***西安财经大学****信息学院*

***姓名***王婧

***学号***1831050171

***班级***软件工程1802

***年级***大二

***指导教师***杨新安

*《数据结构》 课程设计报告*

***课程设计报告名称***电视大赛观众投票及排名 *实验室* 实验楼316 *完成****日期***2019-12-10

|  |
| --- |
| 1. 目的 2. 掌握数据结构的逻辑特点及存储实现； 3. 设计要求（15分） 4. 问题描述（5分）   在很多的电视大赛中，通常当选手表演结束后，现场观众通过手中的按键对参赛选手进行投票，然后对选手获得的票数进行统计，从高到低进行降序排序，从而自动产生冠军、亚军和季军。  要求：首先输入参赛选手的人数（范围为1-9个），然后根据人数通过malloc函数来开辟存放选手信息的顺序表。将选手的编号和姓名依此存入顺序表单元中，观众通过按键进行投票，按’1’为1号选手投票，按’2’为2号选手投票，以此类推，以按’0’作为投票结束标志。投票结束后进行排序，在此采用希尔排序，然后为每个选手计算名次，得票相同的名次也相同。   1. 需求分析。（10分）   **本程序分为四个模块：** 1.主程序模块：实现对函数的调用； 2.顺序表模块：实现对选手信息的存储； 3.投票模块：实现观众对选手的投票； 4.排序模块：实现对选手成绩的排序。  **本程序分为五个模块** a、主程序模块：实现对函数的调用；  b、顺序表模块：实现选手信息存储；  c、投票模块：实现观众对选手的投票；  d、排序模块：实现对选手成绩的排序； e、文件保存模块：实现对选手成绩的保存；   1. 概要设计（20分） 2. **主界面设计（**5分）   6.6   1. **存储结构设计**（5分）   参赛选手信息存储类型的定义：  typedef struct Node  {  char name[9];//姓名  int num;//编号  int score;//得分  int tax;//名次  }Node;   1. **系统功能设计**（10分）   1、函数Menu, 用于进行菜单的显示和选号：void Meun() 2、函数Init\_sq，为选手个数分配动态：void Init\_sq(SeqList \*L,int n) 3、函数Init，用于完成选手信息的输入：void Init(SeqList \*L) 4、Vote函数用以实现投票的功能：void Vote(SeqList \*L) 5、希尔排序：void Shellsort(SeqList \*L) 6、count函数，用以计算选手名次：void count(SeqList \*L) 7、Display函数，用以输出最终结果：void Display(SeqList \*L) 8、Save函数，实现文件的生成：void Save(SeqList \*L)   1. 模块设计（15分） 2. **模块设计**（5分）   1.主程序模块：实现对函数的调用； 2.顺序表模块：实现对选手信息的存储； 3.投票模块：实现观众对选手的投票； 4.排序模块：实现对选手成绩的排序。   1. **系统子程序及功能设计**（5分） 2. 菜单模块：实现对函数的调用；   6.6 b、顺序表模块：实现选手信息存储；  1.1  c、投票模块：实现观众对选手的投票；  2.2 d、排序模块：实现对选手成绩的排序；  3.34.4 e、文件保存模块：实现对选手成绩的保存；  5.5  3、**函数主要调用关系图**（5分）  7   1. 详细设计（15分） 2. 数据类型定义（5分）   8   1. 主要子程序详细设计（10分）   1、顺序表模块：实现选手信息存储；  1.1  2、投票模块：实现观众对选手的投票；  2.2 3、排序模块：实现对选手成绩的排序；  3.34.4  4、文件保存模块：实现对选手成绩的保存；  5.5   1. 测试分析（15分） 2. 在刚开始编程的时候经常会碰到一些低级编写错误例如漏分号大括号不匹配等 经过仔细修改已能正常运行； 2、在程序完成后但由于输入选手姓名后不能根据名字长短自动判断自动判断对齐编号列表，后来经过老师指导并请教同学优化代码，在输入名字后按下TAB键就能自动对齐编号列表，程序更加人性化。   调试结果  （1）输入选手信息界面 1 （2）投票界面 2 （3）排序界面 3 （4）排序结果页面 4 （5）保存到文件 5 （6）保留到文件的界面: 6 经编译完成，此次设计程序已完成选手信息输入、投票、排序及保存到文件等功能   1. 设计总结（10分）  在这之前，我们已经对数据结构这门课程学习了一个学期，对其有了一定的了解，但是更多的还是停留在学习了解的范围，对里面的好多东西还是很陌生，并不是很熟练，有着许多欠缺，更多的在运用起来的时候还是感到很不好动手。数据结构课堂上许多关于数据结构的语法规则，听起来十分枯燥无味，也不容易记住，死记硬背是不可取的。然而要使用数据结构这个工具解决实际问题，又必须掌握它。通过多次上机练习，对于语法知识有了感性的认识，加深对它的理解，在理解的基础上就会自然而然地掌握数据结构的语法规定。对于一些内容自己认为在课堂上听懂了，但上机实践中会发现原来理解的偏差，更加巩固了学过的知识，而且在设计的时候学要系统的知识，也是一个较大的挑战，某一方面知识的欠缺都将影响到整个程序的设计。我所编的虽然只是一个小程序，但是每一步的编的过程，让我懂得很多。重温一下数据结构，因而编程比较容易懂，但是通过练习，发现理论与实际的差距。编程从一开始全部编到一个main函数中到，最后逐个分块，到添加功能，到最后的进行代码的优化，使程序的可用性和智能性都有进一步的提高，一点一点的进步让我很欣喜，对编程的兴趣也更加浓厚。此次实习，不仅了有关数据结构的知识，还掌握了一些现实中编程的一些小技巧，实际的编程能力也得到了历练，本次课程设计是很好的一次锻炼机会！ 2. 附程序清单（10分）   #define \_CRT\_SECURE\_NO\_WARNINGS  #include<iostream>  #include<stdlib.h>  using namespace std;  #define initsize 3  #define increment 3  //结构体定义  typedef struct Node  {  char name[9];//姓名  int num;//编号  int score;//得分  int tax;//名次  }Node;  typedef struct  {  Node \*data;  int len;  int listsize;  }SeqList;  void Init\_sq(SeqList \*L, int n)  {  L->data = (Node\*)malloc(n \* sizeof(Node));  if (!L->data)  exit(1);  L->len = n;  L->listsize = n;  }  void Init(SeqList \*L)  {  Node\* newnode;  int i, n;  cout << "请输入选手个数：";  cin >> n;  Init\_sq(L, n);  cout << "请输入选手的姓名和编号" << endl;  printf("姓名\t 编号\t\n");  if (L->len >= L->listsize)  {  newnode = (Node\*)realloc(L->data, (L->listsize + increment) \* sizeof(Node));  if (!newnode)  exit(1);  L->data = newnode;  L->listsize += increment;  }  for (i = 1; i <= L->len; i++)  {  cin >> L->data[i].name >> L->data[i].num;  L->data[i].score = L->data[i].tax = 0;  }  }  void Vote(SeqList \*L)  {  int i, num;  cout << "请输入你要为几号选手投票:(0结束)" << endl;  do  {  cin >> num;  if (num<0 || num>L->len)  {  cout << "你输入的编号错误请重新输入(1--" << L->len << ")" << endl;  continue;  }  else  {  for (i = 1; i <= L->len; i++)  {  if (L->data[i].num == num)  {  L->data[i].score += 1;  }  }  }  } while (num != 0);  }  void Shellsort(SeqList \*L)  {  int i, j;  int crement = L->len;  do  {  crement = crement / 3 + 1;  for (i = crement + 1; i <= L->len; i++)  {  if (L->data[i].score > L->data[i - crement].score)  {  L->data[0] = L->data[i];  for (j = i - crement; j > 0 && L->data[0].score > L->data[j].score; j -= crement)  {  L->data[j + crement] = L->data[j];  L->data[j + crement] = L->data[0];  }  }  }  } while (crement > 1);  }  void count(SeqList\* L)  {  int i;  L->data[1].tax = 1;  for (i = 2; i <= L->len; i++)  {  if (L->data[i - 1].score == L->data[i].score)  {  L->data[i].tax = L->data[i - 1].tax;  }  else  {  L->data[i].tax = L->data[i - 1].tax + 1;    }  }  cout << "正在计算选手名次请稍候" << endl;  cout << "排序成功" << endl;  }  void Display(SeqList \*L)  {  int i;  printf("姓名\t 编号\t 票数\t 名次\t\n");  for (i = 1; i <= L->len; i++)  {  printf("%s\t%d\t%d\t%d\t\n", L->data[i].name, L->data[i].num, L->data[i].score, L->data[i].tax);    }  cout << endl;  }  void Save(SeqList \*L)  {  FILE\* fp;  int i;  char filename[20];  cout << "请输入目标文件名:" << endl;  cin >> filename;  fp = fopen(filename, "w");  fprintf(fp, "姓名\t编号\t票数\t名次\n");  for (i = 1; i <= L->len; i++)  {  fprintf(fp, "%s\t%d\t%d\t%d\t\n", L->data[i].name, L->data[i].num, L->data[i].score, L->data[i].tax);  }  fclose(fp);  cout << "数据已成功保存！" << endl;  }  void Meun()  {  cout << "\*\*\*\*\*\*\*\*\*\*\*\*电视大赛观众投票及排名系统\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;  cout << "\*\*\*\* 1、输入选手信息(姓名、编号) \*\*\*\*" << endl;  cout << "\*\*\*\* 2、开始投票 \*\*\*\*" << endl;  cout << "\*\*\*\* 3、排序 \*\*\*\*" << endl;  cout << "\*\*\*\* 4、查看结果 \*\*\*\*" << endl;  cout << "\*\*\*\* 5、保存到文件 \*\*\*\*" << endl;  cout << "\*\*\*\* 6、退出 \*\*\*\*" << endl;  cout << "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;  }  int main()  {  int sel;  SeqList L;  while (1)  {  Meun();  printf("\n 请输入您要选择的操作序号，按回车键确认：");  cin >> sel;  while (sel < 1 || sel>6)  {  cout << "输入有误，请重新输入：";  cin >> sel;  }  switch (sel)  {  case 1:  Init(&L);  break;  case 2:  Vote(&L);  break;  case 3:  Shellsort(&L);  count(&L);  break;  case 4:  Display(&L);  break;  case 5:  Save(&L);  break;  case 6:  cout << "谢谢使用，再见！" << endl;  exit(0);  break;  default:  break;  }  }  } |