**计算机程序设计基础（C++)**

**实验报告**

专业班级： 软件工程2402班

学 号： 8209240205

姓 名： 宁帅铭

**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;//此处应进行k的初始化

cin >> k;//此处应给给k赋值

int i = k + 1;

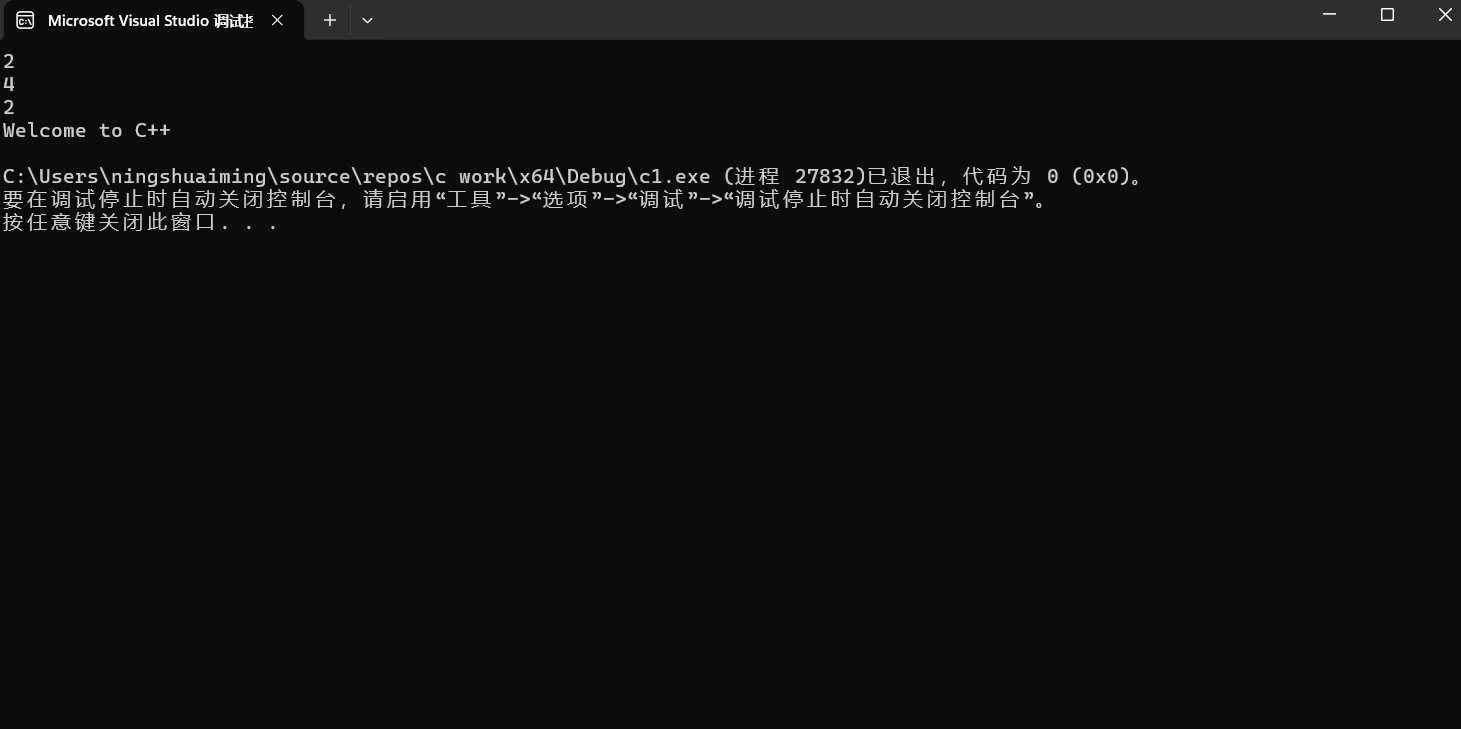
cout << ++i << endl;//此处若为i++，输出的仍为原来的i值。若想输出数值加1后的i值此处应为++i

i = 1;//此处不能加int,否则发生i的重定义

cout << ++i << endl;//此处若为i++，输出的仍为原来的i值，即1。若想输出数值加1后的i值，即2，此处应为++i

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

return 0;

}

2.

#include <iostream>

using namespace std;

int main()

{

const double p = 3.14;//定义圆周率

double a, b, v, s;//分别定义底面半径、锥高、体积、底面面积

cout << "圆周率为p" << endl;//提前声明本程序中圆周率符号为p

cout << "请输入圆锥底面半径" << endl;

cin >> a;//赋值

cout << "请输入锥高" << endl;

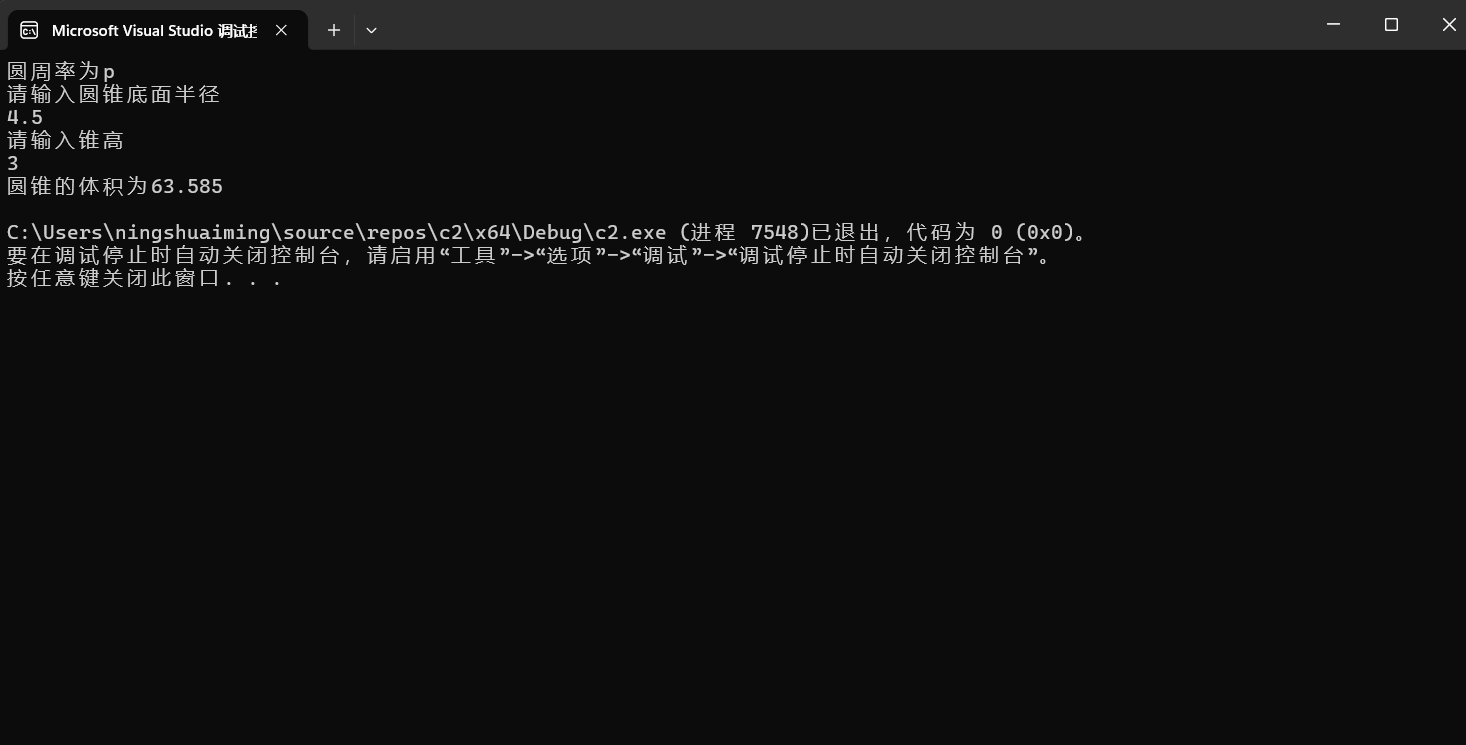
cin >> b;//赋值

s = p \* a \* a;//计算底面面积

v = b \* s / 3;//计算体积

cout << "圆锥的体积为" << v << endl;

return 0;

}

3.

#include <iostream>

using namespace std;

int main()

{

cout << "char lenth:" << sizeof(char) << endl;//输出char类型的的长度

cout << "int lenth:" << sizeof(int) << endl;//输出int类型的的长度

cout << "float lenth:" << sizeof(float) << endl;//输出float类型的的长度

cout << "double lenth:" << sizeof(double) << endl;//输出double类型的的长度

cout << "short lenth:" << sizeof(short) << endl;//输出short类型的的长度

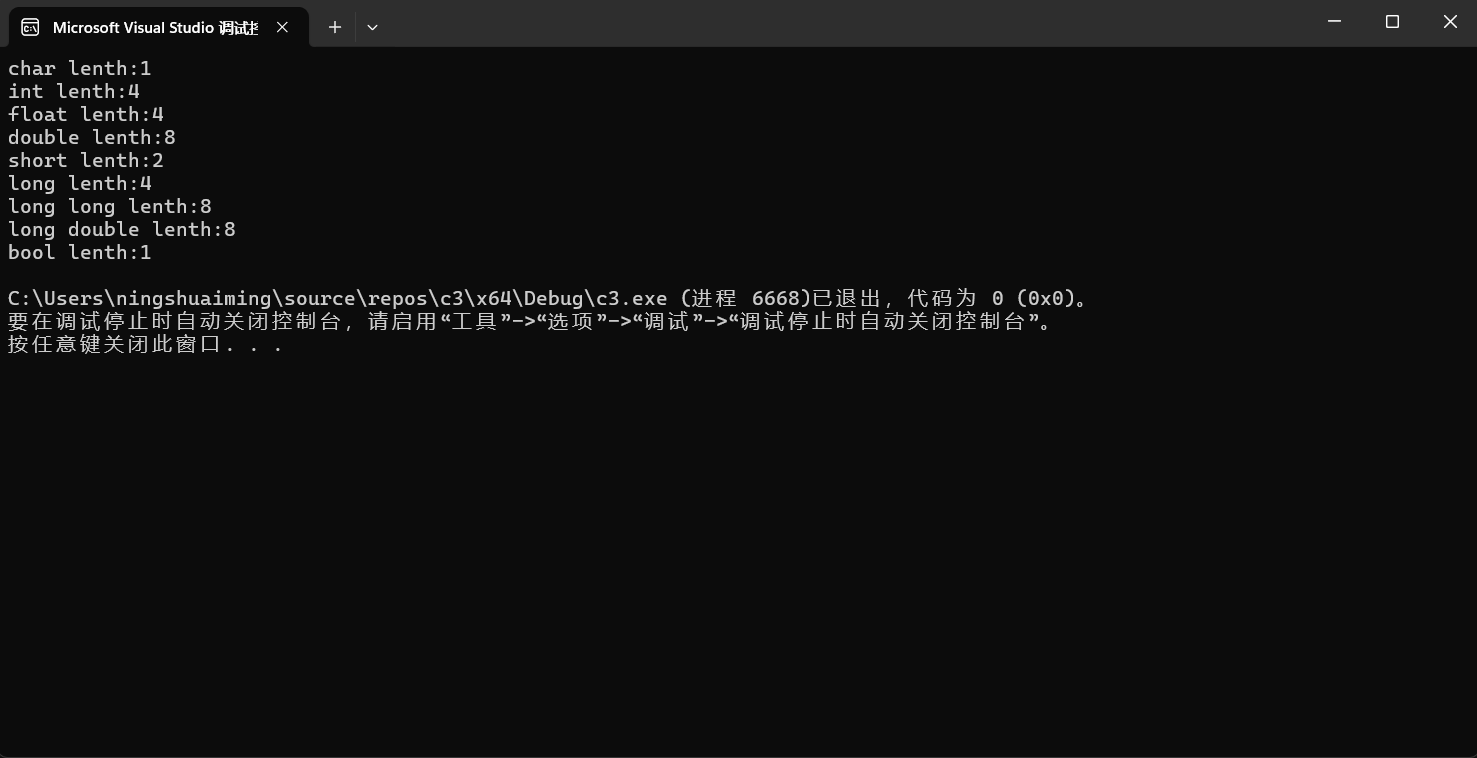
cout << "long lenth:" << sizeof(long) << endl;//输出long类型的的长度

cout << "long long lenth:" << sizeof(long long) << endl;//输出long long类型的的长度

cout << "long double lenth:" << sizeof(long double) << endl;//输出long double类型的的长度

cout << "bool lenth:" << sizeof(bool) << endl;//输出bool类型的的长度

return 0;

}

4.

#include <iostream>

#include <iomanip>

using namespace std;

int main()

{

unsigned int testUnint = 65534;//定义常变量

cout << "output in unsigned int type:" << testUnint << endl; //unsigned int类型输出

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

cout << "output in short type:" << static\_cast<short>(testUnint) << endl;//为什么结果为-2?:因为short 类型的取值范围为-32768~32767，65534超出32767的大小会重新从-32768开始计算，故输出结果为-2

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

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

cout << "output in double type:" << setprecision(4) << static\_cast<double>(testUnint) << endl;//double类型输出，整数位加小数位共保留4位，科学计数法格式

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;//8进制输出

system("pause");

return 0;

}

5.

#include <iostream>

#include <iomanip>

using namespace std;

int main()

{

double a, b;//定义double类型变量

cout << "请输入华氏温度:" << endl;

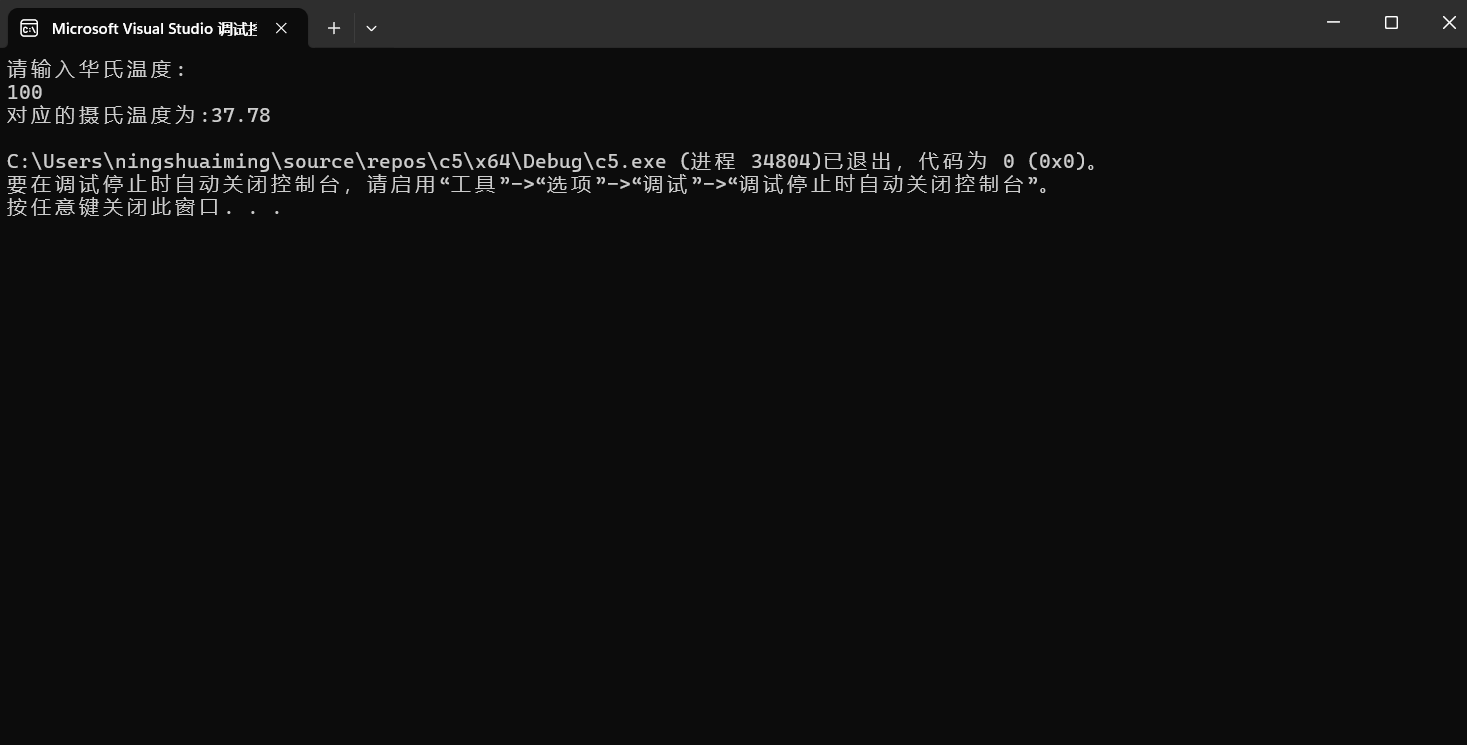
cin >> a;//为a赋值

b = (a - 32) / 1.8;//将华氏温度值转换为摄氏温度并赋值给b

cout << "对应的摄氏温度为:" << fixed << setprecision(2) << b << endl;//将b保留两位小数输出

return 0;

}



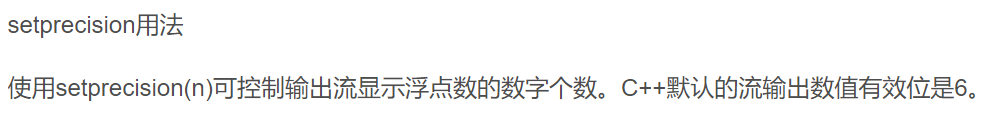
**四、遇到的问题与解决方法**

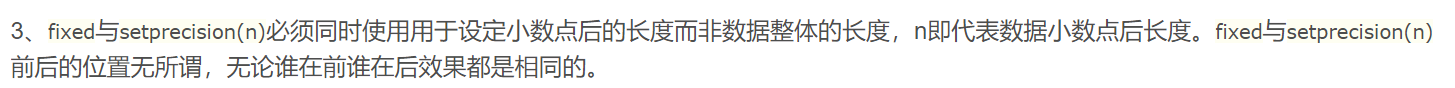
**1.不知道如何保留两位小数**

在编辑华氏度转摄氏度的程序时，体重要求结果保留两位小数。

 首先考虑通过setprecision设置显示浮点型数据的位数。

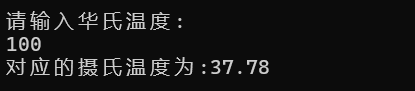
在设置显示位数后，运行程序，发现运行结果不佳。

 随后查阅相关资料，想起单使用setprecision控制的是总的显示位数，不控制小数点后的显示位数

 再次查询相关资料，发现要想保留小数点两位，需要fixed与setprecision结合使用。

遂在代码中加入fixed。





发现代码可以正常运行，两位小数可以正常保留，代码修改成功。

**五、体会**

1.要熟记c++语法要求，减少语法错误

基础知识中的语法规则就像道路上的交通标识，遵守它们才能保证程序“行驶”顺畅。对 C++ 语法的准确记忆能大大减少语法错误的出现频率。比如，知道如何正确声明变量类型、使用标点符号（如分号的正确位置）等。在实验中，我发现很多看似棘手的编译错误，往往是由于对基础知识的疏忽，比如忘记了在一句末添加分号，计算式中运算符的等级关系利用错误等。一旦对这些基础语法规则有深刻的印象，就可以在编写代码时下意识地避免这些错误，节省大量的调试时间。

2.要注意题目要求，按需编程

在编程过程中，要按需编程，如果违反需求，那么编程将是无效的。例如实验5，题目要求将华氏温度转换为摄氏温度，如果疏忽大意，错误编程为摄氏温度转华氏温度，违反了题目需求，将导致此次编程无意义。故应注意题目要求。

**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()

{

int a;

char ch, ch2;//定义字符型变量

cout << "请输入一个字符" << endl;

cin >> ch;//为字符型变量ch赋值

a = int(ch);//将ch强制转换成整形变量，并赋值给a

if (a >= 97 && a <= 122)//判定a的值是否在小写字母对应的ascII码值范围内，若在，则执行下述程序

{

a = a - 32;//因为每个小写字母总比它对应的大写字母的ascII码值大32，故通过a=a-32将a的值重赋为所输小写字母对应的大写字母的ascII码值

ch2 = char(a);//将a转换成字符型变量并赋值ch2,此时ch2即为对应的大写字母

cout << "你输入的为小写字母，其大写为" << ch2 << endl;//输出

}

else//否则a不在上述范围内，即输入的字符不是小写字母

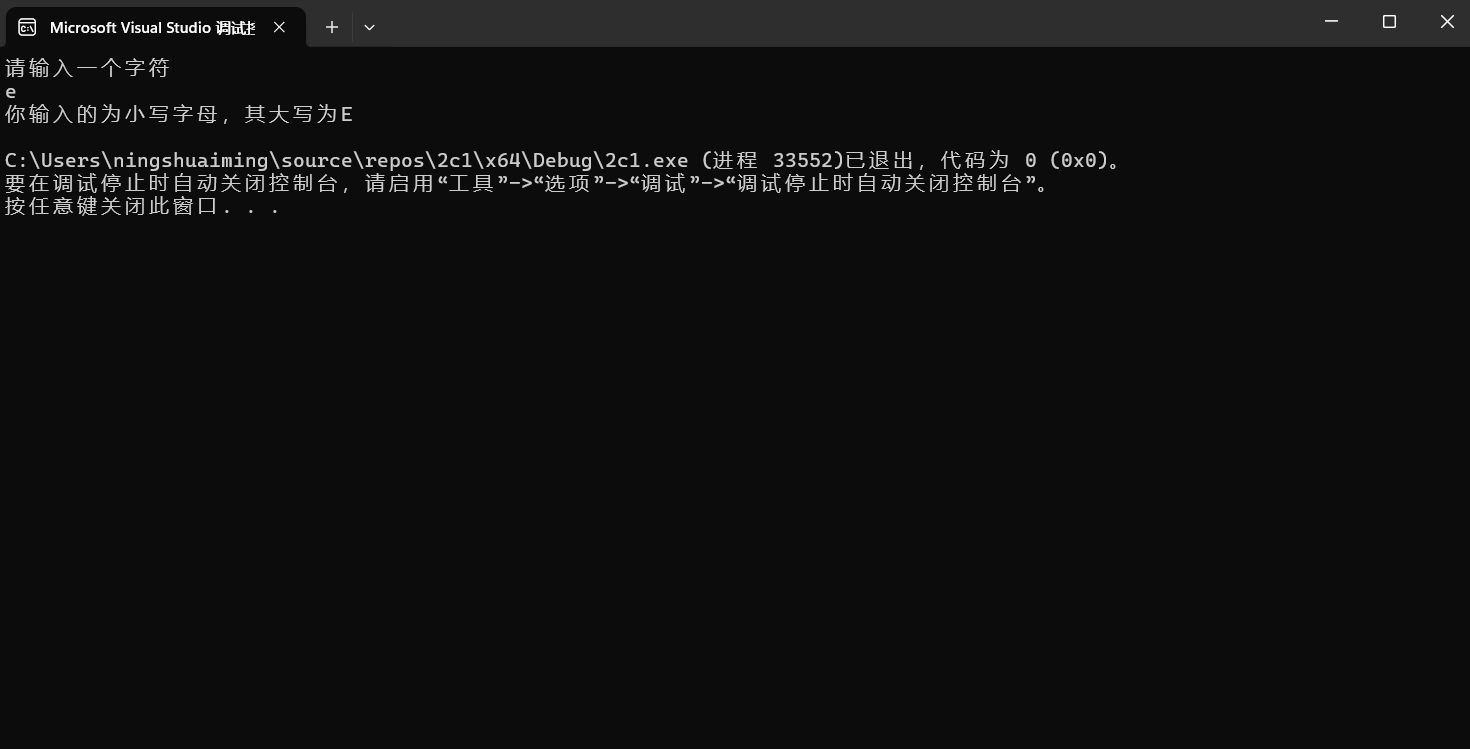
{

cout << "你输入的字符非小写字母，其ascII码为" << a << endl;//声明输入的字符非小写字母，并输出其ascII码值

}

return 0;

}



2.

#include <iostream>

using namespace std;

int main()

{

double x, y;//定义双精度浮点数x,y

cout << "请输入一个大于0小于10的数x" << endl;

cin >> x;

if (x > 0 && x < 1)//第一个判定条件

{

y = 3 - (2 \* x);

cout << "y的值为" << y << endl;

}

else if (x >= 1 && x < 5)//第二个判定条件

{

y = 2 / (4 \* x) + 1;

cout << "y的值为" << y << endl;

}

else if (x >= 5 && x < 10)//第三个判定条件

{

y = x \* x;

cout << "y的值为" << y << endl;

}

else//第四个判定条件

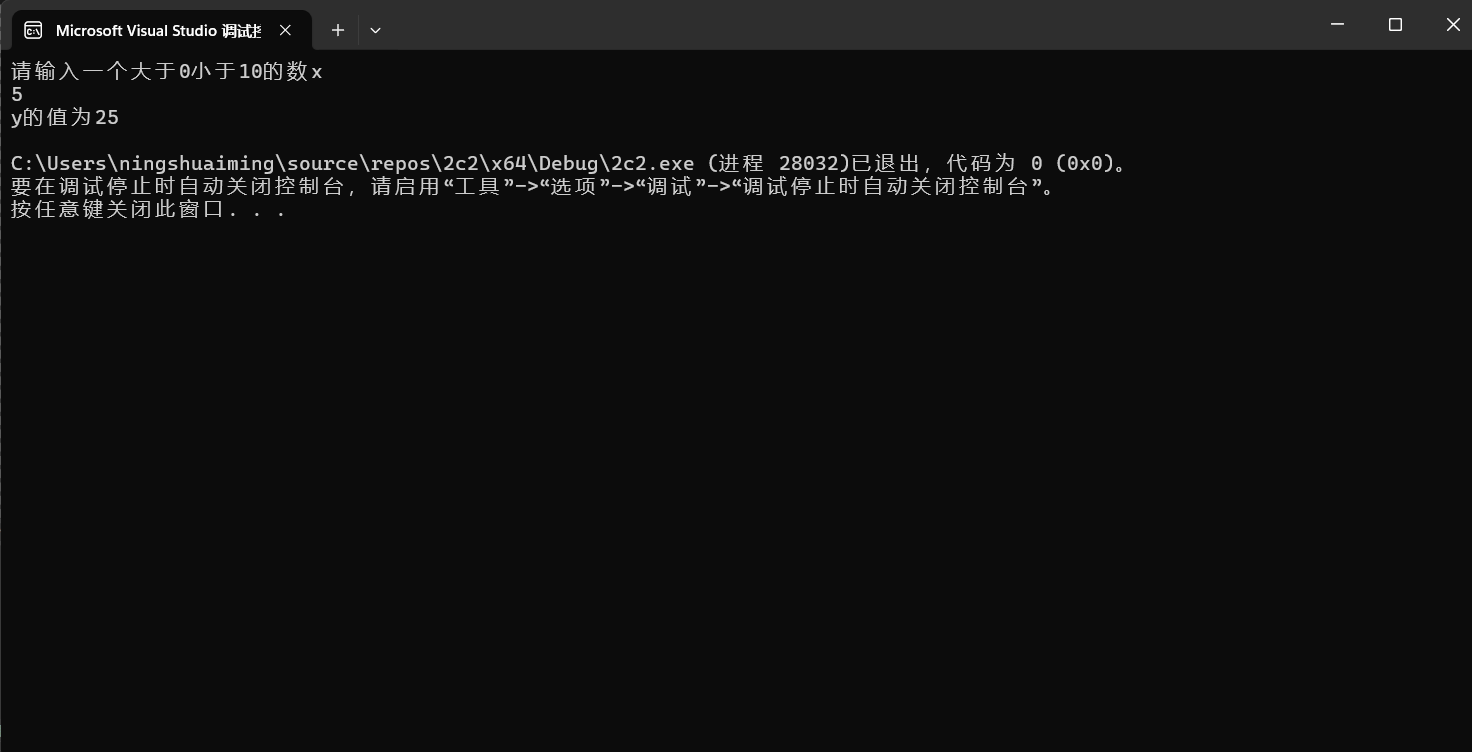
{

cout << "你输入的数不在规定范围中" << endl;

}

return 0;

}



3.

#include <iostream>

using namespace std;

int main()

{

double a, b, c, d;//定义双精度浮点数变量a,b,c,d

cout << "请输入三角形的三边长" << endl;

cin >> a;

cin >> b;

cin >> c;//输入三角形的三边长并分别赋值给a,b,c

if (a + b > c && a + c > b)//三角形的两边之和大于第三边

{

if (a == b || a == c || b == c)//利用或运算符，判定只要有两边相等的三角形即为等腰三角形

{

d = a + b + c;//求三角形周长

cout << "该三角形是等腰三角形，其周长为" << d << endl;

}

else//若没有任何两边相等，则该三角形不是等腰三角形

{

d = a + b + c;//求三角形周长

cout << "该三角形不是等腰三角形，其周长为" << d << endl;

}

}

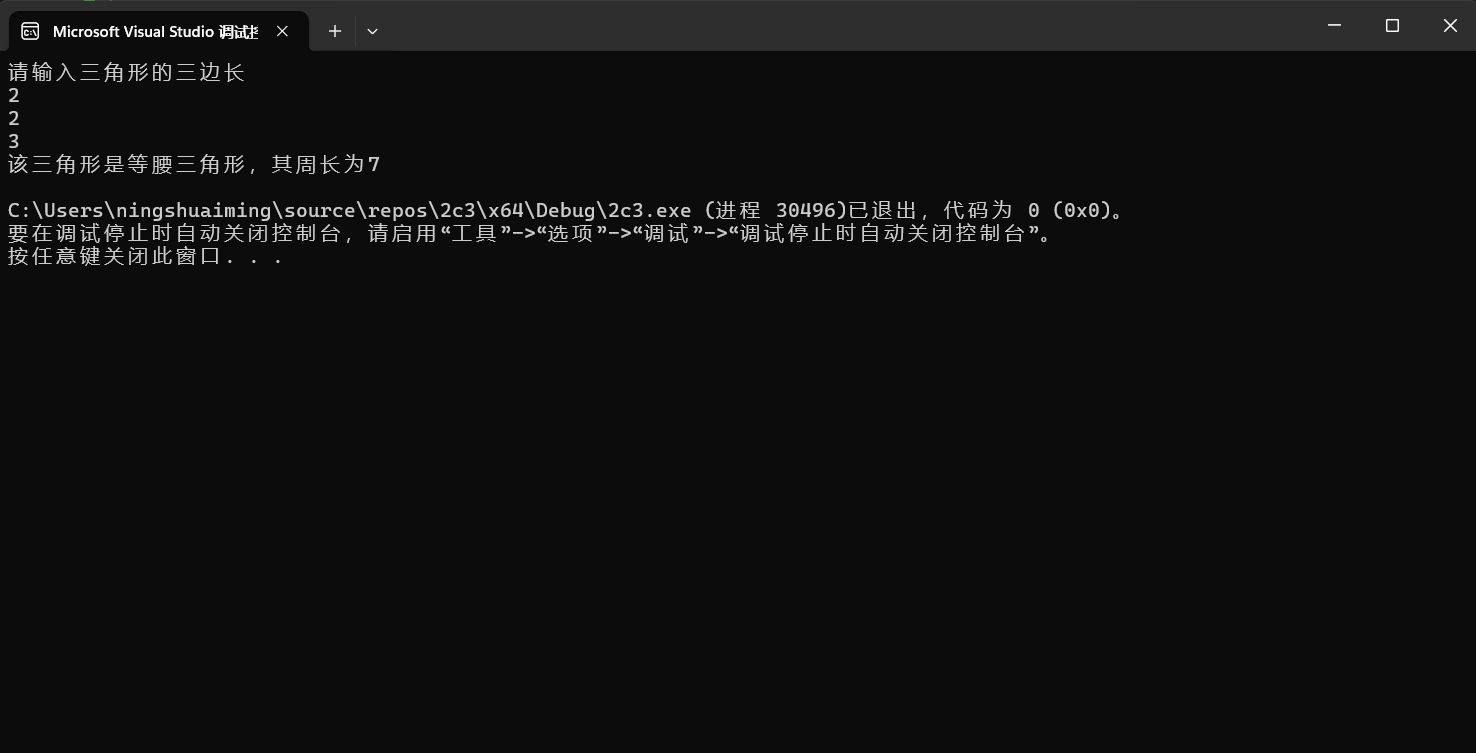
else//不满足两边之和大于第三边，则输入的的三边长无法构成三角形

{

cout << "这样的三边长无法构成三角形" << endl;

}

}



4.

#include <iostream>

using namespace std;

int main()

{

cout << "本程序可以实现+ - \* / %的运算" << endl;//首先声明程序的功能

cout << "求余运算中如果运算数出现小数，将保留整数部分" << endl;//求余运算的两个运算数都要求是整数，故要提前声明若出现小数，将只保留整数进行求余运算

double a, b, c;//定义双精度浮点数a,b,c

char f;//定义字符型变量用来赋予运算符

cout << "请输入第一个运算数" << endl;

cin >> a;//输入第一个运算数

cout << "请输入运算符" << endl;

cin >> f;//输入运算符

cout << "请输入第二个运算数" << endl;

cin >> b;//输入第二个运算数

if (f == '+')//若运算符为加号

{

c = a + b;//进行加法运算，将结果赋值给c

}

else if (f == '-')//若运算符为减号

{

c = a - b; //进行减法运算，将结果赋值给c

}

else if (f == '\*')//若运算符为乘号

{

c = a \* b;//进行乘法运算，将结果赋值给c

}

else if (f == '/')//若运算符为除号

{

c = a / b;//进行除法运算，将结果赋值给c

}

else if (f == '%')//若运算符为求余号

{

c = (int(a)) % (int(b));//进行求余运算，将结果赋值给c

}

else//若输入的运算符不在程序规定的运算符范围内

{

cout << "你输入的运算符非法，不在规定运算符行列内" << endl;//判定输入的运算符非法

}

cout << "运算结果为" << c << endl;//输出运算结果c

return 0;

}



5.

#include <iostream>

using namespace std;

int main()

{

int a;

int eng = 0, spa = 0, num = 0, oth = 0;//分别定义四个整型变量，用来统计所输入的字符中英文字母、空格、数字字符和其他字符的个数

cout << "本程序可以统计你所输入的字符中英文字母、空格、数字字符和其他字符的个数" << endl;//声明本程序的功能

while ((a = cin.get()) != 10)//利用while循环和a=cin.get()逐个读取所输入的字符串中每个字符的ascII码值并赋值给a。只要a的值不等于10，即未读取到回车字符，即可不断进行while循环

{

if ((a >= 65 && a <= 90) || (a >= 97 && a <= 122))//若a值在英文字母的ascII值范围中，说明所读取的字符为英文字母

{

++eng;//给英文字母计数变量eng加1

}

else if (a == 32)//若a的值等于空格字符的ascII值，说明所读取的字符为空格字符

{

++spa;//给空格计数变量spa加1

}

else if (a >= 48 && a <= 57)//若a值在数字字符的ascII值范围中，说明所读取的字符为数字字符

{

++num;//给数字字符技术变量spa加1

}

else//若a的值不在上述任何判定条件范围中，说明所输入的字符不属于任何上述字符种类，故属于其他字符

{

++oth;//给其他字符计数变量oth加一

}

}

cout << "你所输入的字符中英文字母有" << eng << "个，" << "空格有" << spa << "个，" << "数字字符有" << num << "个，" << "其他字符有" << oth << "个。" << endl;//输出统计后英文字母、空格、数字字符和其他字符的个数

return 0;

}



6.

#include <iostream>

using namespace std;

int main()

{

int a, b, c, d;//定义四个整形变量

cout << "请输入两个数，本程序可以计算其最大公约数和最小公倍数" << endl;//声明程序功能

cin >> a;

cin >> b;//输入两个数

if (a > b)

{

c = b;

}

else if (a == b)

{

c = a;

}

else

{

c = a;

}//取得a,b中的较小值，并赋值给c

for (; c <= a && c <= b;)//因为最大公因数一定小于等于两个数中的较小值，即小于等于c,利用for循环，令c不断减1

{

if (a % c == 0 && b % c == 0)

{

break;

}//直到a、b均能整除c,退出循环

else

{

c = c--;

}//否则继续令c减1

}

cout << "这两个数的最大公因数为" << c << endl;//最终c的值即为两个数的最大公因数

if (a > b)

{

d = a;

}

else if (a == b)

{

d = a;

}

else

{

d = b;

}//取a、b中的较大值，并赋值给d

for (; a <= d && b <= d;)//因为最小公倍数一定大于等于两个数中的较大值，即大于等于d,利用for循环，令d不断加1

{

if (d % a == 0 && d % b == 0)

{

break;

}//直到c能够整除a和b，退出循环

else

{

d = d++;

}//否则继续令d加1

}

cout << "这两个数的最小公倍数为" << d << endl;//最终的d值即为两个数的最小公倍数

return 0;

}



7.

#include <iostream>

using namespace std;

int main()

{

int a, b;

for (a = 1; a <= 5; a++)//利用外层for循环以及a的值确定每层中\*的数量，是每层的数量由1增加到5

{

for (b = 1; b <= a; b++)

{

cout << "\*";

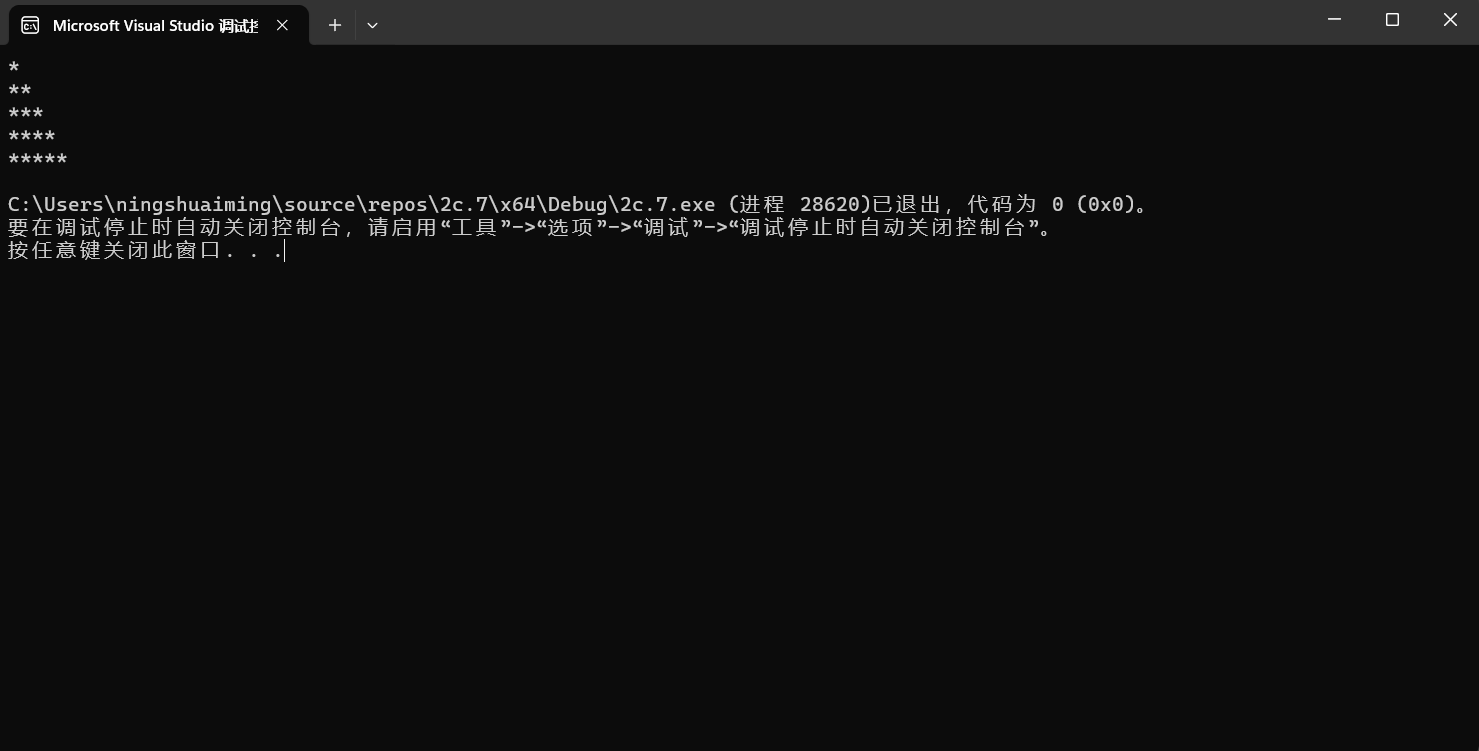
}//嵌套内层for循环,实现\*的输出，使每行\*的数量不超过a

cout << endl;//实现每层结束换行

}

return 0;

}



8.

#include <iostream>

#include <cstdlib>//引入头文件<cstdlib>

using namespace std;

int main()

{

double a, xn, xn1, t, t1;//定义双精度浮点数a, xn, xn1, t, t1

cout << "本程序可以计算你所输入的数a的平方根" << endl;//声明程序的功能

cout << "请输入一个数a" << endl;

cin >> a;//输入一个值，赋给a

if (a >= 0)//如果a的值大于等于0，则可计算平方根

{

xn = a;//令xn的值等于a

xn1 = (xn + a / xn);//利用题中所给公式，第一次算出xn1的值

t = xn1 - xn;

t1 = abs(t);//利用abs（）函数，算出t的绝对值并赋值给t1，t1即为此时的精度

for (; t1 >= 1e-5;)//根据题中要求，利用for循环，若精度仍大于1e-5,就把xn1的值赋给xn，并再次执行题中公式，并计算精度。直到精度小于1e-5,则退出循环

{

xn = xn1;

xn1 = (xn + a / xn) / 2;

t = xn1 - xn;

t1 = abs(t);

}

cout << "平方根为" << xn1 << endl;//根据题中所给信息，最终得到的xn1的值即为输入值的平方根

}

else

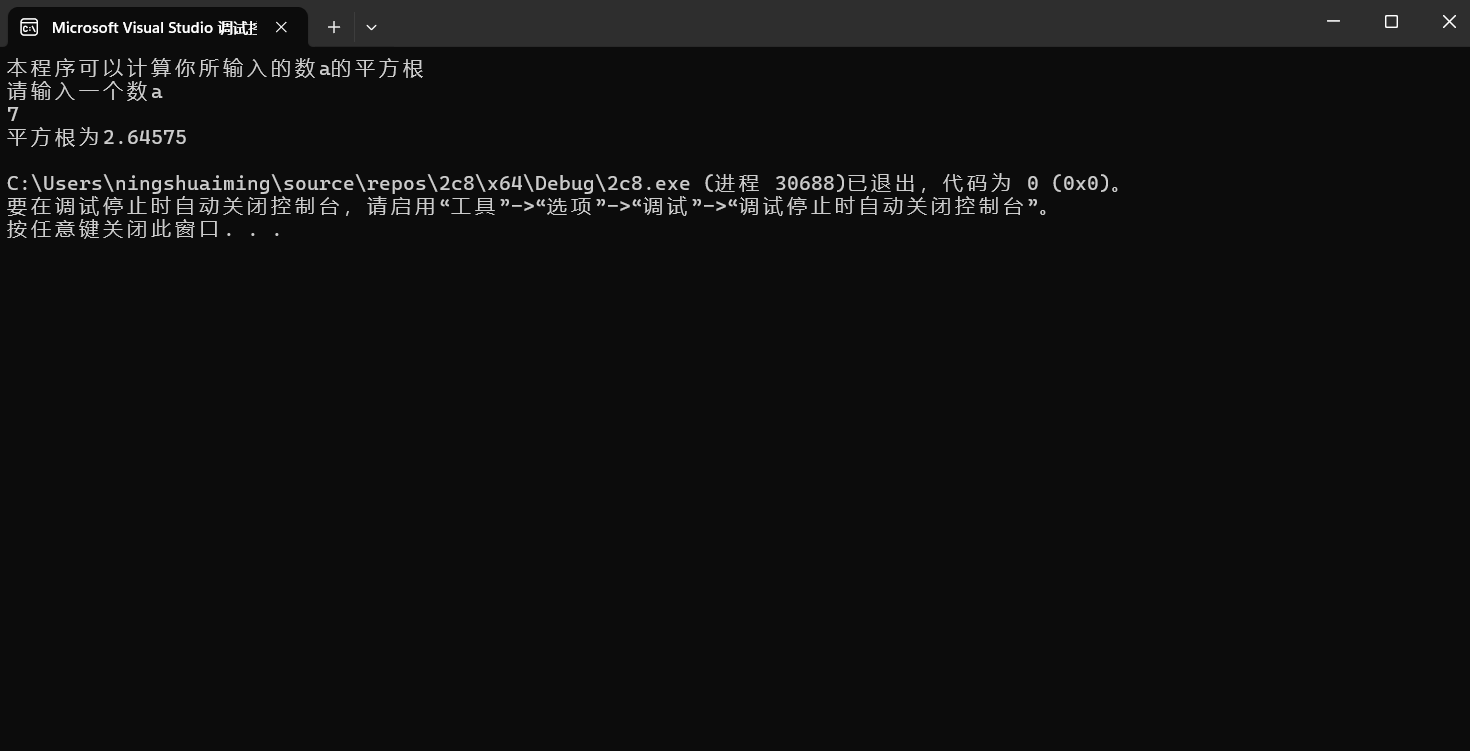
{

cout << "你所输入的数为负数，无法计算平方根" << endl;

}//若a的值为负，则无法计算平方根

return 0;

}



9.

#include <iostream>

using namespace std;

int main()

{

double i, d, m, x;//定义双精度浮点数i,d,m,x。i表示当天买的苹果数，d用于记天数，m表示直到当天共花的钱数，x表示平均花费

for (i = 2, m = 0, d = 0; i <= 100; i = 2 \* i)//利用for循环，只要苹果数不超过100，便不断求得共花的钱数，推进天数。直到购买的苹果数超过100，退出循环，不再记天数和钱数

{

m = m + 0.8 \* i;

d = d++;

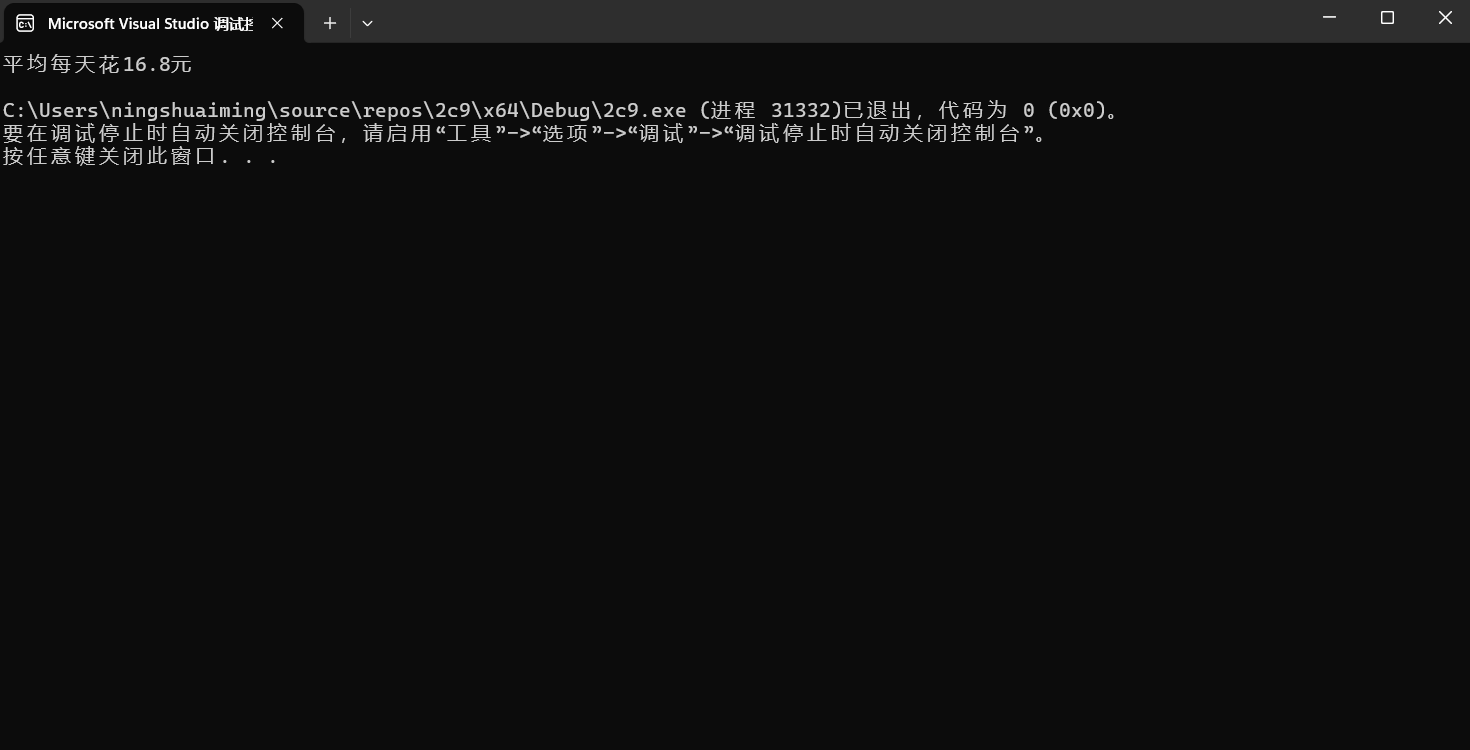
}

x = m / d;//计算平均花费并赋值给x

cout << "平均每天花" << x << "元" << endl;//输出平均花费x

return 0;

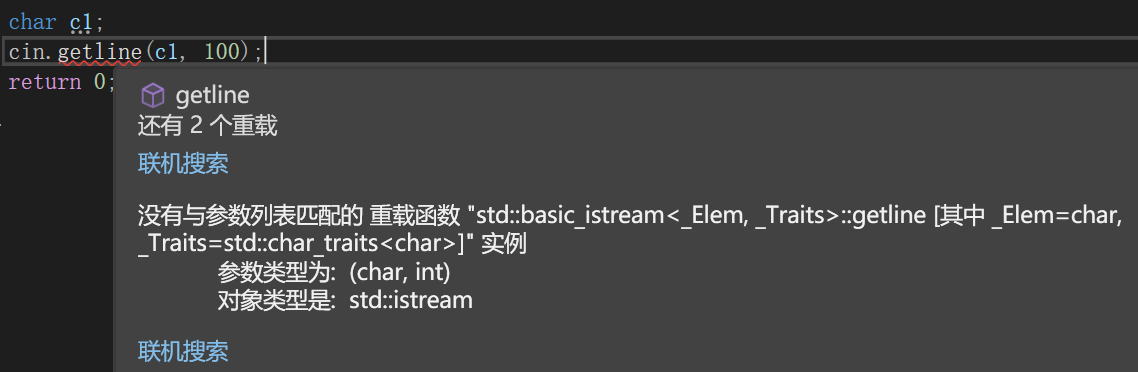
}



**四、遇到的问题与解决方法**

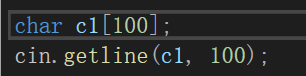
**1.不了解如何读取字符串中每一个字符的ascII码值，尤其是如何录入空格**

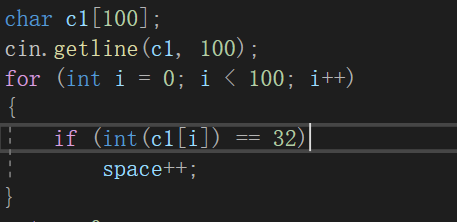
 最开始在互联网上查询相关资料，发现可以使用cin.getline()录入空格。

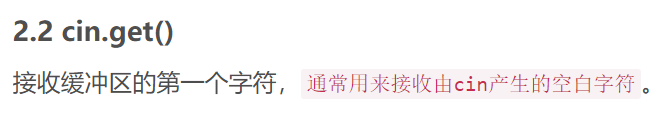
 在实际编程过程中，发现若想使用该函数，需要设置最大可接受字符量。在设置后，编译器报错。

考虑到在使用该函数式时有定义最大可接受字符量，尝试将单个字符变量改编为字符变量类型数组后再次尝试，报错消除。

考虑到定义的是数组，若想要检测每个字符，需要引入循环，尝试利用循环循环。



**** 但是考虑到在此前已经将c1写成了数组，如果在最开始录入的过程中，没有录满我们所设定的100个元素，后面的没有被录入的数组，可能会干扰到最后的运行结果，再加上我目前没有掌握设立动态数组的知识，故只能放弃这一方法。

 再次在互联网上查找其他资料，发现cin.get（）也可以实现相关功能，故决定使用cin.get（）函数。

 但是在使用cin.get()函数时，因为不是很熟悉使用方法，发现在使用cin.get()录入后，再对结果进行输出，只能获得第一个字符。

后通过进一步的查询，了解到要利用cin.get()逐个录入所输入的全部字符并分类,需要将其写入循环当中，利用while循环和a=cin.get()逐个读取所输入的字符串中每个字符的ascII码值并赋值给a。只要a的值不等于10，即未读取到回车字符，即可不断进行while循环。



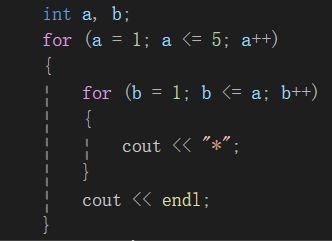
在这样写入后，程序能够对字符串中的每个字符进行判断，从而最终计算出各类型字符的数量。

**2.不熟悉如何利用嵌套循环**

题中要求利用循环输出如下图案

****

不清楚如何利用循环实现如上图案。最初只想到利用单层循环，但经反复尝试，发现无法\*数目逐行递增输出。经与同学讨论请教，了解到，可以利用嵌套循环实现该程序。

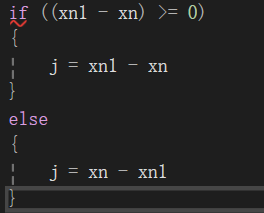


利用外层for循环以及a的值确定每层中\*的数量，是每层的数量由1增加到5，嵌套内层for循环,实现\*的输出，使每行\*的数量不超过a。这样就可以实现\*数量的逐层递增输出了。

**3.不了解如何简便计算绝对值**

题中要求计算|xn+1-xn|的值是否小于规定值，用于判断精确度。而这就要考虑如何计算绝对值。

最初考虑定义变量j代表绝对值，利用if判别进行求绝对值



但这种方法略显麻烦 ，故查询资料寻求更简便的方法。经查询，可以引用头文件<cstdlib>， 并利用abs（）函数更加容易地实现求绝对值



**五、体会**

**1.在c++学习过程中要积极自学，主动查询资料获取新知识、探索解决未知问题**

在 C++实验过程中，我深刻体会到积极自学和查询资料是打开知识宝库的关键钥匙。教材内容往往只是一个基础框架，仅依靠它所涵盖的知识远远不足以应对复杂多样的实验需求。例如，查询得知，求绝对值可以引入头文件<cstdlib>并利用函数abs( ),字符的逐个读取可以利用while循环以及cin.get( )函数，通过自主查询资料，我发现了大量的函数细节、以及最佳实践方法。这种自主探索极大地拓宽了我的知识面，使我能够从更全面的角度去理解和运用 C++，为解决实验中遇到的难题提供了更多思路。

**2.要学会灵活运用已学知识**

比如实验二第7题，我们已经学过for循环，但单纯利用常见的单层for循环是无法输出所求图形的。要巧妙、灵活地利用所学知识，在for循环中再嵌套一层for循环，实现双层要求进行输出。方可得到所需图案。由此可见，单单基础地掌握已学知识是不够的，还要灵活运用，才能更好地实现需求。