

基于 C 的奥运会跳水比赛评分系统设计与实现

唐婷 贺颖 许俊

四川职业技术学院通识教育学院, 中国·四川 遂宁 629000

摘要: 随着计算机技术的发展, 利用计算机编程技术开发一款基于 C 的奥运会跳水比赛评分系统, 实现能够公正、客观地评价运动员的表现, 展现跳水运动的魅力和精彩。本评分系统旨在模拟奥运会跳水比赛的评分流程, 通过编程实现自动去除每位选手的最高分和最低分, 并计算剩余分数的总和作为最终成绩的功能。系统通过 C 语言编写, 能够处理 12 位选手的评分数据, 并输出按总分排名顺序表以及选手的详细信息和得分。开发基于 C 的奥运会跳水比赛评分系统具有提高评分效率和准确性、降低人为因素的影响, 能促进跳水运动的智能化发展以及推动 C 语言及编程技术的发展等多重意义。

关键词: 跳水比赛评分系统; 评分流程; 评分效率; 智能化; C 语言

Design and Implementation of Olympic Diving Competition Scoring System Based on C

Ting Tang Ying He Jun Xu

Sichuan Vocational and Technical College General Education College, Suining, Sichuan, 629000, China

Abstract: With the development of computer technology, a C-based Olympic diving competition scoring system has been developed using computer programming technology, which can fairly and objectively evaluate the performance of athletes and showcase the charm and excitement of diving. This scoring system aims to simulate the scoring process of Olympic diving competitions, automatically removing the highest and lowest scores of each athlete through programming, and calculating the total remaining scores as the final score. The system is written in C language and can process the rating data of 12 contestants, and output a ranking table based on the total score, as well as detailed information and scores of the contestants. The development of a C-based Olympic diving competition scoring system has multiple significance, including improving scoring efficiency and accuracy, reducing the influence of human factors, promoting the intelligent development of diving sports, and advancing the development of C language and programming technology.

Keywords: diving competition scoring system; scoring process; rating efficiency; intelligentization; C language

1 引言

奥运会跳水比赛是奥运会中备受瞩目的水上运动之一, 比赛共设有 8 个奖牌项目, 包括男子和女子的 3 米跳板、10 米跳台, 以及它们的双人项目。跳水比赛动作包括助跑、起跳、空中技巧和入水四个阶段, 裁判组将根据运动员的表现进行打分。跳水比赛以其高度的技巧性、观赏性和挑战性而著称, 运动员需要在空中展现出优雅的动作, 并以完美的入水结束整个动作。随着计算机编程技术的发展, 论文通过使用 C 语言编程技术实现奥运会跳水比赛评分系统的功能, 具有高度的自动化和准确性, 能够模拟奥运会跳水比赛的评分流程, 可以方便地处理大量评分数据, 并快速获取比赛结果。

2 系统开发环境概述

本项目基于 Microsoft Visual Studio 2012 开发环境, 该软件是一款功能强大的集成开发环境 (IDE), 主要用于帮助开发人员创建适用于各种微软支持平台的应用程序。该软件支持多种编程语言, 包括但不限于 C、C#、C++、VB.NET 等,

允许开发者根据自己的需求和习惯选择合适的语言进行开发。它提供了完整的开发生命周期支持, 包括项目管理、构建、调试、部署等各个环节, 还集成了版本控制系统 (如 Git)、自动化测试工具、性能剖析器等其他开发工具。它允许开发者根据自己的喜好和需求自定义开发环境, 包括字体、颜色、菜单、工具栏等都可以进行个性化设置。该软件拥有强大的插件和扩展生态系统, 开发者可以根据自己的需求选择和安装必要的插件和扩展, 从而进一步增强 Visual Studio 的功能和灵活性。综上所述, Microsoft Visual Studio 2012 是一款功能全面、易于使用且高度可定制的集成开发环境, 为开发人员提供了丰富的工具和功能, 帮助他们更高效地开发各种类型的应用程序。

本项目开发语言选用 C 语言, C 语言作为计算机编程技术的入门语言, 在教育领域保持着重要性。C 语言编程技术的发展是一个持续不断的过程, 从诞生到标准化, 再到不断引入新的特性和功能, C 语言始终保持其高效、稳定、跨平台的特性, 并在各个领域得到广泛应用。

使用 C 语言开发奥运会跳水比赛评分系统具有一些特

定的优点，C 语言是一种编译型语言，它可以直接生成机器代码，从而提供出色的性能。在需要处理大量数据或实时响应的评分系统中，C 语言可以确保系统的快速响应和稳定运行。C 语言具有良好的跨平台性，可以在不同的操作系统和硬件上运行，这意味着使用 C 语言开发的跳水比赛评分系统，可以轻松地在各种环境中部署和运行，无需进行大量修改。虽然 C 语言本身不是面向对象的，但它支持结构体 (struct) 和函数指针等特性，可以实现类似面向对象编程的封装、继承和多态等功能。这使得 C 语言开发的系统具有良好的可扩展性，可以方便地添加新的功能和模块。C 语言语法简洁明了，没有多余的冗余。这使得开发者可以编写出高效、易读、易维护的代码。在开发跳水比赛评分系统时，这可以确保代码的质量和可维护性。

3 系统功能设计

本系统是基于 C 语言的奥运会跳水比赛评分系统，主要包含运动员信息管理和运动员评分管理两大模块。运动员信息管理包括录入运动员的序号、国籍、姓名等内容，可以进行运动员信息查询以及修改、删除等操作，以及显示所有参赛运动员的信息列表。运动员评分管理模块主要是录入运动员每轮动作得分明细，根据计算规则统计本轮得分和最终得分，可以进行运动员得分明细查询，可以显示每轮动作得分明细表以及显示最终排名表。系统功能设计如图 1 所示。

该系统功能需求描述：在奥运会跳水比赛中，运动员参加比赛，每个运动员完成 5 轮动作，每一轮跳一个动作，每个动作都要选择难度。当一个运动员跳完一个动作后，7 个评委评分，由程序计算该动作的本轮得分、截至本轮的总分、当前名次，并显示当前名次、运动员国籍、姓名、总分以及 7 个得分、难度、本轮分。当全部运动员的每一轮动作完成后，计算他们的累计总分、排名，按名次顺序显示名次、国籍、姓名、总分。第五轮的总分、名次是运动员的最终总分、

排名。具体要求如下：

- ①有 12 个运动员参加比赛，运动员有编号、姓名、国籍。
- ②每个动作有 7 个分数（分数范围为 0.00~10.00）和难度，均为 2 位小数。
- ③每位运动员的一个动作得分计算，由 7 个分数去掉 2 个最高分、2 个最低分剩余的 3 个之和再乘以难度，2 位小数。
- ④每位运动员的累计总分计算，是这个运动员所有已经跳完的动作得分的累加，如一个运动员跳完了 3 个动作，累计总分就是这 3 个动作得分之和。
- ⑤排名规则，最高分第一名，同分数同名次，符合奥运会跳水比赛排名规则。

4 系统主要功能模块实现

4.1 录入运动员信息

该系统采用模块化设计思想，将系统划分为多个独立的模块，方便未来的扩展和维护。录入运动员信息模块可以实现录入运动员的基本信息（包括编号、姓名、国籍等基本信息），可以建立一个完整的运动员数据库，方便后续的数据查询、修改和删除操作。这里使用结构体 (struct) 来定义运动员的信息，部分代码如下：

```
struct player_type
{
    char sn[10]; // 运动员序号
    char nation[20]; // 国籍
    char name[20]; // 姓名
}player[12];
```

通过 savePlayerInfo() 函数模块实现循环读取输入，将每位选手的信息存储到相应的结构体数组中，并将运动员信息数据写入 player.txt 文件中。本系统设定最多可以录入 12 名选手信息，在后期使用过程中可以修改运动员人数的上限，功能维护非常方便。

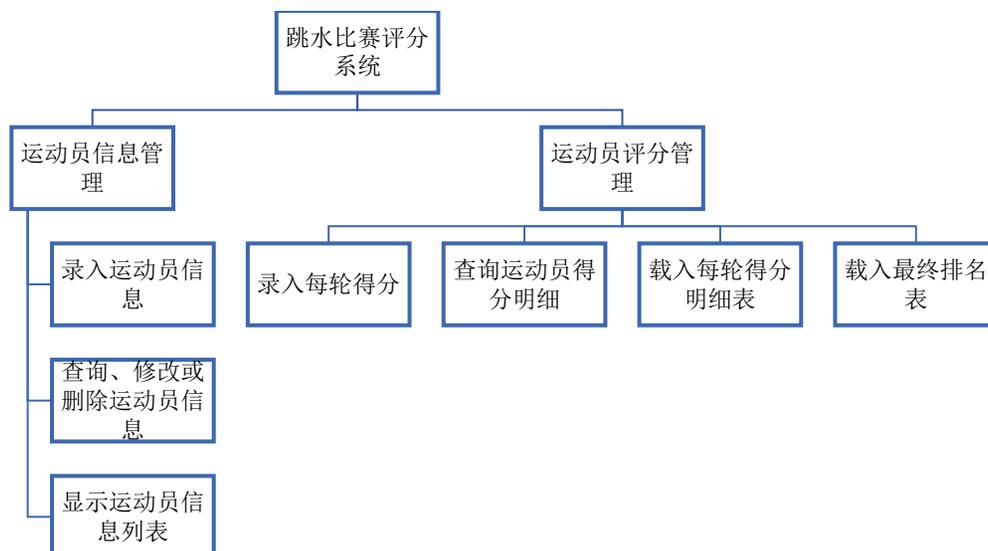


图 1 系统功能设计

4.2 显示运动员信息列表

显示运动员信息列表模块可以实现完整的输出所有参赛选手的详细信息一览表, 这里主要通过 read Player Info (players) 函数模块读取 player.txt 文件中的信息来实现功能需求, 读取文件的一行, 存到运动员数组中, 文件的行数即运动员的个数。部分代码如下:

```
fp = fopen( "player.txt", "r" );
if(fp == NULL){
    printf( "error open file of player.txt.\n" );
    exit(0);
}
while(!feof(fp)){
    fscanf(fp, " %s %s %s\n", players[i].
sn,players[i].nation,players[i].name); // 读文件的一行, 存到数组
    i++;
}
fclose(fp);
return i; // 运动员个数, 即文件行数
```

4.3 录入每轮得分

录入每轮得分模块可以实现输入每位运动员的 5 轮动作得分情况, 每轮可以录入 7 个评委评分, 程序将自动去掉 2 个最高分和 2 个最低分, 并计算剩余分数的总和, 得到最终得分。由程序自动计算该动作的本轮得分、截至本轮的总分, 并按名次顺序显示当前运动员排名、国籍、姓名、总分, 第五轮的总分、名次是运动员的最终总分、排名。这里使用结构体 (struct) 来定义运动员的每轮成绩表, 部分代码如下:

```
struct scores_type
{
    struct sco_type
    {
        int score[8]; // 运动员每轮成绩及总分
    } sco[5];
} scores[12];
```

通过 inputScore(num,round) 函数模块实现循环输入每位运动员每轮的明细成绩, 部分代码如下:

```
printf( "请输入 %d 号运动员的第 %d 轮动作得分:
\n", num, round);
for(i=0; i<7; i++)
{
    scanf( "%d", &sco[i]);
    scores[num-1].sco[round-1].score[i] = sco[i];
}
ranks(sco);
scores[num-1].sco[round-1].score[7] = sco[2] + sco[3] + sco[4];
```

其中, ranks(sco) 函数模块将输入的 7 个评委评分进行分数从小到大的排序, 这里使用的冒泡排序方法, 再实现去掉 2 个最高分和 2 个最低分, 并计算剩余分数的总和, 冒泡

排序函数代码如下:

```
static ranks(int sco[])
{
    int i, j, temp;
    for (i = 0; i < 6; i++)
        for (j = 0; j < 6 - i; j++)
            if (sco[j] > sco[j + 1]) {
                temp = sco[j];
                sco[j] = sco[j + 1];
                sco[j + 1] = temp;
            }
}
```

4.4 载入每轮得分明细表

载入每轮得分明细表模块可以实现查看某一轮的运动员得分明细表, 通过 loadDetails(round) 函数模块来实现在屏幕显示某一轮运动员编号、7 位评委的评分情况、总分、排名等信息。这是该系统的重要模块之一, 载入每轮得分明细表的作用主要是帮助运动员更清晰地了解比赛中每轮的得分情况。通过查看每轮的得分明细, 可以了解每个参赛者或团队在每一轮的具体得分, 以及得分的原因或依据。这有助于比赛过程进行详细的评估和分析, 发现优势和不足, 并为后续的比赛或活动提供有针对性的改进建议。同时, 对于参赛者来说, 查看每轮得分明细也可以帮助他们了解自己在比赛中的表现, 从而调整策略, 提高竞争力。部分代码如下:

```
int i, j;
for(i=0; i<12; i++)
{
    printf( "编号 %d 运动员得分: ", i+1);
    for(j=0; j<7; j++)
    {
        printf( "%d ", scores[i].sco[round-1].score[j]);
    }
    printf( "第 %d 轮总分: %d", round, scores[i].sco
[round-1].score[j]);
    printf( "\n" );
    // 显示某轮排名得分
    printf( "第 %d 轮动作排名:\n" );
    for(i=0; i<12; i++)
    {
        rank[i][0] = i+1;
        rank[i][1] = scores[i].sco[round-1].score[7];
    }
    rankr(rank); // 排名
    for(i=0; i<12; i++)
    {
        printf( "编号 %d 得分 %d\n", rank[i][0], rank[i][1]);
    }
}
```

4.5 载入最终排名表

载入最终排名表模块可以实现在屏幕上显示当前排名情况,这也是该系统的重要模块之一,通过 loadFinal() 函数模块实现显示当前运动员的排名顺序,包括运动员编号、姓名、国籍、当前总分等信息。最终排名表是运动员了解自己成绩和表现的直观方式,它不仅能让运动员对自己的努力有所评估,还能作为衡量整个比赛水平的重要参考。此外,对于组织者而言,最终排名表也是评估比赛成功与否的重要工具,能够帮助他们了解比赛的公正性和准确性。因此,确保最终排名表的准确性和及时性对于所有相关方都至关重要。部分代码如下:

```
int i,j;
for(i=0;i<12;i++)
{
    final[i][0]=i+1;
    final[i][1]=0;
    for(j=0;j<5;j++)
    {
        final[i][1]+=scores[i].sco[j].score[7];
    }
}
rankz(final);// 排名
for(i=0;i<12;i++)
{
    printf(“编号 %d 总分 %d\n”,final[i][0],final[i][1]);
}
```

5 系统功能测试

系统功能测试是确保软件或系统按照预期运行的一系列活动。这通常包括单元测试、集成测试、系统测试和用户验收测试等阶段。通过这些测试,我们可以发现并修复系统中的错误,提高系统的稳定性和可靠性。在进行系统功能测试时,测试人员会模拟用户操作,验证系统的各项功能是否符合需求规格说明书的要求。经过实验数据测试,该系统基本实现录入运动员信息、显示运动员信息列表、录入运动员每轮得分、显示每轮得分明细表、显示最终排名表等这几个主要模块的功能。本系统基于 C 语言实现,响应速度快,运行过程基本稳定,该系统还可以进一步优化和完善,以适

应更多不同的比赛场景和需求。

6 结语

论文主要描述基于 C 的奥运会跳水比赛评分系统设计过程以及主要功能模块实现方法,经实验数据测试,基本实现各项功能模块,达到预期效果。该系统优点是自动化程度高、准确性高、可扩展性强,能够自动处理评分数据,无需人工干预,通过编程实现评分规则,确保评分的准确性,系统可轻松扩展至处理更多选手或评委的数据。

该系统确保评分的公正性,评分系统采用统一的标准和算法进行评分,能够确保每位选手都受到公正的评判。它消除了人为因素在评分过程中的影响,使得比赛结果更加公正、公平。该系统提升观众观赛体验,评分系统能够为观众提供更加直观、易懂的比赛信息。观众可以通过系统实时了解每位选手的得分情况,更好地理解比赛进程和规则,从而提升观赛体验。奥运会是全球最高水平的体育赛事之一,跳水比赛评分系统的开发和应用能够促进国际的交流与合作。各国可以共享评分系统的技术和经验,共同推动跳水运动的发展和提高。

参考文献:

- [1] 吴香艳.基于C#和.NET的班级学生信息管理系统设计与实现[J].无线互联科技,2023,20(22):47-50.
- [2] 余丹丹.基于C#的医技科室业绩考核评分系统的研发[J].电脑知识与技术,2020,16(11):2.
- [3] 王利云.基于C++的高校编程考试自动评分系统[J].信息与电脑(理论版),2019(2):2.
- [4] 邹黎.基于C#平台实现新能源端子压接方案计算系统的开发与设计[J].汽车实用技术,2024(6).
- [5] 黄娟.基于C/S架构的医疗安全不良事件管理系统设计与实现[J].无线互联科技,2024(6).
- [6] 韦业君.基于C#的中职学校1+X证书在线考试系统设计与实现[J].电脑编程技巧与维护,2024(4).
- [7] 朱彦泓.基于C语言的无人机物流小车货物分拣系统设计[J].现代农业,2024(3).

作者简介:唐婷(1983-),女,中国四川遂宁人,本科,讲师,从事办公自动化、C语言编程、数据库技术、Python语言编程研究。