**学生成绩管理系统实验报告**

**一、实验题目解读**

1.信息说明

实验项目的文件均以txt文本文件的形式存储。学生学号、姓名、成绩之间使用空格分隔。

现有文件包括：两个年级学生名单、15门课程的成绩（2门公共课、3门必修课、10门选修课）

成绩文件除了公共课分年级存为不同的文本文件，其他课程的成绩不分年级存为一个文本文件。

公共课：英语、政治；

必修课：数据结构、操作系统、计算机系统；

选修课：程序设计、计算思维、编译原理、软件工程、信息安全、计算机网络、数据库概论、数据挖掘、多媒体技术、人工智能

2.实验要求

功能1：生成成绩汇总文件，包含姓名、学号和各科成绩，格式化输出。

汇总成绩时，可能出现错误，要提示供教务员修改，无成绩或不及格的成绩都记为0分

功能2：生成学分绩排名文件

学分绩数计算公式=（2门公共课成绩+3门必修课成绩+选修课中成绩最高的5门课程成绩）/10 ；分数保留两位小数。

选修课中成绩最高的5门课程成绩：每个学生选修的选修课一般不同，因此一个学生的所有选修课中取分数最高的5门成绩加入排名计算。

排序后输出，要考虑到并列的情况。

功能3：查询功能 查询某个人的成绩信息 查询某门课没有成绩的学生名单 查询学分不够27的学生名单。

**二、需求分析和答题思路**

功能1---->建立一个结构变量 包含各科成绩、学分绩、总学分、排名等信息。

代码如下：

struct inf

{

string number, name;

int Politics, English;

int data, opsystem, csystem;

int programming, compute, compile, softeng, security, network, database, datamining, ai, multimedia;

double credit;

int ranking,sum;

};

在主函数中创建两个结构变量数组grade14[100], grade15[100] 分别储存14、15级学生的信息。打开各文件进行读入。

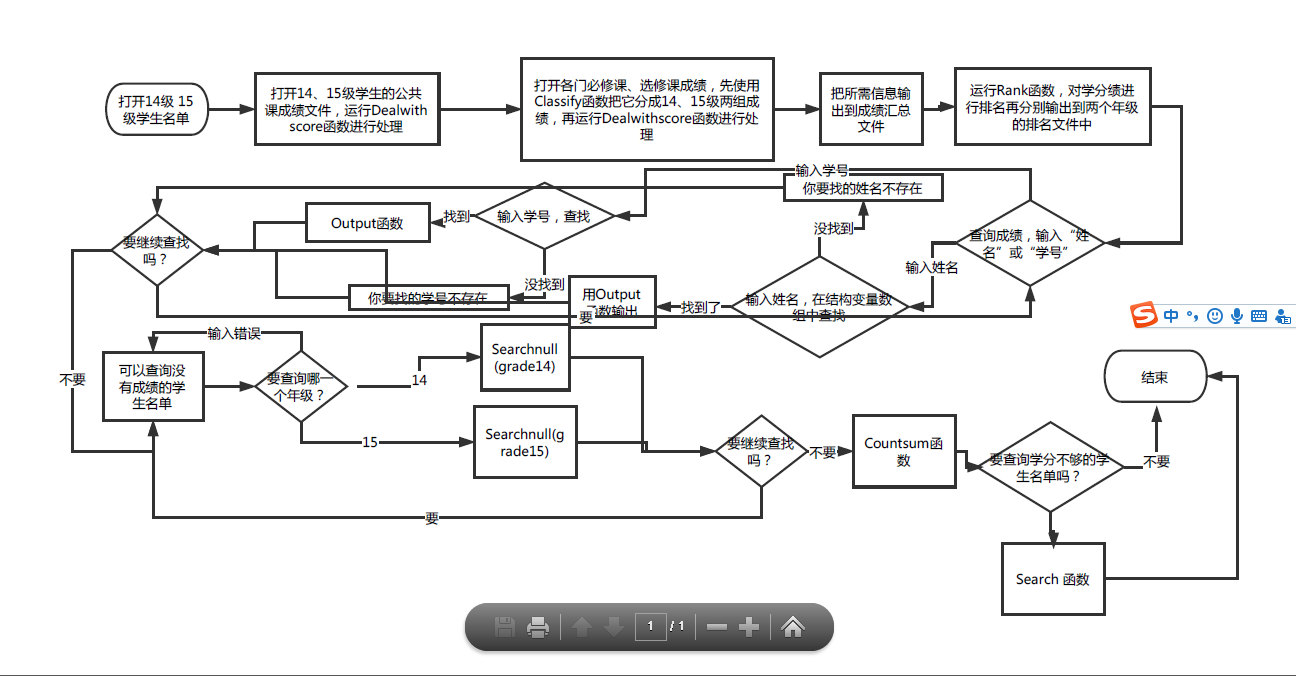
读入过程使用Dealwithscore函数，对数据进行处理，如有错误进行对应的输出。

对于14、15级混在一起的成绩信息，先使用一个classify 函数进行分类，再处理。

功能2---->使用一个Credit 函数，先对一个结构变量内部的选修课成绩进行排序，再取最大的5个值与公共课、必修课成绩一起算出平均学分绩，存到credit中。回到主函数，再使用Rank进行排序，ranking的初始值为1，每找到一个比它大的学分绩，ranking值加一。

最后按照ranking值的大小升序排列输出到指定文件中。

功能3----->分为3块，一是查询成绩信息，可以通过姓名或学号进行查找，通过Output函数进行输出；二是查询没有成绩的学生名单，可以通过Searchnull函数实现，输入要查询的课程名字，输出学生名单；三是查询学分不够的学生名单，先通过void Countsum 函数计算出每个结构变量（学生）的总学分，再把其中不够27的输出。

**三、主函数流程图**

注：处理成绩单中输入的数据时，先建立一些新的结构变量数组（data[100], English[100]…..），把成绩单中的学号、姓名、成绩信息输入到这些数组中，再进行处理，修改时修改这些数组中的数据，最后通过这些数组把新的信息输出到各个成绩单中

**三、各功能函数说明及代码**

1.Dealwithscore函数，输入结构变量数组，处理成绩单中的信息。先搜索，如果姓名一致学号不一致，进行是否是重名的判断，如果不是重名，输出错误提示供用户选择方案 一或方案二进行修改。

再搜索学号一致、姓名不一致的情况。搜索到时，输出错误提示供用户选择方案一或方案二修改

代码如下

void Dealwithscore(inf grade[], inf score[], int &n1, int &n)

{

int i, j, k;

char s[10]; string ss; string ss1="姓名"; string ss2="学号";

char s1[]="方案一"; char s2[]="方案二";

for(i=0; i<n; i++){

for(j=0; j<n1; j++)

if(grade[j].name==score[i].name){

if(grade[j].number!=score[i].number){

for(k=0; k<n1; k++){

if(grade[k].number==score[i].number&&grade[k].name==score[i].name)

cout<<"出现了重名的情况"<<grade[j].number<<"和"<<grade[k].number<<"的姓名都是"<<grade[k].name<<endl;//重名情况判定

}

break;

cout<<"成绩单中的姓名"<<score[i].name<<"的学号"<<score[i].number<<"与名单中的学号"<<grade[j].number<<"不符！"<<endl;//不是重名，输出错误提示，供用户修改

cout<<"现在，有两种解决方案\n"<<"方案一：在成绩单中删除该行信息\n"<<"方案二：成绩不变，根据学生名单中的姓名，提示学号对应的正确学号，供修改。"<<endl;

cout<<"你要选择哪一种？"<<endl;

cin>>s;

if(strcmp(s,s1)==0){

for(k=i; k<n; k++)

score[k]=score[k+1];//选择方案一时，删除成绩结构变量数组中的某个值

n--;

}

else if(strcmp(s,s2)==0){

cout<<"正确的学号是"<<grade[j].number<<"请在下方输入"<<endl;

for(; ; ) {

cin>>ss;

if(ss==grade[j].number) break;//选择方案二，提示正确信息

else cout<<"输入错误，请重新修改！"<<endl;

}

}

}

break;

}

if(j==n1) cout<<"姓名"<<score[i].name<<"不存在于名单中！"<<endl;

}

for(i=0; i<n; i++){

for(j=0; j<n1; j++)

if(grade[j].number==score[i].number){

if(grade[j].name!=score[i].name){

cout<<"成绩单中学号"<<score[i].number<<"的姓名"<<score[i].name<<"与名单中的姓名"<<grade[j].name<<"不符！"<<endl;

cout<<"现在，有两种解决方案\n"<<"方案一：在成绩单中删除该行信息\n"<<"方案二：成绩不变，根据学生名单中的学号，提示学号对应的正确姓名，或者根据姓名给出学号，供修改。"<<endl;

cout<<"你要选择哪一种？"<<endl;

cin>>s;

if(strcmp(s,s1)==0){

for(k=i; k<n; k++)

score[k]=score[k+1];

n--;

}

else if(strcmp(s,s2)==0){

cout<<"正确的姓名是"<<grade[j].name<<"请在下方输入"<<endl;

for(; ; ) {

cin>>ss;

if(ss==grade[j].name) break;

else cout<<"输入错误，请重新修改！"<<endl;

}

}

}

break;

}

if(j==n1) {

cout<<"学号"<<score[i].number<<"不存在！"<<endl;

cout<<"删除成绩单中该行的信息"<<endl;

for(k=i; k<n; k++)

score[k]=score[k+1];

n--;

}

}

}

2.Classify函数，把一个成绩数组分成两个，分别储存14级，15级的成绩信息

void Classify(inf score[], inf score14[], inf score15[], int &m1, int &m2, int &n)

{

for(int i=0; i<n; i++){

if(score[i].number[3]-'0'==4)//通过学号判断是14级还是15级

score14[m1++]=score[i];

else if(score[i].number[3]-'0'==5)

score15[m2++]=score[i];

else

cout<<"成绩单中的学号出现错误。"<<endl;

}

}

3. Credit 函数计算某个人的学分绩

建立一个int数组，储存各门选修课的成绩。对int数组进行升序排序，找出最大的5个数，与公共课、必修课成绩一起计算学分绩并返回

double Credit(inf t)

{

int a[10];

int i,j,k,p,sum;

a[0]=t.ai;

a[1]=t.compile;

a[2]=t.compute;

a[3]=t.database;

a[4]=t.datamining;

a[5]=t.multimedia;

a[6]=t.network;

a[7]=t.programming;

a[8]=t.security;

a[9]=t.softeng;//分别给a[10]数组中的数赋值

for(i=0; i<10; i++){

for(k=i,j=k+1; j<10; j++) if(a[j]<a[k]) k=j;//升序排序

if(k!=i){ p=a[k]; a[k]=a[i]; a[i]=p;}

}

sum=0;

for(i=9; i>4; i--){

sum+=a[i];

}

return (t.English+t.Politics+t.data+t.csystem+t.opsystem+sum)/10.0;//计算学分绩

}

4.Rank函数生成一个排名的值，储存到结构变量的ranking里面。先对输入的结构变量数组中的每一个变量使用Credit函数，计算出学分绩。设置ranking初始值为1，对于每一个变量，在数组中搜索其他变量的credit值，一旦找到比它大的，ranking值就加一，这样最后得到的ranking值就是这个变量（学生）的排名，进行升序排序输出即可。

void Rank(inf grade[], int n)

{

int i,j,k;

inf t;

for(i=0; i<n; i++){

grade[i].credit=Credit(grade[i]);

}

for(i=0; i<n; i++){

grade[i].ranking=1;

for(j=0; j<n; j++){

if((grade[j].credit-grade[i].credit)>1e-7)

grade[i].ranking++;

}

}

for(i=0; i<n; i++){

for(k=i,j=k+1; j<n; j++) if(grade[j].ranking<grade[k].ranking) k=j;

if(k!=i){ t=grade[k]; grade[k]=grade[i]; grade[i]=t;}

}

}

5. 查询某一学生的成绩信息，如果找到，使用output函数进行输出

先计算出每一门课的学分，存到int 数组中方便输出，同时算出总学分

在格式化输出时，我们选择使用不同数量的’\t’，方便上下对齐

void output(inf &t)

{

int a[15];

int i;

t.sum=0;

for(i=0; i<15; i++)

a[i]=0;

if(t.Politics!=0) a[0]=4;

if(t.English!=0) a[1]=4;

if(t.data!=0) a[2]=3;

if(t.opsystem!=0) a[3]=3;

if(t.csystem!=0) a[4]=3;

if(t.programming!=0)a[5]=2;

if(t.compute!=0) a[6]=2;

if(t.network!=0) a[7]=2;

if(t.database!=0) a[8]=2;

if(t.compile!=0) a[9]=2;

if(t.datamining!=0) a[10]=2;

if(t.ai!=0) a[11]=2;

if(t.softeng!=0) a[12]=2;

if(t.security!=0) a[13]=2;

if(t.multimedia!=0) a[14]=2;

for(i=0;i<15;i++){

t.sum+=a[i];

}

cout<<"学号"<<'\t'<<'\t'<<"姓名"<<'\t'<<"政治"<<'\t'<<"政治学分"<<'\t'<<"英语"<<'\t'<<"英语学分"<<'\t'<<"数据结构"<<'\t'<<"数据结构学分"<<endl;

cout<<t.number<<'\t'<<t.name<<'\t'<<t.Politics<<'\t'<<a[0]<<'\t'<<'\t'<<t.English<<'\t'<<a[1]<<'\t'<<'\t'<<t.data<<'\t'<<'\t'<<a[2]<<endl;

cout<<"操作系统"<<'\t'<<"操作系统学分"<<'\t'<<"计算机系统"<<'\t'<<"计算机系统学分"<<'\t'<<"数据结构"<<'\t'<<"数据结构学分"<<endl;

cout<<t.opsystem<<'\t'<<'\t'<<a[3]<<'\t'<<'\t'<<t.csystem<<'\t'<<'\t'<<a[4]<<'\t'<<'\t'<<t.data<<'\t'<<'\t'<<a[5]<<endl;

cout<<"操作系统"<<'\t'<<"操作系统学分"<<'\t'<<"计算机网络"<<'\t'<<"计算机网络学分"<<'\t'<<"数据库概论"<<'\t'<<"数据库概论学分"<<endl;

cout<<t.opsystem<<'\t'<<'\t'<<a[6]<<'\t'<<'\t'<<t.network<<'\t'<<'\t'<<a[7]<<'\t'<<'\t'<<t.database<<'\t'<<'\t'<<a[8]<<endl;

cout<<"编译原理"<<'\t'<<"编译原理学分"<<'\t'<<"数据挖掘"<<'\t'<<"数据挖掘学分"<<'\t'<<"人工智能"<<'\t'<<"人工智能学分"<<endl;

cout<<t.compile<<'\t'<<'\t'<<a[9]<<'\t'<<'\t'<<t.datamining<<'\t'<<'\t'<<a[10]<<'\t'<<'\t'<<t.ai<<'\t'<<'\t'<<a[11]<<endl;

cout<<"软件工程"<<'\t'<<"软件工程学分"<<'\t'<<"信息安全"<<'\t'<<"信息安全学分"<<'\t'<<"多媒体技术"<<'\t'<<"多媒体技术学分"<<endl;

cout<<t.softeng<<'\t'<<'\t'<<a[12]<<'\t'<<'\t'<<t.security<<'\t'<<'\t'<<a[13]<<'\t'<<'\t'<<t.multimedia<<'\t'<<'\t'<<a[14]<<endl;

cout<<"已修总学分"<<'\t'<<"学分绩排名"<<endl;

cout<<t.sum<<'\t'<<'\t'<<t.ranking<<endl;

}

6.查找某一门课没有成绩的学生名单函数Searchnull

先输入要查询的课程的名字，不同的名字判断不同课程的成绩，如果成绩小于60，输出该学生的学号、姓名。

void Searchnull(inf grade[],int n)

{

string s,s1,s2,s3,s4,s5;

int i;

cout<<"要查询哪一门课？"<<endl;

s1="英语"; s2="政治"; s3="数据结构"; s4="计算机系统"; s5="操作系统";

cin>>s;

if(s==s1){//不同的课程进行不同的判断，如果成绩小于60，输出学号姓名

for(i=0; i<n; i++)

if(grade[i].English<60)

cout<<grade[i].number<<'\t'<<grade[i].name<<endl;

}

else if(s==s2){

for(i=0; i<n; i++)

if(grade[i].Politics<60)

cout<<grade[i].number<<'\t'<<grade[i].name<<endl;

}

else if(s==s3){

for(i=0; i<n; i++)

if(grade[i].data<60)

cout<<grade[i].number<<'\t'<<grade[i].name<<endl;

}

else if(s==s4){

for(i=0; i<n; i++)

if(grade[i].csystem<60)

cout<<grade[i].number<<'\t'<<grade[i].name<<endl;

}

else if(s==s5){

for(i=0; i<n; i++)

if(grade[i].opsystem<60)

cout<<grade[i].number<<'\t'<<grade[i].name<<endl;

}

}

7.计算每个同学总学分的函数Countsum

先判断每一门课程是否获得了相应的学分，再把学分加起来存到结构变量的sum 里面。

void Countsum(inf grade[],int n)

{

int i, j;

int a[15];

for(i=0; i<n; i++){

for(j=0; j<15; j++)

a[j]=0;

grade[i].sum=0;

if(grade[i].Politics!=0) a[0]=4;//判断每一门课程是否获得了相应的学分

if(grade[i].English!=0) a[1]=4;

if(grade[i].data!=0) a[2]=3;

if(grade[i].opsystem!=0) a[3]=3;

if(grade[i].csystem!=0) a[4]=3;

if(grade[i].programming!=0)a[5]=2;

if(grade[i].compute!=0) a[6]=2;

if(grade[i].network!=0) a[7]=2;

if(grade[i].database!=0) a[8]=2;

if(grade[i].compile!=0) a[9]=2;

if(grade[i].datamining!=0) a[10]=2;

if(grade[i].ai!=0) a[11]=2;

if(grade[i].softeng!=0) a[12]=2;

if(grade[i].security!=0) a[13]=2;

if(grade[i].multimedia!=0) a[14]=2;

for(j=0;j<15;j++){//学分数相加

grade[i].sum+=a[j];

}

}

}

8.查询并输出学分不够27的学生名单的函数Search

因为已经算出了每个结构变量的sum值（总学分数），只需要遍历数组中的值，如果小于27，就输出学号姓名。

void Search(inf grade[], int n)

{

for(int i=0; i<n; i++)

if(grade[i].sum<27)

cout<<grade[i].number<<'\t'<<grade[i].name<<endl;

}

**四、整体代码展示**

#include<iostream>

#include<fstream>

#include<string>

#include<iomanip>

using namespace std;

struct inf

{

string number, name;

int Politics, English;

int data, opsystem, csystem;

int programming, compute, compile, softeng, security, network, database, datamining, ai, multimedia;

double credit;

int ranking,sum;

};

void Classify(inf score[], inf score14[], inf score15[], int &m1, int &m2, int &n)

{

for(int i=0; i<n; i++){

if(score[i].number[3]-'0'==4)

score14[m1++]=score[i];

else if(score[i].number[3]-'0'==5)

score15[m2++]=score[i];

else

cout<<"成绩单中的学号出现错误。"<<endl;

}

}

void Dealwithscore(inf grade[], inf score[], int &n1, int &n)

{

int i, j, k;

char s[10]; string ss; string ss1="姓名"; string ss2="学号";

char s1[]="方案一"; char s2[]="方案二";

for(i=0; i<n; i++){

for(j=0; j<n1; j++)

if(grade[j].name==score[i].name){

if(grade[j].number!=score[i].number){

for(k=0; k<n1; k++){

if(grade[k].number==score[i].number&&grade[k].name==score[i].name)

cout<<"出现了重名的情况"<<grade[j].number<<"和"<<grade[k].number<<"的姓名都是"<<grade[k].name<<endl;

}

break;

cout<<"成绩单中的姓名"<<score[i].name<<"的学号"<<score[i].number<<"与名单中的学号"<<grade[j].number<<"不符！"<<endl;

cout<<"现在，有两种解决方案\n"<<"方案一：在成绩单中删除该行信息\n"<<"方案二：成绩不变，根据学生名单中的姓名，提示学号对应的正确学号，供修改。"<<endl;

cout<<"你要选择哪一种？"<<endl;

cin>>s;

if(strcmp(s,s1)==0){

for(k=i; k<n; k++)

score[k]=score[k+1];

n--;

}

else if(strcmp(s,s2)==0){

cout<<"正确的学号是"<<grade[j].number<<"请在下方输入"<<endl;

for(; ; ) {

cin>>ss;

if(ss==grade[j].number) break;

else cout<<"输入错误，请重新修改！"<<endl;

}

}

}

break;

}

if(j==n1) cout<<"姓名"<<score[i].name<<"不存在于名单中！"<<endl;

}

for(i=0; i<n; i++){

for(j=0; j<n1; j++)

if(grade[j].number==score[i].number){

if(grade[j].name!=score[i].name){

cout<<"成绩单中学号"<<score[i].number<<"的姓名"<<score[i].name<<"与名单中的姓名"<<grade[j].name<<"不符！"<<endl;

cout<<"现在，有两种解决方案\n"<<"方案一：在成绩单中删除该行信息\n"<<"方案二：成绩不变，根据学生名单中的学号，提示学号对应的正确姓名，或者根据姓名给出学号，供修改。"<<endl;

cout<<"你要选择哪一种？"<<endl;

cin>>s;

if(strcmp(s,s1)==0){

for(k=i; k<n; k++)

score[k]=score[k+1];

n--;

}

else if(strcmp(s,s2)==0){

cout<<"正确的姓名是"<<grade[j].name<<"请在下方输入"<<endl;

for(; ; ) {

cin>>ss;

if(ss==grade[j].name) break;

else cout<<"输入错误，请重新修改！"<<endl;

}

}

}

break;

}

if(j==n1) {

cout<<"学号"<<score[i].number<<"不存在！"<<endl;

cout<<"删除成绩单中该行的信息"<<endl;

for(k=i; k<n; k++)

score[k]=score[k+1];

n--;

}

}

}

double Credit(inf t)

{

int a[10];

int i,j,k,p,sum;

a[0]=t.ai;

a[1]=t.compile;

a[2]=t.compute;

a[3]=t.database;

a[4]=t.datamining;

a[5]=t.multimedia;

a[6]=t.network;

a[7]=t.programming;

a[8]=t.security;

a[9]=t.softeng;

for(i=0; i<10; i++){

for(k=i,j=k+1; j<10; j++) if(a[j]<a[k]) k=j;

if(k!=i){ p=a[k]; a[k]=a[i]; a[i]=p;}

}

sum=0;

for(i=9; i>4; i--){

sum+=a[i];

}

return (t.English+t.Politics+t.data+t.csystem+t.opsystem+sum)/10.0;

}

void Rank(inf grade[], int n)

{

int i,j,k;

inf t;

for(i=0; i<n; i++){

grade[i].credit=Credit(grade[i]);

}

for(i=0; i<n; i++){

grade[i].ranking=1;

for(j=0; j<n; j++){

if((grade[j].credit-grade[i].credit)>1e-7)

grade[i].ranking++;

}

}

for(i=0; i<n; i++){

for(k=i,j=k+1; j<n; j++) if(grade[j].ranking<grade[k].ranking) k=j;

if(k!=i){ t=grade[k]; grade[k]=grade[i]; grade[i]=t;}

}

}

void output(inf &t)

{

int a[15];

int i;

t.sum=0;

for(i=0; i<15; i++)

a[i]=0;

if(t.Politics!=0) a[0]=4;

if(t.English!=0) a[1]=4;

if(t.data!=0) a[2]=3;

if(t.opsystem!=0) a[3]=3;

if(t.csystem!=0) a[4]=3;

if(t.programming!=0)a[5]=2;

if(t.compute!=0) a[6]=2;

if(t.network!=0) a[7]=2;

if(t.database!=0) a[8]=2;

if(t.compile!=0) a[9]=2;

if(t.datamining!=0) a[10]=2;

if(t.ai!=0) a[11]=2;

if(t.softeng!=0) a[12]=2;

if(t.security!=0) a[13]=2;

if(t.multimedia!=0) a[14]=2;

for(i=0;i<15;i++){

t.sum+=a[i];

}

cout<<"学号"<<'\t'<<'\t'<<"姓名"<<'\t'<<"政治"<<'\t'<<"政治学分"<<'\t'<<"英语"<<'\t'<<"英语学分"<<'\t'<<"数据结构"<<'\t'<<"数据结构学分"<<endl;

cout<<t.number<<'\t'<<t.name<<'\t'<<t.Politics<<'\t'<<a[0]<<'\t'<<'\t'<<t.English<<'\t'<<a[1]<<'\t'<<'\t'<<t.data<<'\t'<<'\t'<<a[2]<<endl;

cout<<"操作系统"<<'\t'<<"操作系统学分"<<'\t'<<"计算机系统"<<'\t'<<"计算机系统学分"<<'\t'<<"数据结构"<<'\t'<<"数据结构学分"<<endl;

cout<<t.opsystem<<'\t'<<'\t'<<a[3]<<'\t'<<'\t'<<t.csystem<<'\t'<<'\t'<<a[4]<<'\t'<<'\t'<<t.data<<'\t'<<'\t'<<a[5]<<endl;

cout<<"操作系统"<<'\t'<<"操作系统学分"<<'\t'<<"计算机网络"<<'\t'<<"计算机网络学分"<<'\t'<<"数据库概论"<<'\t'<<"数据库概论学分"<<endl;

cout<<t.opsystem<<'\t'<<'\t'<<a[6]<<'\t'<<'\t'<<t.network<<'\t'<<'\t'<<a[7]<<'\t'<<'\t'<<t.database<<'\t'<<'\t'<<a[8]<<endl;

cout<<"编译原理"<<'\t'<<"编译原理学分"<<'\t'<<"数据挖掘"<<'\t'<<"数据挖掘学分"<<'\t'<<"人工智能"<<'\t'<<"人工智能学分"<<endl;

cout<<t.compile<<'\t'<<'\t'<<a[9]<<'\t'<<'\t'<<t.datamining<<'\t'<<'\t'<<a[10]<<'\t'<<'\t'<<t.ai<<'\t'<<'\t'<<a[11]<<endl;

cout<<"软件工程"<<'\t'<<"软件工程学分"<<'\t'<<"信息安全"<<'\t'<<"信息安全学分"<<'\t'<<"多媒体技术"<<'\t'<<"多媒体技术学分"<<endl;

cout<<t.softeng<<'\t'<<'\t'<<a[12]<<'\t'<<'\t'<<t.security<<'\t'<<'\t'<<a[13]<<'\t'<<'\t'<<t.multimedia<<'\t'<<'\t'<<a[14]<<endl;

cout<<"已修总学分"<<'\t'<<"学分绩排名"<<endl;

cout<<t.sum<<'\t'<<'\t'<<t.ranking<<endl;

}

void Searchnull(inf grade[],int n)

{

string s,s1,s2,s3,s4,s5;

int i;

cout<<"要查询哪一门课？"<<endl;

s1="英语"; s2="政治"; s3="数据结构"; s4="计算机系统"; s5="操作系统";

cin>>s;

if(s==s1){

for(i=0; i<n; i++)

if(grade[i].English<60)

cout<<grade[i].number<<'\t'<<grade[i].name<<endl;

}

else if(s==s2){

for(i=0; i<n; i++)

if(grade[i].Politics<60)

cout<<grade[i].number<<'\t'<<grade[i].name<<endl;

}

else if(s==s3){

for(i=0; i<n; i++)

if(grade[i].data<60)

cout<<grade[i].number<<'\t'<<grade[i].name<<endl;

}

else if(s==s4){

for(i=0; i<n; i++)

if(grade[i].csystem<60)

cout<<grade[i].number<<'\t'<<grade[i].name<<endl;

}

else if(s==s5){

for(i=0; i<n; i++)

if(grade[i].opsystem<60)

cout<<grade[i].number<<'\t'<<grade[i].name<<endl;

}

}

void Countsum(inf grade[],int n)

{

int i, j;

int a[15];

for(i=0; i<n; i++){

for(j=0; j<15; j++)

a[j]=0;

grade[i].sum=0;

if(grade[i].Politics!=0) a[0]=4;

if(grade[i].English!=0) a[1]=4;

if(grade[i].data!=0) a[2]=3;

if(grade[i].opsystem!=0) a[3]=3;

if(grade[i].csystem!=0) a[4]=3;

if(grade[i].programming!=0)a[5]=2;

if(grade[i].compute!=0) a[6]=2;

if(grade[i].network!=0) a[7]=2;

if(grade[i].database!=0) a[8]=2;

if(grade[i].compile!=0) a[9]=2;

if(grade[i].datamining!=0) a[10]=2;

if(grade[i].ai!=0) a[11]=2;

if(grade[i].softeng!=0) a[12]=2;

if(grade[i].security!=0) a[13]=2;

if(grade[i].multimedia!=0) a[14]=2;

for(j=0;j<15;j++){

grade[i].sum+=a[j];

}

}

}

void Search(inf grade[], int n)

{

for(int i=0; i<n; i++)

if(grade[i].sum<27)

cout<<grade[i].number<<'\t'<<grade[i].name<<endl;

}

int main()

{

inf grade15[100], grade14[100], English14[100], Politics14[100], English15[100], Politics15[100],opsystem[200],opsystem14[100],opsystem15[100],compile[200],compile14[100],compile15[100],csystem[200], csystem14[100],csystem15[100],data[200],data14[100],data15[100],database[200],database14[100],database15[100],programming[200],programming14[100],programming15[100],compute[200],compute14[100],compute15[100],softeng[200],softeng14[100],softeng15[100],security[200],security14[100],security15[100],network[200],network14[100],network15[100],datamining[200],datamining14[100],datamining15[100],ai[200],ai14[100],ai15[100],multimedia[200],multimedia14[100],multimedia15[100],t;

int n1,n2,m1,m2,n,i,j,x,y; n1=n2=m1=m2=n=0;

ifstream infile("14级学生名单.txt");

if(!infile)

{ cerr<<"打开文件：14级学生名单.txt 失败"<<endl; exit(0);}

while(infile>>t.number){

infile>>t.name;

grade14[n1++]=t;

}

ifstream infile2("15级学生名单.txt");

if(!infile2)

{ cerr<<"打开文件：15级学生名单.txt 失败"<<endl; exit(0);}

while(infile2>>t.number){

infile2>>t.name;

grade15[n2++]=t;

}

ifstream infile3("14级英语成绩.txt");

if(!infile3)

{ cerr<<"打开文件：14级英语成绩.txt 失败"<<endl; exit(0);}

while(infile3>>t.number){

infile3>>t.name;

infile3>>t.English;

English14[n++]=t;

}

Dealwithscore(grade14, English14, n1, n);

for(j=0; j<n1; j++){

for(i=0; i<n; i++)

if(grade14[j].number==English14[i].number){

if(English14[i].English>=60) grade14[j].English=English14[i].English;

else grade14[j].English=0;

break;

}

if(i==n) grade14[j].English=0;

}

ofstream out("14级英语成绩.txt");

if(!out)

{ cerr<<"打开文件14级英语成绩.txt失败！"<<endl; exit(0);}

for(i=0; i<n; i++){

out<<English14[i].number<<'\t'<<English14[i].name<<'\t'<<English14[i].English<<endl;

}

out.close();

n=0;

ifstream infile4("14级政治成绩.txt");

if(!infile4)

{ cerr<<"打开文件：14级政治成绩.txt 失败"<<endl; exit(0);}

while(infile4>>t.number){

infile4>>t.name;

infile4>>t.Politics;

Politics14[n++]=t;

}

Dealwithscore(grade14, Politics14, n1, n);

for(j=0; j<n1; j++){

for(i=0; i<n; i++)

if(grade14[j].number==Politics14[i].number){

if(Politics14[i].Politics>=60) grade14[j].Politics=Politics14[i].Politics;

else grade14[j].Politics=0;

break;

}

if(i==n) grade14[j].Politics=0;

}

ofstream out2("14级政治成绩.txt");

if(!out2)

{ cerr<<"打开文件14级政治成绩.txt失败！"<<endl; exit(0);}

for(i=0; i<n; i++){

out2<<Politics14[i].number<<'\t'<<Politics14[i].name<<'\t'<<Politics14[i].Politics<<endl;

}

out2.close();

n=0;

ifstream infile5("15级英语成绩.txt");

if(!infile5)

{ cerr<<"打开文件：15级英语成绩.txt 失败"<<endl; exit(0);}

while(infile5>>t.number){

infile5>>t.name;

infile5>>t.English;

English15[n++]=t;

}

Dealwithscore(grade15, English15, n2, n);

for(j=0; j<n1; j++){

for(i=0; i<n; i++)

if(grade15[j].number==English15[i].number){

if(English15[i].English>=60) grade15[j].English=English15[i].English;

else grade15[j].English=0;

break;

}

if(i==n) grade15[j].English=0;

}

ofstream out3("15级英语成绩.txt");

if(!out3)

{ cerr<<"打开文件15级英语成绩.txt失败！"<<endl; exit(0);}

for(i=0; i<n; i++){

out3<<English15[i].number<<'\t'<<English15[i].name<<'\t'<<English15[i].English<<endl;

}

out3.close();

n=0;

ifstream infile6("15级政治成绩.txt");

if(!infile6)

{ cerr<<"打开文件：15级政治成绩.txt 失败"<<endl; exit(0);}

while(infile6>>t.number){

infile6>>t.name;

infile6>>t.Politics;

Politics15[n++]=t;

}

Dealwithscore(grade15, Politics15, n2, n);

for(j=0; j<n1; j++){

for(i=0; i<n; i++)

if(grade15[j].number==Politics15[i].number){

if(Politics15[i].Politics>=60) grade15[j].Politics=Politics15[i].Politics;

else grade15[j].Politics=0;

break;

}

if(i==n) grade15[j].Politics=0;

}

ofstream out4("15级政治成绩.txt");

if(!out4)

{ cerr<<"打开文件15级政治成绩.txt失败！"<<endl; exit(0);}

for(i=0; i<n; i++){

out4<<Politics15[i].number<<'\t'<<Politics15[i].name<<'\t'<<Politics15[i].Politics<<endl;

}

out4.close();

n=m1=m2=0;

ifstream infile7("操作系统成绩.txt");

if(!infile7)

{ cerr<<"打开文件操作系统成绩.txt 失败"<<endl; exit(0);}

while(infile7>>t.number){

infile7>>t.name;

infile7>>t.opsystem;

opsystem[n++]=t;

}

Classify(opsystem, opsystem14, opsystem15, m1, m2, n);

Dealwithscore(grade14,opsystem14, n1, m1);

for(j=0; j<n1; j++){

for(i=0; i<m1; i++)

if(grade14[j].number==opsystem14[i].number){

if(opsystem14[i].opsystem>=60) grade14[j].opsystem=opsystem14[i].opsystem;

else grade14[j].opsystem=0;

break;

}

if(i==m1) grade14[j].opsystem=0;

}

Dealwithscore(grade15, opsystem15, n1, m2);

for(j=0; j<n1; j++){

for(i=0; i<m2; i++)

if(grade15[j].number==opsystem15[i].number){

if(opsystem15[i].opsystem>=60) grade15[j].opsystem=opsystem15[i].opsystem;

else grade15[j].opsystem=0;

break;

}

if(i==m2) grade15[j].opsystem=0;

}

ofstream out5("操作系统成绩.txt");

if(!out5)

{ cerr<<"打开文件操作系统成绩.txt失败！"<<endl; exit(0);}

for(i=0; i<m1; i++){

out5<<opsystem14[i].number<<'\t'<<opsystem14[i].name<<'\t'<<opsystem14[i].opsystem<<endl;

}

for(i=0; i<m2; i++){

out5<<opsystem15[i].number<<'\t'<<opsystem15[i].name<<'\t'<<opsystem15[i].opsystem<<endl;

}

out5.close();

n=m1=m2=0;

ifstream infile8("编译原理成绩.txt");

if(!infile8)

{ cerr<<"打开文件编译原理成绩.txt 失败"<<endl; exit(0);}

while(infile8>>t.number){

infile8>>t.name;

infile8>>t.compile;

compile[n++]=t;

}

Classify(compile, compile14, compile15, m1, m2, n);

Dealwithscore(grade14, compile14, n1, m1);

for(j=0; j<n1; j++){

for(i=0; i<m1; i++)

if(grade14[j].number==compile14[i].number){

if(compile14[i].compile>=60) grade14[j].compile=compile14[i].compile;

else grade14[j].compile=0;

break;

}

if(i==m1) grade14[j].compile=0;

}

Dealwithscore(grade15, compile15, n1, m2);

for(j=0; j<n1; j++){

for(i=0; i<m2; i++)

if(grade15[j].number==compile15[i].number){

if(compile15[i].compile>=60) grade15[j].compile=compile15[i].compile;

else grade15[j].compile=0;

break;

}

if(i==m2) grade15[j].compile=0;

}

ofstream out6("编译原理成绩.txt");

if(!out6)

{ cerr<<"打开文件编译原理成绩.txt失败！"<<endl; exit(0);}

for(i=0; i<m1; i++){

out6<<compile14[i].number<<'\t'<<compile14[i].name<<'\t'<<compile14[i].compile<<endl;

}

for(i=0; i<m2; i++){

out6<<compile15[i].number<<'\t'<<compile15[i].name<<'\t'<<compile15[i].compile<<endl;

}

out6.close();

n=m1=m2=0;

ifstream infile9("计算机系统成绩.txt");

if(!infile9)

{ cerr<<"打开文件计算机系统成绩.txt 失败"<<endl; exit(0);}

while(infile9>>t.number){

infile9>>t.name;

infile9>>t.csystem;

csystem[n++]=t;

}

Classify(csystem, csystem14, csystem15, m1, m2, n);

Dealwithscore(grade14, csystem14, n1, m1);

for(j=0; j<n1; j++){

for(i=0; i<m1; i++)

if(grade14[j].number==csystem14[i].number){

if(csystem14[i].csystem>=60) grade14[j].csystem=csystem14[i].opsystem;

else grade14[j].csystem=0;

break;

}

if(i==m1) grade14[j].csystem=0;

}

Dealwithscore(grade15, csystem15, n1, m2);

for(j=0; j<n1; j++){

for(i=0; i<m2; i++)

if(grade15[j].number==csystem15[i].number){

if(csystem15[i].csystem>=60) grade15[j].csystem=csystem15[i].csystem;

else grade15[j].csystem=0;

break;

}

if(i==m2) grade15[j].csystem=0;

}

ofstream out7("计算机系统成绩.txt");

if(!out7)

{ cerr<<"打开文件计算机系统成绩.txt失败！"<<endl; exit(0);}

for(i=0; i<m1; i++){

out7<<csystem14[i].number<<'\t'<<csystem14[i].name<<'\t'<<csystem14[i].csystem<<endl;

}

for(i=0; i<m2; i++){

out7<<csystem15[i].number<<'\t'<<csystem15[i].name<<'\t'<<csystem15[i].csystem<<endl;

}

out7.close();

n=m1=m2=0;

ifstream infile10("数据结构成绩.txt");

if(!infile10)

{ cerr<<"打开文件数据结构成绩.txt 失败"<<endl; exit(0);}

while(infile10>>t.number){

infile10>>t.name;

infile10>>t.data;

data[n++]=t;

}

Classify(data, data14, data15, m1, m2, n);

Dealwithscore(grade14, data14, n1, m1);

for(j=0; j<n1; j++){

for(i=0; i<m1; i++)

if(grade14[j].number==data14[i].number){

if(data14[i].data>=60) grade14[j].data=data14[i].data;

else grade14[j].data=0;

break;

}

if(i==m1) grade14[j].data=0;

}

Dealwithscore(grade15, data15, n1, m2);

for(j=0; j<n1; j++){

for(i=0; i<m2; i++)

if(grade15[j].number==data15[i].number){

if(data15[i].data>=60) grade15[j].data=data15[i].data;

else grade15[j].data=0;

break;

}

if(i==m2) grade15[j].data=0;

}

ofstream out8("数据结构成绩.txt");

if(!out8)

{ cerr<<"打开文件数据结构成绩.txt失败！"<<endl; exit(0);}

for(i=0; i<m1; i++){

out8<<data14[i].number<<'\t'<<data14[i].name<<'\t'<<data14[i].data<<endl;

}

for(i=0; i<m2; i++){

out8<<data15[i].number<<'\t'<<data15[i].name<<'\t'<<data15[i].data<<endl;

}

out8.close();

n=m1=m2=0;

ifstream infile11("数据库概论成绩.txt");

if(!infile11)

{ cerr<<"打开文件数据库概论成绩.txt 失败"<<endl; exit(0);}

while(infile11>>t.number){

infile11>>t.name;

infile11>>t.database;

database[n++]=t;

}

Classify(database, database14, database15, m1, m2, n);

Dealwithscore(grade14, database14, n1, m1);

for(j=0; j<n1; j++){

for(i=0; i<m1; i++)

if(grade14[j].number==database14[i].number){

if(database14[i].database>=60) grade14[j].database=database14[i].database;

else grade14[j].database=0;

break;

}

if(i==m1) grade14[j].database=0;

}

Dealwithscore(grade15, database15, n1, m2);

for(j=0; j<n1; j++){

for(i=0; i<m2; i++)

if(grade15[j].number==database15[i].number){

if(database15[i].database>=60) grade15[j].database=database15[i].database;

else grade15[j].database=0;

break;

}

if(i==m2) grade15[j].database=0;

}

ofstream out9("数据库概论成绩.txt");

if(!out9)

{ cerr<<"打开文件数据库概论成绩.txt失败！"<<endl; exit(0);}

for(i=0; i<m1; i++){

out9<<database14[i].number<<'\t'<<database14[i].name<<'\t'<<database14[i].database<<endl;

}

for(i=0; i<m2; i++){

out9<<database15[i].number<<'\t'<<database15[i].name<<'\t'<<database15[i].database<<endl;

}

out9.close();

n=m1=m2=0;

ifstream infile12("程序设计成绩.txt");

if(!infile12)

{ cerr<<"打开文件程序设计成绩.txt 失败"<<endl; exit(0);}

while(infile12>>t.number){

infile12>>t.name;

infile12>>t.programming;

programming[n++]=t;

}

Classify(programming, programming14, programming15, m1, m2, n);

Dealwithscore(grade14, programming14, n1, m1);

for(j=0; j<n1; j++){

for(i=0; i<m1; i++)

if(grade14[j].number==programming14[i].number){

if(programming14[i].programming>=60) grade14[j].programming=programming14[i].programming;

else grade14[j].programming=0;

break;

}

if(i==m1) grade14[j].programming=0;

}

Dealwithscore(grade15, programming15, n1, m2);

for(j=0; j<n1; j++){

for(i=0; i<m2; i++)

if(grade15[j].number==programming15[i].number){

if(programming15[i].programming>=60) grade15[j].programming=programming15[i].programming;

else grade15[j].programming=0;

break;

}

if(i==m2) grade15[j].programming=0;

}

ofstream out10("程序设计成绩.txt");

if(!out10)

{ cerr<<"打开文件程序设计成绩.txt失败！"<<endl; exit(0);}

for(i=0; i<m1; i++){

out10<<programming14[i].number<<'\t'<<programming14[i].name<<'\t'<<programming14[i].programming<<endl;

}

for(i=0; i<m2; i++){

out10<<programming15[i].number<<'\t'<<programming15[i].name<<'\t'<<programming15[i].programming<<endl;

}

out10.close();

n=m1=m2=0;

ifstream infile13("计算思维成绩.txt");

if(!infile13)

{ cerr<<"打开文件计算思维成绩.txt 失败"<<endl; exit(0);}

while(infile13>>t.number){

infile13>>t.name;

infile13>>t.compute;

compute[n++]=t;

}

Classify(compute, compute14, compute15, m1, m2, n);

Dealwithscore(grade14, compute14, n1, m1);

for(j=0; j<n1; j++){

for(i=0; i<m1; i++)

if(grade14[j].number==compute14[i].number){

if(compute14[i].compute>=60) grade14[j].compute=compute14[i].compute;

else grade14[j].compute=0;

break;

}

if(i==m1) grade14[j].compute=0;

}

Dealwithscore(grade15, compute15, n1, m2);

for(j=0; j<n1; j++){

for(i=0; i<m2; i++)

if(grade15[j].number==compute15[i].number){

if(compute15[i].compute>=60) grade15[j].compute=compute15[i].compute;

else grade15[j].compute=0;

break;

}

if(i==m2) grade15[j].compute=0;

}

ofstream out11("计算思维成绩.txt");

if(!out11)

{ cerr<<"打开文件计算思维成绩.txt失败！"<<endl; exit(0);}

for(i=0; i<m1; i++){

out11<<compute14[i].number<<'\t'<<compute14[i].name<<'\t'<<compute14[i].compute<<endl;

}

for(i=0; i<m2; i++){

out11<<compute15[i].number<<'\t'<<compute15[i].name<<'\t'<<compute15[i].compute<<endl;

}

out11.close();

n=m1=m2=0;

ifstream infile14("软件工程成绩.txt");

if(!infile14)

{ cerr<<"打开文件软件工程成绩.txt 失败"<<endl; exit(0);}

while(infile14>>t.number){

infile14>>t.name;

infile14>>t.softeng;

softeng[n++]=t;

}

Classify(softeng, softeng14, softeng15, m1, m2, n);

Dealwithscore(grade14, softeng14, n1, m1);

for(j=0; j<n1; j++){

for(i=0; i<m1; i++)

if(grade14[j].number==softeng14[i].number){

if(softeng14[i].softeng>=60) grade14[j].softeng=softeng14[i].softeng;

else grade14[j].softeng=0;

break;

}

if(i==m1) grade14[j].softeng=0;

}

Dealwithscore(grade15, softeng15, n1, m2);

for(j=0; j<n1; j++){

for(i=0; i<m2; i++)

if(grade15[j].number==softeng15[i].number){

if(softeng15[i].softeng>=60) grade15[j].softeng=softeng15[i].softeng;

else grade15[j].softeng=0;

break;

}

if(i==m2) grade15[j].softeng=0;

}

ofstream out12("软件工程成绩.txt");

if(!out12)

{ cerr<<"打开文件软件工程成绩.txt失败！"<<endl; exit(0);}

for(i=0; i<m1; i++){

out12<<softeng14[i].number<<'\t'<<softeng14[i].name<<'\t'<<softeng14[i].softeng<<endl;

}

for(i=0; i<m2; i++){

out12<<softeng15[i].number<<'\t'<<softeng15[i].name<<'\t'<<softeng15[i].softeng<<endl;

}

out12.close();

n=m1=m2=0;

ifstream infile15("信息安全成绩.txt");

if(!infile15)

{ cerr<<"打开文件信息安全成绩.txt 失败"<<endl; exit(0);}

while(infile15>>t.number){

infile15>>t.name;

infile15>>t.security;

security[n++]=t;

}

Classify(security, security14, security15, m1, m2, n);

Dealwithscore(grade14, security14, n1, m1);

for(j=0; j<n1; j++){

for(i=0; i<m1; i++)

if(grade14[j].number==security14[i].number){

if(security14[i].security>=60) grade14[j].security=security14[i].security;

else grade14[j].security=0;

break;

}

if(i==m1) grade14[j].security=0;

}

Dealwithscore(grade15, security15, n1, m2);

for(j=0; j<n1; j++){

for(i=0; i<m2; i++)

if(grade15[j].number==security15[i].number){

if(security15[i].security>=60) grade15[j].security=security15[i].security;

else grade15[j].security=0;

break;

}

if(i==m2) grade15[j].security=0;

}

ofstream out13("信息安全成绩.txt");

if(!out13)

{ cerr<<"打开文件信息安全成绩.txt失败！"<<endl; exit(0);}

for(i=0; i<m1; i++){

out13<<security14[i].number<<'\t'<<security14[i].name<<'\t'<<security14[i].security<<endl;

}

for(i=0; i<m2; i++){

out13<<security15[i].number<<'\t'<<security15[i].name<<'\t'<<security15[i].security<<endl;

}

out13.close();

n=m1=m2=0;

ifstream infile16("计算机网络成绩.txt");

if(!infile16)

{ cerr<<"打开文件计算机网络成绩.txt 失败"<<endl; exit(0);}

while(infile16>>t.number){

infile16>>t.name;

infile16>>t.network;

network[n++]=t;

}

Classify(network, network14, network15, m1, m2, n);

Dealwithscore(grade14, network14, n1, m1);

for(j=0; j<n1; j++){

for(i=0; i<m1; i++)

if(grade14[j].number==network14[i].number){

if(network14[i].network>=60) grade14[j].network=network14[i].network;

else grade14[j].network=0;

break;

}

if(i==m1) grade14[j].network=0;

}

Dealwithscore(grade15, network15, n1, m2);

for(j=0; j<n1; j++){

for(i=0; i<m2; i++)

if(grade15[j].number==network15[i].number){

if(network15[i].network>=60) grade15[j].network=network15[i].network;

else grade15[j].network=0;

break;

}

if(i==m2) grade15[j].network=0;

}

ofstream out14("计算机网络成绩.txt");

if(!out14)

{ cerr<<"打开文件计算机网络成绩.txt失败！"<<endl; exit(0);}

for(i=0; i<m1; i++){

out14<<network14[i].number<<'\t'<<network14[i].name<<'\t'<<network14[i].network<<endl;

}

for(i=0; i<m2; i++){

out14<<network15[i].number<<'\t'<<network15[i].name<<'\t'<<network15[i].network<<endl;

}

out14.close();

n=m1=m2=0;

ifstream infile17("数据挖掘成绩.txt");

if(!infile17)

{ cerr<<"打开文件数据挖掘成绩.txt 失败"<<endl; exit(0);}

while(infile17>>t.number){

infile17>>t.name;

infile17>>t.datamining;

datamining[n++]=t;

}

Classify(datamining, datamining14, datamining15, m1, m2, n);

Dealwithscore(grade14, datamining14, n1, m1);

for(j=0; j<n1; j++){

for(i=0; i<m1; i++)

if(grade14[j].number==datamining14[i].number){

if(datamining14[i].datamining>=60) grade14[j].datamining=datamining14[i].datamining;

else grade14[j].datamining=0;

break;

}

if(i==m1) grade14[j].datamining=0;

}

Dealwithscore(grade15, datamining15, n1, m2);

for(j=0; j<n1; j++){

for(i=0; i<m2; i++)

if(grade15[j].number==datamining15[i].number){

if(datamining15[i].datamining>=60) grade15[j].datamining=datamining15[i].datamining;

else grade15[j].datamining=0;

break;

}

if(i==m2) grade15[j].datamining=0;

}

ofstream out15("数据挖掘成绩.txt");

if(!out15)

{ cerr<<"打开文件数据挖掘成绩.txt失败！"<<endl; exit(0);}

for(i=0; i<m1; i++){

out15<<datamining14[i].number<<'\t'<<datamining14[i].name<<'\t'<<datamining14[i].datamining<<endl;

}

for(i=0; i<m2; i++){

out15<<datamining15[i].number<<'\t'<<datamining15[i].name<<'\t'<<datamining15[i].datamining<<endl;

}

out15.close();

n=m1=m2=0;

ifstream infile18("人工智能成绩.txt");

if(!infile18)

{ cerr<<"打开文件人工智能成绩.txt 失败"<<endl; exit(0);}

while(infile18>>t.number){

infile18>>t.name;

infile18>>t.ai;

ai[n++]=t;

}

Classify(ai, ai14, ai15, m1, m2, n);

Dealwithscore(grade14, ai14, n1, m1);

for(j=0; j<n1; j++){

for(i=0; i<m1; i++)

if(grade14[j].number==ai14[i].number){

if(ai14[i].ai>=60) grade14[j].ai=ai14[i].ai;

else grade14[j].ai=0;

break;

}

if(i==m1) grade14[j].ai=0;

}

Dealwithscore(grade15, ai15, n1, m2);

for(j=0; j<n1; j++){

for(i=0; i<m2; i++)

if(grade15[j].number==ai15[i].number){

if(ai15[i].ai>=60) grade15[j].ai=ai15[i].ai;

else grade15[j].ai=0;

break;

}

if(i==m2) grade15[j].ai=0;

}

ofstream out16("人工智能成绩.txt");

if(!out16)

{ cerr<<"打开文件人工智能成绩.txt失败！"<<endl; exit(0);}

for(i=0; i<m1; i++){

out16<<ai14[i].number<<'\t'<<ai14[i].name<<'\t'<<ai14[i].ai<<endl;

}

for(i=0; i<m2; i++){

out16<<ai15[i].number<<'\t'<<ai15[i].name<<'\t'<<ai15[i].ai<<endl;

}

out16.close();

n=m1=m2=0;

ifstream infile19("多媒体技术成绩.txt");

if(!infile19)

{ cerr<<"打开文件多媒体技术成绩.txt 失败"<<endl; exit(0);}

while(infile19>>t.number){

infile19>>t.name;

infile19>>t.multimedia;

multimedia[n++]=t;

}

Classify(multimedia, multimedia14, multimedia15, m1, m2, n);

Dealwithscore(grade14, multimedia14, n1, m1);

for(j=0; j<n1; j++){

for(i=0; i<m1; i++)

if(grade14[j].number==multimedia14[i].number){

if(multimedia14[i].multimedia>=60) grade14[j].multimedia=multimedia14[i].multimedia;

else grade14[j].multimedia=0;

break;

}

if(i==m1) grade14[j].multimedia=0;

}

Dealwithscore(grade15, multimedia15, n1, m2);

for(j=0; j<n1; j++){

for(i=0; i<m2; i++)

if(grade15[j].number==multimedia15[i].number){

if(multimedia15[i].multimedia>=60) grade15[j].multimedia=multimedia15[i].multimedia;

else grade15[j].multimedia=0;

break;

}

if(i==m2) grade15[j].multimedia=0;

}

ofstream out17("多媒体技术成绩.txt");

if(!out17)

{ cerr<<"打开文件多媒体技术成绩.txt失败！"<<endl; exit(0);}

for(i=0; i<m1; i++){

out17<<multimedia14[i].number<<'\t'<<multimedia14[i].name<<'\t'<<multimedia14[i].multimedia<<endl;

}

for(i=0; i<m2; i++){

out17<<multimedia15[i].number<<'\t'<<multimedia15[i].name<<'\t'<<multimedia15[i].multimedia<<endl;

}

out17.close();

ofstream out18("成绩汇总.txt");

if(!out18)

{ cerr<<"打开文件成绩汇总.txt失败！"<<endl; exit(0);}

out18<<"学号"<<'\t'<<'\t'<<"姓名"<<'\t'<<"政治"<<'\t'<<"英语"<<'\t'<<"程序设计"<<'\t'<<"计算思维"<<'\t'<<"计算机系统"<<'\t'<<"数据结构"<<'\t'<<"操作系统"<<'\t'<<"计算机网络"<<'\t'<<"数据库概论"<<'\t'<<"编译原理"<<'\t'<<"数据挖掘"<<'\t'<<"人工智能"<<'\t'<<"软件工程"<<'\t'<<"信息安全"<<'\t'<<"多媒体技术"<<endl;

for(i=0; i<n1; i++){

out18<<grade14[i].number<<'\t'<<grade14[i].name<<'\t'<<grade14[i].Politics<<'\t'<<grade14[i].English<<'\t'<<grade14[i].programming<<'\t'<<grade14[i].compute<<'\t'<<grade14[i].csystem<<'\t'<<'\t'<<grade14[i].data<<'\t'<<grade14[i].opsystem<<'\t'<<grade14[i].network<<'\t'<<'\t'<<grade14[i].database<<'\t'<<'\t'<<grade14[i].compile<<'\t'<<grade14[i].datamining<<'\t'<<grade14[i].ai<<'\t'<<grade14[i].softeng<<'\t'<<grade14[i].security<<'\t'<<grade14[i].multimedia<<endl;

}

for(i=0; i<n2; i++){

out18<<grade15[i].number<<'\t'<<grade15[i].name<<'\t'<<grade15[i].Politics<<'\t'<<grade15[i].English<<'\t'<<grade15[i].programming<<'\t'<<grade15[i].compute<<'\t'<<grade15[i].csystem<<'\t'<<'\t'<<grade15[i].data<<'\t'<<grade15[i].opsystem<<'\t'<<grade15[i].network<<'\t'<<'\t'<<grade15[i].database<<'\t'<<'\t'<<grade15[i].compile<<'\t'<<grade15[i].datamining<<'\t'<<grade15[i].ai<<'\t'<<grade15[i].softeng<<'\t'<<grade15[i].security<<'\t'<<grade15[i].multimedia<<endl;

}

out18.close();

Rank(grade14,n1);

ofstream out19("14级排名.txt");

if(!out19)

{ cerr<<"打开文件14级排名.txt失败！"<<endl; exit(0); }

out19<<"学号"<<'\t'<<'\t'<<"姓名"<<'\t'<<"学分绩"<<'\t'<<"学分绩排名"<<endl;

for(i=0; i<n1; i++){

out19<<grade14[i].number<<'\t'<<grade14[i].name<<'\t'<<fixed<<setprecision(2)<<grade14[i].credit<<'\t'<<grade14[i].ranking<<endl;

}

out19.close();

Rank(grade15,n2);

ofstream out20("15级排名.txt");

if(!out20)

{ cerr<<"打开文件15级排名.txt失败！"<<endl; exit(0); }

out20<<"学号"<<'\t'<<'\t'<<"姓名"<<'\t'<<"学分绩"<<'\t'<<"学分绩排名"<<endl;

for(i=0; i<n2; i++){

out20<<grade15[i].number<<'\t'<<grade15[i].name<<'\t'<<fixed<<setprecision(2)<<grade15[i].credit<<'\t'<<grade15[i].ranking<<endl;

}

out20.close();

string s,ss;

string s1="姓名"; string s2="学号"; string s3="要";

for( ; ; ){

cout<<"你可以查询成绩单中的信息！"<<endl;

cout<<"是要通过姓名还是学号查询？（请输入“姓名”或“学号”）"<<endl;

cin>>s;

if(s==s1){

x=0;

cout<<"请输入姓名："<<endl;

cin>>ss;

for(i=0; i<n1; i++){

if(grade14[i].name==ss) {output(grade14[i]);x=1;}

}

for(j=0; j<n2; j++){

if(grade15[j].name==ss) {output(grade15[j]);x=1;}

}

if(x==0) cout<<"你要查找的姓名不存在！"<<endl;

}

else if(s==s2){

x=0;

cout<<"请输入学号："<<endl;

cin>>ss;

for(i=0; i<n1; i++){

if(grade14[i].number==ss) {output(grade14[i]); x=1;}

}

for(j=0; j<n2; j++){

if(grade15[j].number==ss) {output(grade15[j]); x=1;}

}

if(x==0) cout<<"你要查找的学号不存在！"<<endl;

}

else{

cout<<"输入错误，请重新输入！"<<endl; continue;

}

cout<<"还要继续查找吗?如果要，请输入“要”"<<endl;

cin>>s;

if(s==s3)

;

else break;

}

for( ; ; ){

cout<<"可以查询某一年级，某门公共课或必修课没有成绩的学生的名单！"<<endl;

cout<<"要查询哪一个年级？（请输入14或15）"<<endl;

cin>>y;

if(y==14) Searchnull(grade14,n1);

else if(y==15) Searchnull(grade15,n2);

else{

cout<<"输入错误，请重新输入！"<<endl; continue;

}

cout<<"还要继续查找吗?如果要，请输入“要”"<<endl;

cin>>s;

if(s==s3)

;

else break;

}

Countsum(grade14,n1);

Countsum(grade15,n2);

for( ; ; ){

cout<<"是否要查询总学分不够27的学生名单？(请输入“要”或“不要”)"<<endl;

cin>>s;

if(s==s3){

Search(grade14,n1); Search(grade15,n2);

}

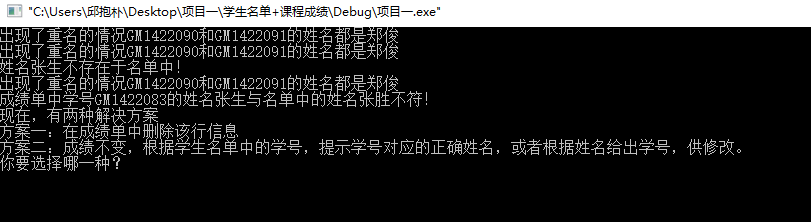
else break;

}

return 0;

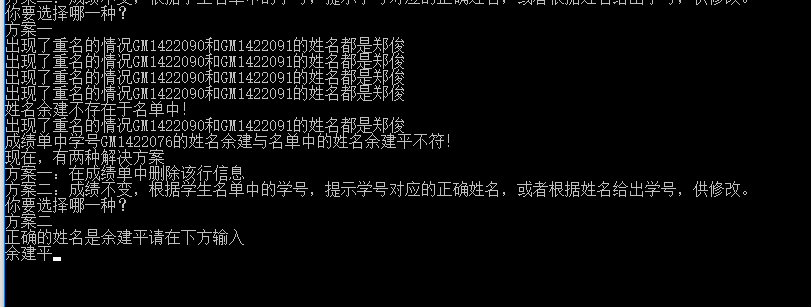
}

**五、程序调试过程**

运行程序

查询过程中，出现了姓名相同，学号不同的情况，经验证，发现是出现了重名，所以输出“出现了重名的情况”。

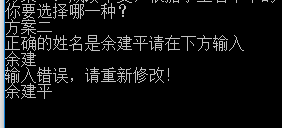
而出现学号相同，姓名不同的情况时，就出现了上图的情况



选择方案一，自动删除信息。

之后在其他的成绩单中也发现了重名的问题，输出提示。

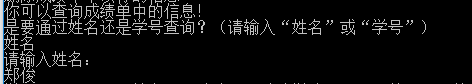
如果选择方案二，就提示正确姓名，在下方输入



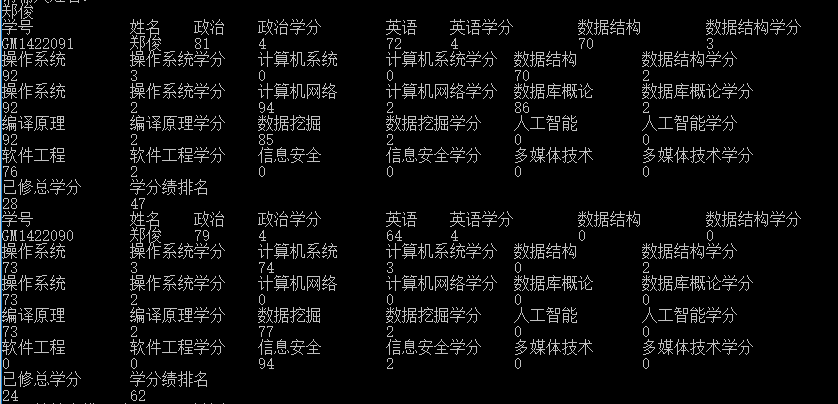
输入错误时，提示重新修改。



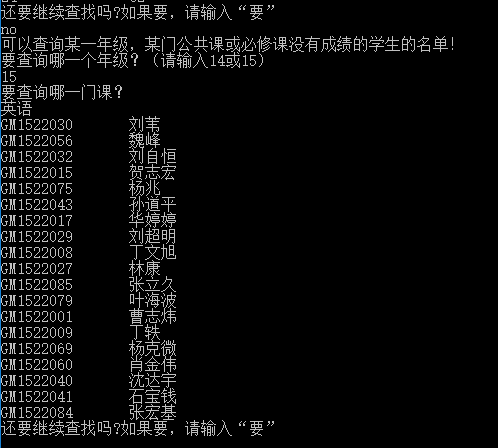
学号不存在时，也输出错误提示并删除



查询成绩信息，可以按照姓名查询，这里我们输入刚才重名的同学的名字

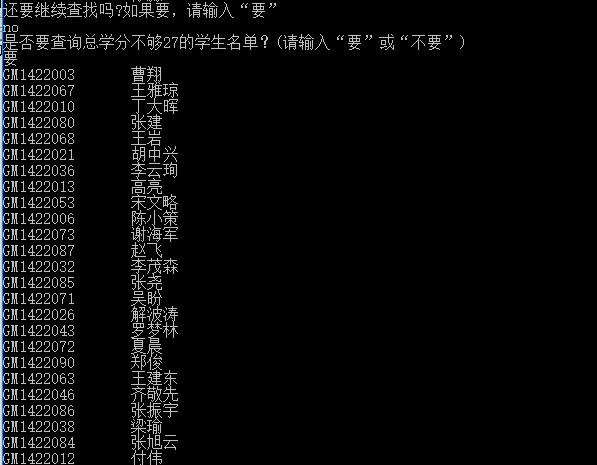


两人的信息都输出了



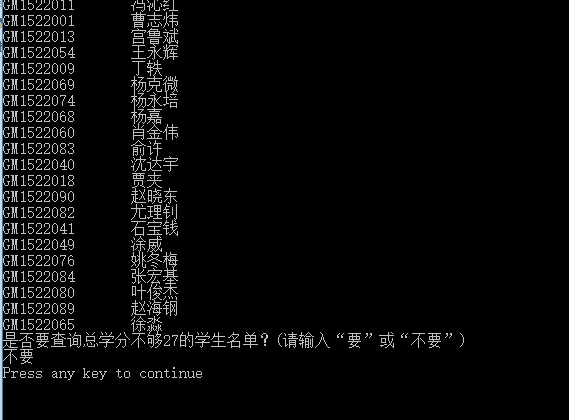
输入no结束这一块的查找

之后查询15级英语课没有成绩的学生名单，输出结果如上。

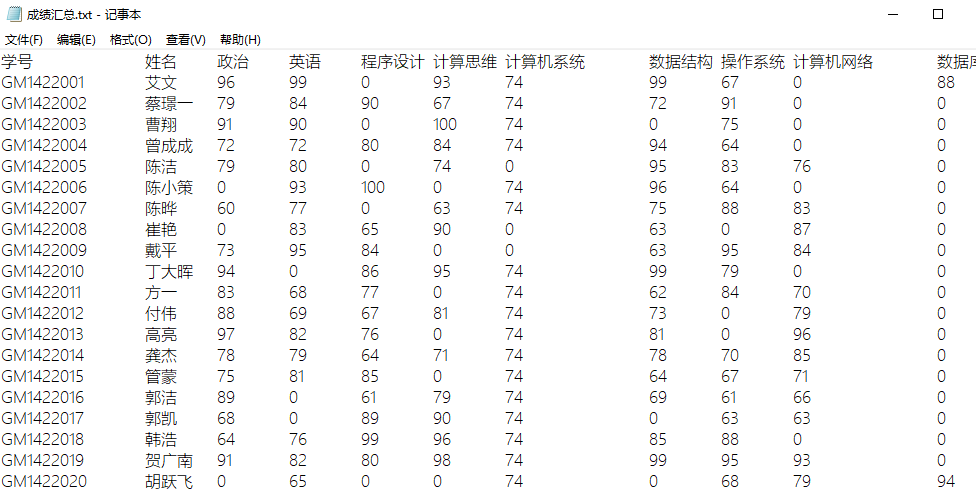


输入no结束这一块的查找。

之后查询学分不够27的学生名单，输出这个名单



输入“不要”结束查找，退出程序



成绩汇总文件





两个年级的排名文件

**六、思考与改进**

注意到这个程序代码比较长，达到了1000行。那么有没有可以简化代码的方法？

我认为改进方向大体有两点。

1.程序中打开了20多个文件，因为每个文件名字不同，所以没法套用统一的函数，有没有解决的办法？

2.注意到Output函数与Countsum函数中有重复的片段，可否设计一个处理学分的函数，在函数中计算出总学分并赋值给struct inf t.sum，然后再输出一个包含各课程修得学分数的int型数组？