# C++实验五

**李嘉康**

### 题目1 图形类设计

主函数代码：

#include <iostream>

#include<cmath>

#include"Shape.h"

#define pi 3.14

using namespace std;

int main() {

//圆的部分测试：用引用来进行值的传递

bool flag;

for (int i = 1; i <= 3; i++) {

cout << "请输入圆的坐标和半径：" << endl;

float x1, y1, r1;

cin >> x1 >> y1 >> r1;

if (r1 < 0)

{

cout << "invalid input!" << endl;

continue;

}

Circle c1(x1, y1, r1);

c1.Show();

}

//矩形部分的测试：

for (int i = 1; i <= 3; i++) {

cout << "请输入矩形的宽和长：" << endl;

float wide1, len1;

cin >> wide1 >> len1;

if (wide1 <= 0 || len1 <= 0)

{

cout << "invalid input!" << endl;

continue;

}

Rectangle r2(wide1, len1);

r2.Show();

}

//三角形的测试：

for (int i = 1; i <= 3; i++) {

cout << "请输入三角形顶点的三个坐标：" << endl;

float coordinate[7];

Triangle t1;

calculation(coordinate, t1);

t1.Show();

}

return 0;

}

头文件

#pragma once

class Circle {

public:

//~Circle();

//Circle() { }//missing initializing

Circle() { //non-argumental initializing

x = 0;

y = 0;

r = 0;

}

Circle(float x0) {//partly initializing

x = x0;

y = 1;

r = 1;

}

Circle(float x0, float y0, float r0)//standard initializing

{

x = x0;

y = y0;

r = r0;

}

float GetArea();

void Show();

void Set(float x0, float y0, float r0);

private:

float x;

float y;

float r;

};

class Rectangle {

public:

//~Rectangle();//析构

//Rectangle();//无操作

Rectangle() {//无参数初始化

wide = 0;

len = 0;

}

Rectangle(float wide0) {//部分参数初始化

wide = wide0;

len = 0;

}

Rectangle(float wide0, float len0) {//满参数初始化

wide = wide0;

len = len0;

}

float GetArea();//求面积

void Show();//展示基本信息

void Set(float wide0, float len0);//修改参数

private:

float wide;

float len;

};

class Triangle {

public:

//~Triangle();

//Triangle();//无操作

Triangle() {//无参数

a = 0;

b = 0;

c = 0;

}

Triangle(float a0) {//部分参数

a = a0;

b = 0;

}

Triangle(float a0, float b0, float c0) {//无参数

a = a0;

b = b0;

c = c0;

}

float GetArea();

void Show();

void Set(float A, float B, float C) {

a = A;

b = B;

c = C;

}

private:

float a;

float b;

float c;

};

void calculation(float\* arr, Triangle& t);

头文件.cpp

#include<iostream>

#include "Shape.h"

using namespace std;

#define pi 3.14

float Circle::GetArea() {

return pi \* r \* r;

}

void Circle::Show() {

cout << "圆心坐标：" << x << "," << y << endl;

cout << "直径:" << 2 \* r << endl;

cout << "周长:" << 2 \* pi \* r << endl;

cout << "面积:" << GetArea() << endl;

}

void Circle::Set(float x0, float y0, float r0) {

x = x0;

y = y0;

r = r0;

cout << "Now it's fully updated as below:" << endl;

Show();

}

float Rectangle::GetArea() {

return wide \* len;

}

void Rectangle::Show() {

cout << "长，宽：" << len << "," << wide << endl;

cout << "周长:" << 2 \* (wide + len) << endl;

cout << "面积:" << GetArea() << endl;

}

void Rectangle::Set(float x0, float y0) {

wide = x0;

len = y0;

cout << "Now it's fully updated as below:" << endl;

Show();

}

float Triangle::GetArea() {

float temp = (a + b + c) / 2;

return sqrt(temp \* (temp - a) \* (temp - b) \* (temp - c));//用公式

}

void Triangle::Show() {

cout << "三边长：" << a << b << c << endl;

cout << "周长:" << a + b + c << endl;

cout << "面积:" << GetArea() << endl;

}

void calculation(float\* arr, Triangle& t) {//形参为引用，此函数的目的是将坐标转换为边，并判断输入是否合理

for (int i = 0; i < 6; i++)

cin >> arr[i];//一次性输入坐标，先输x再输y

float A, B, C;//三条边的长度。

A = sqrt(pow((arr[0] - arr[2]), 2) + pow((arr[1] - arr[3]), 2));

B = sqrt(pow((arr[0] - arr[4]), 2) + pow((arr[1] - arr[5]), 2));

C = sqrt(pow((arr[2] - arr[4]), 2) + pow((arr[3] - arr[5]), 2));

bool flag = true;

if (A + B <= C)flag = 0;//依次判断三条边是否符合题意

if (A + C <= B)flag = 0;

if (B + C <= A)flag = 0;

if (flag == true) {

t.Set(A, B, C);

}

else {

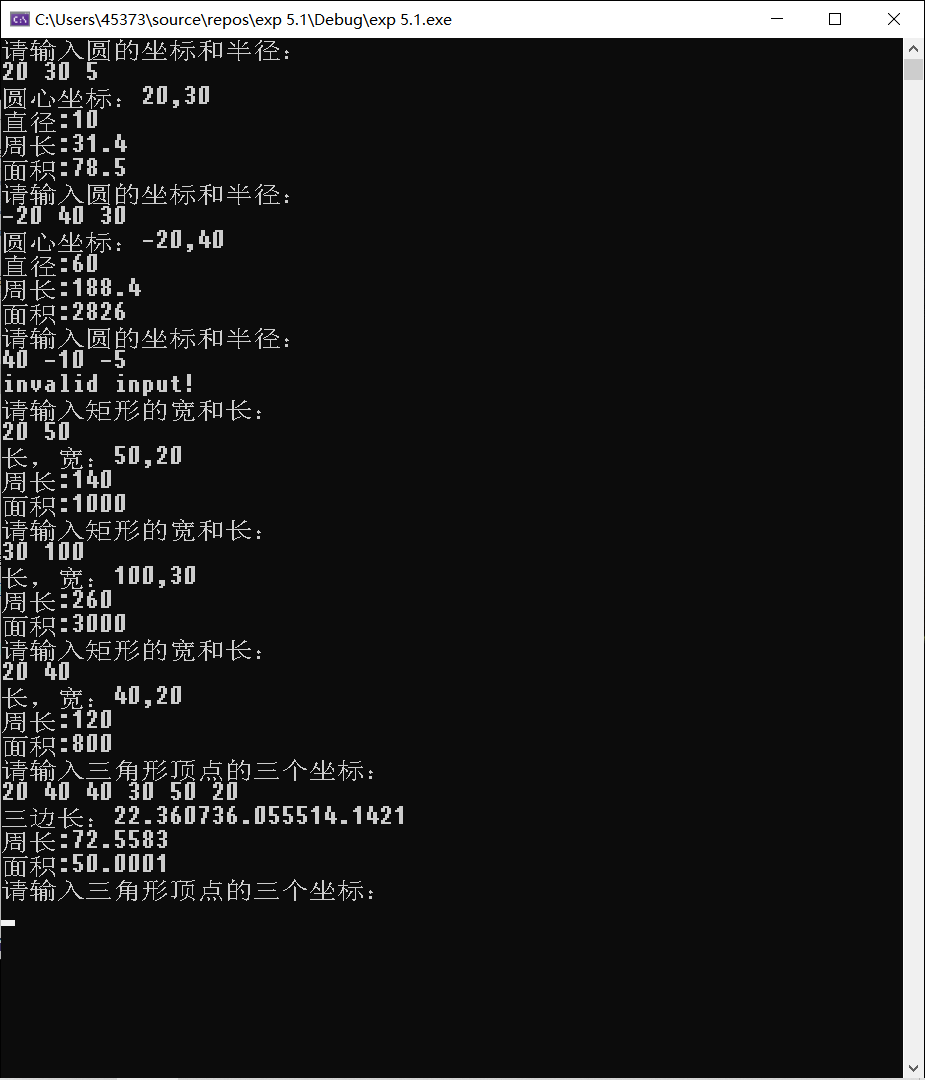
cout << "输入的坐标无法构成三角形，请重新输入坐标:";

calculation(arr, t);

}

}

运行截图：



**题目2 学生类**

主函数：

#include "Student.h"

#include <iostream>

#include <string>

using namespace std;

int main() {

cout << "请输入你想输入的学生的个数" << endl;

int n;

cin >> n;

cout << endl;

student stu[50];

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

cout << "请输入第" << i + 1 << "个学生的学号、姓名、性别、出生日期，年级、班级、院系、专业；";

string stuno2;

string name2;

string sex2;

string birthdate2;//please input something like this 20010527

string grade2;

string class\_2;

string department2;

string major2;

cout << "Please input valid data,for sex it could only be 女性 or 男性,birthdate for eg be “85年07月”,grade be 05级，class be jy01，department be computer，major be application" << endl;

cin >> stuno2 >> name2 >> sex2 >> birthdate2 >> grade2 >> class\_2 >> department2 >> major2;

if (sex2 != "女" && sex2 != "男") {//这里不好限制，故仅仅做了这一个限制

cout << "invalid attempt,Please try again for student" << i + 1 << endl;

cin >> stuno2 >> name2 >> sex2 >> birthdate2 >> grade2 >> class\_2 >> department2 >> major2;

}

stu[i].SetInfo(stuno2, name2, sex2, birthdate2, grade2, class\_2, department2, major2);

cout << endl;

}

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

cout << "第" << i + 1 << "个学生的学号、姓名、性别、出生日期 年级、班级、院系、专业；";

stu[i].Show();

cout << endl;

}

int x = 1;

cout << "please input the order number of the student for updating and input ctrl + Z to stop" << endl;

while (cin >> n) {

string stuno2;

string name2;

string sex2;

string birthdate2;//please input something like this 20010527

string grade2;

string class\_2;

string department2;

string major2;

cout << "Please input valid data, for sex it could only be 女性 or 男性, birthdate for eg be “85年07月”, grade be 05级，class be jy01，department be computer，major be application" << endl;

cin >> stuno2 >> name2 >> sex2 >> birthdate2 >> grade2 >> class\_2 >> department2 >> major2;

if (sex2 != "女" && sex2 != "男") {

cout << "invalid attempt,Please try again for student" << n << endl;

cin >> stuno2 >> name2 >> sex2 >> birthdate2 >> grade2 >> class\_2 >> department2 >> major2;

}

stu[x - 1].SetInfo(stuno2, name2, sex2, birthdate2, grade2, class\_2, department2, major2);

cout << endl;

cout << "updated!The information is updated as:" << endl;

stu[x - 1].Show();

cout << "please input the order of the student for updating and input ctrl + Z to stop" << endl;

}

return 0;

}

头文件.h

#pragma once

#include<string>

#include<string.h>

#include <iostream>

using namespace std;

class student {

public:

student();

student(string stuno1, string name1);

//~Student();

student(string stuno1, string name1, string sex1, string birthdate1, int grade1, string class\_1, string department1, string major1);//带完整参数的构造函数

void SetInfo(string stuno1, string name1, string sex1, string birthdate1, string grade1, string class\_1, string department1, string major1);

void Show();

private:

string stuno;

string name;

string sex;

string birthdate;//please input something like this 20010527

string grade;

string class\_;

string department;

string major;

};

头文件.cpp

#include "Student.h"

#include <iostream>

using namespace std;

student::student() {

stuno = "320190941301";

}//默认

student::student(string stuno1, string name1) {//部分

stuno = stuno1;

name = name1;

}

//~Student();

student::student(string stuno1, string name1, string sex1, string birthdate1, int grade1, string class\_1, string department1, string major1) {//带参数

stuno = stuno1;

name = name1;

sex = sex1;

birthdate = birthdate1;

grade = grade1;

class\_ = class\_1;

department = department1;

major = major1;

}//带完整参数的构造函数

void student::SetInfo(string stuno1, string name1, string sex1, string birthdate1, string grade1, string class\_1, string department1, string major1) {//带参数

stuno = stuno1;

name = name1;

sex = sex1;

birthdate = birthdate1;

grade = grade1;

class\_ = class\_1;

department = department1;

major = major1;

}

void student::Show() {

cout << endl;

cout << "stu number:" << stuno << endl;

cout << "stu name:" << name << endl;

cout << "stu sex:" << sex << endl;

cout << "stu birthdate:" << birthdate << endl;

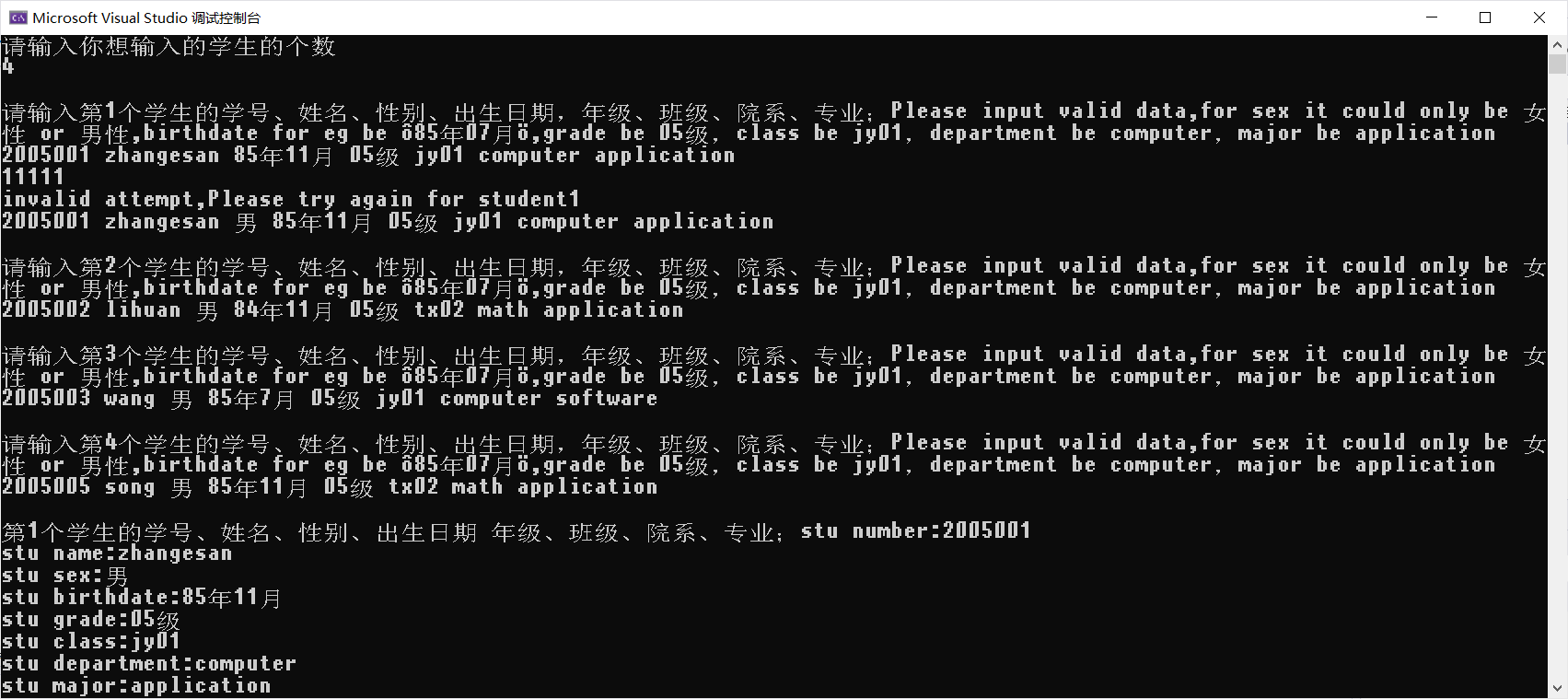
cout << "stu grade:" << grade << endl;

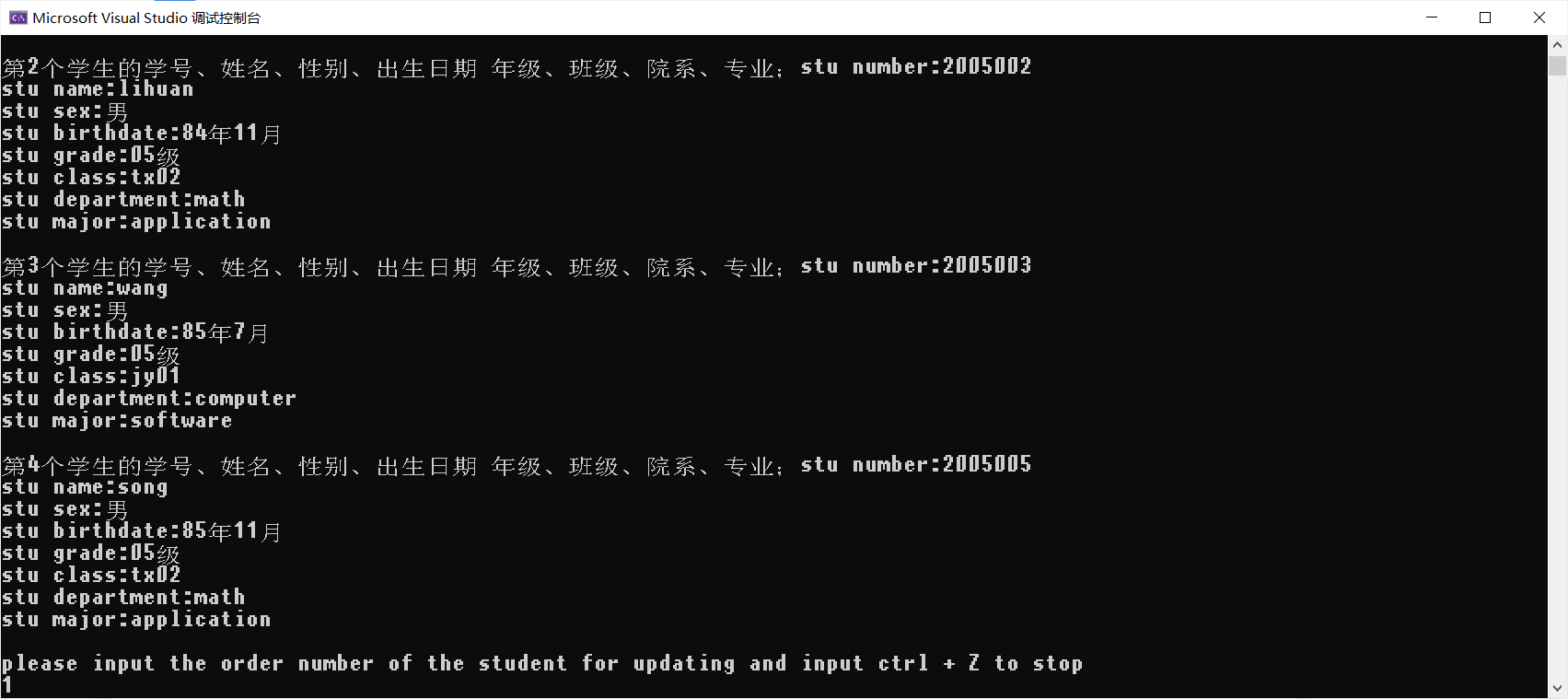
cout << "stu class:" << class\_ << endl;

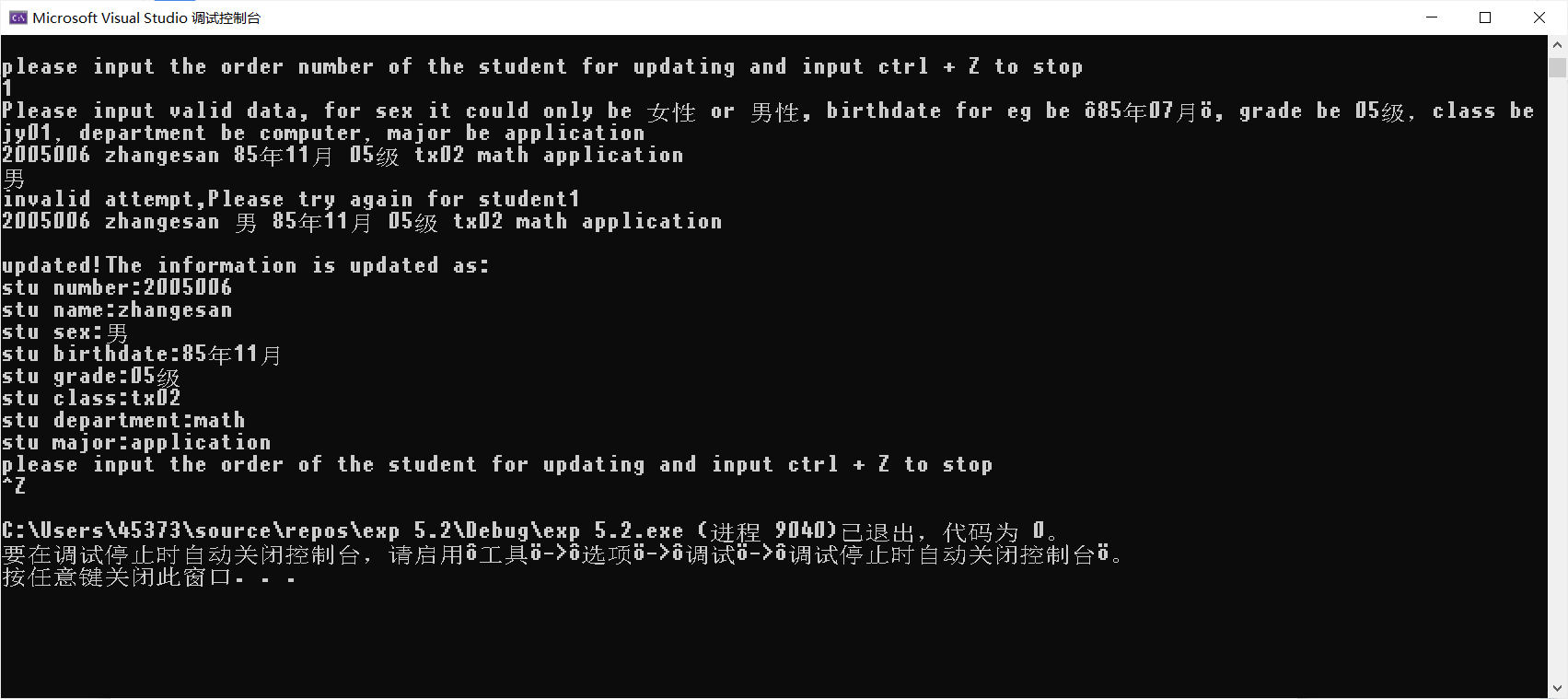
cout << "stu department:" << department << endl;

cout << "stu major:" << major << endl;

}







输入数据为报告所要求，不合理的数据输入也被要求重新输入，并且在输入后更改了数值。

**[**思考与扩展**]**

在SetInfo里加一个i表示是第几个学生输入次序，自动赋值学号：”200500” +”i”即可

**题目3 人员管理**

主函数

#include<iostream>

#include<string>

#include"employee.h"

using namespace std;

/\*employee::employee() {

num = 1000;

name = "xiaoming";

sex = 'f';

Date birthday(1900, 1, 1);

position = "worker";

}\*/

int main() {

cout << "请输入你想输入的员工的个数" << endl;

int n;

cin >> n;

cout << endl;

//初始化：

employee emp[80];

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

cout << "请输入第" << i + 1 << "个员工的编号、姓名、性别、出生年、月、日、职位"<<endl;

string num1, name1, sex1, position1;

int year, month, day;

cin >> num1 >> name1 >> sex1 >> year >> month >> day>>position1;

emp[i].SetInfo(num1, name1, sex1,year ,month,day, position1);

cout << endl;

}

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

emp[i].Show();

cout << endl;

}

//修改信息：

cout << "请输入你想修改的上面员工的序号" << endl;

int x;

while (cin >> x && x) {

string num2, name2, sex2, position2;

int year, month, day;

cout << "输入其新的编号，姓名、性别、出生年、月、日、职位" << endl;

cin >>num2 >> name2 >> sex2 >> year >> month >> day>>position2;

emp[x-1].SetInfo(num2, name2, sex2,year,month,day,position2);

cout << "若您还想修改信息，请输入上面展示的员工的序号，并输入新的姓名和职位等信息，否则输入0结束" << endl;

}

cout << "---------------------------------------------------" << endl << endl;

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

emp[i].Show();

cout << endl;

}

cout << endl;

cout << "输入1用编号查询员工信息" << endl;

cout << "输入2用姓名查询员工信息" << endl;

cout << "输入其他值，结束。" << endl;

cin >> x;

switch (x) {

case 1: {

string num;

cout << "请输入你要找的员工的编号：";

cin >> num;

emp[0].SearchNum(num, emp, n);

break;

}

case 2: {

string name;

cout << "请输入你要找的员工的姓名：";

cin >> name;

emp[0].SearchName(name, emp, n);

break;

}

}

return 0;

}

头文件.h

#pragma once

#include<string>

#include<string.h>

#include <iostream>

using namespace std;

class Date {

public:

/\*Date(int y = 1900, int m = 1, int d = 1) {//带默认参数的构造函数

year = y;

month = m;

day = d;

}\*/

Date();

Date(int y0, int m0, int d0);

//Date();//缺失构造函数

friend class employee;

void Show();

void SetYear(int y) {

year = y;

}

void SetMonth(int m) {

month = m;

}

void SetDay(int d) {

day = d;

}

int GetYear() {

return year;

}

int GetMonth() {

return month;

}

int GetDay() {

return day;

}

private:

int year;

int month;

int day;

};

class employee {

public:

employee();//缺省构造函数

employee(string n, string nam, char s, Date bir, string pos);//带参数构造函数，这里Date是引用&

//employee(int ,string ,char,Data&,string="worker");

void SetInfo(string n, string nam, string s, int year, int month, int day, string pos);

friend class Date;

void Show();

string GetNum() { return num; }

string GetName() { return name; }

void SearchNum(string, employee\*, int);

void SearchName(string, employee\*, int);

private:

string num;

string name;

string sex;

Date birthday;

string position;

};

头文件.cpp

#include "employee.h"

#include <iostream>

using namespace std;

employee::employee() {

num = "2001";

}//缺省构造函数

employee::employee(string n, string nam, char s, Date bir, string pos) {

num = n;

name = nam;

sex = s;

Date birthay = bir;

position = pos;

}//带参数构造函数，这里Date是引用&

//employee(int ,string ,char,Data&,string="worker");

void employee::SetInfo(string n, string nam, string s, int year, int month, int day, string pos) {

num = n;

name = nam;

sex = s;

birthday.SetDay(day);

birthday.SetMonth(month);

birthday.SetYear(year);

position = pos;

}void employee::Show() {

cout << "number,name,sex,birthday,position依次为:" << endl;

cout << "number:" << num << endl;

cout << "name:" << name << endl;

cout << "sex:" << sex << endl;

birthday.Show();

cout << "position:" << position << endl;

}

void employee::SearchNum(string num, employee\* p, int n) {

bool flag = 1;

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

if (num == p[i].GetNum()) {

cout << "所找的员工的信息为：" << endl;

p[i].Show();

flag = 0;

break;

}

}

if (flag)cout << "没有编号为" << num << "的员工，请确认编号输入是否正确！" << endl;

}

void employee::SearchName(string name, employee\* p, int n) {

bool flag = 1;

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

if (name == p[i].GetName()) {

cout << "所找的员工的信息为：" << endl;

p[i].Show();

flag = 0;

break;

}

}

if (flag)cout << "没有姓名为" << name << "的员工，请确认姓名输入是否正确！" << endl;

}

/\*Date(int y = 1900, int m = 1, int d = 1) {//带默认参数的构造函数

year = y;

month = m;

day = d;

}\*/

Date::Date() {

year = 1;

month = 1;

day = 1;

}

Date::Date(int y0, int m0, int d0) {//全参数

year = y0;

month = m0;

day = d0;

}

//Date();//缺失构造函数

void Date::Show() {

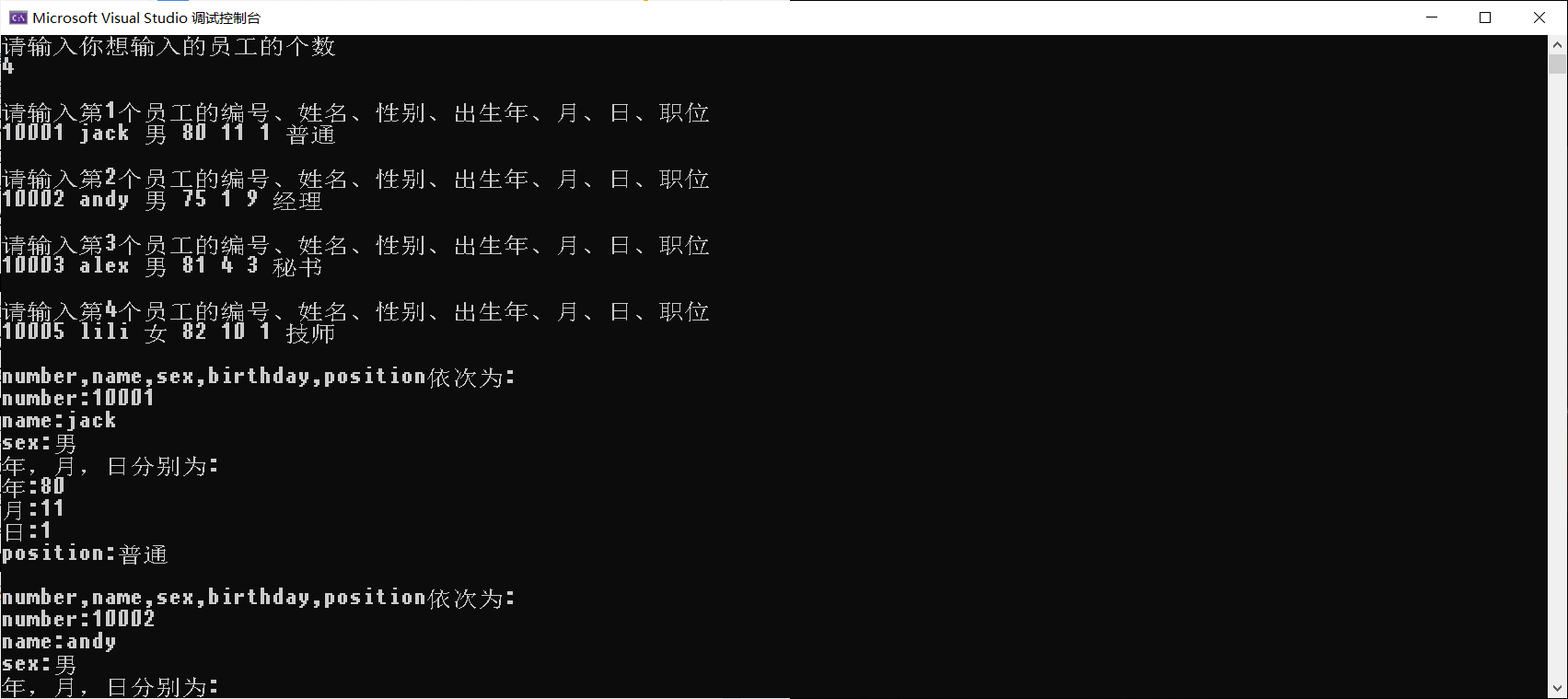
cout << "年，月，日分别为:" << endl;

cout << "年:" << year << endl;

cout << "月:" << month << endl;

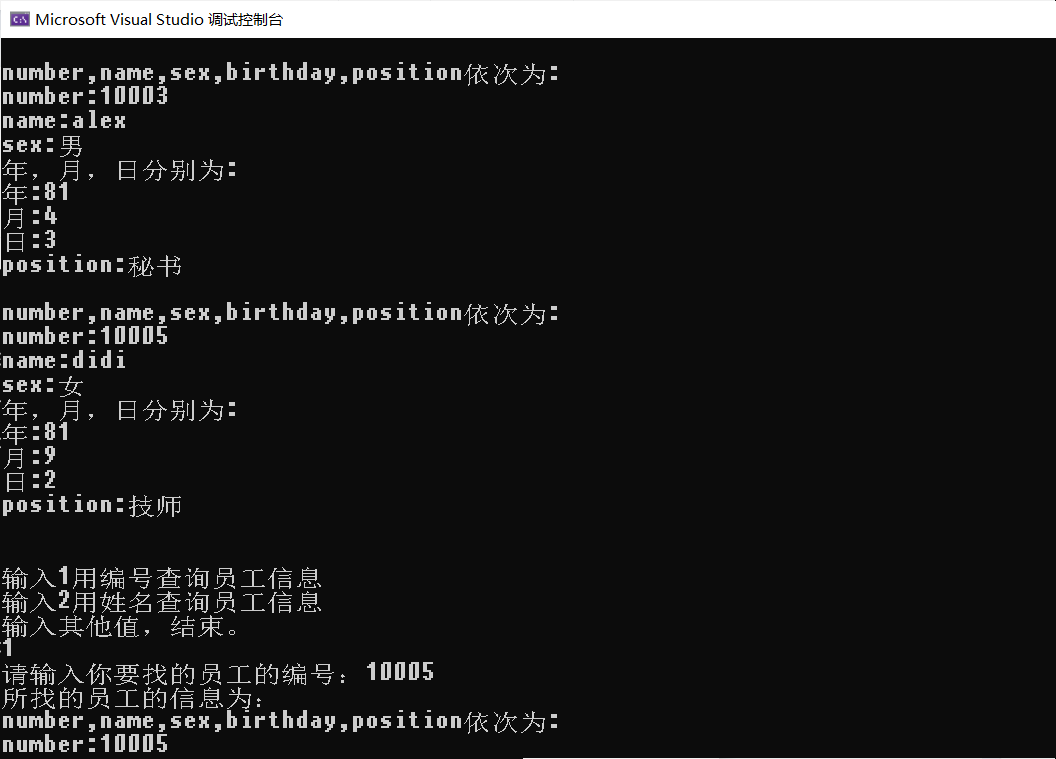
cout << "日:" << day << endl;

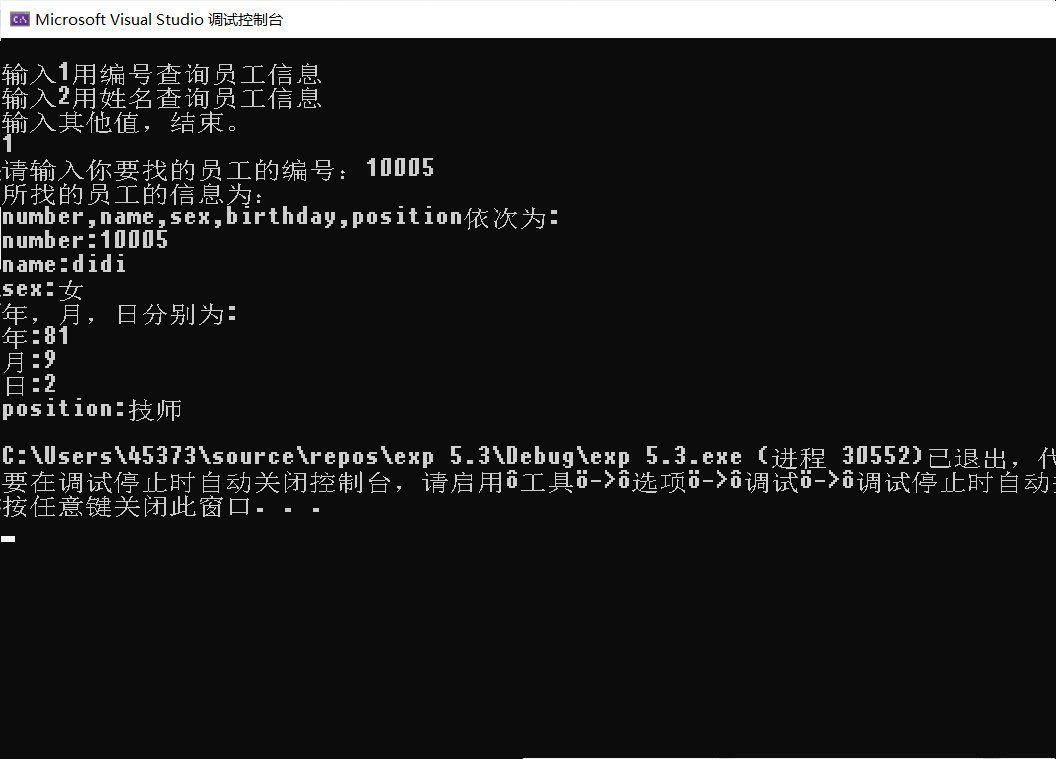
}











**[**思考与扩展**]**

如何使用链表方式来保存已输入职工对象?查询函数该如何来写？

单个链表结构体里面的变量定义为俩个类和一个结构体指针即可，输入构建和链表的构建一样。

查询函数即依次遍历整个链表来对比所查询的值与每个结构体结点里面查询的值是否相符即可