**计算机程序设计基础（C++)**

**实验报告**

专业班级： 软件工程2306班

学 号： 8209230626

姓 名： 赵云

**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");

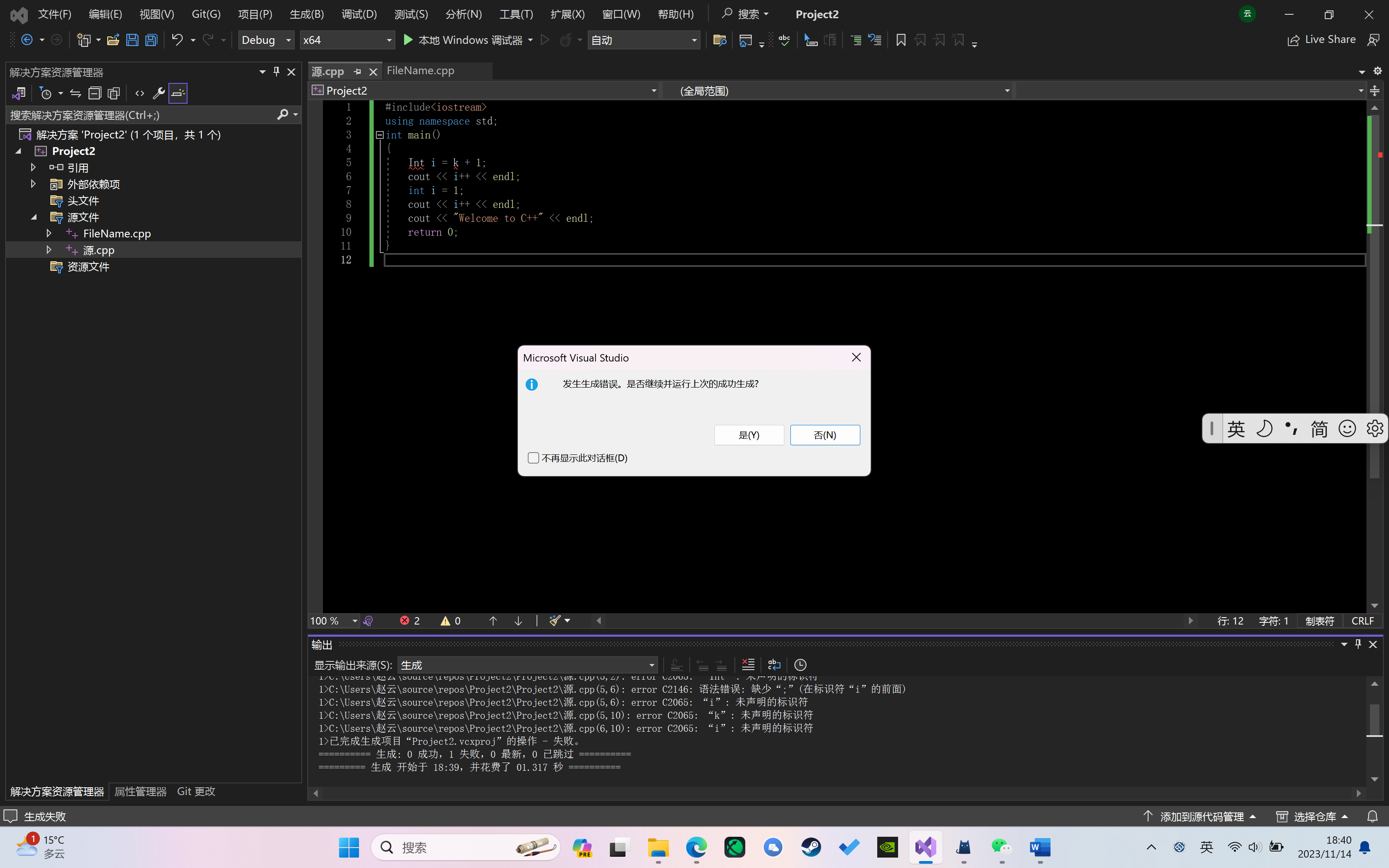
return 0;

**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;

int i = k + 1;

cout << i++ << endl;

i = 1;

cout << i++ << endl;

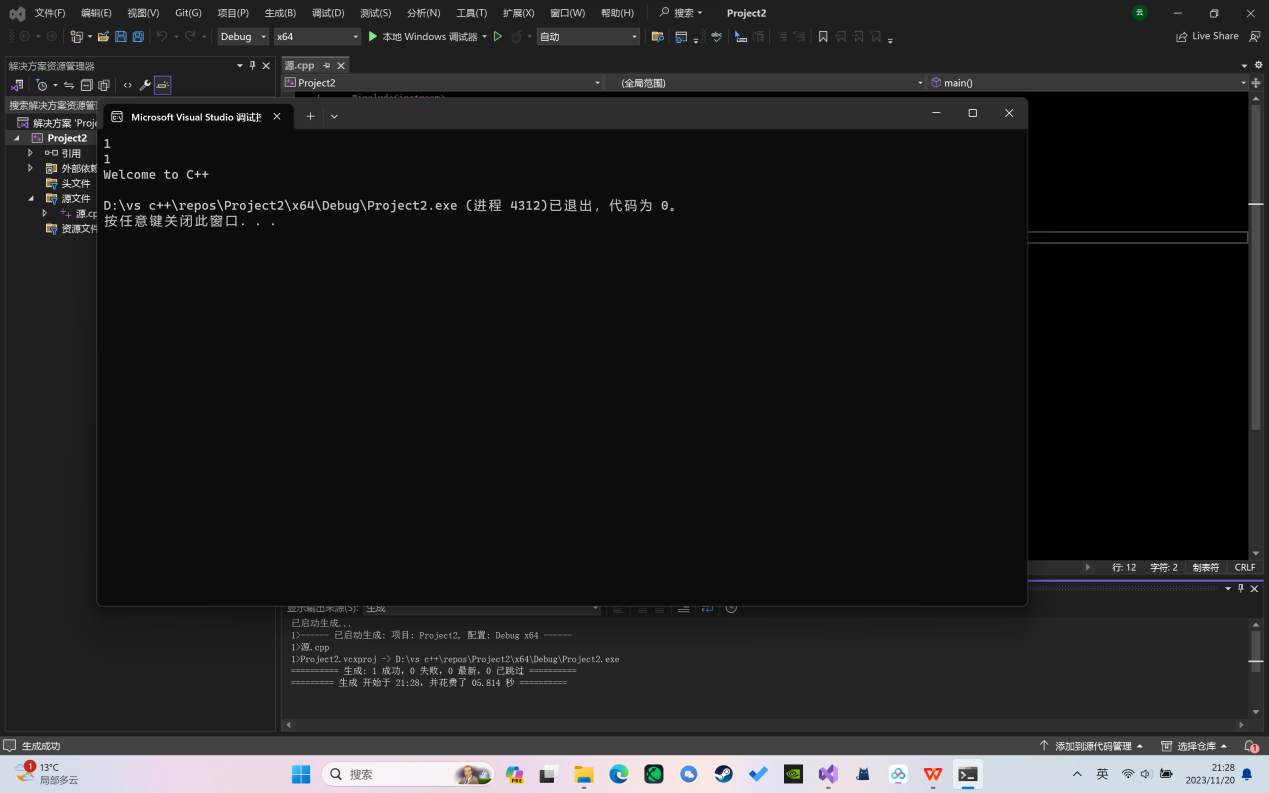
cout << "Welcome to C++" << endl;

return 0;

}

错误：

1. 原代码中“int”的i大写；
2. k未定义且未初始化；
3. i定义了两遍。

输出结果：

2.#include <iostream>

using namespace std;

const double pi = 3.14;

int main(){

double r, h,V;

cout << "请输入圆锥底的半径:" << endl;

cin >> r;

cout << "锥高:" << endl;

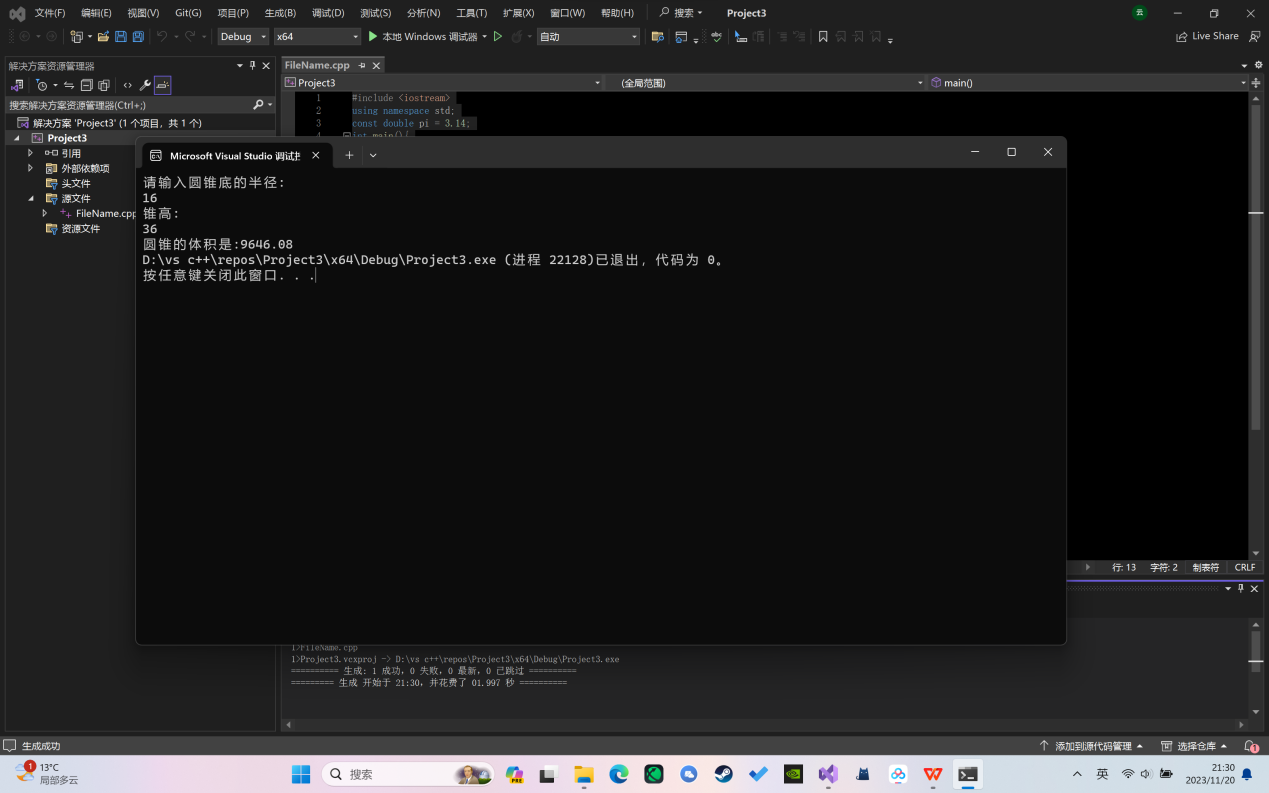
cin >> h;

V = pi \* r \* r \* h / 3;

cout << "圆锥的体积是:" << V;

return 0;

}



3.#include<iostream>

using namespace std;

int main()

{

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

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

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

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

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

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

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

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

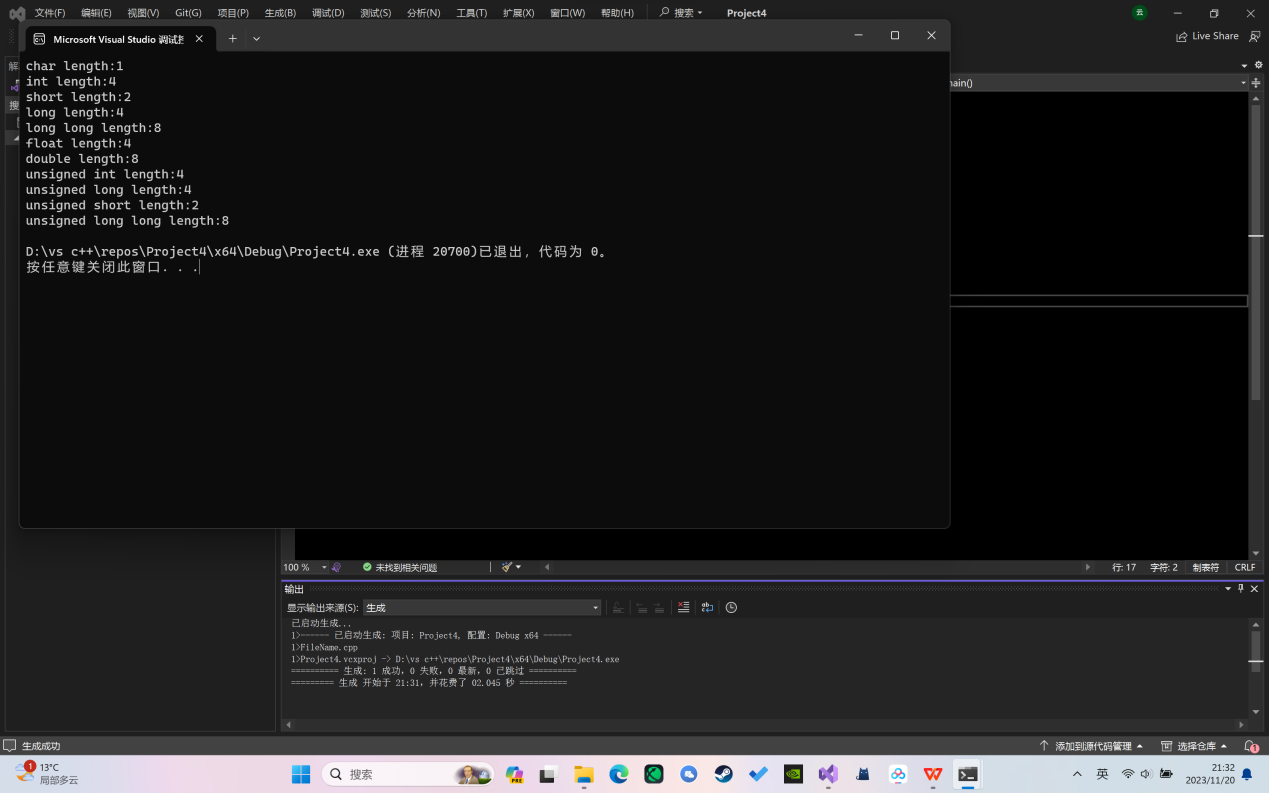
cout << "unsigned long length:" << sizeof(unsigned long) << endl;

cout << "unsigned short length:" << sizeof(unsigned short) << endl;

cout << "unsigned long long length:" << sizeof(unsigned long long) << endl;

return 0;

}



1. （1）改成输出八进制后：

#include <iostream>

#include <iomanip>

using namespace std;

int main() {

unsigned int testUnint = 65534;//oxfffe

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

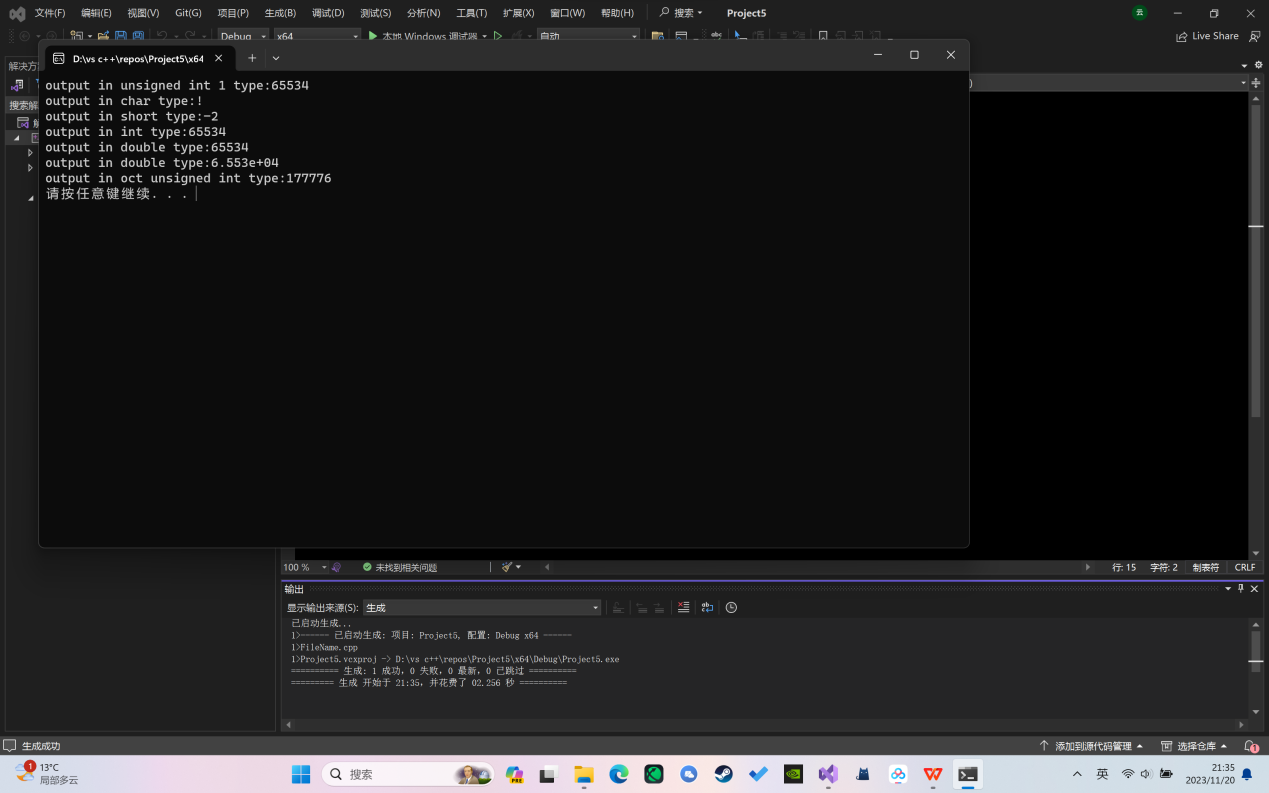
cout << "output in double type:" << setprecision(4) << static\_cast<double>(testUnint) << endl;

cout << "output in oct unsigned int type:" << oct << testUnint << endl; //8进制输出

system("pause");

return 0;

}



1. 为什么转化成short型后变成了-2？

因为65534的二进制原码为0111 1111 1111 1110，在转换为无符号short型时需要补码，

补码后的二进制表示为1000 0000 0000 0010，最高位的1表示负号，第二低位的1表示2的1次方，

所以就表示为-2。

（3）将实数转化为int型，会忽略小数点后所有的数。

5.#include <iostream>

#include <iomanip>

using namespace std;

double F, C;

int main() {

cout << "请输入华氏摄氏度:℉=" << endl;

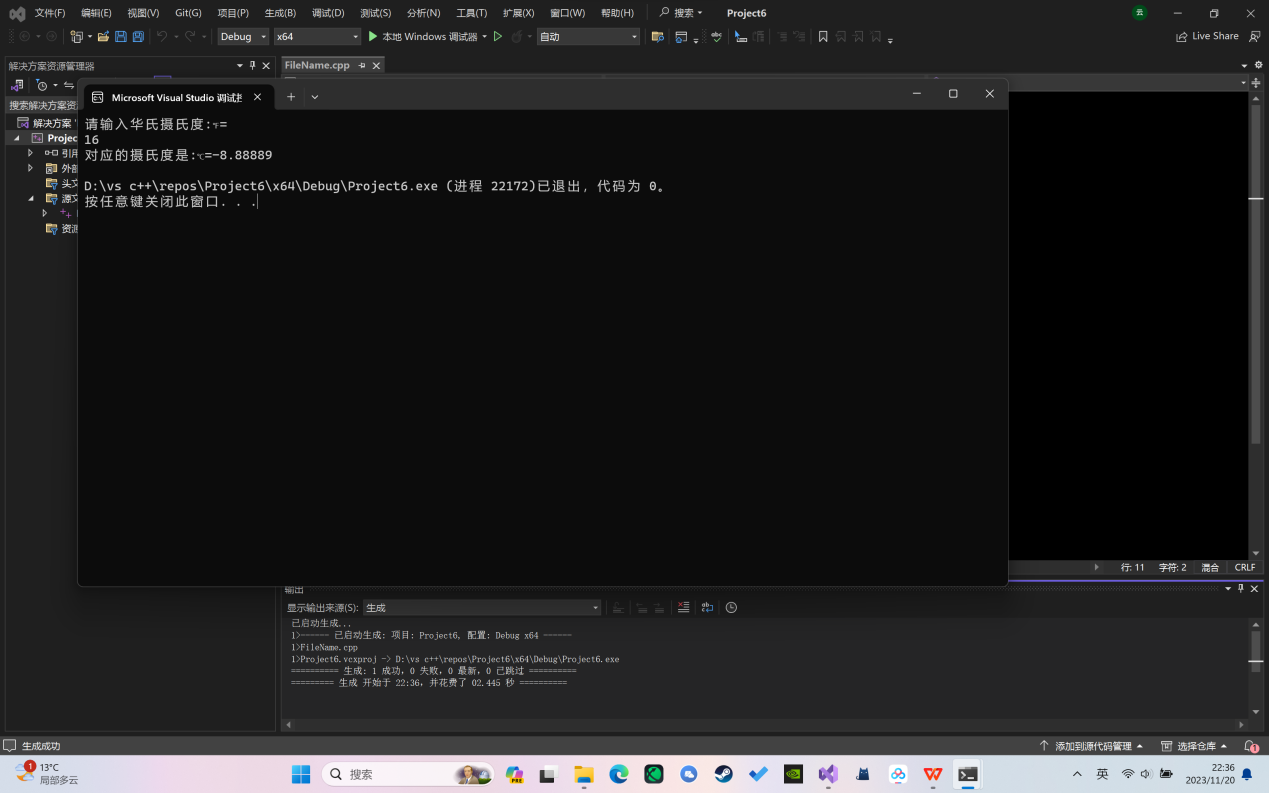
cin >> F;

C = 5.0 / 9.0 \* (F - 32);

cout << "对应的摄氏度是:℃=" << C << endl;

return 0;

}



1. **遇到的问题与解决方法**

**在解决第四题时，因为不明白为什么unsigned int转化为short会变成-2，**

**所以去网上查了资料。了解了计算机的底层逻辑，最后自己解决了这个问题。**

1. **体会**

**学会了计算机的原码、反码和补码。**

**提升了自主解决问题的能力。**

**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;

char achar,bchar;

int a;

int main() {

cout << "请输入一个小写的英文字母：" << endl;

cin >> achar;

a = achar;

if (a < 123 && a > 96) {

bchar = a - 32;

cout <<"它的大写形式是:"<< bchar << endl;

return 0;

}

else {

bchar = a + 1;

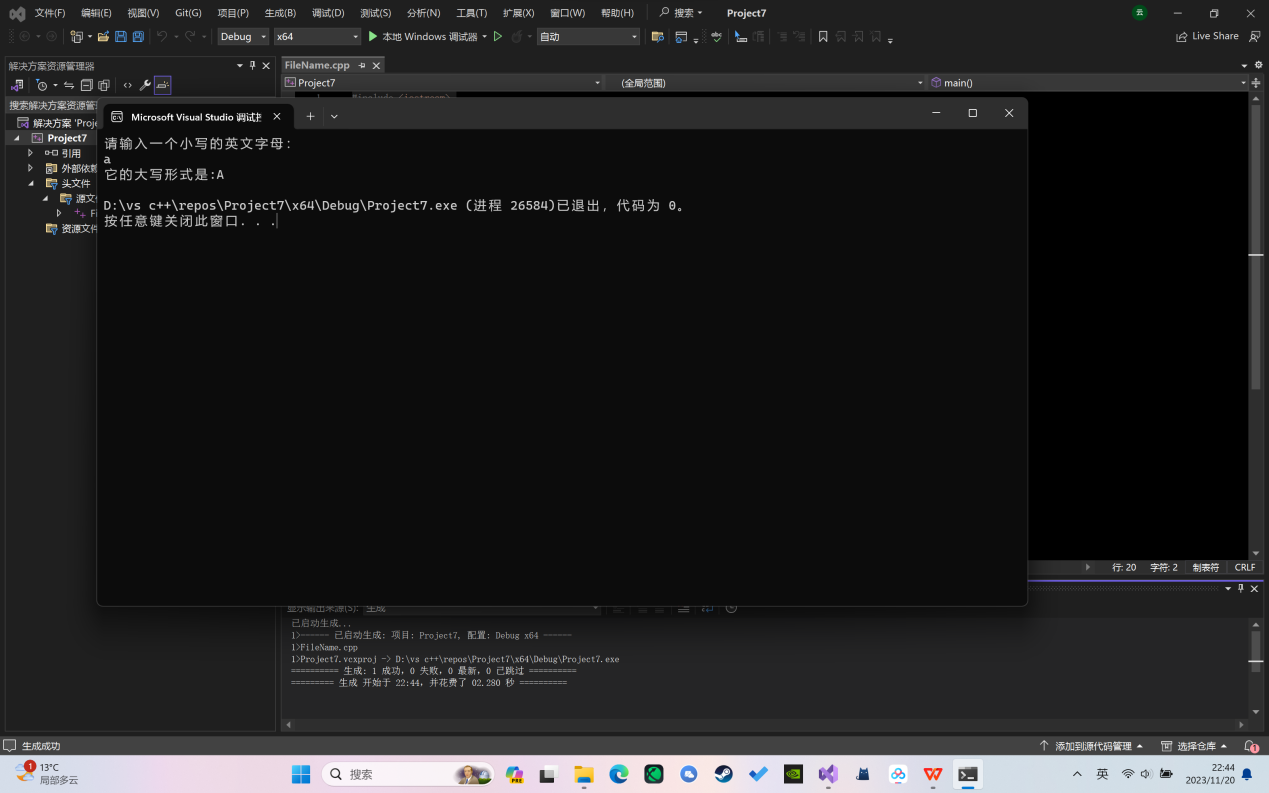
cout << "你输入的不是小写字母！\n这个字符的后继字符的ASCII码是:" << static\_cast<int>(bchar) << endl;

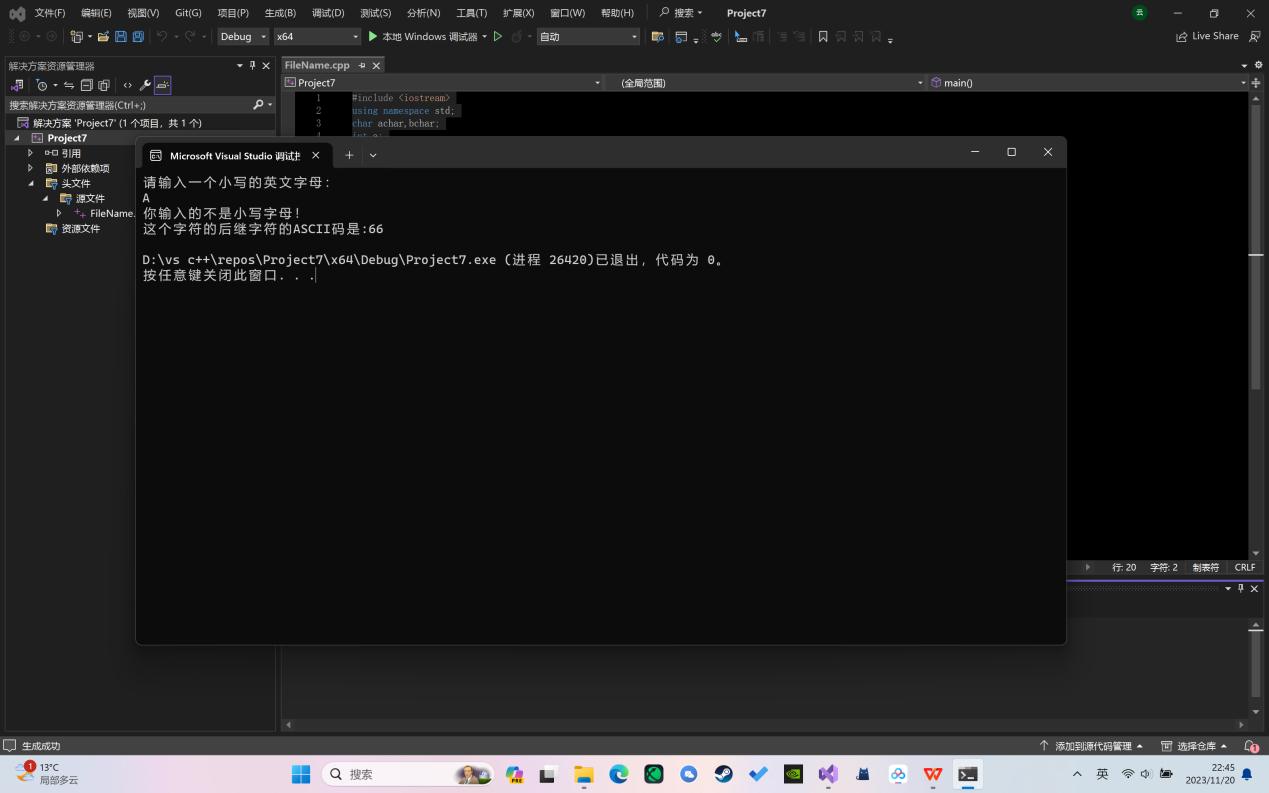
return 0;

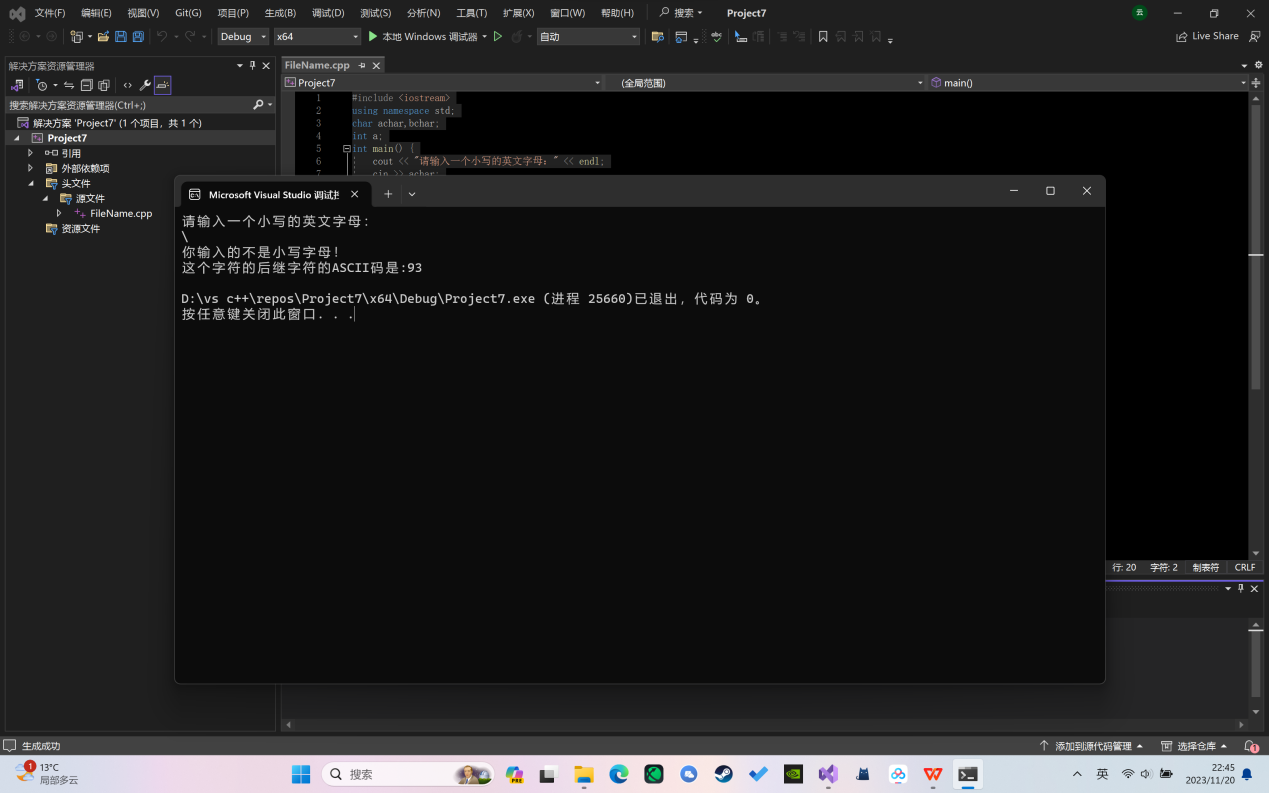
}

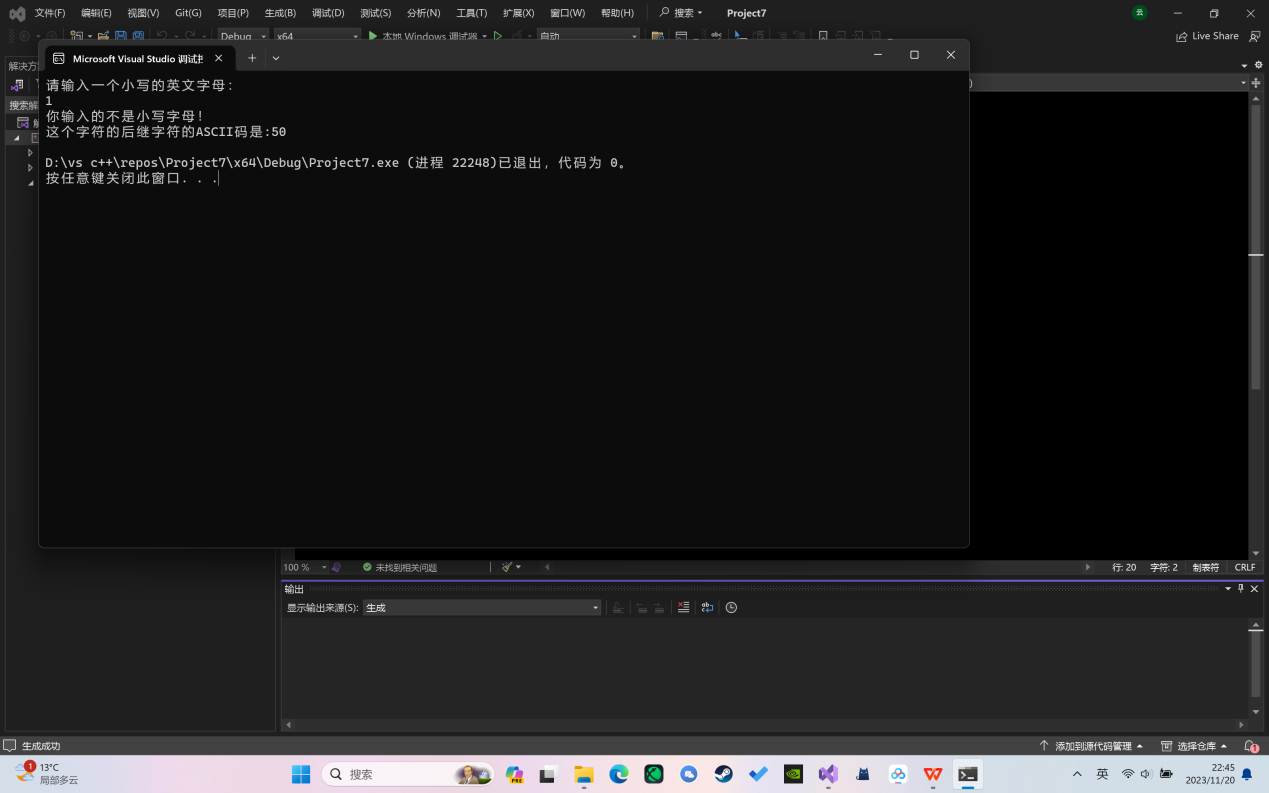
return 0;

}









2.#include <iostream>

using namespace std;

double x, y;

int main() {

cout << "输入x,范围为大于0小于10：" << endl;

cin >> x;

if (x > 0 && x < 1) {

y = 3 - (2.0\* x);

cout << "y=" << y << endl;

}

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

y = 2.0 / (4.0 \* x) + 1;

cout << "y=" << y << endl;

}

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

y = x \* x;

cout << "y=" << y << endl;

}

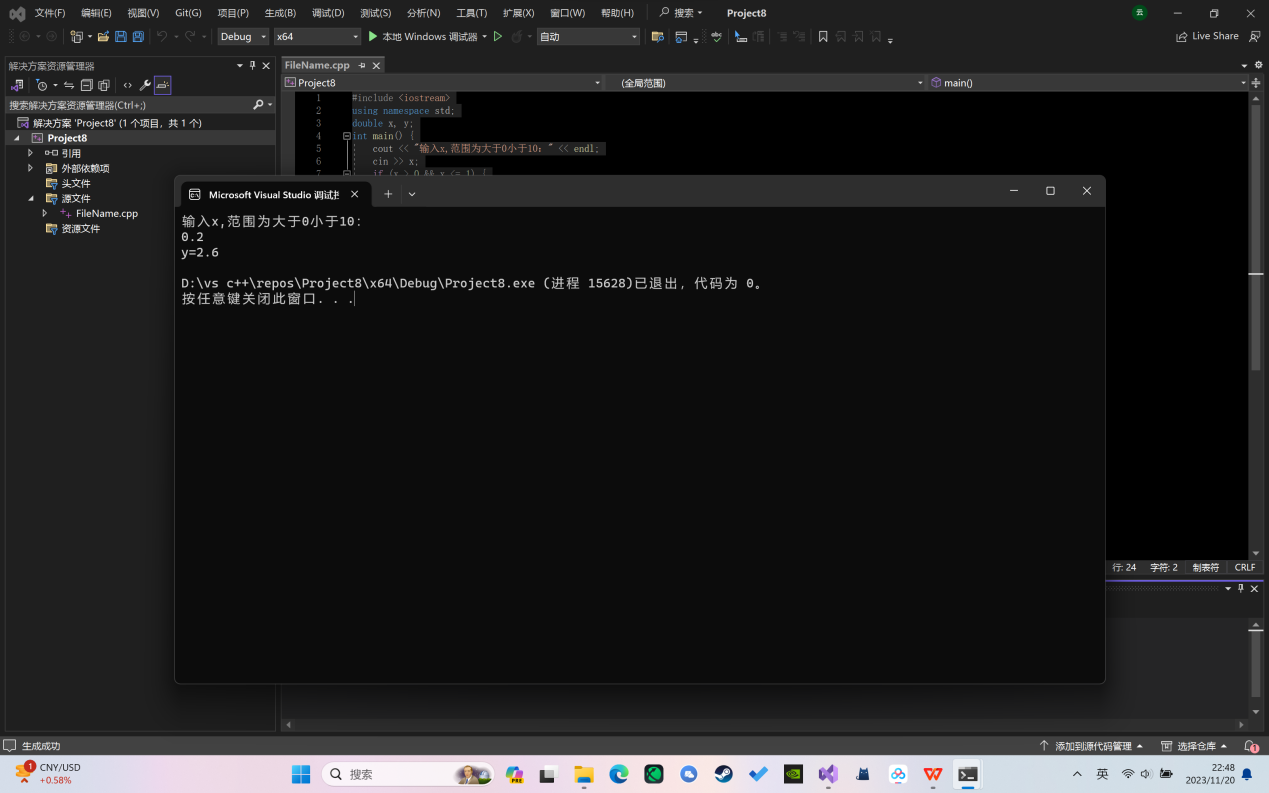
else {

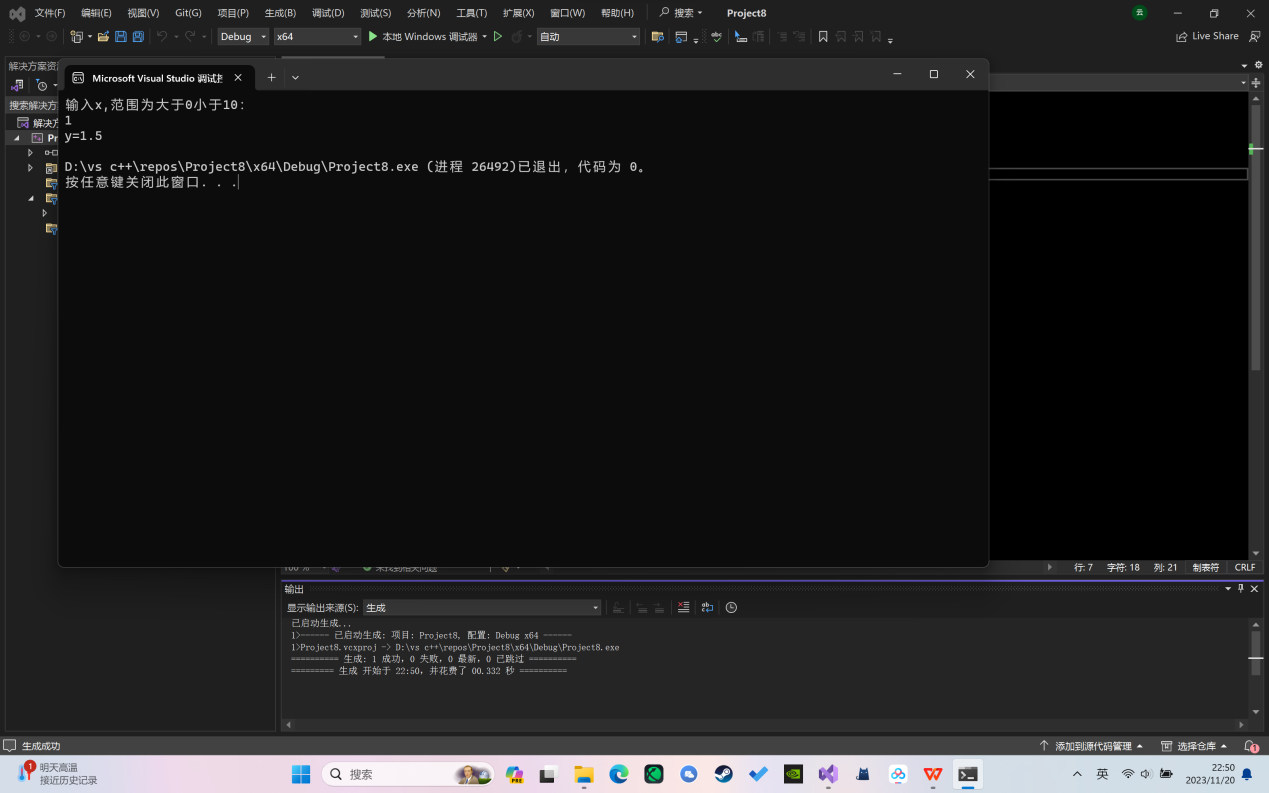
cout << "你输入的数字超出范围了！" << endl;

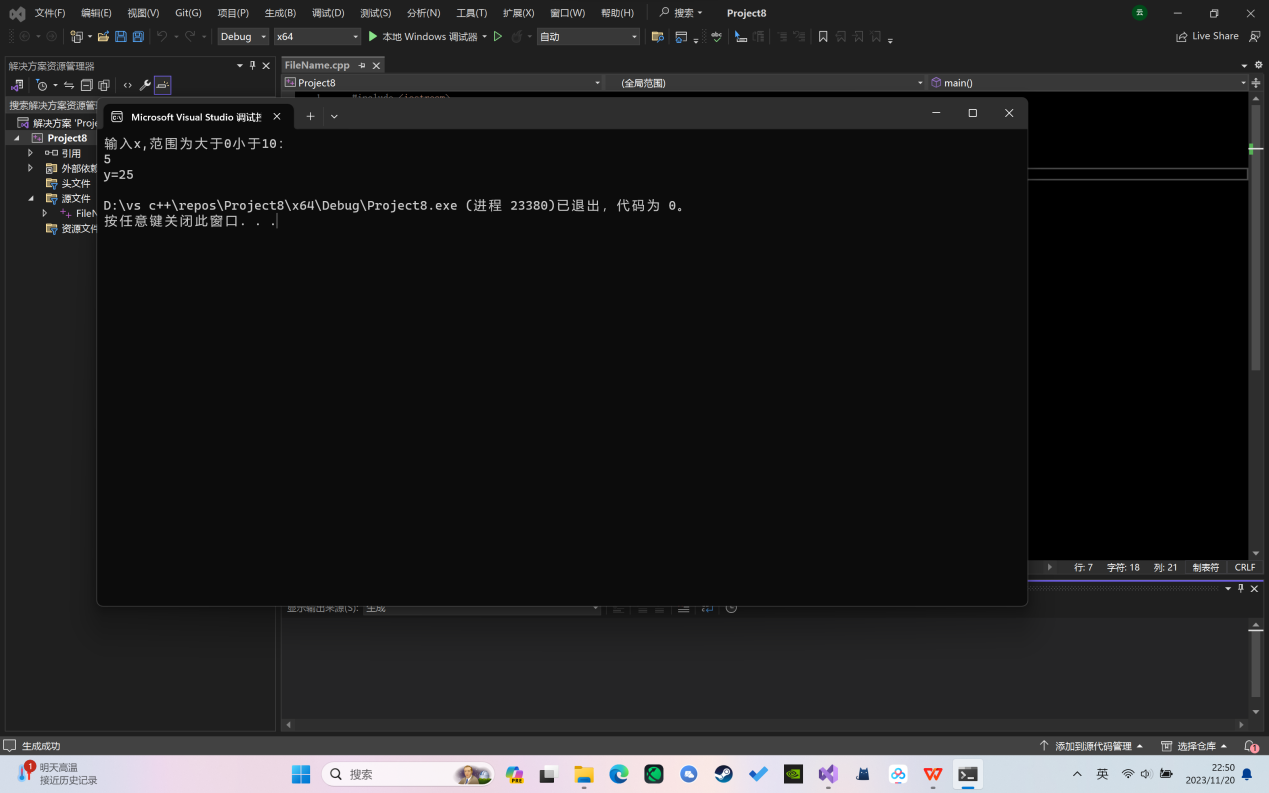
}

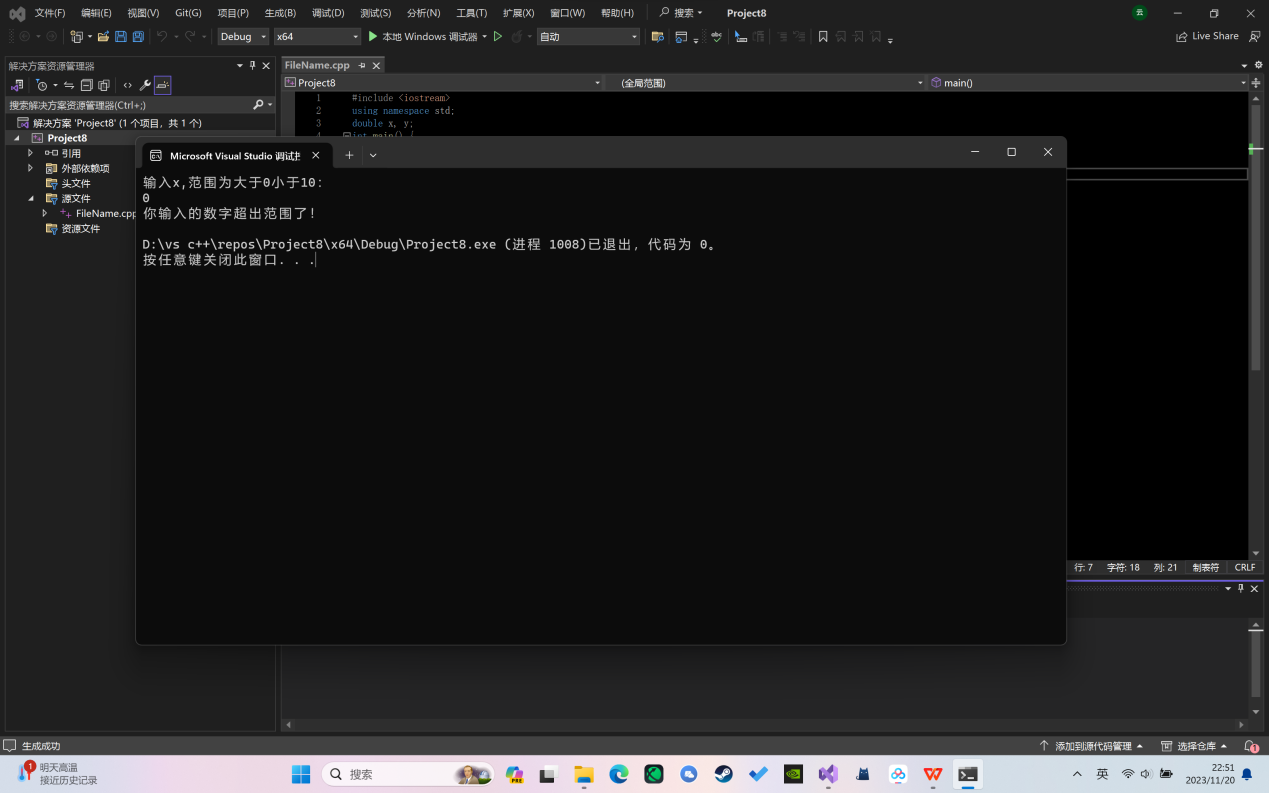
return 0;

}









3.#include <iostream>

using namespace std;

int main() {

double a, b, c,C;

cout << "请输入三角形的三条边：" << endl;

cin >> a >> b >> c;

if (a + b > c && b + c > a && c + a > b) {

C = a + b + c;

if (a == b || a == c || b == c) {

cout << "这是一个等腰三角形\n它的周长为" << C << endl;

}

else {

cout << "这不是一个等腰三角形\n它的周长为" << C << endl;

}

return 0;

}

else {

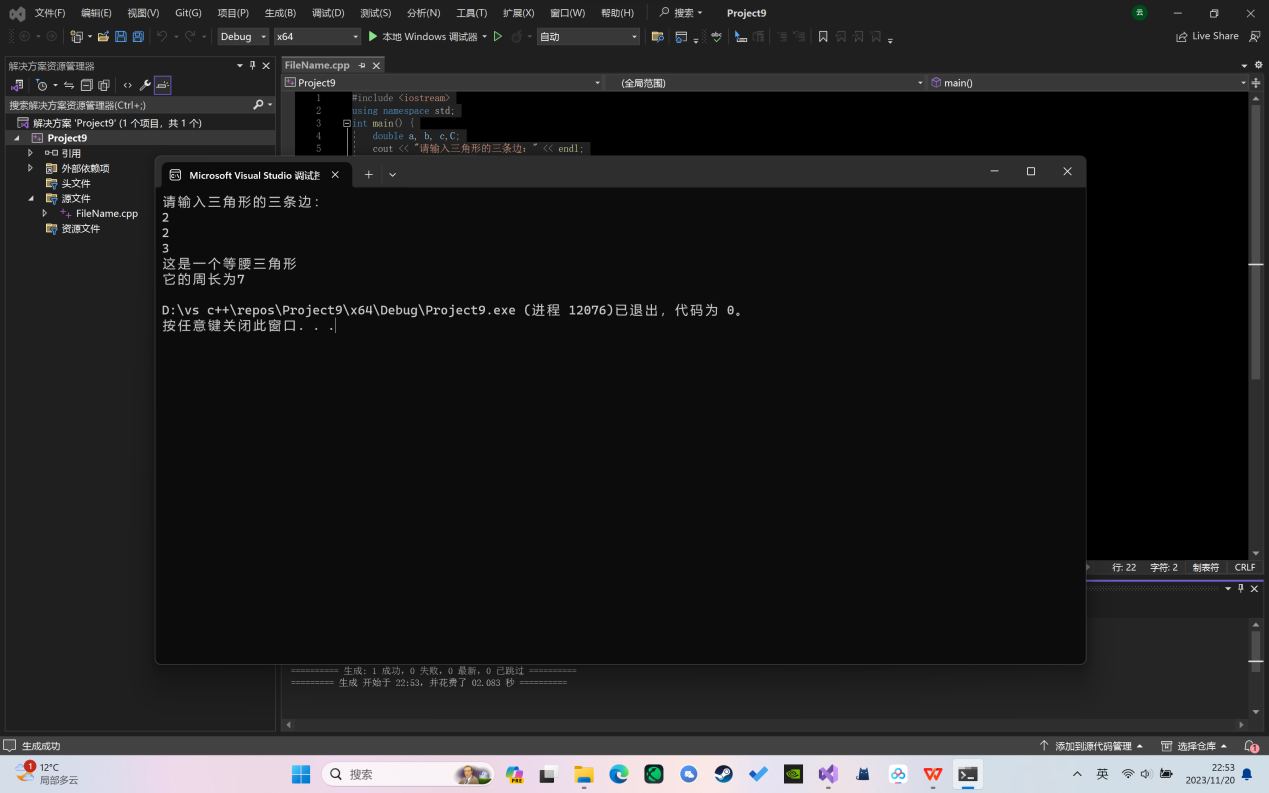
cout << "这不是一个三角形！" << endl;

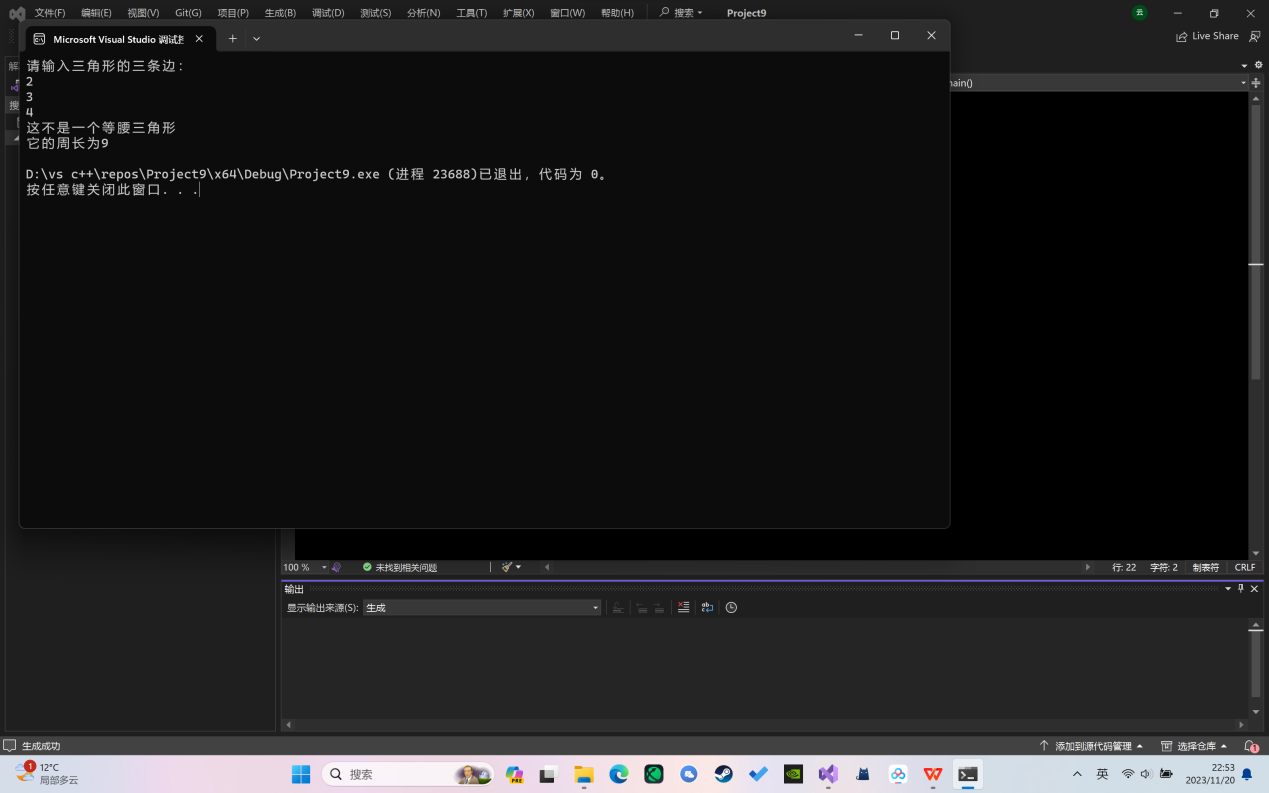
return 0;

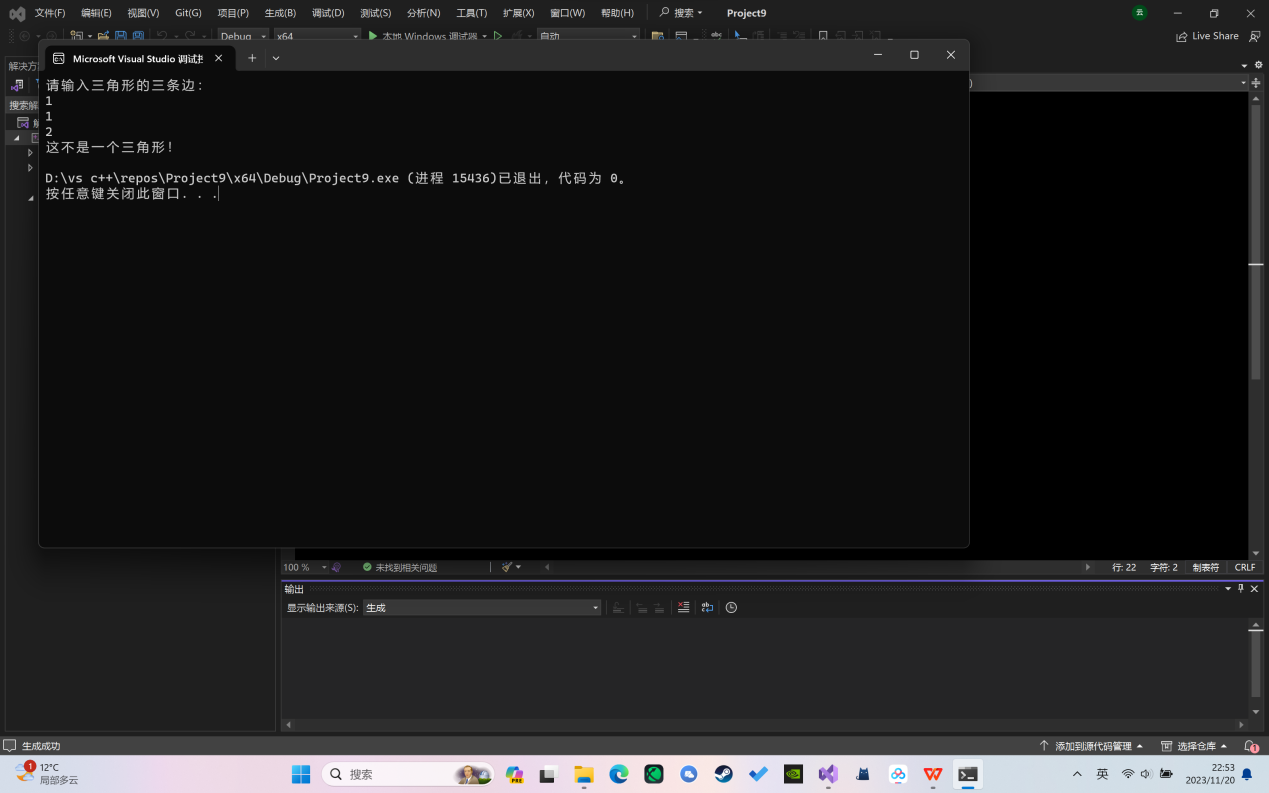
}

return 0;

}







4.#include <iostream>

using namespace std;

int a, b;

double c;

int main() {

char fuhao;

cout << "要进行哪种运算？" << endl;

cin >> fuhao;

cout << "请输入数字:" << endl;

cin >> a >> b;

if (fuhao != '+' && fuhao != '-' && fuhao != '\*' && fuhao != '/' && fuhao != '%') {

cout << "你输入不是正常的数学运算符！" << endl;

return 0;

}

else if (fuhao == '+'){

c = a + b;

cout << "结果是:" << c << endl;

}

else if (fuhao == '-') {

c = a - b;

cout << "结果是:" << c << endl;

}

else if (fuhao == '\*') {

c = a \* b;

cout << "结果是:" << c << endl;

}

else if (fuhao == '/') {

if (b != 0) {

c = a / b;

cout << "结果是:" << c << endl;

}

else {

cout << "除数不可以等于0！" << endl;

}

}

else if (fuhao == '%') {

if (b != 0) {

c = a % b;

cout << "结果是:" << c << endl;

}

else {

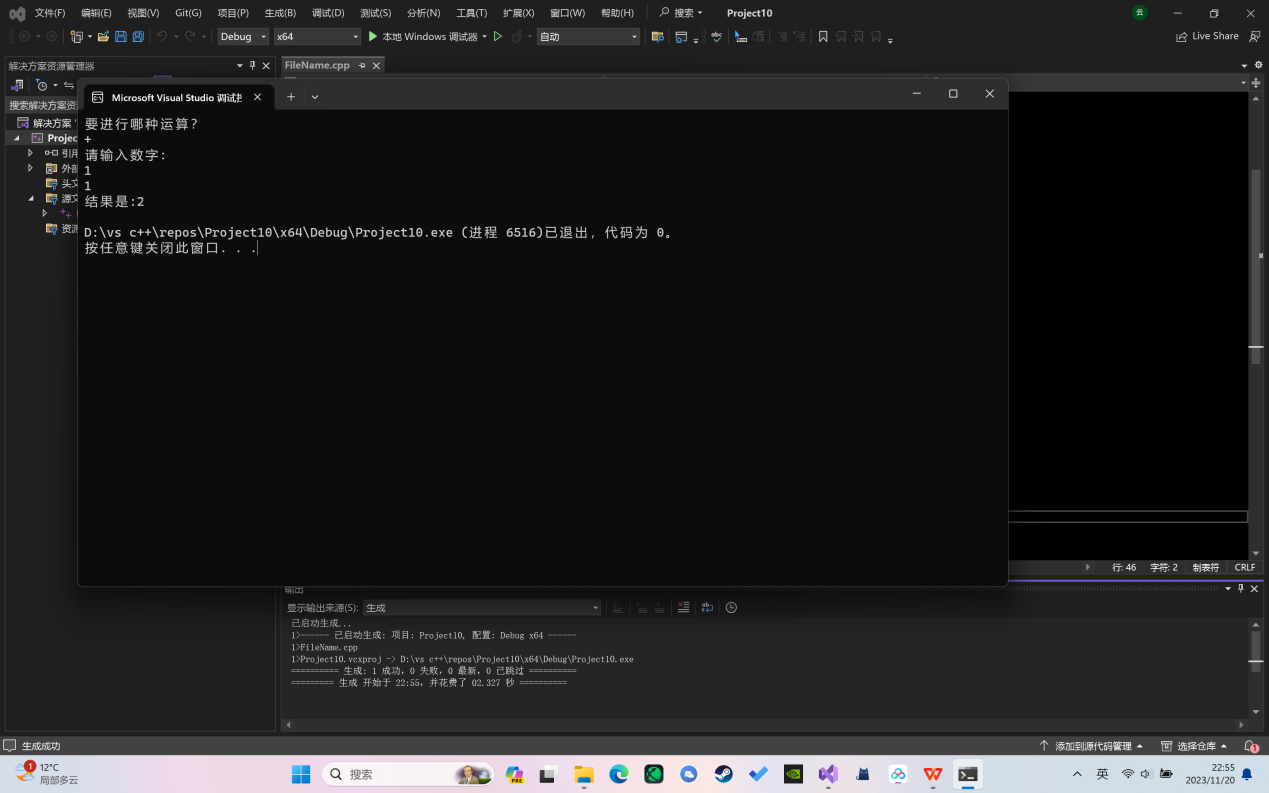
cout << "除数不可以等于0！" << endl;

}

}

return 0;

}



（-、\*、/以此类推，由于篇幅有限，避免冗长，源代码在上方，程序已发至github）

5.#include <iostream>

#include<stdio.h>

using namespace std;

int main() {

char sen[100];

int da = 0, xiao = 0, shu = 0, kong = 0, qita = 0;

cout << "请输入一段不超过100字的字符：" << endl;

scanf\_s("%[^\n]", &sen, sizeof(sen));

for (int i = 0; i < sizeof(sen); i++) {

int a = static\_cast<int>(sen[i]);

if (a > 64 && a < 91) {

da++;

}

else if (a > 96 && a < 123) {

xiao++;

}

else if (a > 47 && a < 58) {

shu++;

}

else if (a == 32) {

kong++;

}

else {

qita++;

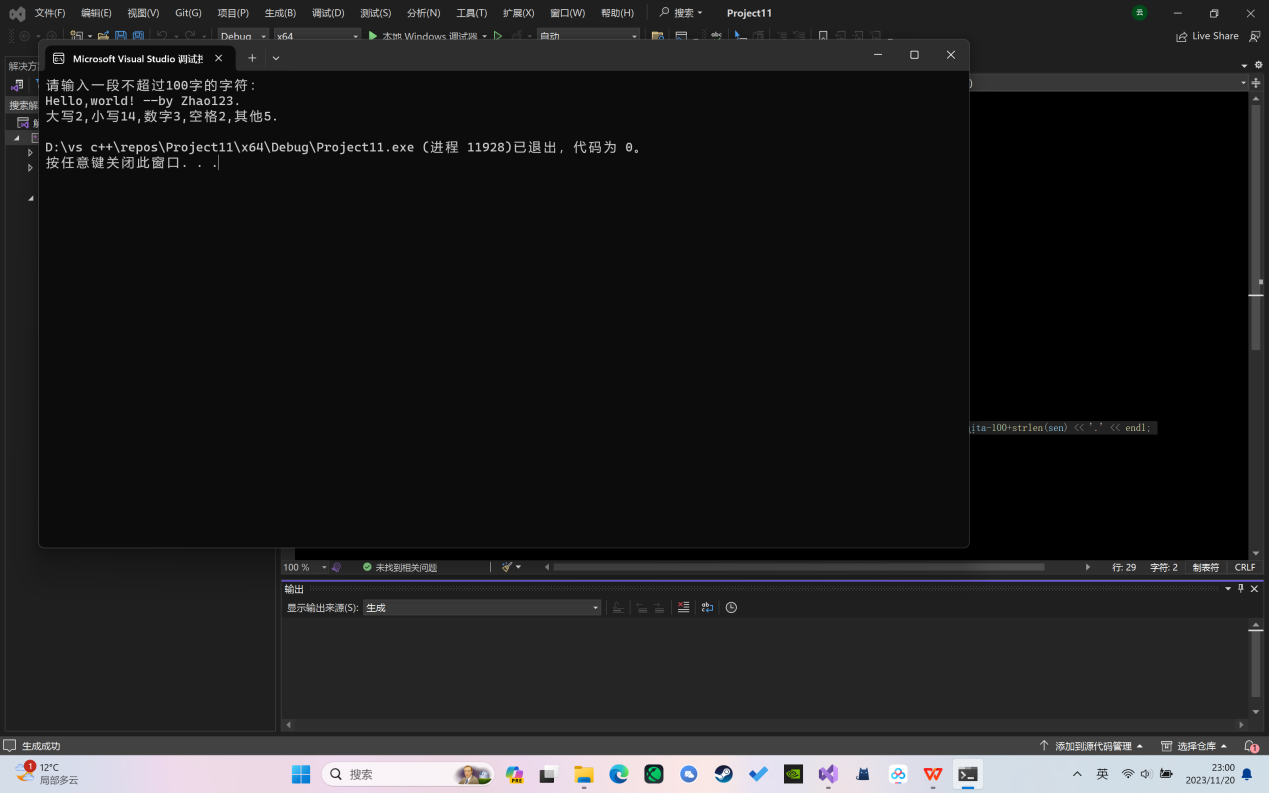
}

}

cout << "大写" << da << ',' << "小写" << xiao << ',' << "数字" << shu << ',' << "空格" << kong << ',' << "其他" << qita-100+strlen(sen) << '.' << endl;

return 0;

}



6.#include<iostream>

using namespace std;

int main() {

int a, b;

cout << "请输入两个整数:" << endl;

cin >> a >> b;

if (a < b) {

int t,s;

t = a;

s = b;

b = t;

a = s;

}

int max,min;

for (max = b; a % max != 0 || b % max != 0; max--) {

}

cout << "a，b的最大公约数为：" << max << endl;

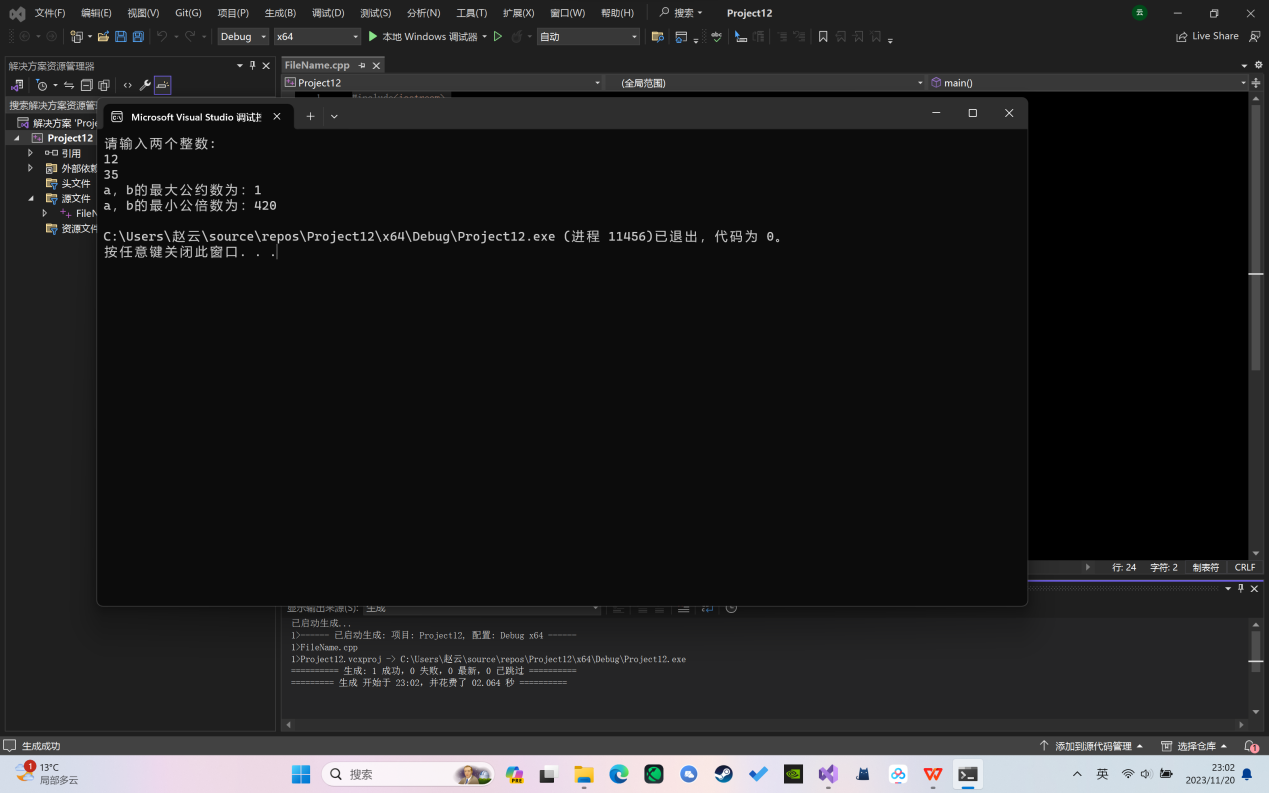
for (min = a; min % a != 0 || min % b != 0; min++) {

}

cout << "a，b的最小公倍数为：" << min << endl;

return 0;

}



7.#include<iostream>

using namespace std;

int main() {

string xing;

int i ;

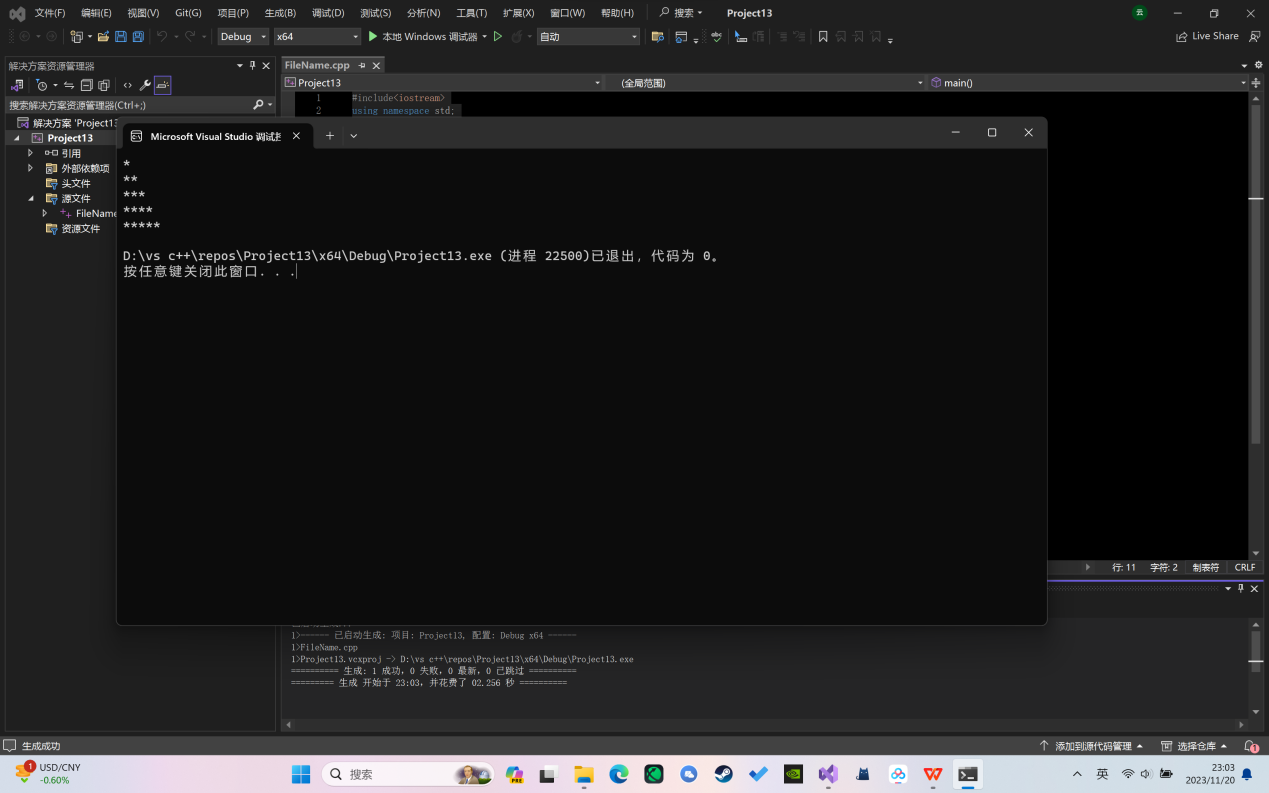
for (i = 0; i < 5; i++) {

xing += '\*';

cout << xing << endl;

}

}



1. （1）原题

#include<iostream>

#include<cmath>

#include<iomanip>

using namespace std;

int main() {

cout << "请输入一个数：" << endl;

double a;

cin >> a;

double m=a,n;

do {

n = m;

m = 0.5 \* (n + a / n);

} while (fabs(m - n) >= 1e-5);

cout <<"这个数的平方根约为："<< m << endl;

}

1. 改题（处理负数，引入虚数单位i）

#include<iostream>

#include<cmath>

using namespace std;

int main() {

cout << "请输入一个数：" << endl;

double a;

cin >> a;

double m = a, n;

if (a > 0) {

do {

n = m;

m = 0.5 \* (n + a / n);

} while (fabs(m - n) >= 1e-5);

cout << "这个数的平方根约为：" << m << endl;

}

else if (a < 0) {

n = -a;

do {

n = m;

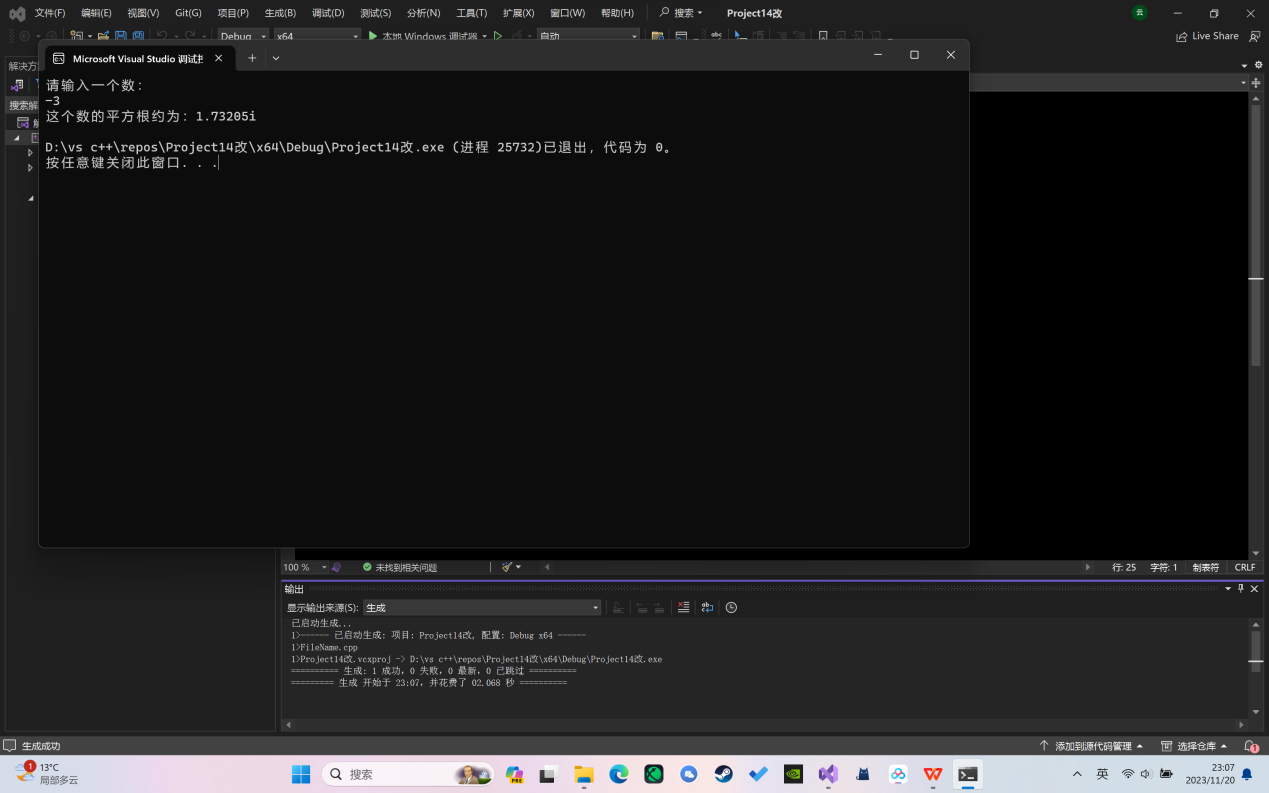
m = 0.5 \* (n + ( - a) / n);

} while (fabs(m - n) >= 1e-5);

cout << "这个数的平方根约为：" << fabs(m) << 'i' << endl;

}

}



1. 可以改为-10次方，但是没有意义，因为只保留小数点后5位，除非使用setprecision，否则

没有必要。

9.#include <iostream>

using namespace std;

int sum(int apple,int day) {

if (apple > 50) {

return apple;

}

else {

return apple + sum(apple\*2, day + 1);

}

}

int main() {

int tian=1;

for (int a = 2; a < 50;a\*=2) {

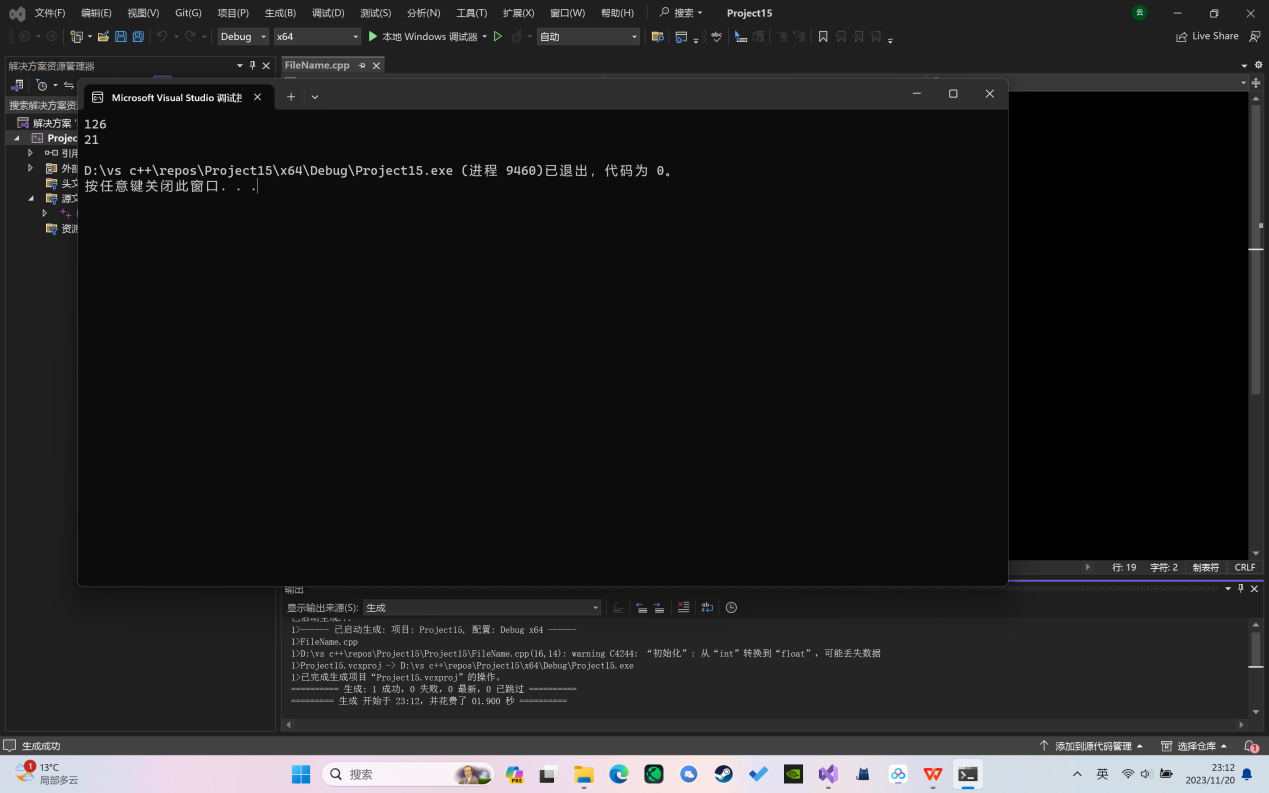
tian++;

}

float averge=sum(2,0)/tian;

cout <<sum(2,0)<<'\n'<< averge << endl;

}



**四、遇到的问题与解决方法**

**1.在判断输入是否为大写、小写字母时，利用了字符的ascii码进行判断；**

**2.第二题的if判断有些冗长，暂时没有想出好的优化方案；**

**3.在解决第三题时查找了有关scanf函数的相关资料，学会了如何正常的输入空格。但是编写代码的途中，原本最后“qita”后面的句点是中文句号，但却始终显示41379（应该是中文句号的16进制的ASCII码转化成了10进制），导致我花了将近一下午时间来检查为什么其他字符的值这么大，最后才发现是中文句号的问题\（^\_^）/;**

**4.倒数第二题在使用迭代法计算平方根时，引用了fabs（）取绝对值函数和虚数单位i。一开始用的是for循环，但是for循环是在循环开始前检验循环条件，这就导致循环只能进行一次，所以最后使用了do……while循环。**

**五、体会**

**提升了自己的逻辑思维，缜密思考的能力；**

**获得了细心检查的教训。**