printf("                        ------------------\n\n");
    printf("            ******************************************************\n\n");
    printf("            ******************************************************\n\n");
    printf("            || 1-课程分数段统计      2-课程平均分分数段统计 ||\n\n");
    printf("            || 3-学生个人排名统计    4-宿舍成绩统计分析     ||\n\n");
    printf("            || 5-分学期挂科统计      6-学生进退分析         ||\n\n");
    printf("            || 0-退出系统                                   ||\n\n");
```

```
        printf("                || 0-退出系统                                   ||\n\n");
        printf("                ******************************************************\n\n");
        printf("                ******************************************************\n\n");
        printf("请输入所选择的序号:");
        scanf("%d",&choice);
        system("cls");
        switch (choice)
        {
            case 0: y = 0; break;
            case 1: countSubScore(); getchar(); break;
            case 2: countAveScore(); getchar(); break;
            case 3: countStScore(); getchar(); break;
            case 4: countDorRank(); getchar(); break;
            case 5: countFail(); getchar(); break;
            case 6: countChange(); getchar(); break;
            default:
                printf("              输入错误，请重新输入！\n");
                getchar();
                break;
        }//switch(choice)
        getchar();
    }//while(y)
}

//信息排序
void ssort()
{
    int choice, y = 1;

    while (y)
    {
        system("cls");
        printf("                          ------------------\n");
        printf("                           *班主任管家系统*\n");
        printf("                             *信息排序*\n");
        printf("                          ------------------\n\n");
        printf("                ******************************************************\n\n");
        printf("                ******************************************************\n\n");
        printf("                || 1-学生课程成绩排序       2-学生挂科门次排序 ||\n\n");
        printf("                || 3-课程挂科人次排序       4-宿舍成绩排序     ||\n\n");
        printf("                || 5-学生奖励情况排序       6-学生惩处情况排序 ||\n\n");
        printf("                || 0-退出系统                                  ||\n\n");
        printf("                ******************************************************\n\n");
        printf("                ******************************************************\n\n");
        printf("请输入所选择的序号:");
        scanf("%d",&choice);
        getchar();
        system("cls");
        switch (choice)
        {
            case 0:  y = 0; break;
            case 1: subStScoreSort(); getchar(); break;
            case 2: failStcntSort(); getchar(); break;
            case 3: failSubSort(); getchar(); break;
            case 4: dorScoreSort(); getchar(); break;
            case 5: awdStSort(); getchar(); break;
            case 6: punStScore(); getchar(); break;
            default:
                printf("              输入错误，请重新输入！\n");
                getchar();
                break;
        }//switch(choice)
    }//while(y)
}//ssort

//信息删除
```

```
//自定义删除
void ddelete()
{
    int choice, y = 1;

    while (y)
    {
        system("cls");
        printf("                          ----------\n");
        printf("                           *班主任管家系统*\n");
        printf("                             *信息删除*\n");
        printf("                          ----------\n\n");
        printf("              ************************************************\n\n");
        printf("              ************************************************\n\n");
        printf("              || 1-学生基本信息删除        2-课程信息删除 ||\n\n");
        printf("              || 3-课程成绩删除            4-品行成绩删除 ||\n\n");
        printf("              || 5-综合信息删除            0-退出系统     ||\n\n");
        printf("              ************************************************\n\n");
        printf("              ************************************************\n\n");
        printf("请输入所选择的序号:");
        scanf("%d",&choice);
        system("cls");
        switch (choice)
        {
            case 0: y = 0; break;
            case 1: deleteStudent(); getchar(); break;
            case 2: deleteSubject(); getchar(); break;
            case 3: deleteStScore(); getchar(); break;
            case 4: deleteStCondSc(); getchar(); break;
            case 5: deleteCominfo(); getchar(); break;
            default:
                printf("              输入错误，请重新输入！\n");
                getchar();
                break;
        }//switch(choice)
        getchar();
    }//while(y)
}//modify

//判断学号是否已存在
//存在返回1，不存在返回0
int isNumExist(struct student *head, char *num)
{
    struct student *q = head;
    while (q != NULL)
    {
        if (strcmp(q ->stnum, num) == 0)
            return 1;
        q = q ->next;
    }
    return 0;
}

//将学生信息写入文件
void fprintfStudent(struct student *head)
{
    FILE *fp;
    struct student *p;
    if ((fp = fopen("student.txt", "w")) == NULL)
    {
        printf("Can't open the file!\n");
        return ;
    }
    p = head;
    while (p != NULL)
    {
```

```
    {
        fprintf(fp, "%s\t%s\t%s\t%d\t%s\n", p ->stnum, p ->stname, p ->stsex, p ->stage, p ->dornum);
        p = p ->next;
    }
    fclose(fp);
}//fprintfStudent

//从文件中读取学生信息
struct student* loadStudent()
{
    FILE *fp;
    struct student *head, *tail, *p;

    head = tail = NULL;
    if ((fp = fopen("student.txt", "r")) == NULL)
    {
        printf("Can't open the file!\n");
        exit(0);
    }
    while (!feof(fp))
    {
        p = (struct student *)malloc(sizeof(struct student));
        fscanf(fp, "%s\t%s\t%s\t%d\t%s\n",  p ->stnum, p ->stname, p ->stsex, &p ->stage, p ->dornum);
        if (head == NULL)
            head = p;
        else
            tail ->next = p;
        tail = p;
    }
    tail ->next = NULL;
    fclose(fp);
    return head;
}//loadStudent

//录入学生基本信息
struct student* creatStudent()
{
    struct student *head, *tail, *p;
    head = (struct student *)malloc(sizeof(struct student));
    int age, key, y = 1;
    char num[12], sex[6], dn[10], name[20];
    char kk[3] = "0";
    char ss1[5] = "男", ss2[5] = "女";

    head = tail = NULL;
    printf("请输入学生信息，以学号0为结束！\n");
    while (y)
    {
        printf("\n");
        do
        {
            printf("请输入学生学号：");
            scanf("%s", num);
            if (strlen(num) != 10 && strcmp(kk, num) != 0)
            {
                printf("学号格式输入错误，请重新输入！\n");
                key = 1;
            }
            else
            {
                key = isNumExist(head, num);
                if (key == 1)
                    printf("该学号已存在，请重新输入！\n");
                if (strcmp(kk, num) == 0)
                    y = 0;
            }
```

```
        }while (key == 1);
        if (y != 0)
        {
            printf("请输入学生姓名：");
            scanf("%s", name);
            do
            {
                key = 1;
                printf("请输入学生性别：");
                scanf("%s", sex);
                if (strcmp(sex, ss1)==0 || strcmp(sex, ss2)==0)
                    key = 0;
                else
                    printf("性别输入错误，请重新输入！\n");
            }while(key == 1);
            printf("请输入学生年龄：");
            scanf("%d", &age);
            //getchar();
            printf("请输入学生宿舍号：");
            scanf("%s", dn);

            printf("是否要存储该学生信息?\n");
            printf("  1-确定   0-取消\n");
            int k;
            scanf("%d", &k);
            if (k == 1)
            {
                p = (struct student *)malloc(sizeof(struct student));
                strcpy(p ->stname, name);
                strcpy(p ->stnum, num);
                strcpy(p ->stsex, sex);
                strcpy(p ->dornum, dn);
                p ->stage = age;
                p ->next = NULL;
                if(head == NULL)
                    head = p;
                else
                    tail ->next = p;
                tail = p;
            }//if
        }//if
    }//while(y)
    printf("是否要覆盖原有信息，并将其存入文件？");
    printf("  1-确认   0-取消\n");
    printf("请选择：");
    scanf("%d", &y);
    if (y)
    {
        fprintfStudent(head);
        printf("\n录入学生基本信息成功！\n");
    }
    printf("\n **按Enter返回！");
    return head;
}//creatStudent

//添加学生基本信息
struct student* insertStudent(struct student *head)
{
    struct student *tail, *p;
    int age, key, y = 1;
    char num[12], sex[6], dn[6], name[20];
    char kk[3] = "0";

    tail = head;
    while (tail!=NULL && tail ->next!=NULL)
```

```
    {
        tail = tail ->next;
    }

    printf("请输入学生信息，以学号0为结束！\n");
    while (y)
    {
        printf("\n");
        do
        {
            printf("请输入学生学号：");
            scanf("%s", num);
            if (strlen(num) != 10 && strcmp(kk, num) != 0)
            {
                printf("学号格式输入错误，请重新输入！\n");
                key = 1;
            }
            else
            {
                key = isNumExist(head, num);
                if (key == 1)
                    printf("该学号已存在，请重新输入！\n");
                if (strcmp(kk, num) == 0)
                    y = 0;
            }
        }while (key == 1);
        if (y != 0)
        {
            printf("请输入学生姓名：");
            scanf("%s", name);
            printf("请输入学生性别：");
            scanf("%s", sex);
            printf("请输入学生年龄：");
            scanf("%d", &age);
            printf("请输入学生宿舍号：");
            scanf("%s", dn);

            printf("是否要存储该学生信息?\n");
            printf("  1-确定   0-取消\n");
            int k;
            scanf("%d", &k);
            if (k == 1)
            {
                p = (struct student *)malloc(sizeof(struct student));
                strcpy(p ->stname, name);
                strcpy(p ->stnum, num);
                strcpy(p ->stsex, sex);
                strcpy(p ->dornum, dn);
                p ->stage = age;
                p ->next = NULL;
                if(head == NULL)
                    head = p;
                else
                    tail ->next = p;
                tail = p;
            }//if
        }//if
    }//while(y)
    fprintfStudent(head);
    printf("添加学生基本信息成功！\n");
    printf("\n **按Enter返回！");
    return head;
}//insertStudent

//判断学号是否已存在
```

```
//存在返回1，不存在返回0
int isNumExist(struct subject *head, char *num)
{
    struct subject *q = head;
    while (q != NULL)
    {
        if (strcmp(q ->subNum, num) == 0)
            return 1;
        q = q ->next;
    }
    return 0;
}//isNumExist

//将课程信息写入文件
void fprintfsubject(struct subject *head)
{
    FILE *fp;
    struct subject *p;
    if ((fp = fopen("subject.txt", "w")) == NULL)
    {
        printf("Can't open the file!\n");
        return ;
    }
    p = head;
    while (p != NULL)
    {
        fprintf(fp, "%s\t%s\t%s\t%g\t%d\n", p ->subNum, p ->subName, p ->cate, p ->credit, p ->term);
        p = p ->next;
    }
    fclose(fp);
}//fprintfsubject

//从文件中读取课程信息
struct subject* loadSubject()
{
    FILE *fp;
    struct subject *head, *tail, *p;

    head = tail = NULL;
    if ((fp = fopen("subject.txt", "r")) == NULL)
    {
        printf("Can't open the file!\n");
        exit(0);
    }
    while (!feof(fp))
    {
        p = (struct subject *)malloc(sizeof(struct subject));
        fscanf(fp, "%s\t%s\t%s\t%g\t%d\n",  p ->subNum, p ->subName, p ->cate, &p ->credit, &p ->term);
        if (head == NULL)
            head = p;
        else
            tail ->next = p;
        tail = p;
    }
    tail ->next = NULL;
    fclose(fp);
    return head;
}//loadSubject

//录入课程信息
struct subject* creatSubject()
{
    struct subject *head, *tail, *p;
    head = (struct subject *)malloc(sizeof(struct subject));
    int key, y = 1;
```

```
    char num[12], cate[6], name[20];
    int term;
    float credit;
    char kk[3] = "0";

    head = tail = NULL;
    printf("请输入课程信息，以编号0为结束！\n");
    while (y)
    {
        printf("\n");
        do
        {
            printf("请输入课程编号：");
            scanf("%s", num);
            key = isNumExist(head, num);
            if (key == 1)
                printf("该编号已存在，请重新输入！\n");
            if (strcmp(kk, num) == 0)
                y = 0;
        }while (key == 1);
        if (y != 0)
        {
            printf("请输入课程名称：");
            scanf("%s", name);
            printf("请输入课程类别：");
            scanf("%s", cate);
            printf("请输入课程学分：");
            scanf("%f", &credit);
            printf("请输入课程所在学期：");
            scanf("%d", &term);

            printf("是否要存储该课程信息?\n");
            printf("  1-确定   0-取消\n");
            int k;
            scanf("%d", &k);
            if (k == 1)
            {
                p = (struct subject *)malloc(sizeof(struct subject));
                strcpy(p ->subName , name);
                strcpy(p ->subNum, num);
                strcpy(p ->cate, cate);
                p ->term = term;
                p ->credit = credit;
                p ->next = NULL;
                if(head == NULL)
                    head = p;
                else
                    tail ->next = p;
                tail = p;
            }//if
        }//if
    }//while(y)
    printf("是否要覆盖原有信息，并将其存入文件？");
    printf("  1-确认   0-取消\n");
    printf("请选择：");
    scanf("%d", &y);
    if (y)
    {
        fprintfsubject(head);
        printf("\n录入课程信息成功！\n");
    }
    printf("\n **按Enter返回！");
    return head;
}//creatsubject
```

```
//添加课程信息
struct subject* insertSubject(struct subject *head)
{
    struct subject *tail, *p;
    int key, y = 1;
    char num[12], cate[6], name[20];
    int term;
    float credit;
    char kk[3] = "0";

    tail = head;
    while (tail!=NULL && tail ->next!=NULL)
    {
        tail = tail ->next;
    }

    printf("请输入课程信息，以编号0为结束！\n");
    while (y)
    {
        printf("\n");
        do
        {
            printf("请输入课程编号：");
            scanf("%s", num);
            key = isNumExist(head, num);
            if (key == 1)
                printf("该编号已存在，请重新输入！\n");
            if (strcmp(kk, num) == 0)
                y = 0;
        }while (key == 1);
        if (y != 0)
        {
            printf("请输入课程名称：");
            scanf("%s", name);
            printf("请输入课程类别：");
            scanf("%s", cate);
            printf("请输入课程学分：");
            scanf("%f", &credit);
            printf("请输入课程所在学期：");
            scanf("%d", &term);

            printf("是否要存储该课程信息?\n");
            printf("  1-确定   0-取消\n");
            int k;
            scanf("%d", &k);
            if (k == 1)
            {
                p = (struct subject *)malloc(sizeof(struct subject));
                strcpy(p ->subName , name);
                strcpy(p ->subNum, num);
                strcpy(p ->cate, cate);
                p ->term = term;
                p ->credit = credit;
                p ->next = NULL;
                if(head == NULL)
                    head = p;
                else
                    tail ->next = p;
                tail = p;
            }//if
        }//if
    }//while(y)
    fprintfsubject(head);
    printf("添加课程信息成功！\n");
    printf("\n **按Enter返回！");
```

```
    return head;
}//insertStudent

//判断学号与成绩是否已存在
//存在返回1，不存在返回0
int isNumExist(struct stscore *a, char *stnum, int n)
{
     for (int i = 0; i < n; i++)
    {
        if (strcmp(a[i].stnum, stnum)==0)
            return 1;
    }
    return 0;
}//isNumExist

//将课程成绩写入文件
void fprintfstscore(struct stscore *head, char *subname)
{
    FILE *fp;
    struct stscore *p;
    if ((fp = fopen(subname, "w")) == NULL)
    {
        printf("Can't open the file!\n");
        return ;
    }
    p = head;
    while (p != NULL)
    {
        fprintf(fp, "%s\t%s\t%d\t%s\n", p ->stnum, p ->subNum, p ->mark, p ->repair);
        p = p ->next;
    }
    fclose(fp);
}//fprintfstscore

//录入课程成绩
struct stscore * creatSubScore()
{
    struct subject *head_sub, *r, *a[20];
    struct stscore *head_sc, *p, *tail;
    struct student *head_st, *q;
    char term[3];
    char subname[20];
    int cnt;
    int mark, k, t;
    head_sc = (struct stscore *)malloc(sizeof(struct stscore));
    head_st = loadStudent();
    r = head_sub = loadSubject();

    tail = head_sc;
    printf("请输入学期：");
    scanf("%s", term);
    t = 0 + term[0] - '0';
    printf("          第%d学期课程\n", t);
    printf("    ========================\n");
    cnt = 1;
    while (r)
    {
        if (r ->term == t)
        {
            a[cnt] = r;
            printf("     %d -- %s\n", cnt++, r ->subName);
        }
        r = r ->next;
    }
    printf("     请选择：");
```

```
    scanf("%d", &t);
    strcpy(subname, a[t] ->subName);
    strcat(subname, "_");
    strcat(subname, term);
    strcat(subname, ".txt");

    system("cls");
    printf("     %s 课程成绩录入\n", a[t] ->subName);
    printf("   ========================\n");
    printf("请输入对应学号的成绩\n");
    printf("     学  号          成绩\n");
    q = head_st;
    while (q)
    {
        p = (struct stscore *)malloc(sizeof(struct stscore));
        tail ->next = p;
        tail = p;
        printf("   %s  :    ", q ->stnum);
        strcpy(p ->stnum ,q ->stnum);
        strcpy(p ->subNum, a[t] ->subNum);
        do
        {
            k = 1;
            scanf("%d", &mark);
            if (mark < 0 || mark > 100)
            {
                printf("成绩输入错误，请重新输入:");
                k = 0;
            }
        }while (k == 0);
        p ->mark = mark;
        if (p ->mark < 60)
            strcpy(p ->repair, "Y");
        else
            strcpy(p ->repair, "N");
        q = q ->next;
    }//while
    tail ->next = NULL;

    fprintfstscore(head_sc ->next, subname);
    printf("\n录入课程成绩成功！\n");
    printf("\n **按Enter返回！");
    return head_sc;
}//creatSubScore

//判断学号是否已存在
//存在返回1，不存在返回0
int isNumExist(struct stCondSc *head, char *stNum)
{
    struct stCondSc *q = head;
    while (q != NULL)
    {
        if (strcmp(q ->stnum, stNum) == 0)
            return 1;
        q = q ->next;
    }
    return 0;
}//isNumExist

//将学生品行成绩信息写入文件
void fprintfstCondSc(struct stCondSc *head)
{
    FILE *fp;
    struct stCondSc *p;
    if ((fp = fopen("stCondSc.txt", "w")) == NULL)
```

```
    {
        printf("Can't open the file!\n");
        return ;
    }
    p = head;
    while (p != NULL)
    {
        fprintf(fp, "%s\t%d\t%d\t%d\t%d\n", p ->stnum, p ->coumark, p ->headmark, p ->classsmark, p ->conductmark);
        p = p ->next;
    }
    fclose(fp);
}//fprintfstCondSc

//录入学生品行成绩信息
struct stCondSc* creatConductScore()
{
    struct stCondSc *head, *tail, *p;
    struct student *head_st, *q;
    int mark, k;
    head = (struct stCondSc*)malloc(sizeof(struct stCondSc));
    q = head_st = loadStudent();
    tail = head;

    printf("请输入学生品行成绩信息\n");
    while (q)
    {
        p = (struct stCondSc *)malloc(sizeof(struct stCondSc));
        tail ->next = p;
        tail = p;
        printf("学号：%s\n", q ->stnum);
        strcpy(p ->stnum ,q ->stnum);
        printf("辅导员打分：");
        do
        {
            k = 1;
            scanf("%d", &mark);
            if (mark < 0 || mark > 100)
            {
                printf("成绩输入错误，请重新输入:");
                k = 0;
            }
        }while (k == 0);
        p ->coumark = mark;

        printf("班主任打分：");
        do
        {
            k = 1;
            scanf("%d", &mark);
            if (mark < -1 || mark > 100)
            {
                printf("成绩输入错误，请重新输入:");
                k = 0;
            }
        }while (k == 0);
        p ->headmark = mark;

        printf("班级打分：");
        do
        {
            k = 1;
            scanf("%d", &mark);
            if (mark < 0 || mark > 100)
            {
                printf("成绩输入错误，请重新输入:");
```

```
            k = 0;
        }
    }while (k == 0);
    p ->classsmark = mark;
    p ->conductmark = p ->classsmark + p ->coumark + p ->headmark;
    printf("\n");
    q = q ->next;
  }//while
  fprintfstCondSc(head ->next);
  printf("\n录入品行成绩成功！\n");
  printf("\n **按Enter返回！");
  return head;
}//creatConductScore

//判断学号是否已存在
//存在返回1，不存在返回0
int isNumExist(struct cominfo *head, char *stNum)
{
  struct cominfo *q = head;
  while (q != NULL)
  {
    if (strcmp(q ->stnum, stNum) == 0)
      return 1;
    q = q ->next;
  }
  return 0;
}//isNumExist

//将学生综合信息写入文件
void fprintfStCominfo(struct cominfo *head)
{
  FILE *fp;
  struct cominfo *p;
  if ((fp = fopen("cominfo.txt", "w")) == NULL)
  {
    printf("Can't open the file!\n");
    return ;
  }
  p = head;
  while (p != NULL)
  {
    fprintf(fp, "%s\t%s\t%d\n", p ->stnum, p ->stname, p ->getcredit);
    fprintf(fp, "%s\t%d-%d-%d\t%d\n", p ->awdlev, p ->ayear, p ->amon, p ->aday, p ->amark);
    fprintf(fp, "%s\t%d-%d-%d\t%d\n", p ->puncate, p ->pyear, p ->pmon, p ->pday, p ->pmark);
    p = p ->next;
  }
  fclose(fp);
} //fprintfAwdPunScore

//录入学生综合信息
struct cominfo* creatAwdPunScore()
{
  struct cominfo *head, *tail, *p;
  struct student *head_st, *q;
  int t;
  head = (struct cominfo *)malloc(sizeof(struct cominfo));
  tail = head;
  q = head_st = loadStudent();

  printf("请输入学生综合信息\n");
  while (q)
  {
    p = (struct cominfo *)malloc(sizeof(struct cominfo));
    tail ->next = p;
    tail = p;
```

```
        printf("学号：%s\n", q ->stnum);
        printf("姓名：%s\n", q ->stname);
        strcpy(p ->stnum, q ->stnum);
        strcpy(p ->stname, q ->stname);
        printf("是否有奖励记录：");
        printf("1 - 是   0 - 否\n");
        scanf("%d", &t);
        if (t)
        {
            printf("请输入奖励类别：");
            scanf("%s", p ->awdlev);
            printf("请输入获奖时间(年 月 日)：");
            scanf("%d %d %d", &p ->ayear, &p ->amon, &p ->aday);
            printf("请输入奖励分值：");
            scanf("%d", &p ->amark);
        }
        else
        {
            strcpy(p ->awdlev, "无");
            p ->aday = 0;
            p ->amon = 0;
            p ->aday = 0;
            p ->amark = 0;
        }
        printf("是否有惩处记录：");
        printf("1 - 是   0 - 否\n");
        scanf("%d", &t);
        if(t)
        {
            printf("请输入惩处类别：");
            scanf("%s", p ->puncate);
            printf("请输入惩处时间：");
            scanf("%d %d %d", &p ->pyear, &p ->amon, &p ->pday);
            printf("请输入惩罚分值：");
            scanf("%d", &p ->pmark);
        }
        else
        {
            strcpy(p ->puncate, "无");
            p ->pyear = 0;
            p ->pmon = 0;
            p ->pday = 0;
            p ->pmark = 0;
        }
        printf("\n");
        q = q ->next;
    }//while

    fprintfStCominfo(head ->next);
    printf("录入学生综合信息成功！\n");
    printf("\n **按Enter返回！");
    return head;
}//creatAwdPunScore

//修改学生基本信息
struct student* modifyStudent()
{
    char stnum[15];
    char sex[6];
    struct student* head, *p;
    int key = 1, t;
    char ss1[5] = "男", ss2[5] = "女";

    printf("请输入要修改学生的学号：");
    do
```

```
    {
        scanf("%s", stnum);
        if (strlen(stnum) != 10)
        {
            printf("学号格式输入错误，请重新输入！\n");
            printf("请输入学号：");
        }
        else
            key = 0;
    }while (key == 1);

    head = loadStudent();
    p = head;
    while (p!=NULL && strcmp(p ->stnum,stnum)!=0)
    {
        p = p ->next;
    }
    if (p == NULL)
        printf("未找到，修改失败！\n");
    else
    {
        printf("请输入修改后的信息：\n");
        printf("请输入姓名：");
        scanf("%s", p ->stname);
        do
        {
            key = 1;
            printf("请输入性别：");
            scanf("%s", sex);
            if (strcmp(sex, ss1)==0 || strcmp(sex, ss2)==0)
                key = 0;
            else
                printf("性别输入错误，请重新输入！\n");
        }while(key == 1);
        strcpy(p ->stsex, sex);
        printf("请输入年龄：");
        scanf("%d", &p ->stage);
        printf("请输入宿舍号：");
        scanf("%s", p ->dornum);
        printf(" *确认修改? \n");
        printf("  1-确认  0-取消\n  ");
        scanf("%d", &t);
        if (t)
        {
            fprintfStudent(head);
            printf("修改成功！\n");
        }
    }//else
    printf("\n **按Enter返回！");
    return head;
}//modifyStudent

//修改课程信息
struct subject * modifySubject()
{
    struct subject *head, *p;
    char subnum[20];

    printf("请输入要修改课程的课程号：");
    scanf("%s", subnum);

    head = loadSubject();
    p = head;
    while (p!=NULL && strcmp(p ->subNum,subnum)!=0)
    {
```

```
        p = p ->next;
    }
    if (p != NULL)
    {
        printf("请输入修改后的课程信息：\n");
        printf("请输入课程名称：");
        scanf("%s", p ->subName);
        printf("请输入课程类别：");
        scanf("%s", p ->cate);
        printf("请输入课程学分：");
        scanf("%f", &p ->credit);
        printf("请输入课程所在学期：");
        scanf("%d", &p ->term);
        fprintfsubject(head);
        printf("修改成功！\n");
    }//if
    else
        printf("未找到，修改失败！\n");
    printf("\n **按Enter返回！");
    return head;
}//modifySubject

//从文件中读取课程成绩
struct stscore * loadStscore(char *modname)
{
    FILE *fp;
    struct stscore *head, *tail, *p;

    head = tail = NULL;
    if ((fp = fopen(modname, "r")) == NULL)
    {
        printf("Can't open the file!\n");
        exit(0);
    }
    while (!feof(fp))
    {
        p = (struct stscore *)malloc(sizeof(struct subject));
        fscanf(fp, "%s\t%s\t%d\t%s\n",  p ->stnum, p ->subNum, &p ->mark, p ->repair);
        if (head == NULL)
            head = p;
        else
            tail ->next = p;
        tail = p;
    }//while
    tail ->next = NULL;
    fclose(fp);
    return head;
}//loadSubject

//判断学号是否存在
int isNumExist(struct stscore *head, char *num)
{
    struct stscore *q = head;
    while (q != NULL)
    {
        if (strcmp(q ->stnum , num) == 0)
            return 1;
        q = q ->next;
    }
    return 0;
}//isNumExist

//修改课程成绩信息
struct stscore * modifySubScore()
{
```

```
    struct stscore *head, *p;
    struct subject *head_sub, *r, *a[10];
    char subname[20], stnum[15];
    int mark;
    int t, cnt;
    char term[3];

    printf("请输入学期：");
    scanf("%s", term);
    t = 0 + term[0] - '0';
    r = head_sub = loadSubject();
    printf("          第%d学期课程\n", t);
    printf("   ========================\n");
    cnt = 1;
    while (r)
    {
        if (r ->term == t)
        {
            a[cnt] = r;
            printf("     %d -- %s\n", cnt++, r ->subName);
        }
        r = r ->next;
    }
    printf("     请选择：");
    scanf("%d", &t);
    strcpy(subname, a[t] ->subName);
    strcat(subname, "_");
    strcat(subname, term);
    strcat(subname, ".txt");

    system("cls");
    printf("请输入学号：");
    scanf("%s", stnum);
    head = loadStscore(subname);
    p = head;
    while (p!=NULL && strcmp(p ->stnum, stnum) != 0)
    {
        p = p ->next;
    }
    if (p != NULL)
    {
        printf("请输入修改后的信息：\n");
        printf("请输入成绩：");
        scanf("%d", &mark);
        while (mark>100 || mark < 0)
        {
            printf("成绩格式错误，请重新输入！\n");
            printf("请输入成绩：");
            scanf("%d", &mark);
        }
        p ->mark = mark;
        if (p ->mark >= 60)
            strcpy(p ->repair, "N");
        else
            strcpy(p ->repair, "Y");
        fprintfstscore(head, subname);
        printf("\n **修改成功！\n");
    }//if
    else
        printf(" **未找到，修改失败！\n");
    printf("\n **按Enter返回！");
    return head;
}//modifySubScore

//从文件中读入学生品行信息
```