**计算机程序设计基础（C++)**

**实验报告**

专业班级： 软工2402

学 号： 8209240223

姓 名： 王仪杰

**实验报告成绩：**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **实验** | **实验一** | **实验二** | **实验三** | **实验四** | **实验五** | **总评** |
| **成绩** |  |  |  |  |  |  |

**批阅教师：**

**实验一、实验环境与简单程序设计**

**一、实验目的**

1、掌握集成开发环境，掌握C++程序的基本要素以及完整的C++程序开发过程。

2、掌握基本数据类型、运算符和表达式的使用。理解隐式转换和强制转换，理解数据超过该数据类型

表示范围时的溢出。掌握不同数据之间的混合算术运算中数据类型的转换。

3、变量的定义与常量的使用。

4、输入、输出的实现。

5、编译信息的理解与错误的修改。

6、简单程序的设计。

**二、实验内容**

熟悉C++编程环境，可以使用VS；对已经能熟练掌握C++开发环境的同学，可

以跳过本部分内容）

**1.编辑输入下列程序，找出下面代码的错误并改正：**

#include<iostream>

using namespace std;

int main()

{

Int i = k + 1;

cout << i++ << endl;

int i = 1;

cout << i++ << endl;

cout << "Welcome to C++"<<endl;

return 0;

}

**2.求圆锥的体积：要求键盘输入圆锥底的半径、锥高，使用标识符常量定义圆周率。**

(1)创建一个控制台项目

(2)在文件中输入程序内容，存盘

(3)编译、连接、运行；观察结果

**3**.**通过下面程序验证你所使用系统上运行的C++编译器中每个基本数据类型的长度。**

#include<iostream>

using namespace std;

int main()

{

cout << "char length:" << sizeof(char) << endl;

cout << "int length:" << sizeof(int) << endl;

}

**4.观察下面程序的执行结果。**

#include <iostream>

#include <iomanip>

using namespace std;

int main()

unsigned int testUnint=65534;//oxfffe

cout << "output in unsigned int 1 type:"" << testUnint<< end;//<<oct;

cout << "output in char type:!" << static\_ cast<char>(testUnint)<< endl;

cout << "output in short type:" << static\_ cast<short>(testUnint)<< endl;//为什么结果为-2?:

cout << "output in int type:" << static\_ cast<int>. (testUnint)<< endl;

cout << "output in double type:"<< static cast<double>(testUnint)<< endl;

cout << "output in double type:" <<setprecision(4)<< static\_ cast<double>(testUnint)<< endl;

cout << "output in Hex unsigned int type:" <<hex<< testUnint<< endl; //16进制输出

system("pause");

return 0;

**自己编程测试一下将testUnint按8进制输出<<oct;je\_将一个实数转换成int,观察结果。**

**5.编程，输入华氏温度，将其转换为摄氏温度后输出（保留两位小数）。**

**三、算法分析，程序结果**

1.#include<iostream>

using namespace std;

int main()

{

int k = 0;//未定义变量k

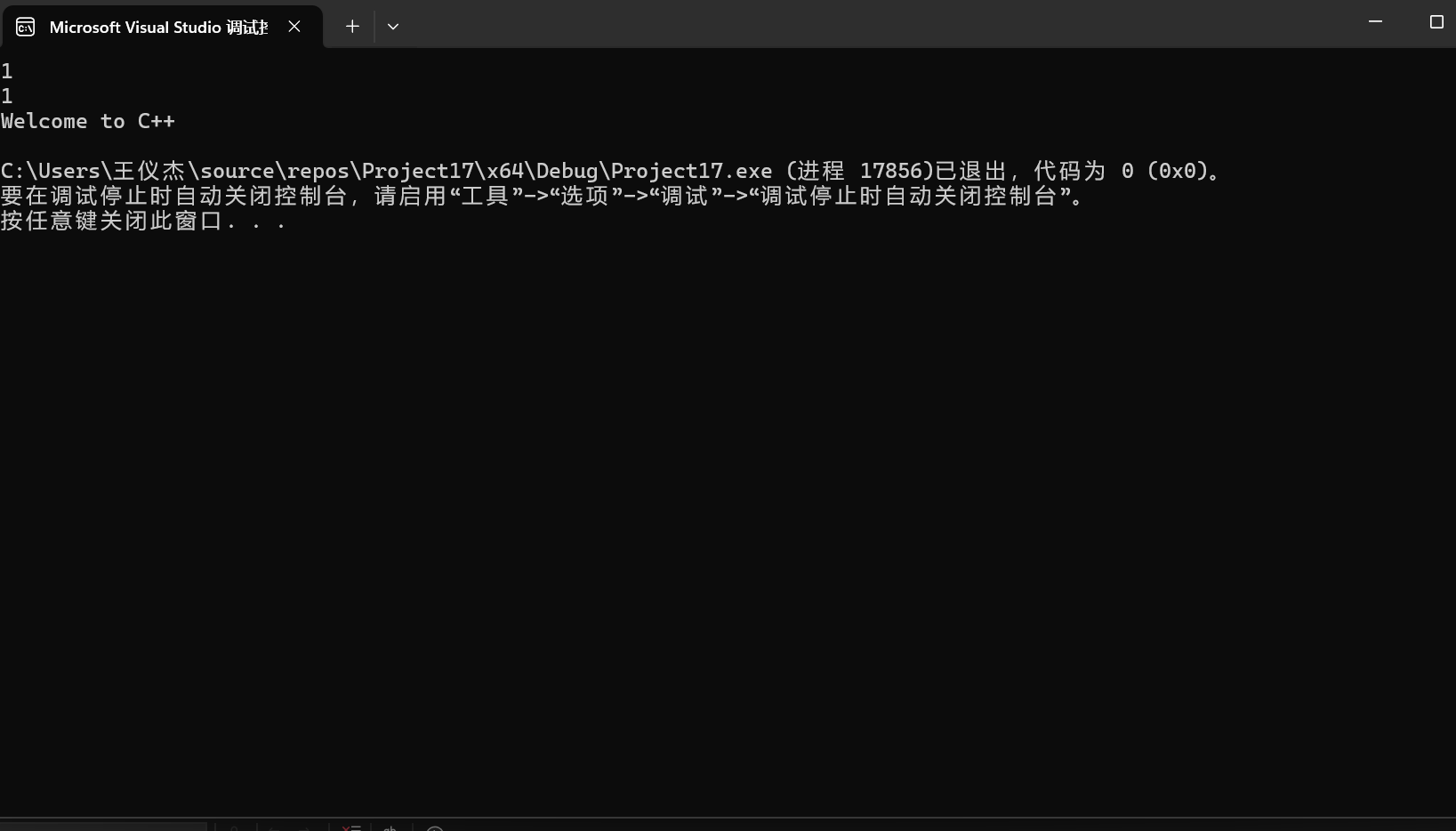
int i = k + 1;

cout << i++ << endl;

i = 1;//变量i重复定义

cout << i++ << endl;//i++表示先进行运算再给i加一，即先输出i在给i加一，因此这行代码不会输出 无意义

cout << "Welcome to C++" << endl;

return 0;

}

2.#include<iostream>

using namespace std;

int main()

{

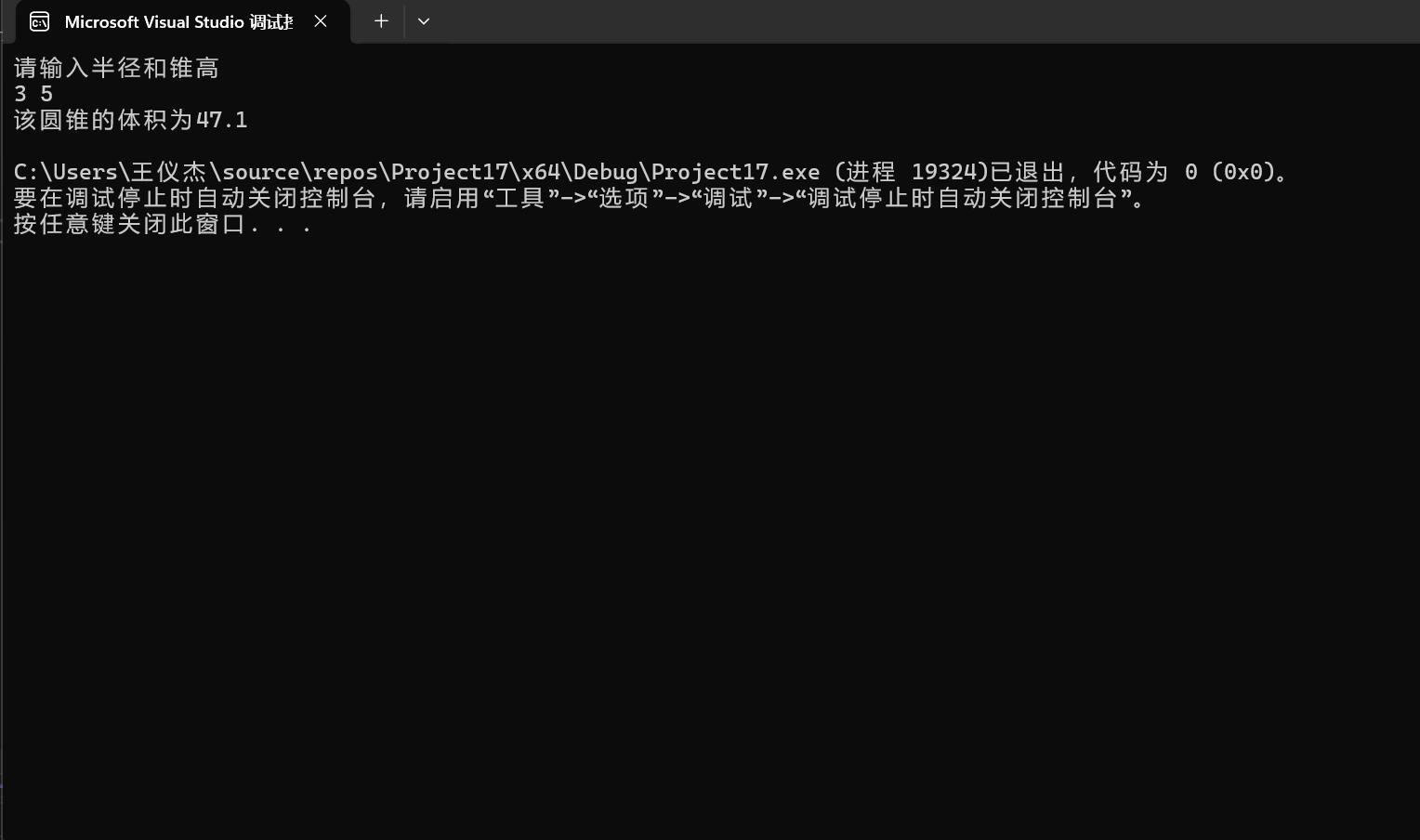
int a = 0, b = 0;

const double c = 3.14;//定义一个常量c 用define a 3.14 也可以

cout << "请输入半径和锥高" << endl;//提示输入

cin >> a >> b;

cout << "该圆锥的体积为" << c \* a \* a \* b / 3 << endl;//用公式计算圆锥体积

return0;

}

3#include<iostream>

using namespace std;

int main()

{

cout << "char length : " << sizeof(char) << endl;//输出char类型数据的长度

cout << "int length : " << sizeof(int) << endl;//输出int类型数据的长度

cout << "float length : " << sizeof(float) << endl;//输出float类型数据的长度

cout << "double length : " << sizeof(double) << endl;//输出double类型数据的长度

cout << " string length : " << sizeof(string) << endl;//输出string类型数据的长度

cout << "bool length :" << sizeof(bool) << endl;//输出bool类型数据的长度

return 0;

}



4.#include <iostream>

#include <iomanip>//使用setprecision时要用的头文件

using namespace std;

int main()

{

unsigned int testUnint = 65534;//oxfffe

cout << "output in unsigned int 1 type:" << testUnint << endl;//直接输出

cout << "output in char type:!" << static\_cast<char>(testUnint) << endl;//转变为字符型变量

cout << "output in short type:" << static\_cast<short>(testUnint) << endl;//为什么结果为-2?:因为此时65534超出了short的范围超出了32767后会从-32768（即从能表示的最小数开始）开始向上计数因此会输出-2

cout << "output in int type:" << static\_cast<int> (testUnint) << endl;//用整型输出

cout << "output in double type:" << static\_cast<double>(testUnint) << endl;//强转为double类型输出

cout << "output in double type:" << setprecision(4) << static\_cast<double>(testUnint) << endl;//不使用fix的情况下setprecision会以科学计数法的方式保留4位

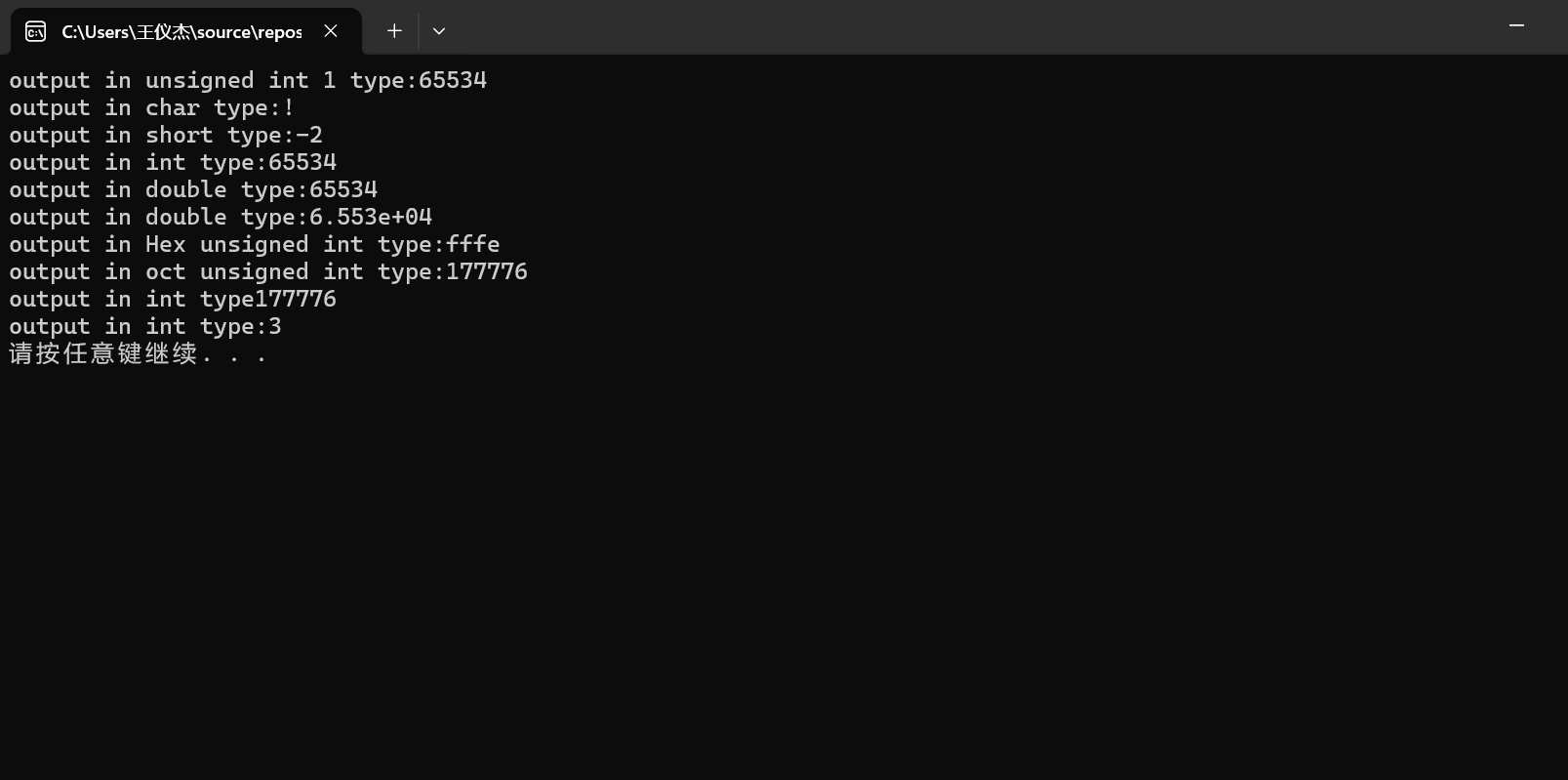
//这样写cout << "output in double type:" <<fixed<< setprecision(4) <<static\_cast<double>(testUnint) << endl;可以真正的保留小数点后四位

cout << "output in Hex unsigned int type:" << hex << testUnint << endl; //16进制输出

cout << "output in oct unsigned int type:" << oct << testUnint << endl;

cout << "output in int type" << static\_cast<int>(testUnint) << endl;

system("pause");

return 0;

}

5.#include <iostream>

#include <iomanip>

using namespace std;

int main()

{

double a = 0;

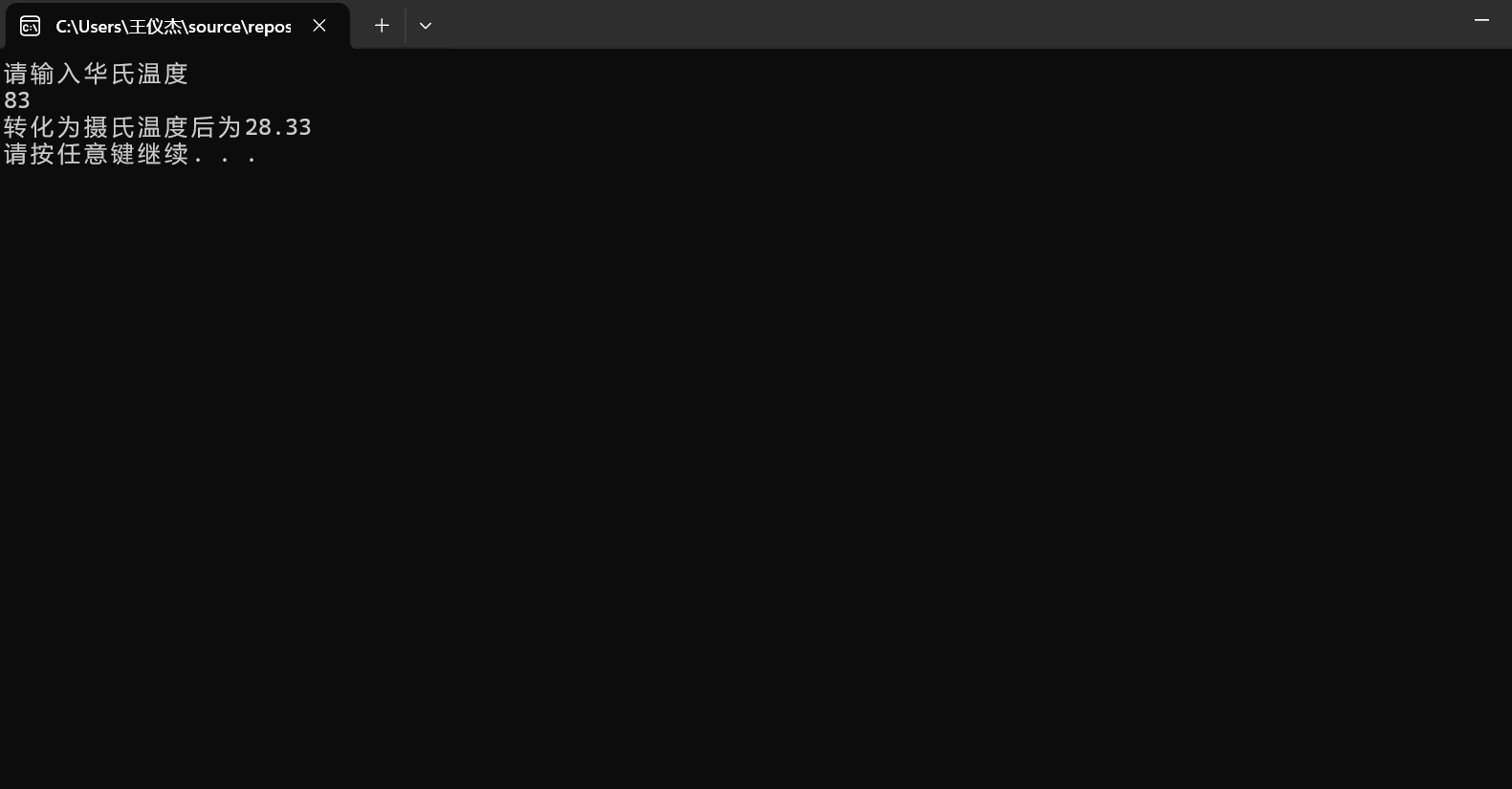
cout << "请输入华氏温度" << endl;

cin >> a;//输入这个华氏温度

cout << "转化为摄氏温度后为" << fixed << setprecision(2) << 5 \* (a - 32) / 9 << endl;//一定使用fixed进行保留两位小数 只用setprecision只能用科学计数法表示

system("pause");

return 0;



}

#### 遇到的问题与解决方法

**一 2.**注意用define a 3.14定义时不用加等号

一 4.精确小数点后位数的函数setprecision:

不使用fix的情况下setprecision会以科学计数法的方式保留4位如3.1415e3代表3141.5

这样写cout << "output in double type:" <<fixed<< setprecision(4) << static\_cast<double>(testUnint) << endl;可以真正的保留小数点后四位

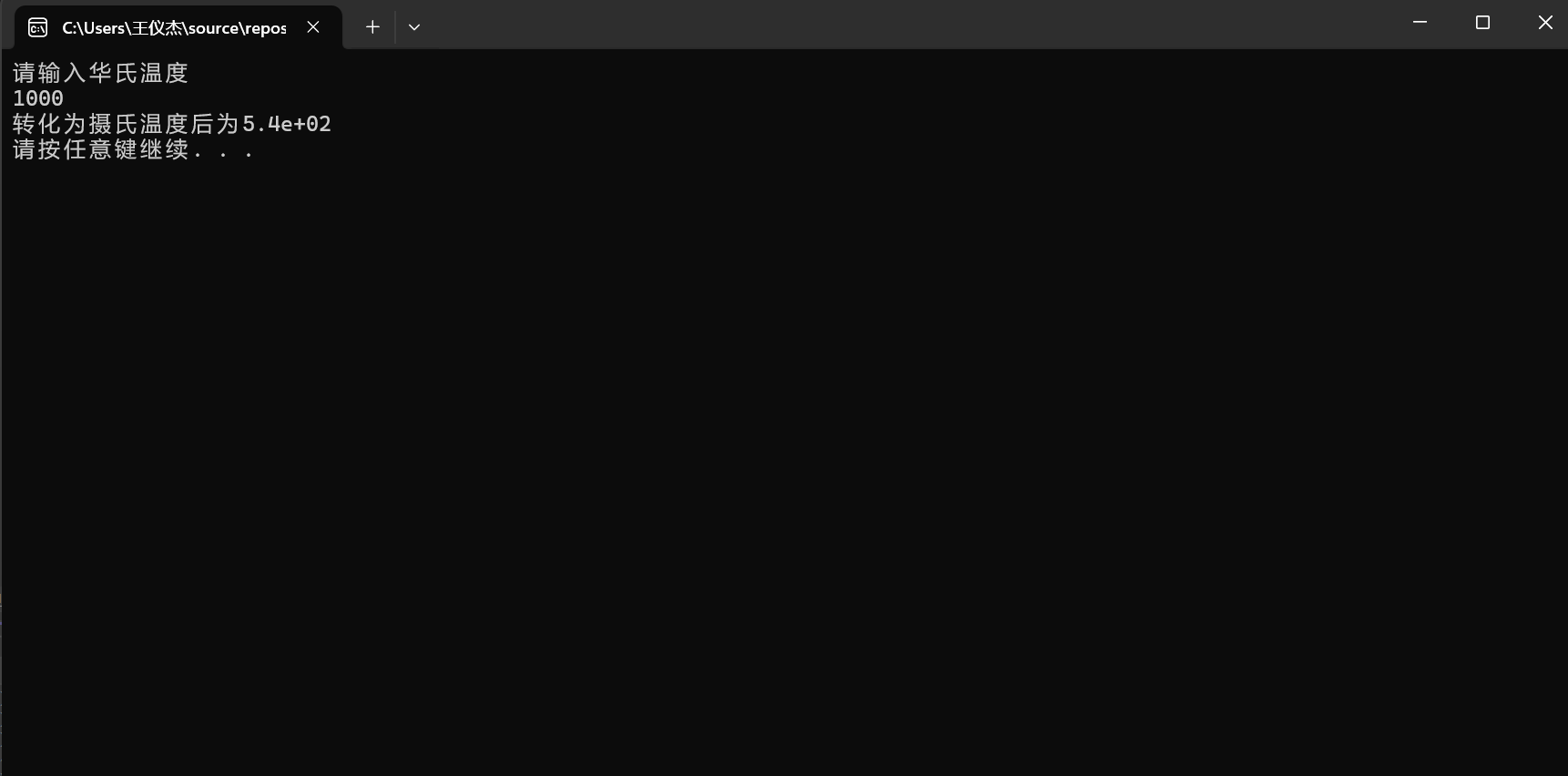
1. 在用字符型变量输出时会输出一个！

屏幕截图 2024-11-12 125603

上网查询可知不同的编译器会对超出char范围的值做特定的处理，因此会输出一个！

#### 一 5.

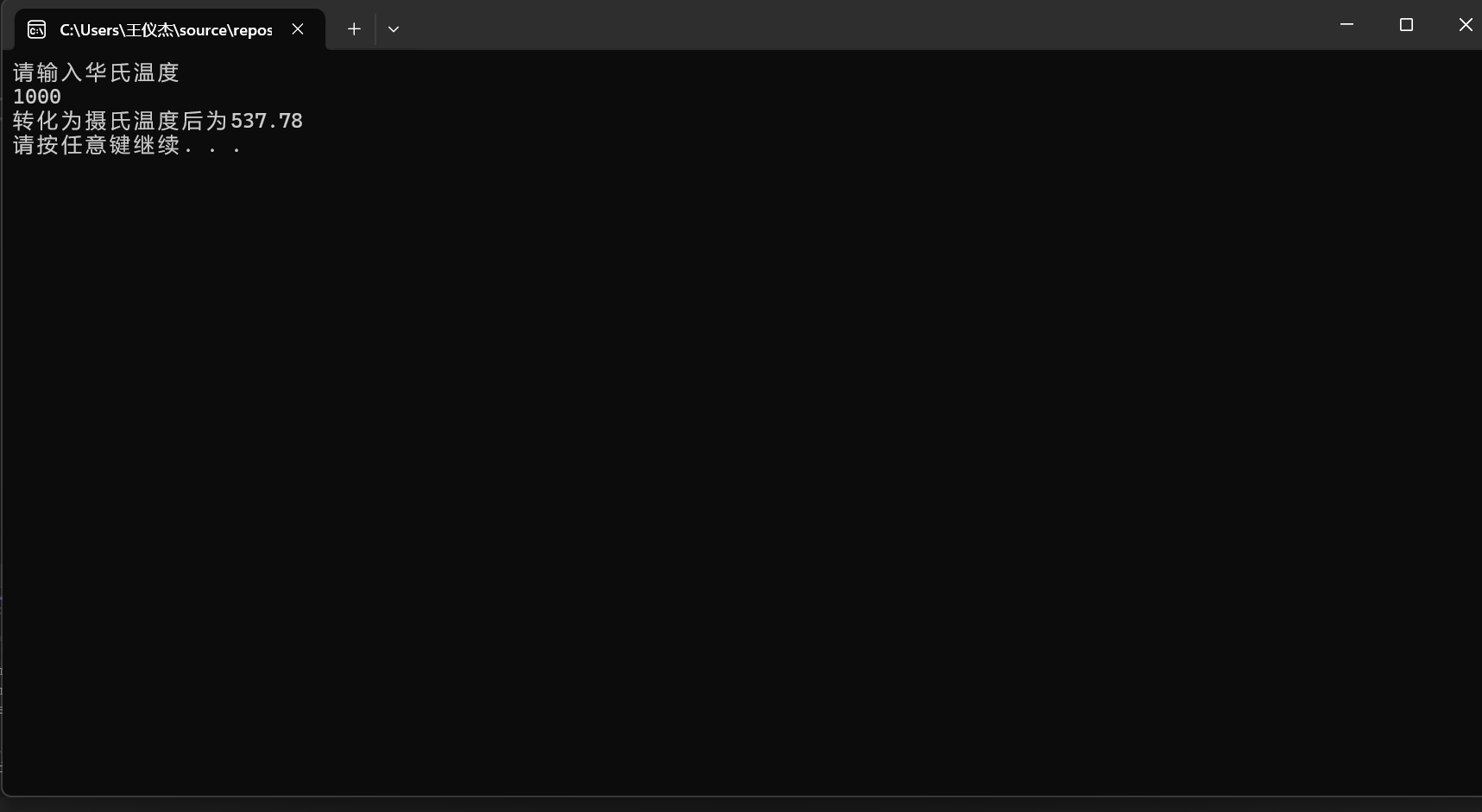
最开始直接使用setprecision(2)在输出时会出现以下情况，经过资料查询可知

****

这样并不是真正的转化为了保留两位小数，而只是运用了科学计数法保留了小数点后两位，查询可知，应该是使用fixed来固定小数点后的位数代码如图：

屏幕截图 2024-11-09 100028

最终输出结果为

****

1. **体会**
2. **在进行实践的过程中应该更加注重细节**

比如在一开始我并未注意到1.1中int i重复定义的问题 ，而在编译器报错时才发现。可能在日常的编程中，自己比较少遇到这一类错误，所以无法立即发现题中的这一点问题。1.2中运用default 时加了=号。并且经常习惯性将之定义为整形int，以后在设置变量时应该注意题目中的要求。而不能根据平时的习惯。我希望自己能够利用好这次实验的机会增强动手能力，并在平时的练习中更加注重细节问题。老师在介绍运算原理时，解释了很多相关原理及方式，但是感觉自己对于这方面还没有完全掌握。希望这一次经历能够加深我对这一类知识点的理解。

1. **在自学的过程中不应只一味追求速度，更要经常复习**

在学习的过程中经常会发生遗忘的现象，比如在实验时忘记了类型强制转换以及进制转换的语法该如何使用。只能参考原代码中给出的方式才能成功实现进制转换。

以及忘记了精确到小数点后几位时的函数setprecision，和他该使用那个头文件

#include <iomanip>//使用setprecision时要用的头文件

只能通过上网查询的方式才能确定使用那种头文件

对于这种情况应该在平时学习的过程中及时对知识点进行复习，也更加需要利用这次实验的机会复习一些已经忘记的知识。

1. **对一些语句的格式并不清楚**

比如在进行类型强制转换时，知道要用到static但对具体的语句怎么写会出现遗忘的情况，对于这种问题应该在平时多加练习，把语句具体怎么写弄清楚，而不应该只是大概知道。

**实验二、数据结构**

**一、实验目的**

1、学习与掌握逻辑运算与逻辑表达式。

2、熟练掌握if、switch、while、do-while，for语句的语法结构与执行过程。

3、掌握选择、循环程序的设计方法

**二、实验内容**

1、输入一个字符，如果为小写，转换为大写输出，否则，输出其后继字符的ASCII码值。

2、输入x计算表达式的值：



分别输入 0.2, 1, 5 , 0,观察输出结果。

1. 输入三角形的三条边，求周长，并判断该三角形是否为等腰三角形（提示：要三边是否可以构成三角形）。

4、 完成计算器程序，实现（+ - \* / %）运算。考虑除数为 0 与运算符非法的情况。

5、输入一行字符，分别统计出其中英文字母、空格、数字字符和其它字符的个数。

提示：从键盘上读入一个字符给变量 c，判断 c 是属于哪种字符并计数，循环读入下个字符，直到回车换行字符'\n'为止。

cin，scanf（）都不能读入空格以及‘\n’字符，查找资料解决输入这两个字符的方法。

（这个题训练大家自主学习能力以及如何获取新知识、探索解决未知问题的能力。）

6、编写一个程序：从键盘上输入两个正整数，求 a 和 b 的最大公约数与最小公倍数。

7、使用循环结构输出下列图形：

\*

\*\*

\*\*\*

\*\*\*\*

\*\*\*\*\*

8、从键盘输入 a，用迭代法求 a 的平方根 x= *a* 。求平方根的迭代公式为：



要求精确到|xn+1 - xn|<10 -5。

提示：迭代法是把 xn代入迭代公式右边，计算出 xn+1来，然后把 xn+1 作为新的 xn ，计算出新的 xn+1，如此重复，直到|xn+1 - xn|<10 -5 时，xn+1 为所求的平方根。可以把 a 作为 xn 的初始值。

思考：（1）如果输入 a 为负，在运行时会出现什么情况? 修改程序使之能处理任何的 a 值。

(2）能否|xn+1 -xn|<10 -10或更小? 为什么? 请试一下。

9、苹果每个 0.8 元，第一天买 2 个，第二天开始，每天买前天的 2 倍，直到购买的苹果数不超过100的最大值，求每天平均花多少钱。

**三、算法分析，程序结果**

**1.**#include<iostream>

using namespace std;

int main()

{

char a;//创建一个字符型变量

cin >> a;

if (a >= 'a' && a <= 'z')//这些字符会转为asc2码从a到z逐渐变大

{

cout << static\_cast<char>(a - 'a' + 'A') << endl;//直接输出会输出为asc2码 要进行类型的强制转换 从a到z的asc2码是连续的因此算出相差多少再加到A的asc2码上即可

}

else

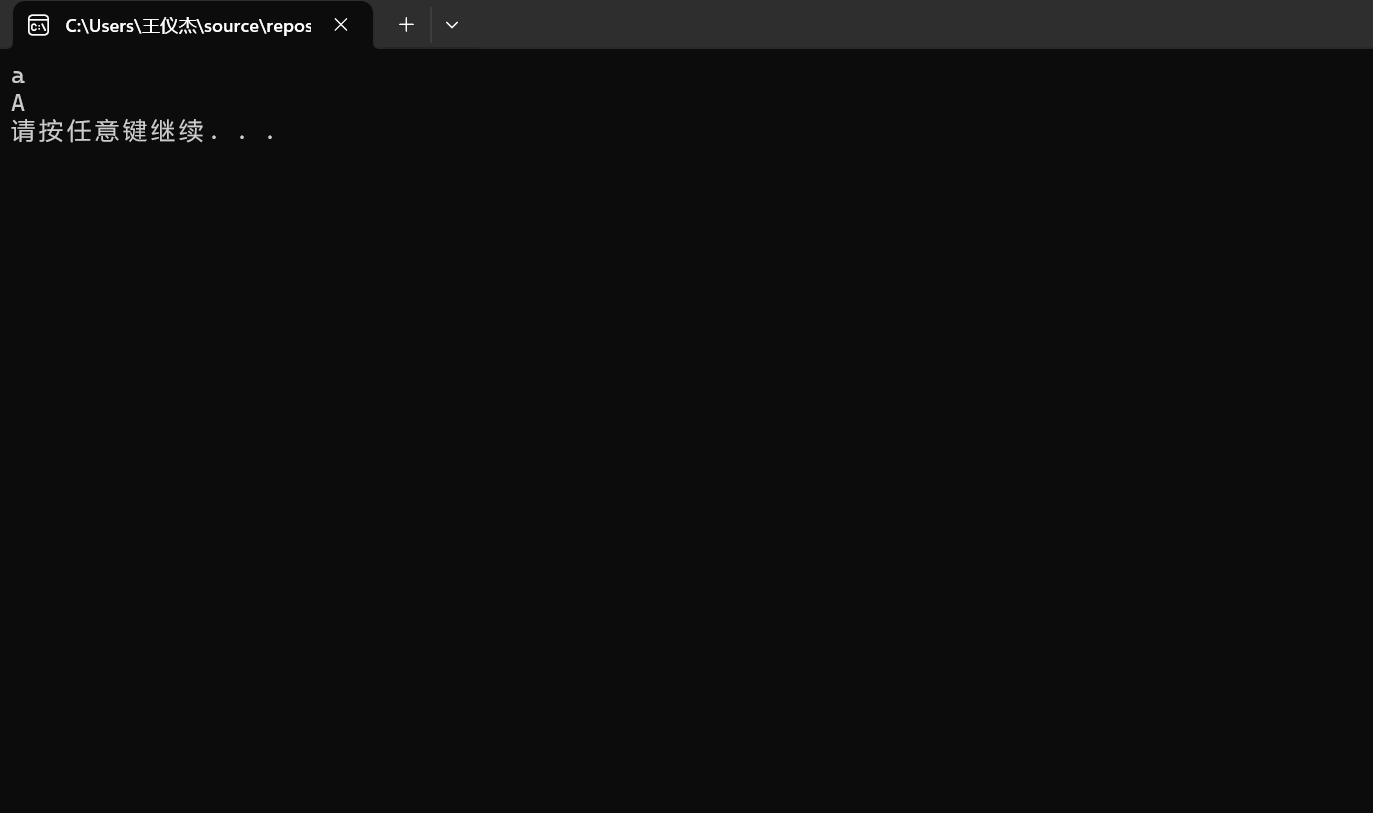
{

cout << (a + 1) << endl;

}

system("pause");

return 0;



}

**2.**#include<iostream>

using namespace std;

int main()

{

double x;

cout << "请输入x的值" << endl;

cin >> x;

if (x > 0 && x < 1)

{

cout << 3 - 2 \* x << endl;

}

else if (x >= 1 && x < 5)

{

cout << 2 / (4 \* x) + 1 << endl;

}

else if (x >= 5 && x < 10)

{

cout << x \* x << endl;

}

else

{

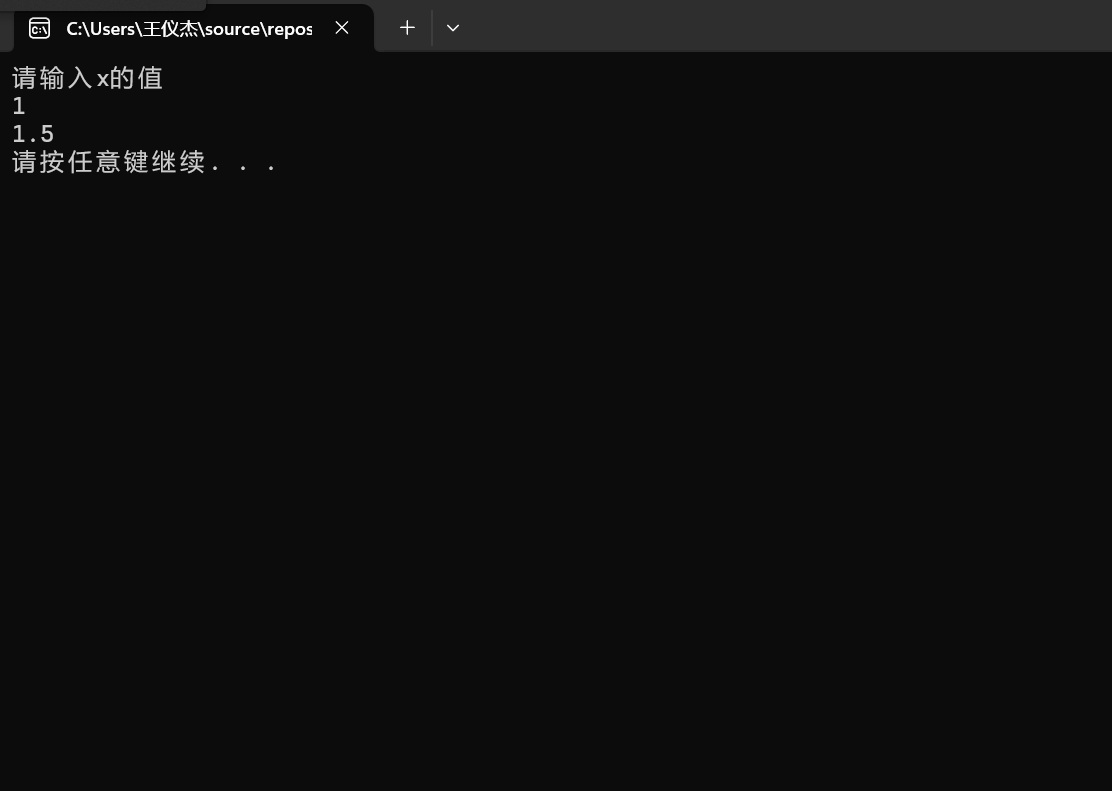
cout << "不在范围内" << endl;

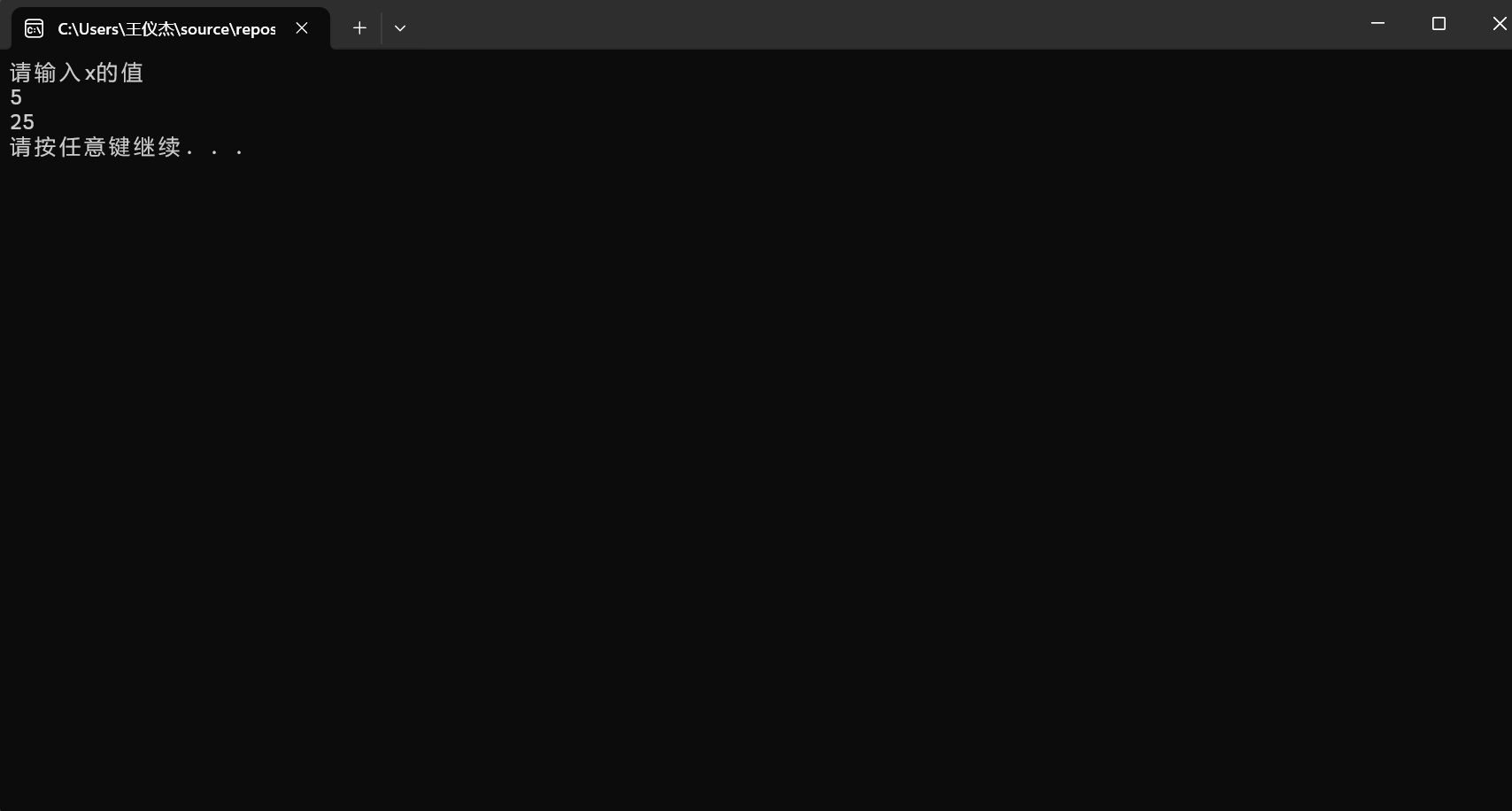
}//分类进行输出即可

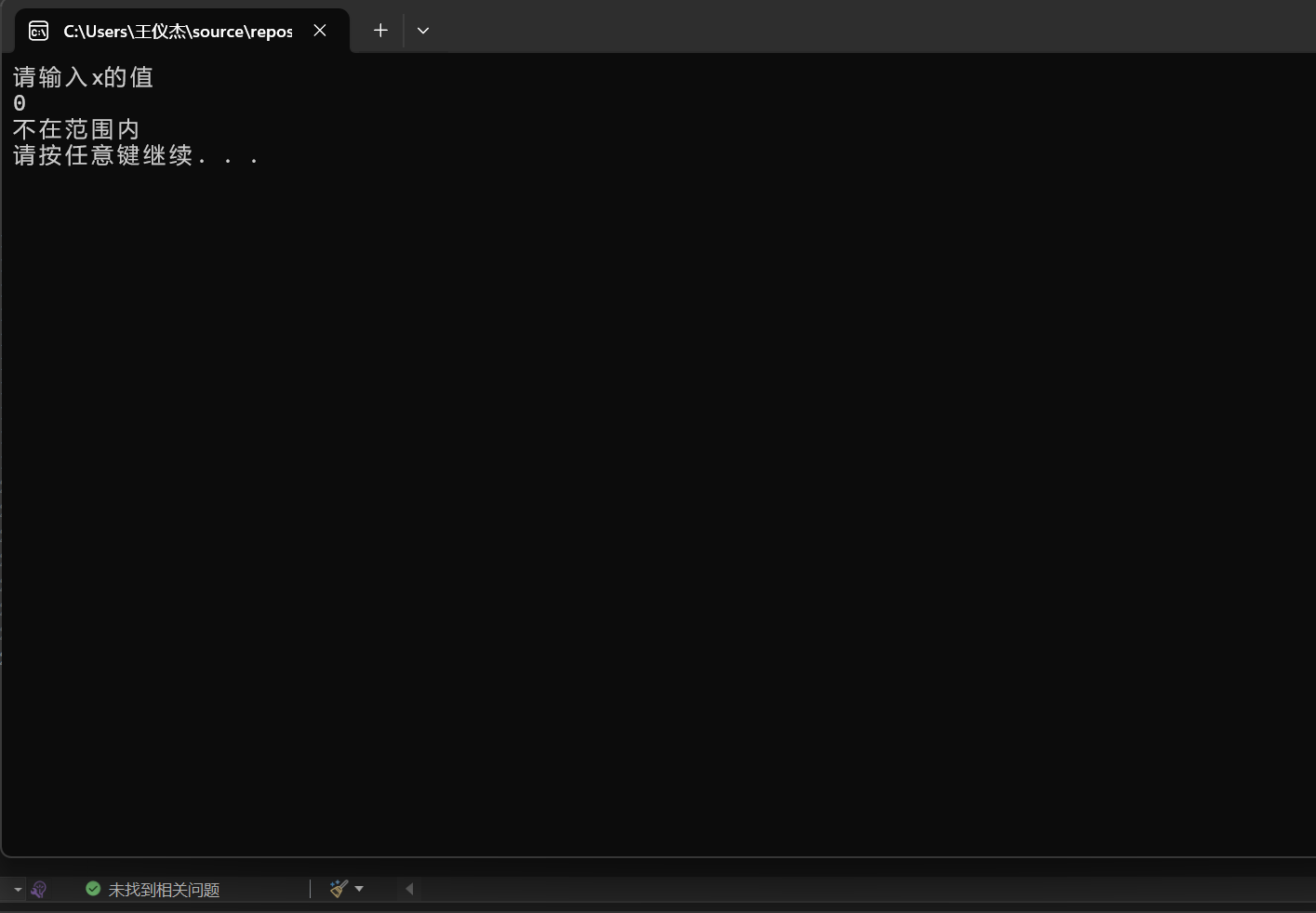
system("pause");

return 0;

}

****

****

****

**3.**#include<iostream>

using namespace std;

int main()

{

int a = 0, b = 0, c = 0;

cout << "请输入三角形的三边长" << endl;

cin >> a >> b >> c;

if ((a < b + c) && (b < a + c) && (c < b + a))//判断能否组成三角形

{

if ((a == b) || (a == c) || (b == c))//判断边长是否相等

{

cout << "该三角形是等腰三角形" << endl;

}

else

{

cout << "该三角形不是等腰三角形" << endl;

}

}

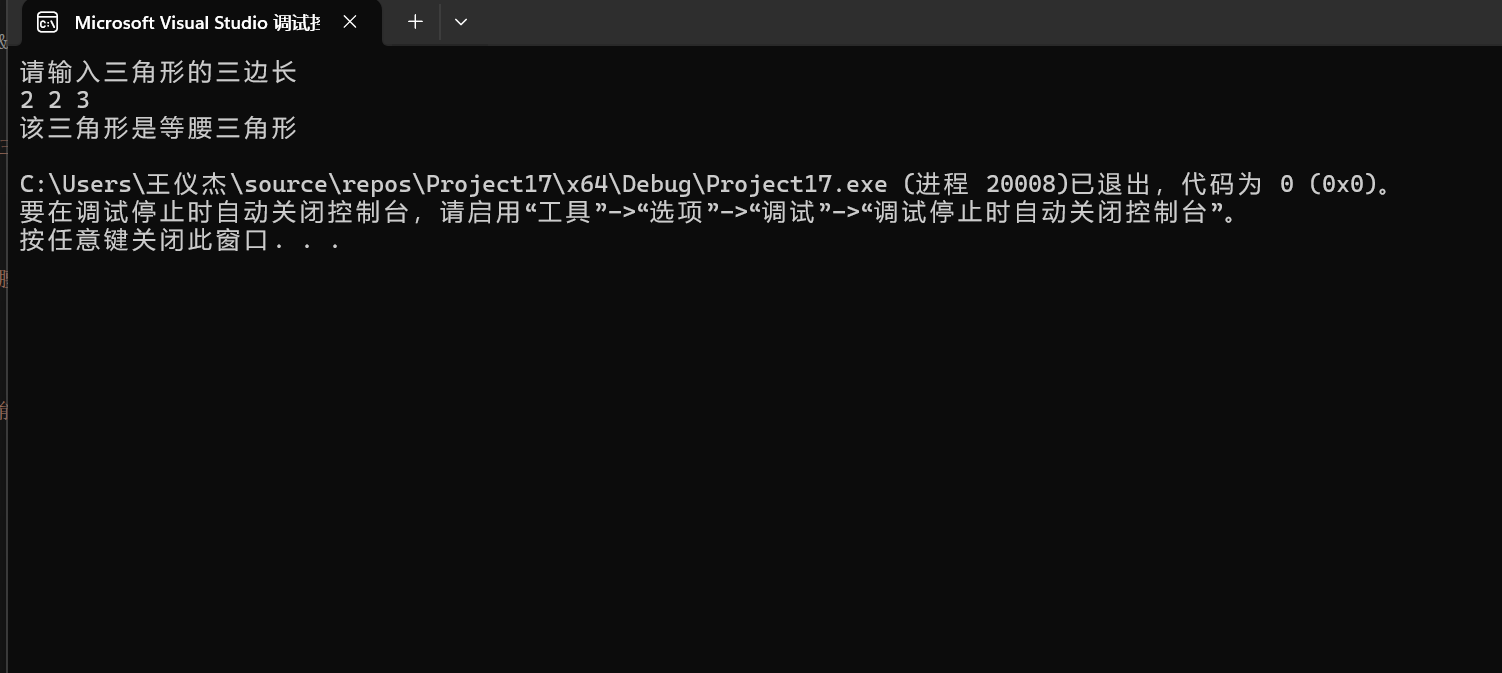
else

{

cout << "您输入的三边长度不能构成三角形" << endl;

}

return 0;

}****

**4.**#include<iostream>

using namespace std;

int main()

{

int num1 = 0, num2 = 0;

char ch;//注意字符型变量不只能够储存字母还可以储存运算符号

cin >> num1 >> ch >> num2;

switch (ch)//利用switch分为四类

{

case '+': cout << num1 + num2 << endl; break;

case '-': cout << num1 - num2 << endl; break;

case'\*':cout << num1 \* num2 << endl; break;

if (num2 != 0)//利用if判断分母是否为零

{

case'/':cout << num1 / num2 << endl; break;

case'%':cout << num1 % num2 << endl; break;

}

else

{

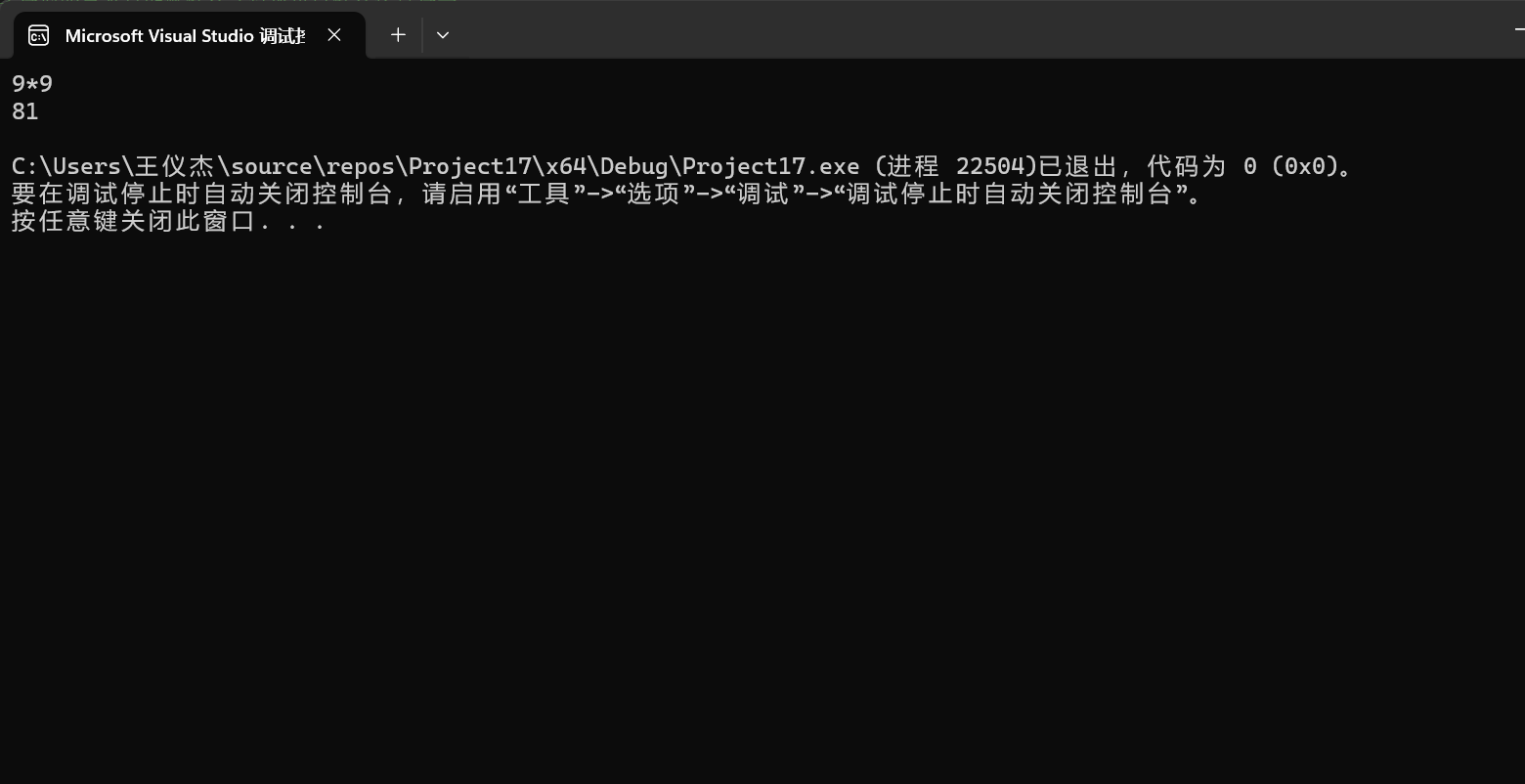
cout << "您输入的除数为零" << endl;

}

default:cout << "您输入的运算符非法" << endl;

}

return 0;

}****

**5.**#include<iostream>

#include<string>//使用getline时需要用到这个头文件

using namespace std;

int main()

{

string input;//创建一个字符串变量但没有初始化，string类型的长度可以根据输入字符长度的变化而变化char【】必须提前确定长度

cout << "请随意输入一行字符" << endl;

getline(cin, input);//将输入的内容存放的input中，更重要的是可以将空格已及特殊字符存放其中

int a = input.length();//将输入的一串字符串的长度存放在a中

int letter = 0, k = 0, num = 0, q = 0;

for (int i = 0; i < a; i++)//注意这里没有等号 数组【】里会少一个

{

char b = input[i];//把第一项给字符型变量b,利用for遍历input中的字符

if ((b >= 'a' && b <= 'z') || (b >= 'A' && b <= 'Z'))

{

letter++;//计算字母的数量

}

else if (b == ' ')

{

k++;//计算空格的数量

}

else if (b >= '0' && b <= '9')

{

num++;//计算数字的数量

}

else

{

q++;//计算其他字符的数量

}

}

cout << "这一行字符中有数字" << num << "个" << endl;

cout << "这一行字符中有字母" << letter << "个" << endl;

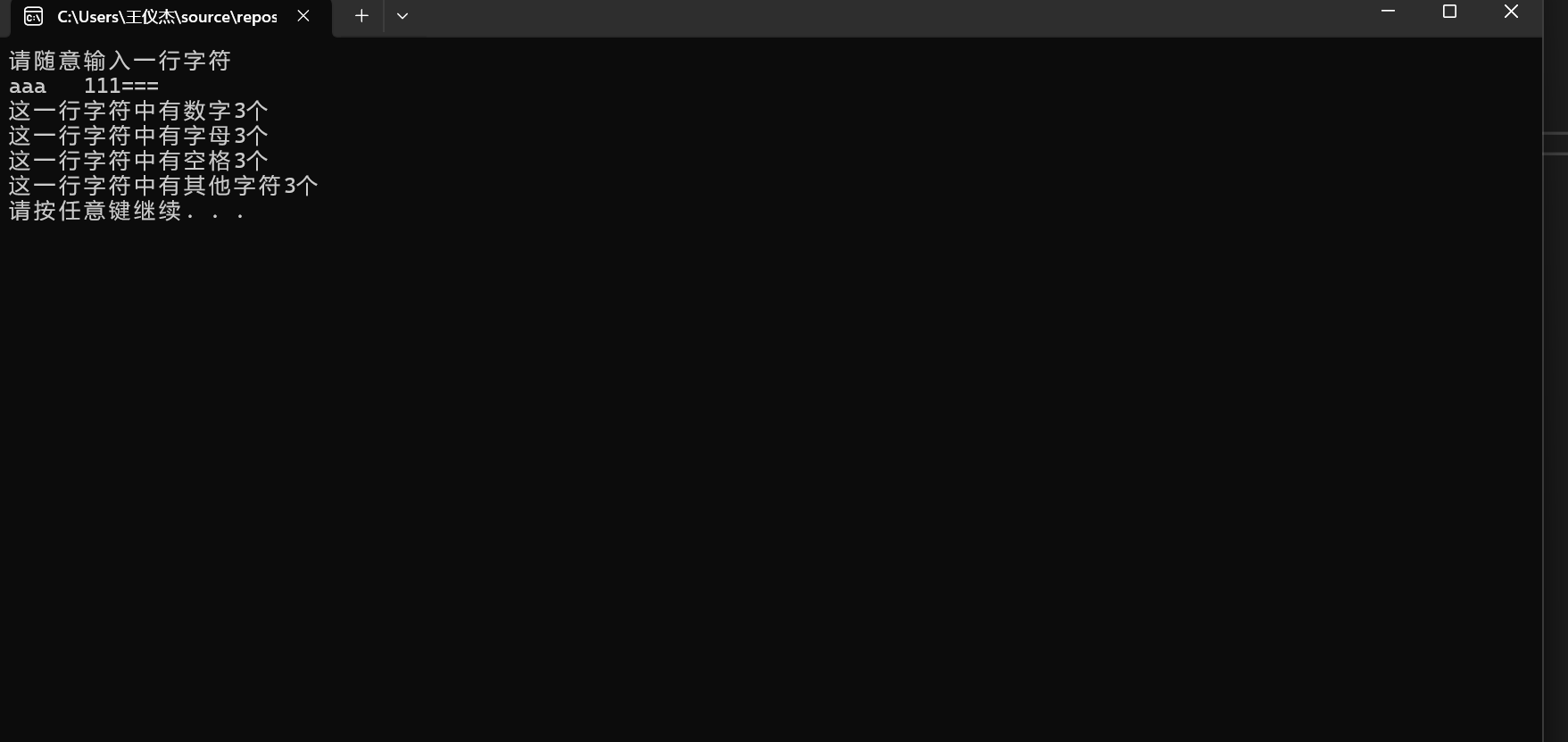
cout << "这一行字符中有空格" << k << "个" << endl;

cout << "这一行字符中有其他字符" << q << "个" << endl;

system("pause");

return 0;

}



6.#include<iostream>

using namespace std;

int gcd(int m, int n)//创建一个求最大公约数的函数

{

int temp = 0;//创建临时变量

for (int i = 1; i <= m && i <= n; i++)//最大公约数的循环范围

{

if ((m % i == 0) && (n % i == 0))//最大公约数的特性，找出所有公约数

{

temp = i;//通过循环把最大的一个赋值给temp

}

else {}

}

return temp;//如果不是用void定义则一定要有返回值

}

int lcm(int x, int y, int z)//创建最小公倍数函数，因为最大公约数与最小公倍数的乘积等于原来两个数的乘积

{

int e = x \* y / z;//所以可以直接采取这样的定义方式

return e;

}

int main() {

int a = 0, b = 0;

cout << "请输入两个正整数" << endl;

cin >> a >> b;

int c = gcd(a, b);//两个函数的调用

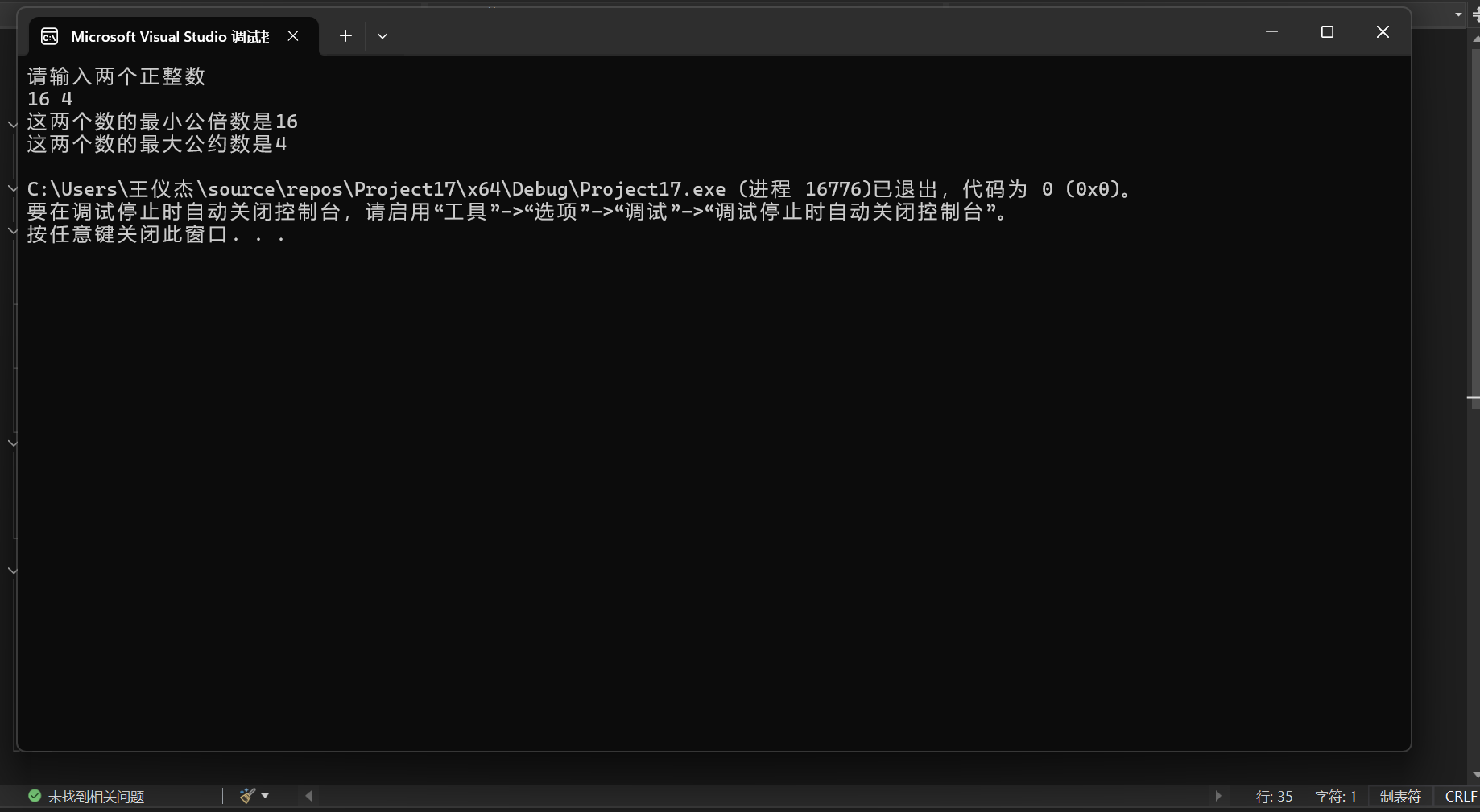
int d = lcm(a, b, c);

cout << "这两个数的最小公倍数是" << d << endl;

cout << "这两个数的最大公约数是" << c << endl;

system("pause");

return 0;

}

**7.**#include<iostream>

using namespace std;

int main()

{

for (int i = 0; i < 5; i++)//第一个循环表示需要5行

{

for (int m = 0; m <= i; m++)//第二个循环表示每行的\*数量小于他的行数

{

cout << "\*";

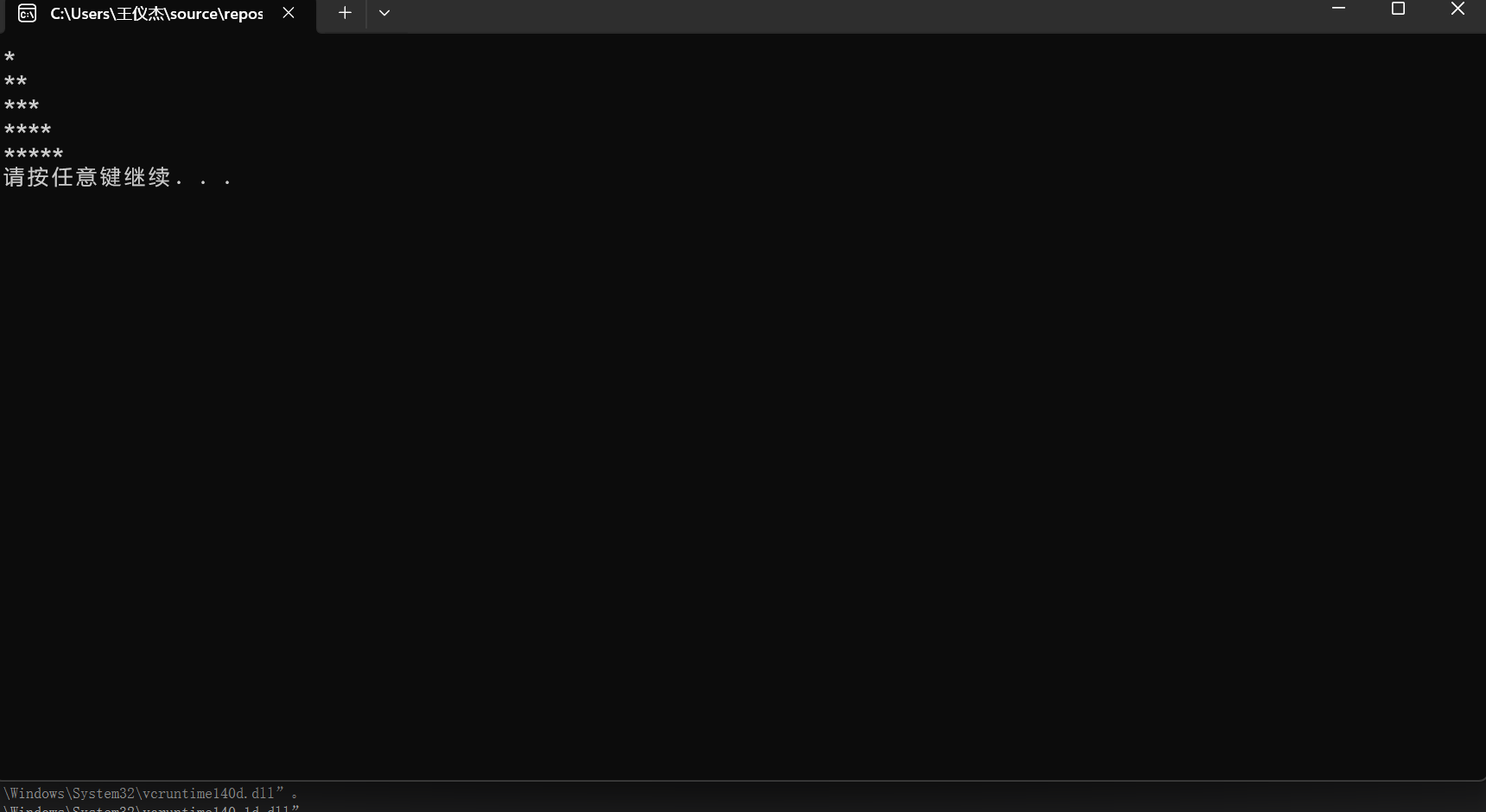
}

cout << endl;//在每一行的结尾进行换行

}

system("pause");

return 0;}



**8.**#include <iostream>

using namespace std;

int main()

{

double a = 0, xn = 0, xnp1 = 0;//定义所需的三个变量

cout << "请输入一个您要对其开平方根的数" << endl;

cin >> a;

xn = a;//a作为初始值

while (1)//设置为死循环

{

xnp1 = 0.5 \* (xn + a / xn);//运用迭代公式

if ((xnp1 - xn >= 0 && xnp1 - xn < 0.00001) || (xnp1 - xn && xn - xnp1 < 0.00001))//设置跳出循环的条件

{

cout << xnp1 << endl;//输出最终的开方值

break;

}

else

{

}

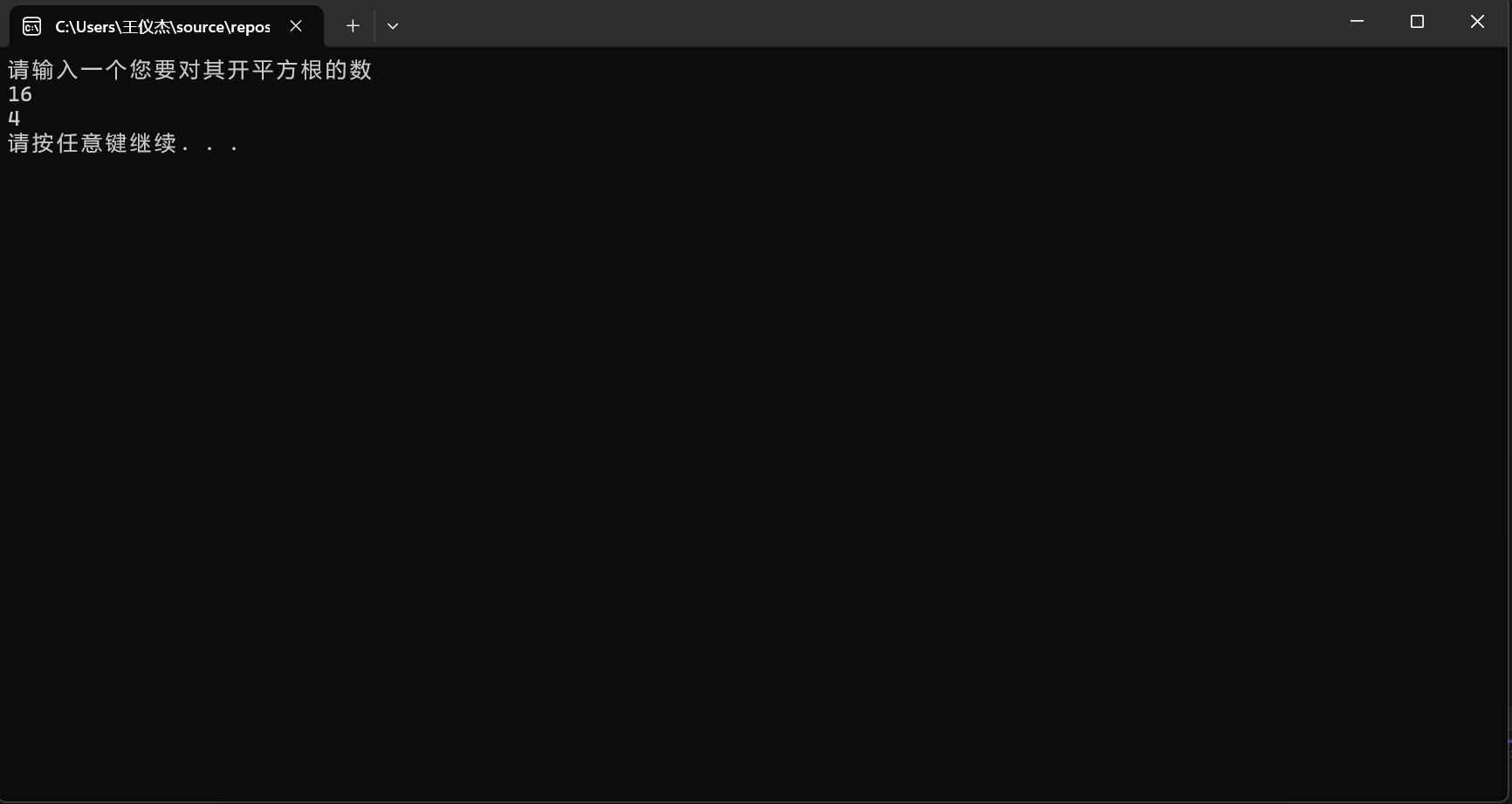
xn = xnp1;//如果比满足条件就把xnp1重新赋值给xn

}

system("pause");

return 0;

}



9.#include <iostream>

using namespace std;

int main()

{

double money = 0;

double day = 0;//设置两个变量并进行初始化

for (int i = 2; i <= 100; i \*= 2)//设置循环看一共会买多少个苹果

{

day++;//计算循环了多少次，即用了多少天

money += i \* 0.8;//计算总钱数

}

cout << "平均每天花钱" << endl;

cout << money / day << endl;//计算平均花钱

system("pause");

return 0;

#### }屏幕截图 2024-11-09 094002

1. **遇到的问题与解决方法**

**二 1.**

开始时我使用小写字母asc2码的值的范围来进行条件判断

后来想起输入字符型变量时编译器本来就会把他转化为asc2码所以我想到

这样写可以避免每次都去查asc2码表

**屏幕截图 2024-11-09 103845**

此处注意输入字符型变量时一定要加单引号

然后我不知道如何将小写字母变为大写，经过上网查询可知因为大小写字母的asc2码是连续的所以可以先算出输入的小写字母距a的asc2码的差值再加到A的asc2码上

**屏幕截图 2024-11-09 104428**

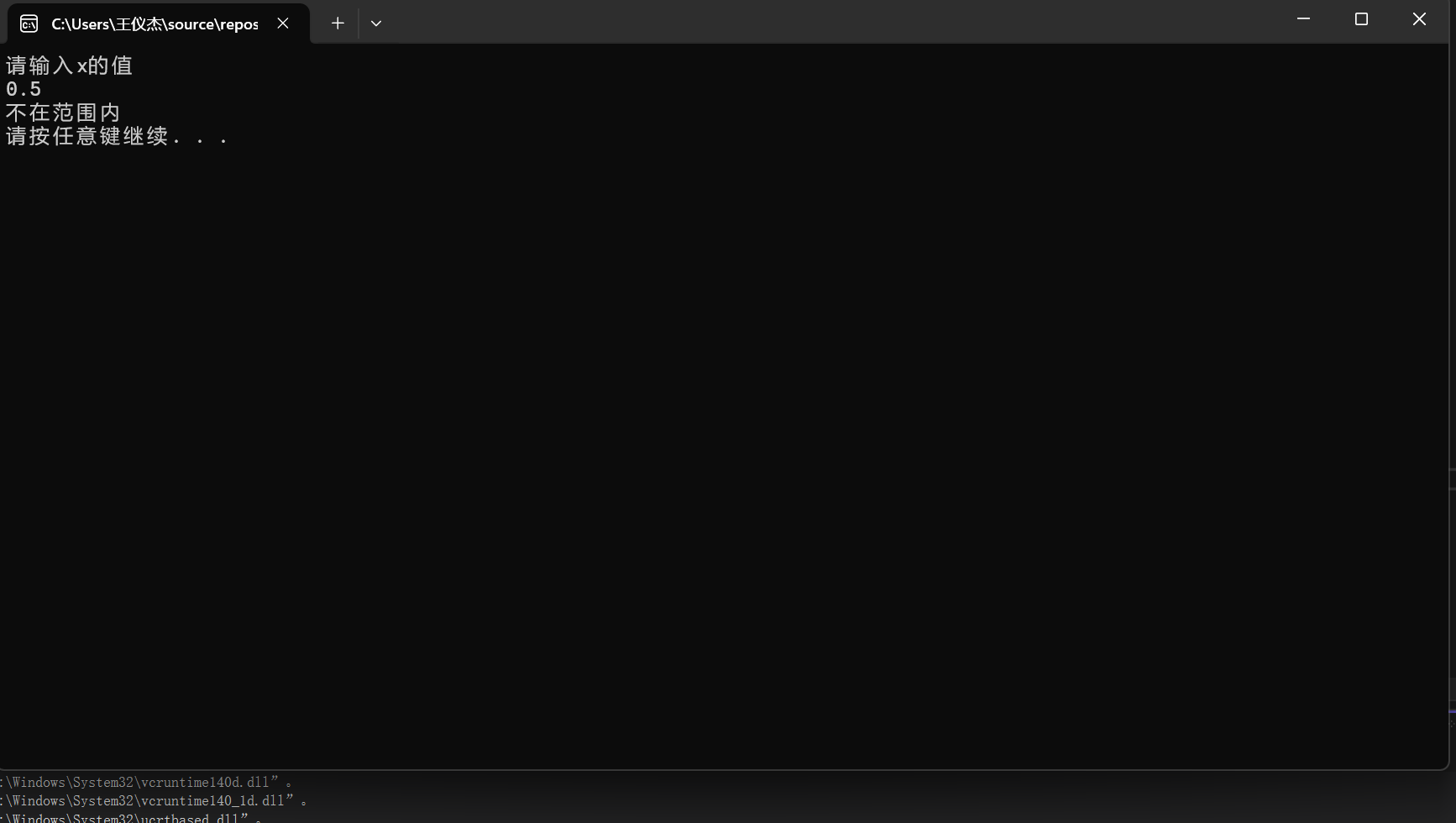
注意如果不进行类型的转换，输出的是asc2码值而非大写字母

**二 2.**

最开始我根据题干直接写连续的大于号

**屏幕截图 2024-11-09 105017**

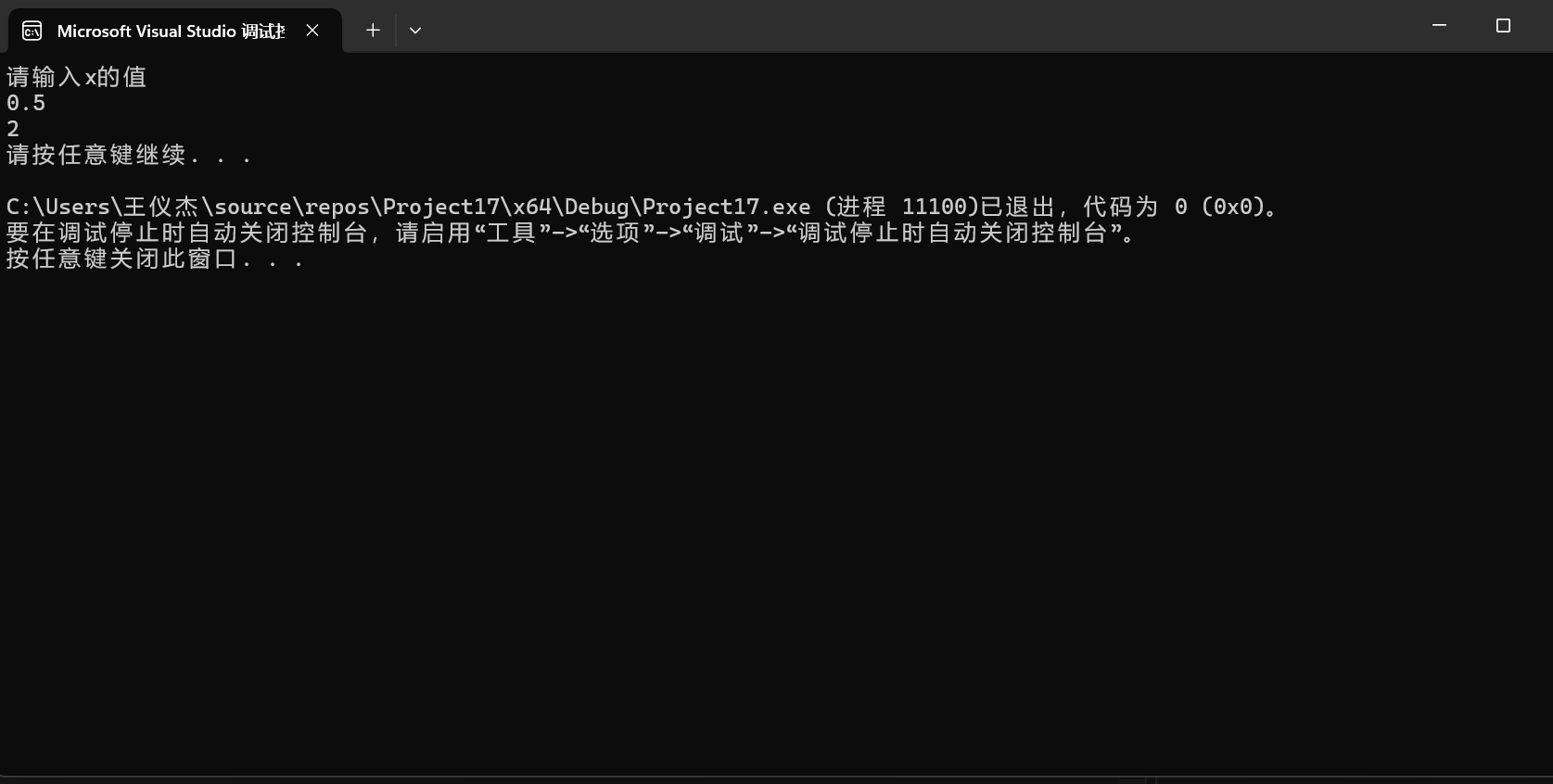
然而在运行后发现出现如下情况



推测报出不在范围内的原因是在利用if判断时条件错误，查询后可知cpp不会识别连续的大于或小于号，应改用逻辑判断的语句

**屏幕截图 2024-11-09 105603**

**最后运行结果为**

****

可见运行结果正常

**二 4.**

**1.**首先要注意char类型不仅可以储存字母含可以储存加减等运算符。

**2.**注意case后面的运算符要加单引号。

**二 5.**

1. 刚开始我想直接用char[]来定义一个字符型数组来储存输入的一行字符，但此时会报错

**屏幕截图 2024-11-09 111606**

即使用char时必须直接规定好字符型数组的长度

所以这时应该用string,string能够储存可变长度的字符序列



1. 如何储存空格

经过在网上查询可知使用getline函数可以实现此功能，getline会遍历这一串字符串直到出现换行为止与cin不同cin只要遇到空格便会停止无法储存空格

1. getline的具体使用语法getline()

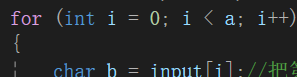
屏幕截图 2024-11-09 113916

接下来在使用for遍历这一行字符时要知道长度

在网上查询可知使用length可以取长度



1. 接下来进行for循环时不知道如何表示数组的第一个元素，然后想到用string定义的字符型变量是否也可以看成一个数组，于是我这样写



5.识别字母时利用asc2码 与之前的题一样

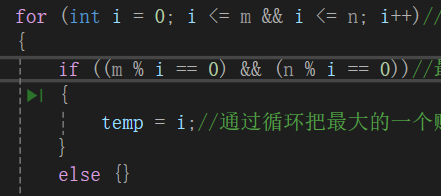


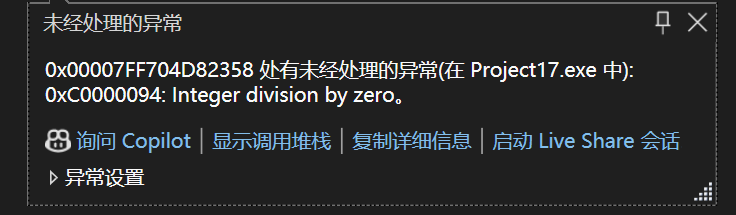
1. 空格也算作是字符两侧要加上单引号



**二 6.**

1. 刚开始再写for循环时我直接让i的初值为0，如图

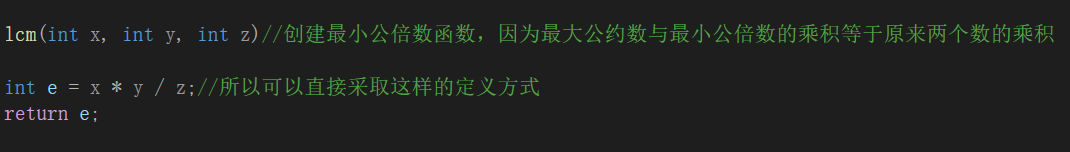


此时会报错

根据之前的经验可知这个错误是说除数为零，于是我更改一下i的初值直接让它从一开始，即可避免这一错误。

1. 在定义最大公约数时采用穷举法每次循环后把最新的值也是最大的值替换给temp

最后把这个值输出即可，在定义最小公倍数时其实也可以采用类似的方式进行枚举最终把最大的值赋给temp。但是观察到两数的乘积等于两数的最大公约数和最小公倍数的乘积，所以将两数相乘并处以它的最大公约数即可得出它的最小公倍数如下图所示



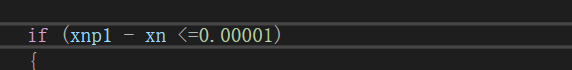
注意要设好返回值。然后进行调用即可。

**二 7.**

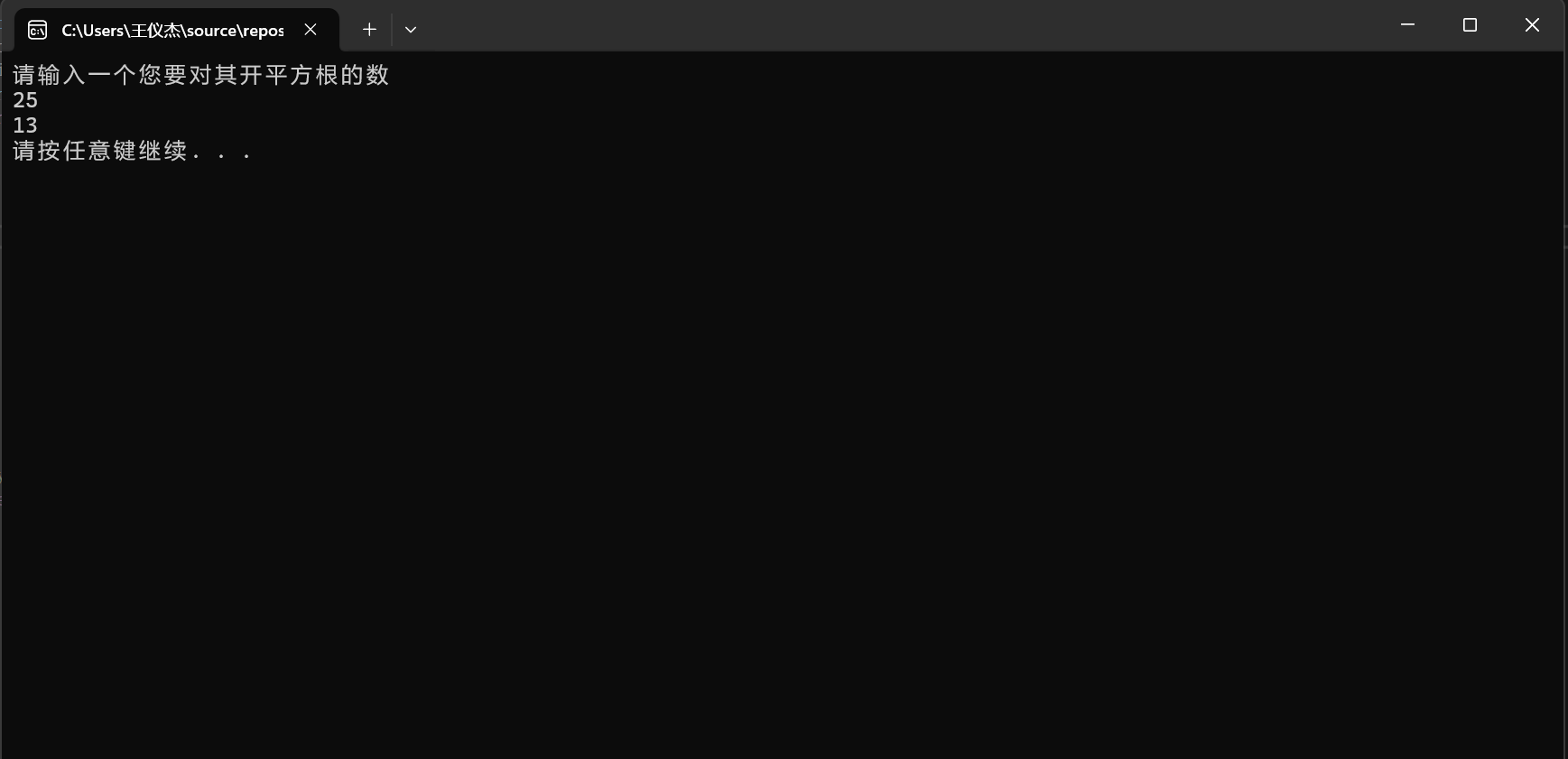
1. 注意每个for表示的都是什么，并且在合适的时候进行换行

**二 8.**

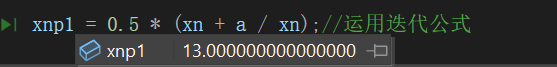
1. 刚开始我把给xn赋初值的步骤写在了循环里面，这导致每进行一次循环都会重置一次xn的值，显然是错误的。
2. 开始时根据题中的条件我直接把if中的条件设为

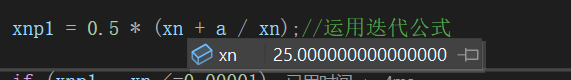


在这种情况下输出的结果为



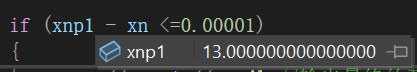
可以看到结果误差极大。

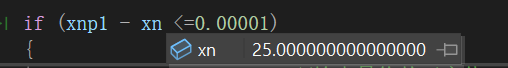
现在使用逐行检验的方法对变量的值进行检验，进入while循环后执行第一步时变量的值如图所示：



此时数值计算正确并未出现任何问题。

接下来在进入if循环之后变量的值如图：





可以看到这是发现问题所在二者相减为负数，判断if条件为真直接输出了xp1。

仔细看题可以发现题上所说的时xnp1-xn的绝对值的范围，于是可以对代码稍作修改

屏幕截图 2024-11-09 191708

这样进行逻辑判断即可实现绝对值的功能。

关于思考的问题，不能使|xn+1 -xn|<10 -10或更小，由于浮点数的精度限制以及误差，应用中无法总是达到10的-10次方的精度。

1. **体会**
2. 对字符型变量以及它所对应的asc2码的相关内容不太熟练比如在2.1中意识到字母的asc2码是连续的这样就可以实现大小写字母的相互转换，另外要注意在使用字符型变量时一定要加单引号，字符型变量都会被计算机转换为asc2码所以如果你直接对其进行输出 ，会输出对应的asc2码而不会输出字母一定需要进行类型的强制转换才会变为字母（转换为char类型）我需要加强这方面的知识。
3. 对char字符型变量储存的东西不够清晰（不仅可以储存字母还可以储存加减这类运算符）
4. 注意switch 的形式，以字符为标准时要在case后面加单引号，每个case后都要加break。switch函数平时使用较少因此以往较多，平时应多加练习。
5. 对录入空格字符串的函数应该进行更多了解这次我使用的时getline函数在查阅资料时发现还可以使用cin.getline来进行空格字符串的识别说明这个方法是可以进行尝试的。在下面一段时间我应该自己学习相关的知识
6. 在编程时也更应注重细节。比如在2.6中忘记给定义的函数设置返回值，忽略分母可能是0的问题，编程时也应注意要求是什么，比如在2.8中忘记考虑绝对值这一条件
7. 对于基本的函数类型的方法有所遗忘比如switch函数，平时应多加练习，及时复习所学习的各种函数，并用其进行实践