

重庆师范大学

《数据结构课程设计》

课程名称： 数据结构

题 目： 运动会计分系统

学 院： 计算机与信息科学学院

专业年级： 计算机科学与技术（3+2）2019 级

小组组长： 张鉴鑫

小组成员： 张鉴鑫、樵宝民、颜之灏、胡方元、邵珣、张聆钰

指导教师： 张万里 职称： 讲师

2022 年 1 月 6 日

目录

一、设计目的.....	2	-
二、问题描述.....	3	-
三、基本要求.....	3	-
四、概要设计.....	3	-
4.1 算法思路.....	3	-
4.2 工作分配.....	4	-
4.3 程序模块.....	4	-
4.4 程序功能图及功能说明.....	5	-
五、程序测试.....	9	-
5.1 多组数据输入与输出.....	9	-
5.2 操作过程.....	14	-
六、总结.....	15	-
七、附程序源码.....	16	-

运动会计分系统

一、 设计目的

数据结构作为一门学科主要研究数据的各种逻辑结构和存储结构，以及对数据的各种操作。因此，主要有三个方面的内容：数据的逻辑结构；数据的物理存储结构；对数据的操作（或算法）。通常，算法的设计取决于数据的逻辑结构，算法的实现取决于数据的物理存储结构。数据结构是信息的一种组织方式，其目的是为了提高算法的效率，它通常与一组算法的集合相对应，通过这组算法集合可以对数据结构中的数据进行某种操作。

在当今信息时代，信息技术已成为当代知识经济的核心技术。我们时刻都在和数据打交道。比如人们在外出工作时找最短路径，在银行查询存款、通过互联网查新闻、以及远程教育报名等，所有这些都在与数据发生关学校。实际上，现实世界中的实体经过抽象以后，就可以成为计算机上所处理的数据。

数据结构课程主要是研究非数值计算的程序设计问题中所出现的计算机操作对象以及它们之间的关学校和操作的学科。数据结构是介于数学、计算机软件 and 计算机硬件之间的一门计算机专业的核心课程，它是计算机程序设计、数据库、操作学校统、编译原理及人工智能等的重要基础，广泛的应用于信息学、学校统工程等各种领域。

学习数据结构是为了将实际问题中所涉及的对象在计算机中表示出来并对它们进行处理。通过课程设计可以提高学生的思维能力，促进学生的综合应用能力和专业素质的提高。课程设计的目的：

- 1) 了解并掌握数据结构与算法的设计方法，具备初步的独立分析和设计能力。
- 2) 提高程序设计和调试能力。学生通过上机实习，验证自己设计的算法的正确性。学会有效利用基本调试方法，迅速找出程序代码中的错误并且修改。
- 3) 培养算法分析能力。分析所设计算法的时间复杂度和空间复杂度，进一步提高程序设计水平。
- 4) 初步掌握软件开发过程的问题分析、学校统设计、程序编码、测试等基本方法和技能。

二、 问题描述

参加运动会有 n 个学校，学校编号为 $1.....n$ 。比赛分成 m 个男子项目，和 w 个女子项目。项目编号为男子 $1.....m$ ，女子 $m+1.....m+w$ 。不同的项目取前五名或前三名积分；取前五名的积分分别为：7、5、3、2、1，前三名的积分分别为：5、3、2；

设计程序达到以下要求：

- 1) 产生各院系的成绩单，内容包括各院系取得的每项成绩的项目号、名次（成绩）、姓名和得分；
- 2) 产生团体总分报表，内容包括院系编号、男子团体总分和团体总分。

三、基本要求

- 1) 能够方便快捷的输入各个项目的前三名或前五名的成绩
- 2) 能够按学校编号、学校总分、男女团体总分进行排序并输出
- 3) 能够精确的统计各学校的总分
- 4) 能够按学校编号查询学校某个项目的基本情况,能够按项目编号查询取得前三或前五名的学校

四、概要设计

4.1 算法思路

- 1) 可以输入各个项目的前三名或前五名的成绩
- 2) 可以按学校编号、学校总分、男女团体总分排序输出
- 3) 可以统计出各学校的总分
- 4) 可以按学校编号查询学校某个项目的情况
- 5) 可以按项目编号查询取得前三或前五名的学校

- 6) 数据存入文件中，并且可以随时查询
- 7) 学校统操作规定：
 - i. 输入数据形式和范围：可以输入学院（学校）的名称和运动会项目的名称
 - ii. 输出形式：有中文提示，各学院（学校）分数为整形
- 8) 界面要求：有合理的操作提示，每个功能都可以设立菜单，并且根据提示，可以完成相关的功能要求
- 9) 存储结构：学生自己可以根据学校统功能要求自己设计，但是要求运动会的相关数据要存储在数据文件中，实时保存

4.2 工作分配

- 1) 框架设计：由颜之灏进行构思，同时参考其他人的意见。
- 2) 系统设计：由樵宝民进行框架的编写。
- 3) 程序设计：由胡方元进行各项功能的编辑。
- 4) 程序调试：由张鉴鑫进行程序的调试与调整。
- 5) 文档制作，由邵琦完成文档的编辑工作。
- 6) 文档制作，由张聆钰完成 ppt 的编辑工作。

4.3 程序模块

(1)运动项目定义:

```
typedef struct Sport    //运动项目结构
{
    char name[20];        //运动项目名称
    int isboy;            //0 为女项目,1 为男项目
    int is3;              //0 为取前五名,1 为取前五名
    int number;           //项目编号
    int first;            //第一名学校的编号
    int second;           //第二名学校的编号
    int third;            //第三名学校的编号
```

```

        int fourth;           //第四名学校的编号
        int fifth;           //第五名学校的编号
        Sport *next;

    }Sport;

```

运动项目结构体定义如下：char 型的数组，int 型的 number, first, second, third, fourth, fifth 以及 sport 类型的指针 next。

(2)学校的定义:

```

typedef struct College      //学校的结构
{
    char name[20];         //学校的名称
    int number;            //学校的编号
    int boy;               //男子团体总分
    int girl;              //女子团体总分
    College *next;

}College;

```

将学校自定义为一个结构体命名为 College；学校的结构体定义如下：char 型的数组，int 型的变量 unumber ， int 型的变量 boy,girl 以及 College 类型的指针 next。

4.4 程序功能图及功能说明

1) 程序功能图

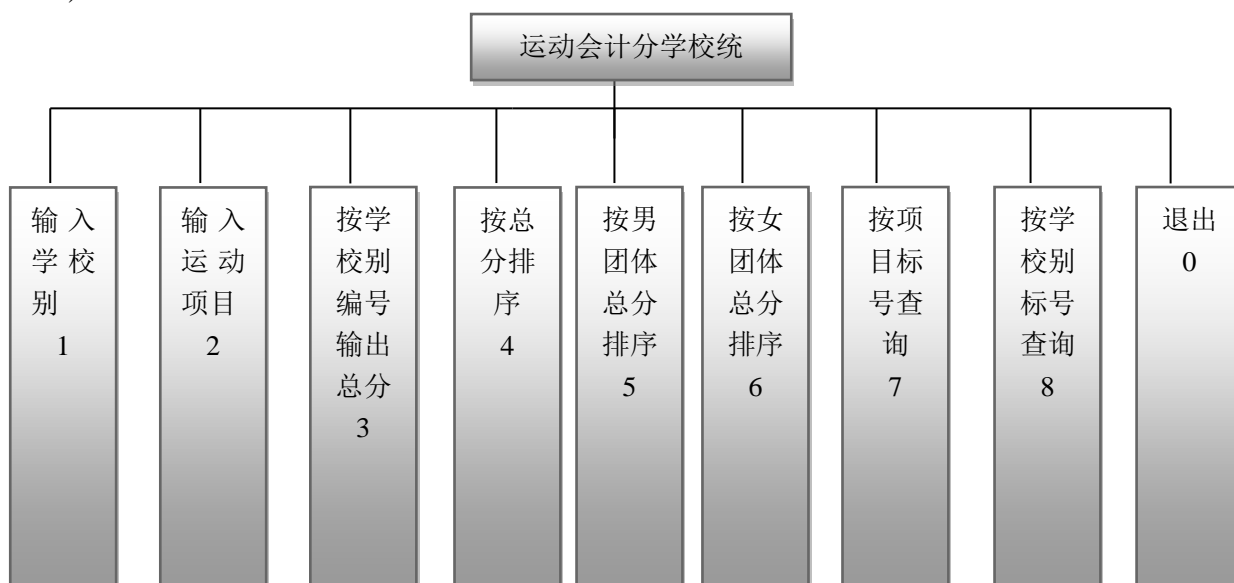


图 1 程序功能图

2) 程序功能说明

如上图标示：

- 1 直接输入学校别名称
- 2 直接输入运动项目的名称，编号，及男女项目代码，以便之后执行的简单快捷
- 3 直接跳转到“按学校别输出总分”的界面，查看按学校别编号输出的总分
- 4 直接跳转到“按总分排序”的界面，查看按总分排列的顺序
- 5 直接跳转到“按男团体输出总分”的界面，查看按男团体总分排列的顺序
- 6 直接跳转到“按女团体输出总分”的界面，查看按女团体总分排列的顺序
- 7 输入所要查出的项目的编号，即可查看所排列的顺序
- 8 输入所要查出的学校别的标号，即可查看所排列的顺序
- 0 退出键，保存文本，关闭程序

3) 程序流程图

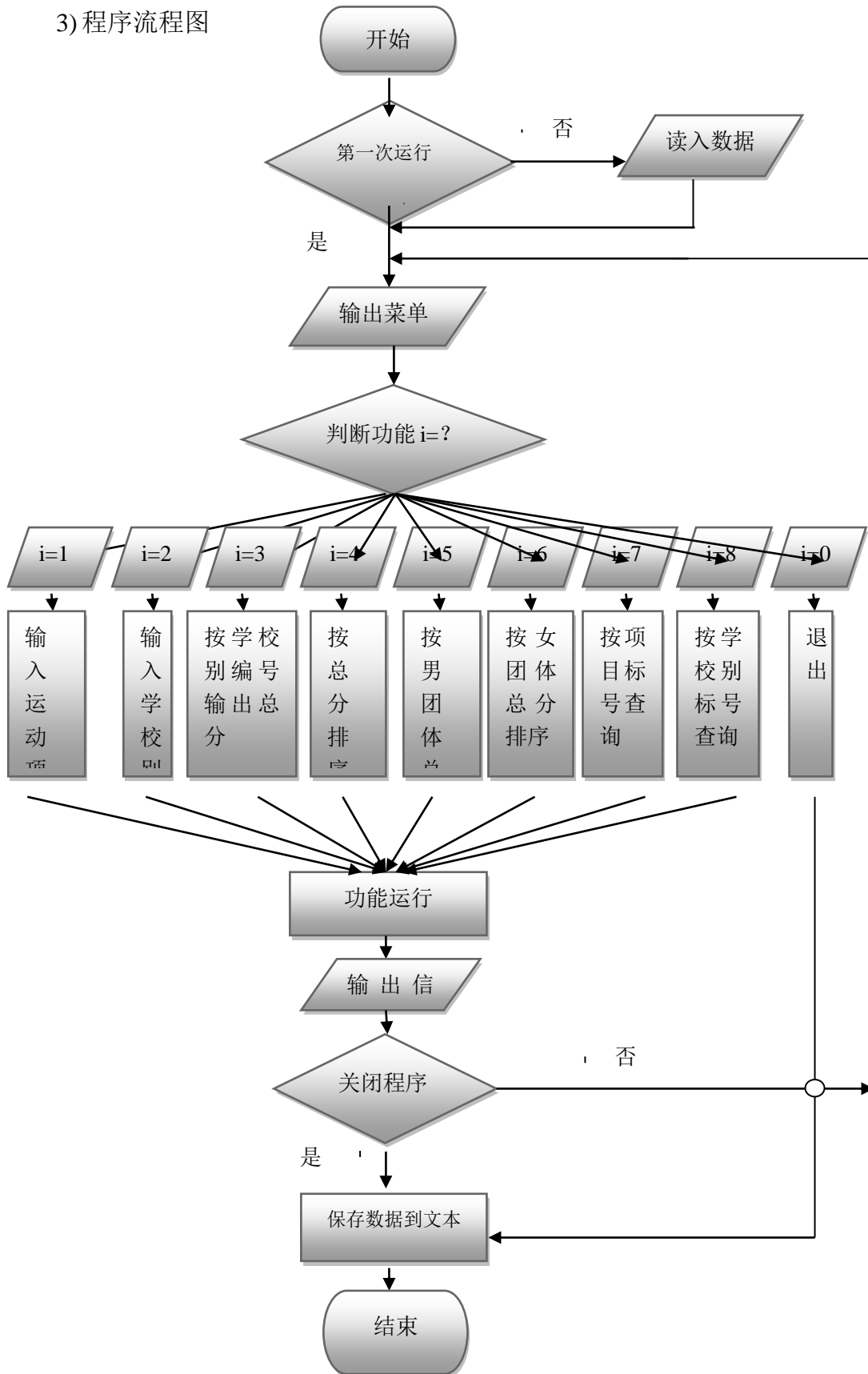


图 2 程序流程图

如上图所示：本学校统在第一次运行时，将直接输出菜单，如果不是第一次运行，则在学校统运行的初期会进行数据的读取。然后，可以根据输出的菜单对需要进行的功能进行选择，选择完毕后将该功能进行运行。例如：输入运动项目，输入学校别，按学校别编号出总分，按男女团体总分数输出，按项目标号查询，按学校别标号查询，退出。运行完毕后，将数据输出到屏幕，如果要结束学校统则自动将数据保存到相应的文本，否则继续输出菜单。

4) 算法设计

主要算法的设计思想：

本程序主要是使用链表来实现操作。一个运动会包括运动项目和参加运动会的成员。因此构造两个链表 **College**, **Sport**。根据输入的不同选择不同的功能，有输入运动项目，输入学校别，按学校别编号输出总分，按总分排序，按男团体总分排序，按女团体总分排序，按项目标号查询，按学校别标号查询，退出。为了操作的方便，并且能够保存输入数据，所以通过文件操作来实现数据的写和读。每次添加新的数据后都要输入 0 退出，数据才能保存。对于总分的排序使用了冒泡排序。为了使整个程序看起来更加友好，又添加了程序启动画面。

5) 相关知识

链表是一种物理存储单元上非连续、非顺序的存储结构，数据元素的逻辑顺序是通过链表中的指针链接次序实现的。链表由一学校列结点（链表中每一个元素称为结点）组成，结点可以在运行时动态生成。每个结点包括两个部分：一个是存储数据元素的数据域，另一个是存储下一个结点地址的指针域。相比于线性表顺序结构，链表比较方便插入和删除操作。

冒泡排序，是指计算机的一种排序方法，它的时间复杂度为 $O(n^2)$ ，虽然不及堆排序、快速排序的 $O(n \log n)$ ，底数为 2，但是有两个优点：1. “编程复杂度”很低，很容易写出代码；2. 具有稳定性，这里的稳定性是指原序列中相同元素的相对顺序仍然保持到排序后的序列，而堆排序、快速排序均不具有稳定性。不过，一路、二路归并排序、不平衡二叉树排序的速度均比冒泡排序快，且具有稳定性，但速度不及堆排序、快速排序。冒泡排序是经过 $n-1$ 趟子排序完成的，第 i 趟子排序从第 1 个数至第 $n-i$ 个数，若第 i 个数比后一个数大（则升序，小则降序）则交换两数。

五、程序测试

5.1 多组数据输入与输出

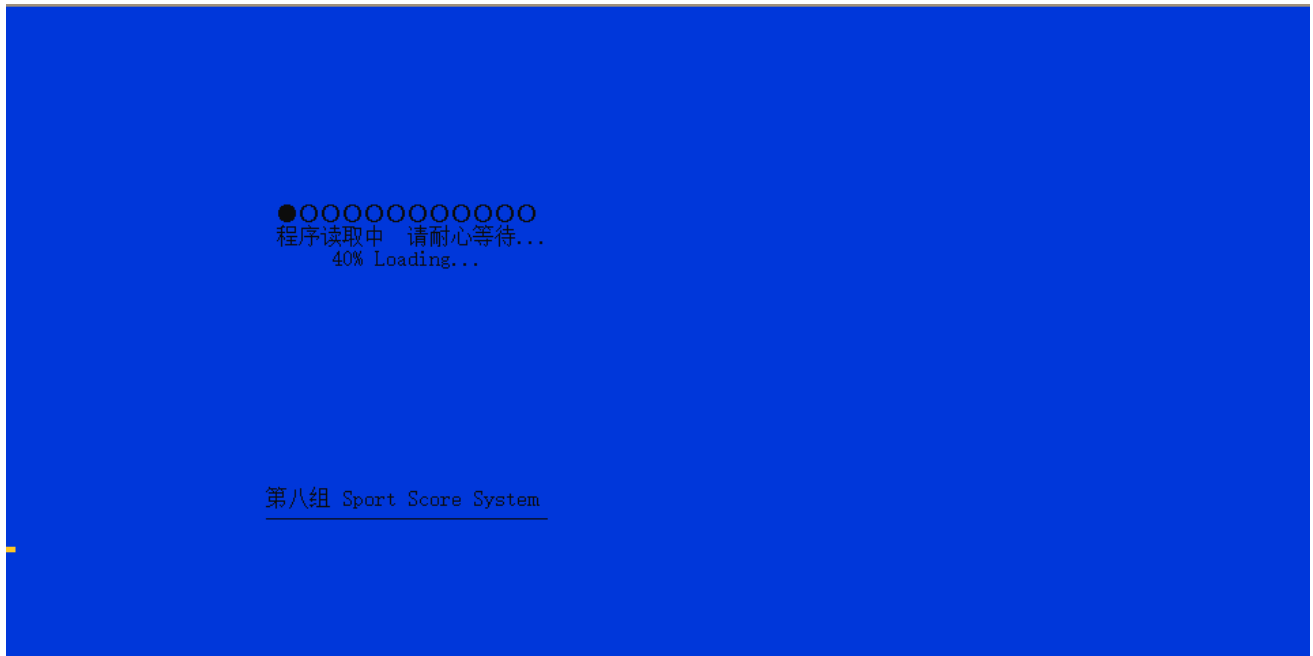


图 3 程序页面图

图 3 为本程序的页面图，为了使程序更加人性化，加入了“程序读取，请耐心等待”的字样。



图 4 程序菜单图

图 4 为本程序的菜单，显示了本程序的主要功能，可以让使用者轻松，快捷的，选择所要进行的操作。



图 5 程序输入学校别图

如图 5 所示，在菜单中选择“1”按回车键，弹出输入一个学校，输入“计算机”，编号“1”单击回车键，返回菜单。



图 6 程序输入运动项目图



图 7 程序输入运动项目图

如图 6, 图 7 所示, 在菜单中选择“2”, 按回车键, 弹出“输入一个运动项目”, 输入“100 米”, 编号“1” 类型“0” 等操作之后。按回车键, 返回菜单。

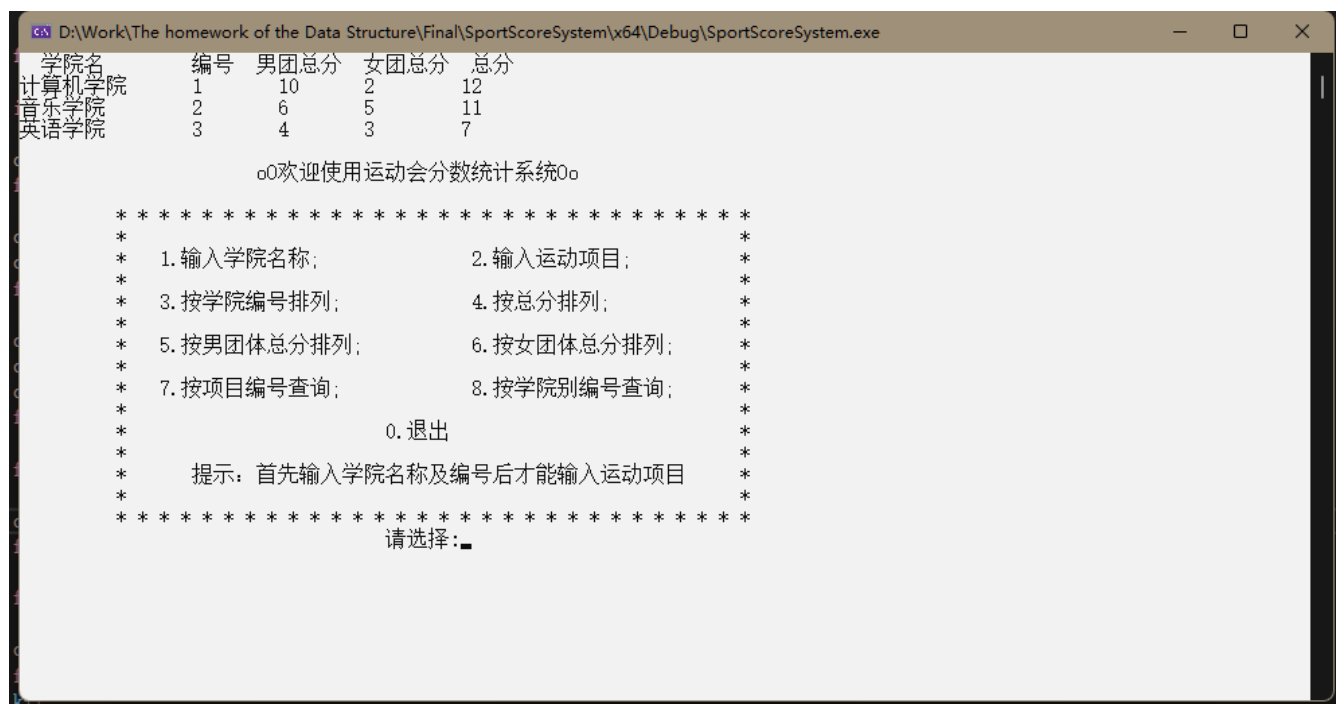


图 8 程序输入学校别编号查询图

如图 8 所示，在菜单中选择“3”，按回车键，展示按学院的编号排列。

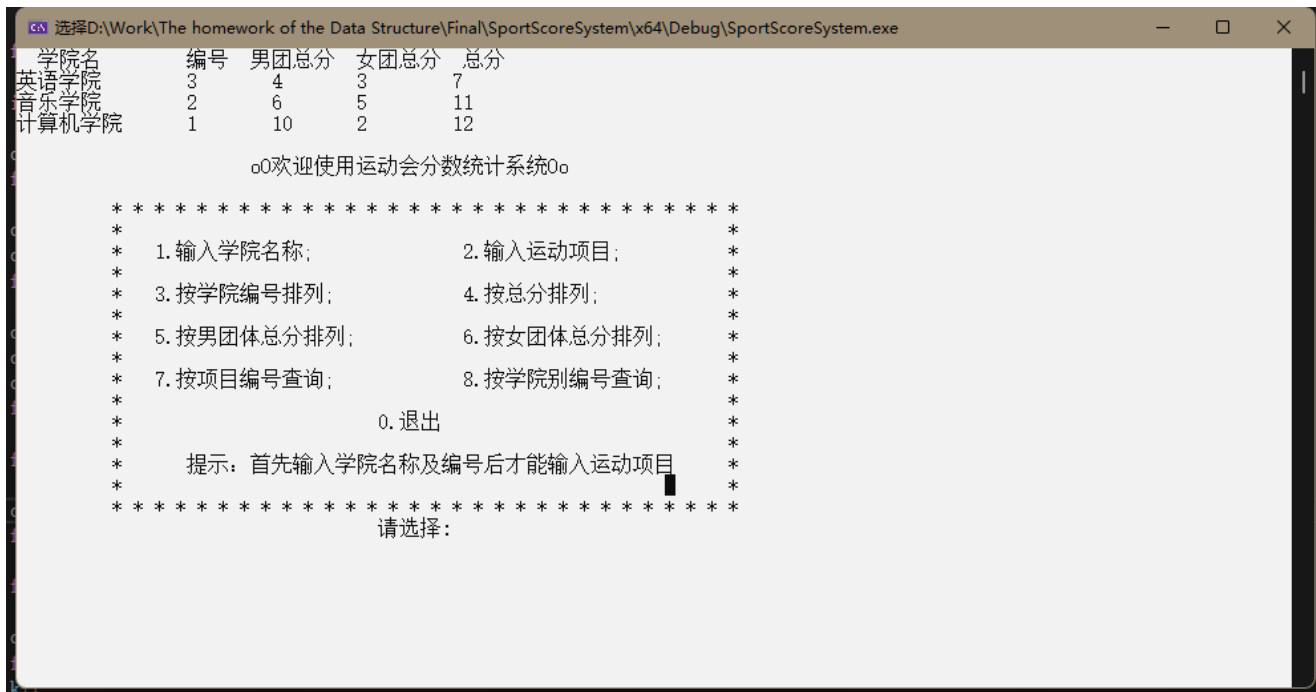


图 9 程序输入总分排列查询图

如图 9 所示，在菜单中选择“4”，按回车键，展示按总分排列。

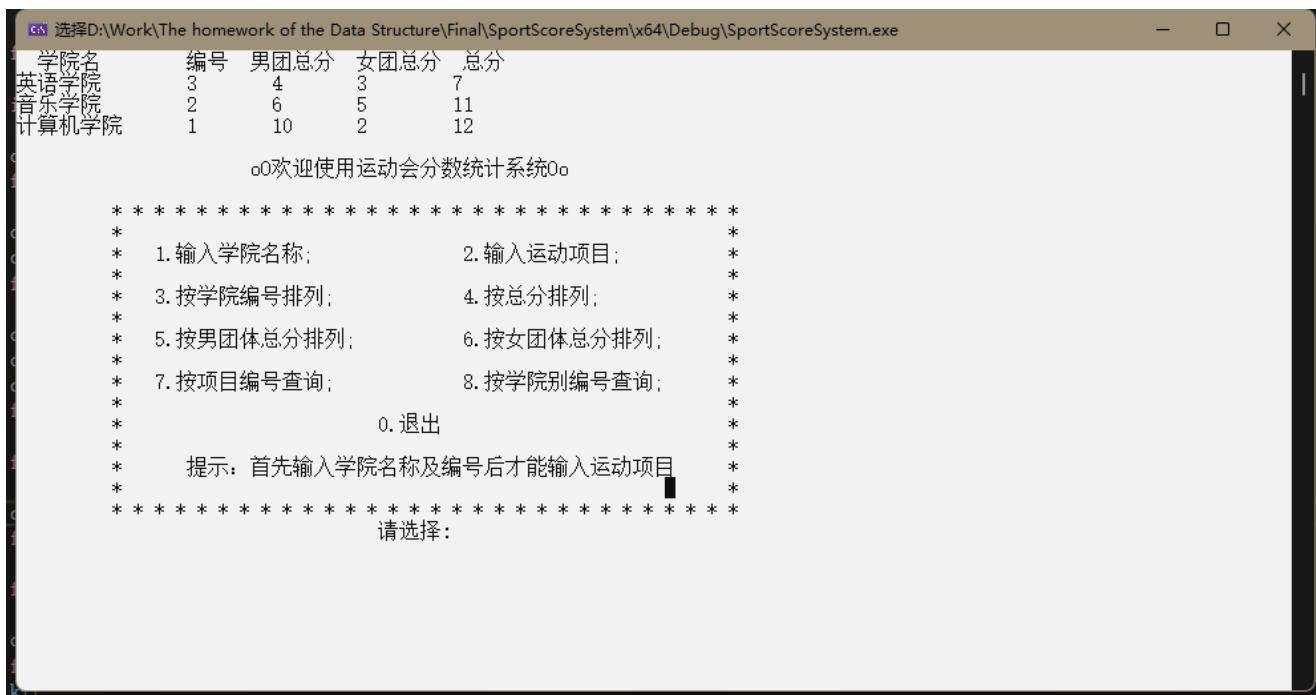


图 10 程序输入男团体排列查询图

如图 10 所示，在菜单中选择“5”，按回车键，展示男团体总分排列。

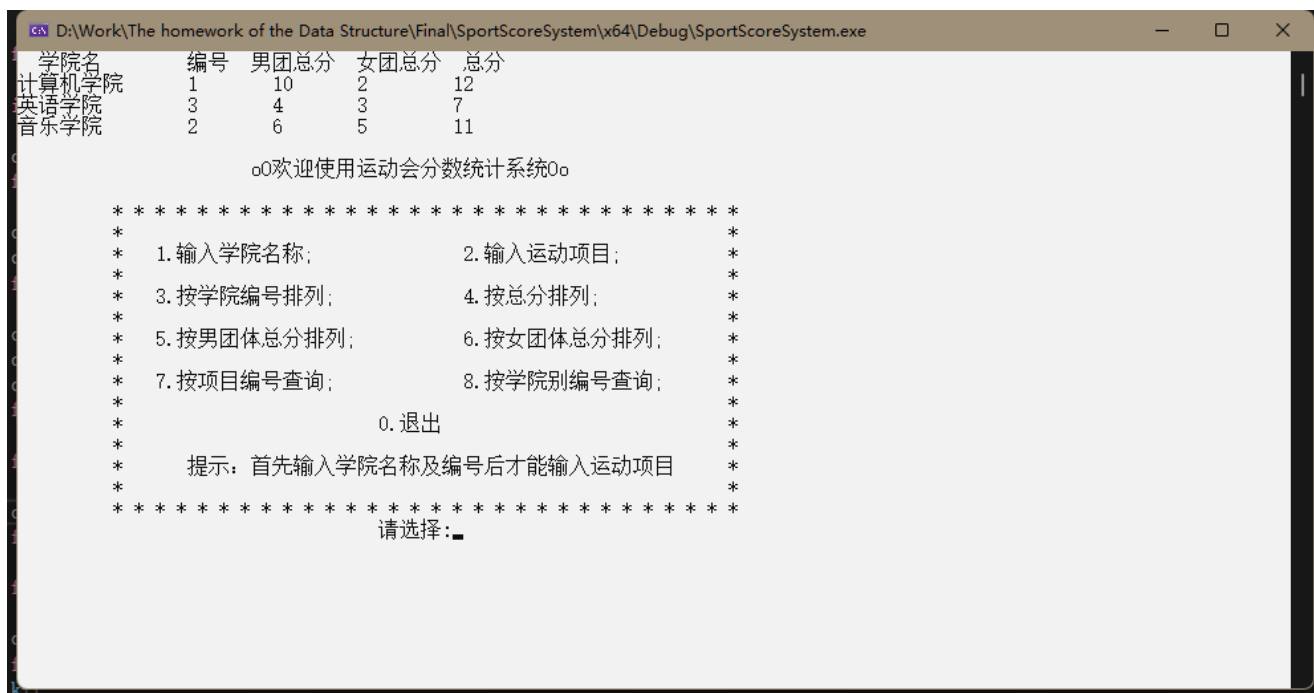


图 11 程序输入女团体排列查询图

如图 11 所示，在菜单中选择“6”，按回车键，展示女团体总分排列。



图 12 程序输入项目编号排列查询图

如图 12 所示，在菜单中选择“7”，按回车键，展示项目编号查询排列。



图 13 程序输入学院编号排列查询图

如图 12 所示，在菜单中选择“7”，按回车键，展示学院编号查询排列。

5.2 操作过程

(1) 用户进入程序启动界面, 等 load 完成后进入主菜单. 主菜单有以下 9 个选项。

1. 输入学校别
2. 输入运动项目
3. 按学校别编号输出总分
4. 按总分排序
5. 按男团体总分排序
6. 按女团体总分排序
7. 按项目编号查询
8. 按学校别编号查询
0. 退出

(2) 输入 0 至 8 中的一个数。

(3) 输入 1, 进行添加学校操作, 分别输入学校的名称和编号。

(4) 输入 2, 进行添加运动项目操作, 分别输入项目名称, 编号, 项目类型, 名次选取, 排名情况。

- (5) 输入 0, 则退出学校统, 并且保存输入的信息. 每次添加完都要输入 0 退出学校统才有效。
- (6) 输入 3, 输出按学校编号总分排名。
- (7) 输入 4, 输出按总分排名。
- (8) 输入 5, 输出按男团体总分排序。
- (9) 输入 6, 输出按女团体总分排序。
- (10) 输入 7, 进入按项目编号查询界面. 输入要查询的项目编号。
- (11) 输入 8, 进入按学校别编号查询界面, 输入要查询的学校的编号。

六、总结

两天的课程设计结束了, 在这次的课程设计中不仅检验了我们所学习的知识, 也培养了我们如何去把握一件事情, 如何去做一件事情, 又如何完成一件事情。

这次的课程设计, 加强了我们动手、思考和解决问题的能力。巩固和加深了对数据结构的理解, 提高综合运用本课程所学知识的能力。培养了我们选用参考书, 查阅手册及文献资料的能力。培养独立思考, 深入研究, 分析问题、解决问题的能力。通过实际编译学校统的分析设计、编程调试, 掌握应用软件的分析方法和工程设计方法。通过课程设计, 培养了我们严肃认真的工作作风, 逐步建立正确的生产观念、经济观念和全局观念。而且做课程设计同时也是对课本知识的巩固和加强, 平时看课本时, 有些问题就不是很能理解, 做完课程设计, 那些问题就迎刃而解了。而且还可以记住很多东西。认识来源于实践, 实践是认识的动力和最终目的, 实践是检验真理的唯一标准。所以这个期末测试之后的课程设计对我们的作用是非常大的。

这次的课程设计使我们懂得了理论与实际相结合是很非常重要的, 只有理论知识是远远不够的, 只有把所学的理论知识与实践结合起来, 从理论中得出结论, 才能真正为社会服务, 从而提高自己的实际动手能力和独立思考的能力。在整个设计过程中, 构思是很花费时间的。调试时经常会遇到这样那样的错误, 有的是因为粗心造成的语法错误。当然, 很多也时用错了方法, 总是实现不了。同时在设计的过程中发现了自己的不足之处, 对以前所学过的知识理解得不够深刻, 掌握得不够牢固。

通过这次的课程设计, 让我们更加了解到数据结构的重要性。以及它对我们专业的发展发挥的作用。对我们而言, 知识上的收获很重要, 但精神上的丰收更加可喜。让我们知道了学无止境的道理。我们每一个人永远不能满足于现有的成就, 人生就像在爬山, 一座山峰的

后面还有更高的山峰在等着你。挫折是一份财富，经历是一份拥有。这次课程设计必将成为我人生旅途上一个非常美好的回忆！同时在做课程设计时要能够从多方面去考虑，去研究，用多种算法去实现要求。此次课程设计，学到了很多课内学不到的东西，比如独立思考解决问题，出现差错的随机应变，这些都让我们受益非浅，今后的制作应该能够更轻松，自己也都能够解决并高质量的完成项目。

七、附程序源码

```
#include<iostream>

#include<iomanip>

#include<string.h>

#include<fstream>

#include<stdlib.h>

#include<ctype.h>

#include<stdio.h>

#include<conio.h>

using namespace std;

typedef struct College    //学院的结构
{
    char name[20];        //学院的名称
    int number;           //学院的编号
    int boy;              //男子团体总分
    int girl;             //女子团体总分
    College* next;
}College;

typedef struct Sport      //运动项目结构
```

```

{

    char name[20];          //运动项目名称

    int isboy;              //0 为女项目,1 为男项目

    int is3;                //0 为取前五名,1 为取前五名

    int number;             //项目编号

    int first;              //第一名学院的编号

    int second;             //第二名学院的编号

    int third;              //第三名学院的编号

    int fourth;             //第四名学院的编号

    int fifth;              //第五名学院的编号

    Sport* next;

}Sport;

int getint(int a)          //字符转换成数字

{

    return (int)(a - '0');

}

College* head1;

//-----启动画面函数-----

void Cover()

{

    system("color 10");

    char line[] = { "程序读取中 请耐心等待..." };

    char bar[] = { "...." };

    int i, j, k = 0, x = 0, y = 0;

    for (i = 0; i <= strlen(line) / 2;)

    {

        system("cls");
    }
}

```

```

for (j = 0; j < 9; j++)    //改变行坐标

    cout << endl;

for (j = 0; j < (75 - strlen(line)) / 2; j++) //改变列坐标

    cout << " ";

for (j = 1; j <= i; j++)    //进度显示器

    cout << "●";

for (x = strlen(line) / 2; x > i; x--)

    cout << "○";

if (k == 4)

    i++;

cout << endl;

for (j = 0; j < (75 - strlen(line)) / 2; j++)    //行坐标定位

    cout << " ";

cout << line;    //输出线条

cout << endl;

for (j = 0; j < (65 - strlen(bar)) / 2; j++)

    cout << " ";

cout << (i + 7) * 5 << "% Loading";

cout.write(bar, k);

cout << endl;

for (j = 0; j < 10; j++)

    cout << endl;

for (j = 0; j < 24; j++)

    cout << " ";

cout << "第八组 Sport Score System" << endl;

for (j = 0; j < 24; j++)

    cout << " ";

```

```

        for (j = 0; j <= 24; j++)

            cout << "—";

        cout << endl;

        for (j = 0; j < 10000000; j++); //延时效果

        k++;

        if (k > 4)

            k = 0;

    }

}

```

```

void college_add()           //添加学院
{

    College* p;

    int mark = 0;

    p = new College;

    cout << "请输入学院的名称:";

    cin >> p->name;

    char c;

    while (mark != 1)

    {

        cout << "请输入学院的编号:";

        cin >> c;

        if (!isdigit(c)) //是否为数字

        {

            cout << "数据非法" << endl;

        }

        else

```

```

        {

            mark = 1;

            p->number = c;

        }

    }

    p->boy = 0;

    p->girl = 0;

    p->next = head1->next;

    head1->next = p;

    cout << "成功添加了一个学院" << endl;

}

int college_getlong(College* first)//得到链表长度

{

    int i = 0;

    while (first->next != NULL)

    {

        i++;

        first = first->next;

    }

    return i;

}

void college_write()//将学院数据写入文本

{

    College* p;

    p = head1;

    p = p->next;

    ofstream outfile("College.txt", ios::out);

```

```

    outfile << college_getlong(p) + 1 << " ";

    while (p != NULL)
    {
        outfile << p->name << " " << p->number << " " << p->boy << " " << p->girl << " ";

        p = p->next;
    }

    outfile.close();

    cout << "学院数据保存成功!" << endl;
}

void college_read()//从文本读入学院数据
{
    int i;

    ifstream infile("College.txt", ios::in);

    infile >> i;

    while (i > 0)
    {
        College* p;

        p = new College;

        infile >> p->name >> p->number >> p->boy >> p->girl;

        p->next = head1->next;

        head1->next = p;

        i--;
    }

    cout << "学院数据读取成功!" << endl;
}

void college_output(College* p)//输出学院
{

```

```

    cout << "  学院名          编号  男团总分  女团总分  总分\t\n";

    while (p)

    {

        cout << p->name << "          \t" << getint(p->number) << "\t" << p->boy << "\t" << p->girl << "\t"
" << (p->girl + p->boy) << endl;

        p = p->next;

    }

}

int college_isexist(int a)//检验学院是否存在

{

    int b = 0;

    College* p;

    p = head1;

    p = p->next;

    while (p)

    {

        if (p->number == a)

        {

            return 1;

        }

        p = p->next;

    }

    return 0;

}

void college_show(int a)//输出所有学院

{

    College* p;

```

```

p = head1;

p = p->next;

while (p)
{
    if (p->number == a)
    {
        cout << p->name << " ";

        return;
    }

    p = p->next;
}

cout << " 无 ";
}

```

void college_search(int a)//按编号搜索学院

```

{
    College* p;

    p = head1;

    p = p->next;

    while (p)
    {
        if (p->number == a)
        {
            cout << "学院名:" << p->name << " " << "男子团体总分:" << p->boy << " " << "女子团
体总分:" << p->girl << " " << "总分:" << (p->boy + p->girl) << " ";

            return;
        }

        p = p->next;
    }
}

```



```

    }

    cout << "无此编号";

}

void college_addmark(int a, int b, int c)//a 为分数,b 为学院编号,c=1 表示男,c=0 表示女
{
    College* p;

    p = head1;

    p = p->next;

    while (p)
    {
        if (p->number == b)
        {
            if (c == '1')
            {
                p->boy = p->boy + a;
            }

            else
            {
                p->girl = p->girl + a;
            }
        }

        p = p->next;
    }
}

void college_order(College* temp, int type) //type=0 按总分,type=1 按男总分,type=2 按女总分,
{
    College* p, * q, * small, * temp1;

```

```

temp1 = new College;

temp1->next = NULL;

p = temp;

while (p)
{
    small = p;

    q = p->next;

    while (q)
    {
        switch (type)
        {
            case 0:
                if ((q->boy + q->girl) < (small->girl + small->boy))
                {
                    small = q;
                }

                break;

            case 1:
                if (q->boy < small->boy)
                {
                    small = q;
                }

                break;

            case 2:
                if (q->girl < small->girl)
                {
                    small = q;
                }

```

```

    }

    break;

default:

    cout << "error" << endl;

}

if (small != p)
{
    temp1->boy = p->boy;

    p->boy = small->boy;

    small->boy = temp1->boy;

    temp1->girl = p->girl;

    p->girl = small->girl;

    small->girl = temp1->girl;

    strcpy(temp1->name, p->name);

    strcpy(p->name, small->name);

    strcpy(small->name, temp1->name);

    temp1->number = p->number;

    p->number = small->number;

    small->number = temp1->number;    //将学院的名字互换

}

q = q->next;

}

p = p->next;

}

}

Sport* head2;

```

```
int sport_isexist(int a)    //检查运动项目(编号)是否已经存在
```

```
{
```

```
    int b = 0;
```

```
    Sport* p;
```

```
    p = head2;
```

```
    p = p->next;
```

```
    while (p)
```

```
    {
```

```
        if (p->number == a)
```

```
        {
```

```
            return 1;
```

```
        }
```

```
        p = p->next;
```

```
    }
```

```
    return 0;
```

```
}
```

```
void sport_add()           //添加项目
```

```
{
```

```
    Sport* p;
```

```
    int mark = 0;
```

```
    p = new Sport;
```

```
    cout << "请输入项目名称:";
```

```
    cin >> p->name;
```

```
    char c;
```

```
    while (mark != 1)
```

```
    {
```

```
        cout << "请输入项目编号:";
```

```

    cin >> c;

    if (!isdigit(c))
    {
        cout << "数据非法" << endl;
    }
    else
    {
        if (sport_isexist(c))
        {
            cout << "该编号已存在" << endl;
        }
        else
        {
            mark = 1;

            p->number = c;
        }
    }
}

mark = 0;

while (mark != 1)
{
    cout << "请输入项目类型(0 为女子项目,1 为男子项目):";

    cin >> c;

    p->isboy = (int)(c - '0');//字符转换成数字

    if (!isdigit(c))
    {
        cout << "数据非法" << endl;
    }
}

```

```

    }

    else if (p->isboy < 0 || p->isboy>1)
    {
        cout << "数据非法" << endl;
    }

    else
    {
        mark = 1;

        p->isboy = c;
    }
}

mark = 0;

while (mark != 1)
{
    cout << "请输入项目名次情况(0 为取前 3 名,1 为取前 5 名):";

    cin >> c;

    p->is3 = (int)(c - '0');

    if (!isdigit(c))
    {
        cout << "数据非法" << endl;
    }

    else if (p->is3 < 0 || p->is3>1)
    {
        cout << "数据非法" << endl;
    }

    else
    {

```

```

        mark = 1;

        p->is3 = c;
    }
}

mark = 0;

while (mark != 1)
{
    cout << "请输入第一名的学院的编号:";

    cin >> c;

    if (!isdigit(c))
    {
        cout << "数据非法" << endl;
    }
    else
    {
        if (!college_isexist(c))
        {
            cout << "该学院不存在,请先添加";
        }
        else
        {
            mark = 1;

            p->first = c;

            if (p->is3 == '0')

                college_addmark(5, c, p->isboy);

            else

                college_addmark(7, c, p->isboy);
        }
    }
}

```

```

        }

    }

}

mark = 0;

while (mark != 1)

{

    cout << "请输入第二名的学院的编号:";

    cin >> c;

    if (!isdigit(c))

    {

        cout << "数据非法" << endl;

    }

    else

    {

        if (!college_isexist(c))

        {

            cout << "该学院不存在,请先添加";

        }

        else

        {

            mark = 1;

            p->second = c;

            if (p->is3 == '0')

                college_addmark(3, c, p->isboy);

            else

                college_addmark(5, c, p->isboy);

        }

    }

}

```



```

    }
}

mark = 0;

while (mark != 1)
{
    cout << "请输入第三名的学院的编号:";

    cin >> c;

    if (!isdigit(c))
    {
        cout << "数据非法" << endl;
    }
    else
    {
        if (!college_isexist(c))
        {
            cout << "该学院不存在,请先添加";
        }
        else
        {
            mark = 1;

            p->third = c;

            if (p->is3 == '0')
                college_addmark(2, c, p->isboy);
            else
                college_addmark(3, c, p->isboy);
        }
    }
}

```

```

}

mark = 0;

if (p->is3 == '1')
{
    while (mark != 1)
    {
        cout << "请输入第四名的学院的编号:";

        cin >> c;

        if (!isdigit(c))
        {
            cout << "数据非法" << endl;
        }
        else
        {
            if (!college_isexist(c))
            {
                cout << "该学院不存在,请先添加";
            }
            else
            {
                mark = 1;

                p->fourth = c;

                college_addmark(2, c, p->isboy);
            }
        }
    }

    mark = 0;
}

```

```

while (mark != 1)
{
    cout << "请输入第五名的学院的编号:";

    cin >> c;

    if (!isdigit(c))
    {
        cout << "数据非法" << endl;
    }
    else
    {
        if (!college_isexist(c))
        {
            cout << "该学院不存在,请先添加" << endl;
        }
        else
        {
            mark = 1;

            p->fifth = c;

            college_addmark(1, c, p->isboy);
        }
    }
}

else
{
    p->fourth = '0';

    p->fifth = '0';
}

```

```

    }

    p->next = head2->next;

    head2->next = p;

    cout << "成功添加了一个运动项目" << endl;

}

int sport_getlong(Sport* first)                //得到项目链表长度
{
    int i = 0;

    while (first->next != NULL)

    {

        i++;

        first = first->next;

    }

    return i;

}

void sport_write()                            //将项目数据写入文本文档
{

    Sport* p;

    p = head2;

    p = p->next;

    ofstream outfile("Sport.txt", ios::out);

    outfile << sport_getlong(p) + 1 << " ";

    while (p != NULL)

    {

        outfile << p->name << " " << p->number << " " << p->isboy << " " << p->is3 << " " << p->first
        << " " << p->second << " " << p->third << " " << p->fourth << " " << p->fifth << " ";

        p = p->next;
    }
}

```

```

    }

    outfile.close();

    cout << "体育项目数据保存成功!" << endl;
}

void sport_read()                                //从文本读取项目数据
{
    int i;

    ifstream infile("Sport.txt", ios::in);

    infile >> i;

    while (i > 0)
    {
        Sport* p;

        p = new Sport;

        infile >> p->name >> p->number >> p->isboy >> p->is3 >> p->first >> p->second >> p->third >>
p->fourth >> p->fifth;

        p->next = head2->next;

        head2->next = p;

        i--;
    }

    cout << "体育项目数据读取成功!" << endl;
}

void sport_output(Sport* p)                      //输出项目的情况
{
    cout << "name  " << "\t" << "Num" << "    " << "B/G" << "    " << "  3/5" << "    " << "
first" << "    " << "second" << "    " << "third" << "    " << "fourth" << "    " << "fifth" << "    "
<< endl;

    while (p)

```

```

{
    cout << p->name << "\t" << " " << getint(p->number) << " " << getint(p->isboy) << "
" << getint(p->is3) << " " << " ";

    college_show(p->first);

    college_show(p->second);

    college_show(p->third);

    college_show(p->fourth);

    college_show(p->fifth);

    //printf("\n");

    p = p->next;

    cout << "\n";

}

cout << endl;
}

void sport_search(int a)                //搜索项目
{
    Sport* p;

    p = head2;

    p = p->next;

    while (p)
    {
        if (p->number == a)
        {
            cout << "项目名称:" << p->name << endl << "项目类型:";

            if (p->isboy == 1)
            {

```

```

        cout << "男子项目";

    }

    else

    {

        cout << "女子项目";

    }

    cout << endl << "第一名:";

    college_show(p->first);

    cout << endl << "第二名:";

    college_show(p->second);

    cout << endl << "第三名:";

    college_show(p->third);

    cout << endl << "第四名:";

    college_show(p->fourth);

    cout << endl << "第五名:";

    college_show(p->fifth);

    return;

}

p = p->next;

}

cout << "无此编号";

}

void main()                                //运动会程序主函数

{

    Cover();

    system("color F0");    //改变背景,前景色

    head1 = new College;

```

```

head1->next = NULL;

head2 = new Sport;

head2->next = NULL;

//school_add();

sport_read();

college_read();

//sport_add();

College* p1;

Sport* p2;

p1 = head1;

p1 = p1->next;

p2 = head2;

p2 = p2->next;

char choose;

char temp;

//string ch=" ";

int a = 1;

while (a != 0)

{

    cout << endl;

    cout << "                                oO 欢迎使用运动会分数统计系统 Oo                                \n" <<

endl;

    cout << "                                *****" << endl;

    cout << "                                *                                *" <<

endl;

    cout << "                                *                                1.输入学院名称;                                2.输入运动项目;                                *" << endl;

    cout << "                                *                                *" <<

```



```

endl;

cout << "          *    3.按学院编号排列;          4.按总分排列;          *" << endl;
cout << "          *                                *" <<

endl;

cout << "          *    5.按男团体总分排列;          6.按女团体总分排列;          *" << endl;
cout << "          *                                *" <<

endl;

cout << "          *    7.按项目编号查询;          8.按学院别编号查询;          *" << endl;
cout << "          *                                *" <<

endl;

cout << "          *                                0.退出                                *" <<

endl;

cout << "          *                                *" <<

endl;

cout << "          *          提示: 首先输入学院名称及编号后才能输入运动项目          *" <<

endl;

cout << "          *                                *" <<

endl;

cout << "          *          *****" << endl;

cout << "                                请选择:";

//cin>>ch;

//choose=int(ch[0])+int(ch[1])-'0';    //处理异常状态

cin >> choose;

if (!isdigit(choose))

{

    system("cls");

    cout << "操作非法 1" << endl;

```

```

}

else
{
    switch (getint(choose))
    {

    case 1:

        system("cls");

        college_add();

        break;

    case 2:

        system("cls");

        cout << "当前项目:" << endl;

        sport_output(p2);

        cout << "当前学院:" << endl;

        college_output(p1);

        sport_add();

        break;

    case 3:

        system("cls");

        college_output(p1);

        break;

    case 4:

        system("cls");

        college_order(p1, 0);

        college_output(p1);

        break;

```

case 5:

```
system("cls");  
  
college_order(p1, 1);  
  
college_output(p1);  
  
break;
```

case 6:

```
system("cls");  
  
college_order(p1, 2);  
  
college_output(p1);  
  
break;
```

case 7:

```
system("cls");  
  
cout << "请输入项目编号:";  
  
cin >> temp;  
  
sport_search(temp);  
  
break;
```

case 8:

```
system("cls");  
  
cout << "请输入学院的编号:";  
  
cin >> temp;  
  
college_search(temp);  
  
break;
```

case 0:

```
system("cls");  
  
a = 0;  
  
break;
```

default:

```
        system("cls");  
        cout << "操作非法\n";  
    }  
}  
  
college_write();  
sport_write();  
system("exit");  
}
```