|  |  |  |  |
| --- | --- | --- | --- |
| **学号** | 2106410217 | **成绩** |  |





**课程设计说明书**

**设计名称 C++课程设计**

**设计题目 题目2：学生成绩管理系统**

**设计时间 2022.6.6—2022.6.10**

**学 院 信息与控制工程学院**

**专 业 计算机科学与技术**

**班 级 计算机2102班**

**姓 名 杨 树 昆**

**指导教师 宋晓宇 师金钢**

**2022 年 6 月 6 日**

C++课程设计说明书

**目录**

1. 设计任务说明

2. 开发工具

3. 设计思路

4. 系统效果图（每个图要有文字说明，可附多图。）

5. 源代码

6. 参考文献

7. 设计体会

1. **设计任务说明**

功能：学生成绩管理系统，每个学生是一个记录，包括学号、姓名、3 门课程成绩。

系统要求实现以下功能：

1、信息录入：录入学生成绩信息（包括学生学号、姓名、各门课程的成绩等），并将信

息写入文件中，要求有文件读写操作；

2、信息查询：输入学号，查询学生各门课程的成绩，并显示；

3、排序：按各门课程的成绩平均分进行排序，并显示；

4、信息删除与修改——输入学号，修改或删除该学生的成绩信息。

**二、开发工具**

Visual Studio 2022

1. **设计思路**

分步实施：1、初步完成总体设计，搭好框架，确定人机对话的界面，确定函数个数。2、完成信息的录入和显示（即通过编程创建文件，输入学生信息并可将文件内容显示出来）3、完成修改功能，即输入学号，可修改该学生的信息

要求：1、用C++语言实现程序设计；

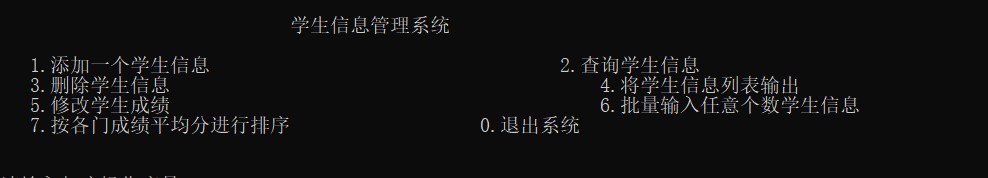
2、利用结构体数组、链表等实现学生信息表达、查询等，充分体现数据结构的知识；

3、系统的各个功能模块要求用函数的形式实现；

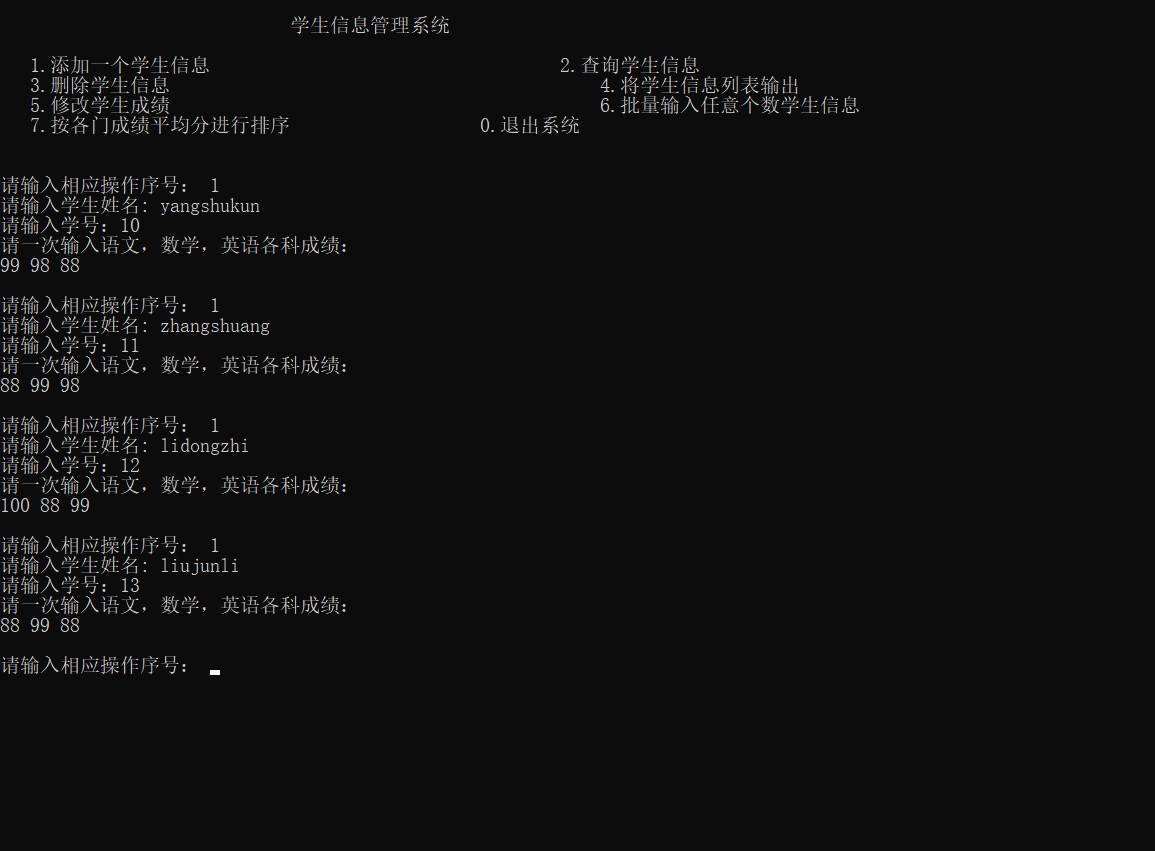
4、界面友好（良好的人机交互），程序要有注释。

5、程序中所用到的排序查找方法需采用数据结构中所学方法实现，要求方法实用效率高。

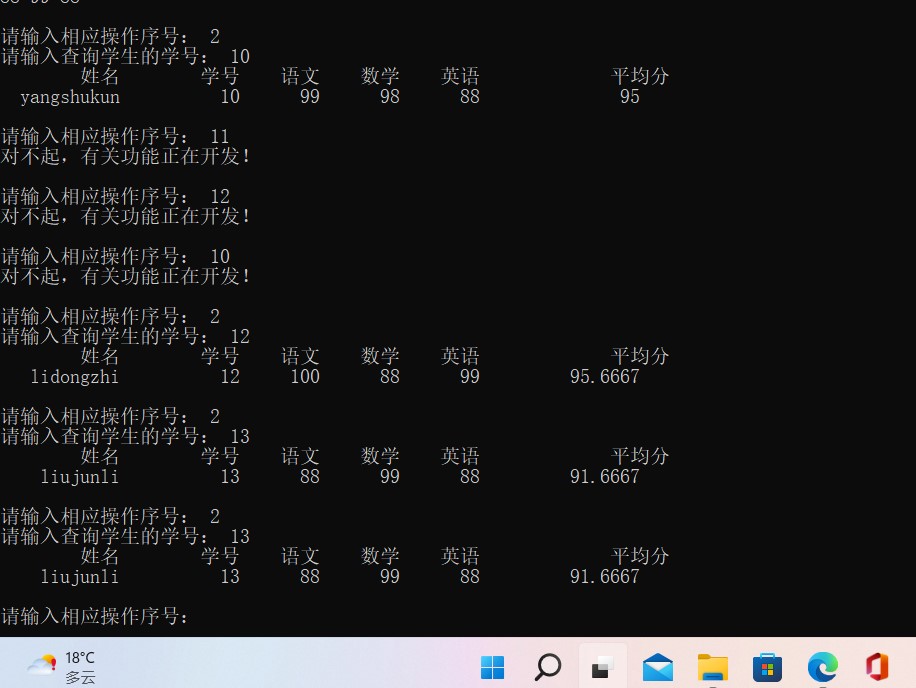
**四、系统效果图**

**首页**

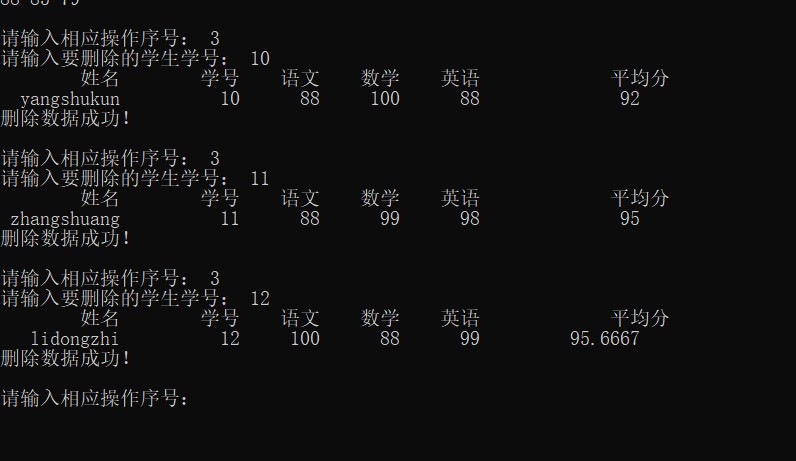
**功能1、按个录入信息**

****

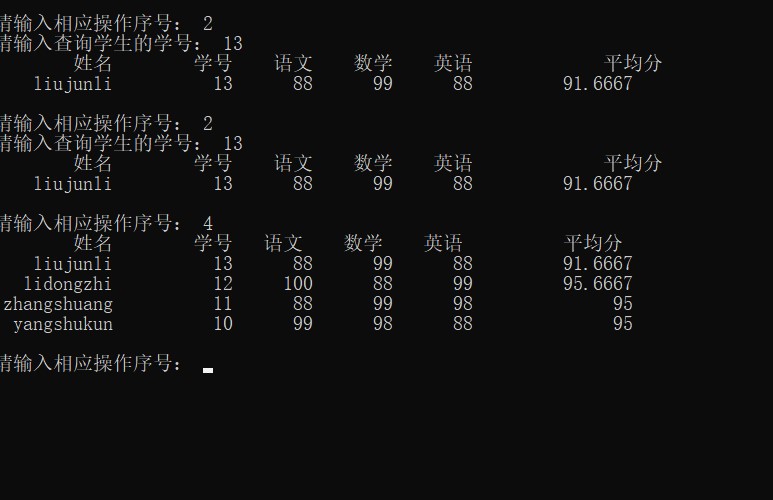
**功能2、查询学生信息**

****

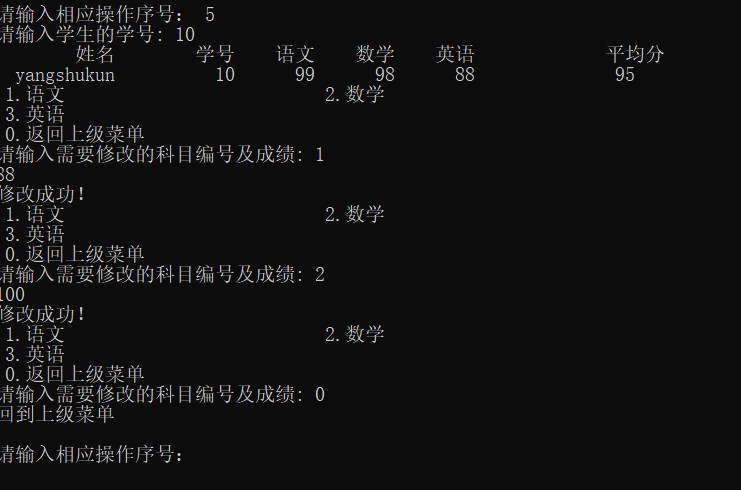
**功能3、删除学生信息**

****

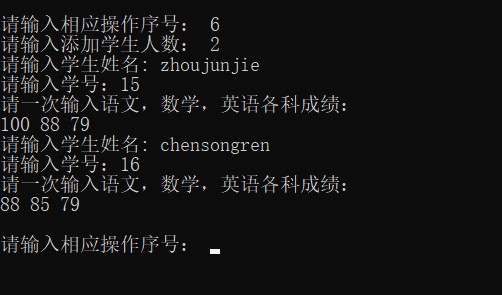
**功能4、成绩表输出**

****

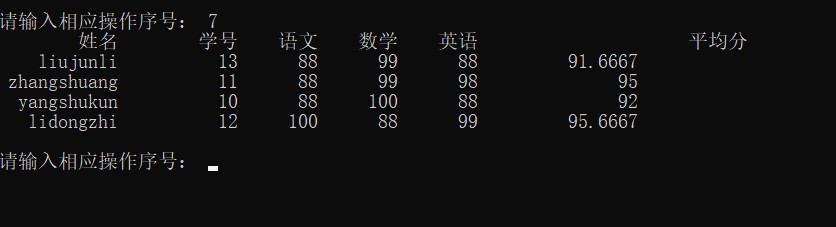
**功能5、修改学生成绩**

****

**功能6、批量输入多个学生信息**

****

**功能7、按平均分进行排序**

****

1. **源代码**

#include<iostream>

#include <string>

#include<iomanip>

using namespace std;

class StudentData

{//建立学生类

public:

string name; double average;

double math, english, chinese, number;

StudentData\* next;

StudentData\* pre;

void insertdate()

{

cin >> chinese >> math >> english ;

}

void setlang() { cin >> chinese; }

void setmath() { cin >> math; }

void setenglish() { cin >> english; }//输入各科成绩

};

struct linklist

{//带头指针和尾指针的双向链表

StudentData\* head, \* present;//头指针和当前指针

void insert\_one();

void delete\_data();

void setdata();

bool check();

void creat();

void output();

void setmore();

void averagerank();

}X;

void linklist::output()

{//打印链表，输出学生信息

cout << " 姓名 学号 语文 数学 英语 平均分" << endl;

present = head;

while (present->next != NULL)

{//不取尾结点，尾节点为空

present->average = (present->chinese + present->math + present->english ) / 3;

cout << setw(12) << present->name << setw(12) << present->number << setw(8) << present->chinese << setw(8) << present->math << setw(8) << present->english

<< setw(16) << present->average << endl;

present = present->next;

}

}

void linklist::creat()

{//创建空结点

present = new StudentData;

head = present;

present->next = NULL;

present->pre = NULL;

}

void linklist::insert\_one()

{

present = new StudentData;

head->pre = present;

present->next = head;

head = present;

present->pre = NULL;//新加一个学生数据加在表头

cout << "请输入学生姓名: ";

cin >> present->name;

cout << "请输入学号：";

cin >> present->number;

cout << "请一次输入语文，数学，英语各科成绩：" << endl;

present->insertdate(); //输入成绩

}

void linklist::delete\_data()

{//删除学生信息

//string studentnam;

int sigh = 0;

cout << "请输入要删除的学生学号： ";

if (check())

{

if (present->pre == NULL)

{

head = present->next;

delete present;

present = head;

}

if (present->pre != NULL)

{

present->pre->next = present->next;

present->next->pre = present->pre;

head = present->pre;

delete present;

}

cout << "删除数据成功！" << endl;

}

else cout << "删除失败。" << endl;

}

bool linklist::check()

{//按学号查找并输出学生信息

present = head;

int idnumber;

cin >> idnumber;

while (present->next != NULL)

{

if (idnumber == present->number)

{

present->average = (present->chinese + present->math + present->english ) / 3;

cout << " 姓名 学号 语文 数学 英语 平均分" << endl;

cout << setw(12) << present->name << setw(12) << present->number << setw(8) << present->chinese << setw(8) << present->math << setw(8) << present->english

<< setw(16) << present->average << endl;

return true;

}

else

{

present = present->next;

continue;

}

}

cout << "查无此人！" << endl;

return false;

}

void linklist::setdata()

{//修改学生成绩

int subject;

cout << "请输入学生的学号: ";

if (check())

{

while (1)

{//设置个循环修改成绩，按0或者超出选项退出循环

cout << " 1.语文 2.数学" << endl

<< " 3.英语 "<<endl

<< " 0.返回上级菜单 " << endl;

cout << "请输入需要修改的科目编号及成绩: ";

cin >> subject;

switch (subject)

{

case 1:present->setlang(); break;

case 2:present->setmath(); break;

case 3:present->setenglish(); break;

case 0:cout << "回到上级菜单" << endl;; return;

default:

{

cout << "回到上级菜单。" << endl;//这行可以不要

return;

}

}

cout << "修改成功！" << endl;

}

}

}

void linklist::setmore()

{//通过循环次数来控制插入人数

cout << "请输入添加学生人数： ";

int n;

cin >> n;

while (n--) insert\_one();

}

void linklist::averagerank()

{//注意链表的尾结点是默认值，因为在创建链表的时候是在空节点上逆序创建的

int h = 0, k = 0; present = head;

L:

while (present->next->next != NULL)

{

h = k;

if (present->average <= present->next->average)

{

present = present->next;

}//无需互换

if (present->average > present->next->average && present->pre == NULL)

{

k++;

present->next = present->next->next;

present->next->pre->next = present;

present->pre = present->next->pre;

present->next->pre = present;

present->pre->pre = NULL;

}//头节点处链表间互换

if (present->average > present->next->average && present->pre != NULL)

{

if (present->next->next != NULL)

{//最后一个空结点不纳入考虑

k++;

present->pre->next = present->next;

present->next = present->next->next;

present->pre->next->next = present;

present->pre->next->pre = present->pre;

present->pre = present->pre->next;

present->next->pre = present;

}

}

while (h != k)

{//若一次完整循环下来没有顺序变化，退出

while (present->pre != NULL) { present = present->pre; } goto L;

}

}

while (present->pre != NULL) { present = present->pre; }

head = present;//因为头结点不为空，修改完后可能会被改变，此处从新定位头结点

cout << " 姓名 学号 语文 数学 英语 平均分" << endl;

do {

present->average = (present->chinese + present->math + present->english ) / 3;

cout << setw(12) << present->name << setw(12) << present->number << setw(8) << present->chinese << setw(8) << present->math << setw(8) << present->english

<< setw(16) << present->average << endl;

present = present->next;

} while (present->next != NULL);

}

int main()

{

cout << " " << endl;

cout << " 学生信息管理系统 " << endl;

cout << " " << endl;

cout << " 1.添加一个学生信息 2.查询学生信息" << endl;

cout << " 3.删除学生信息 4.将学生信息列表输出" << endl;

cout << " 5.修改学生成绩 6.批量输入任意个数学生信息" << endl;;

cout << " 7.按各门成绩平均分进行排序 0.退出系统 " << endl;

cout << " " << endl;

X.creat();//先建空表

while (true) {

int choise;

cout << endl << "请输入相应操作序号： ";

cin >> choise;

switch (choise)

{

case 1: X.insert\_one(); break;

case 2:

{

cout << "请输入查询学生的学号： ";

X.check(); break;

}

case 3: X.delete\_data(); break;

case 4: X.output(); break;

case 5: X.setdata(); break;

case 6: X.setmore(); break;

case 7: X.averagerank(); break;

case 0:

break;

default:

cout << "对不起，有关功能正在开发！" << endl; break;

}

if (!choise) break;

}

return 0;

}

1. **参考文献**

**C++面向对象程序设计**

1. **设计体会**

经过这次课程设计，我学到了很多东西：

①巩固和加深了对编程的理解，提高综合运用本课程所学知识的能力。

②培养了我选用参考书，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。

③经过实际编译系统的分析设计、编程调试，掌握应用软件的分析方法和工程设计方法。

④能够按要求编写课程设计报告书，能正确阐述设计和实验结果，正确绘制系统和程序框图。

这次高级语言程序设计，学生成绩管理系统，让我对高级语言有了进一步的认识。首先，想要学好这门课程，主要在于实践，要通过不断地上机操作才能更好的学习它。

之后，课程设计期间，在调试出现很多错误，很多警告的过程中我的知识也在得到不断地巩固；当你把代码输入电脑，并用编译器将其运行，发现通过不了，再来检查找出问题，这是一件非常辛苦的事情，也很浪费时间。于是在课程设计的时候，我花了好几天的时间来规划与写代码，将要实现的内容分析清楚，才把代码输入电脑。我们觉得写程序，应该先找到该程序中的核心地方，用多种方法来实现该核心，这才可能避免等到发现逻辑上或者编译器不支持上的错误，才来想补救的措施，这样花费时间在想补救措施是很不值得的。也让我学到了很多人生的哲理：懂得怎样去制定计划，怎样去实现这个计划，并掌握了在执行过程中怎样去克服心理上的不良情绪。我们对于高级语言的理解和使用还处在一个初级的阶段，我们要想掌握它，要学习的还很多，同时要懂得创新，许多的编程方法都是人们一步一步总结出来的，要努力拓展自己的思维，努力去寻求新的方法，以创造出更为完善的应用程序。

课程设计是把我们所学的理论知识进行系统的总结并应用于实践的良好机会，有利于加强我们用知识理论来分析实际问题的能力，进而加强了我们对知识认识的实践度，巩固了我们的理论知识，深化了对知识的认识，并为走向社会打下一个良好的基础。