实验二 控制结构

一、实验目的与要求

1、学习与掌握逻辑运算与逻辑表达式。

2、熟练掌握 if、switch 语句的语法结构与执行过程。

3、掌握选择程序的设计方法

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：

3、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：

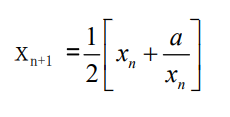
\*

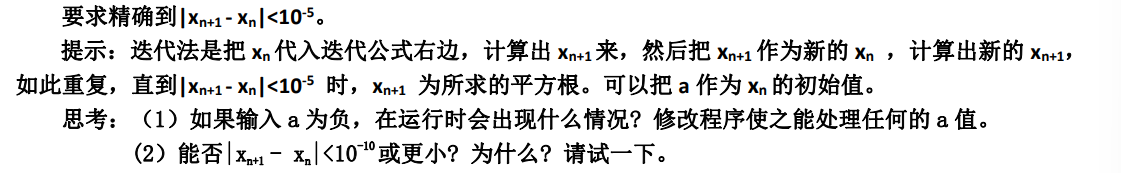
\*\*

\*\*\*

\*\*\*\*

\*\*\*\*\*

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：

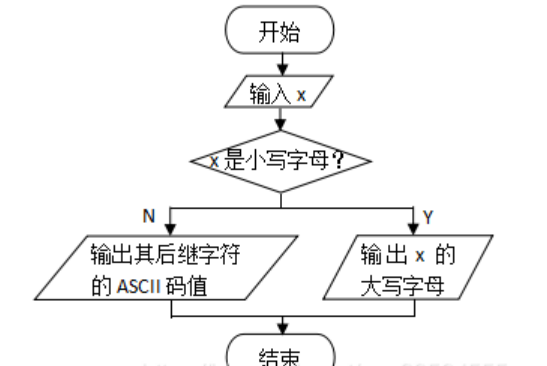


9、苹果每个 0.8 元，第一天买 2 个，第二天开始，每天买前天的 2 倍，直到购买的苹果数不超过 100的最大值，求每天平均花多少钱。

三、算法分析、程序与结果

1.

算法如流程图所示



程序：

#include<iostream>

#include<string>

using namespace std;

int main()

{

char x;

cin >> x;

if ('a' <= x&& x <= 'z')

{

x= x - 32;

cout << x << endl;

}

else

{

x = x + 1;

cout << x << endl;

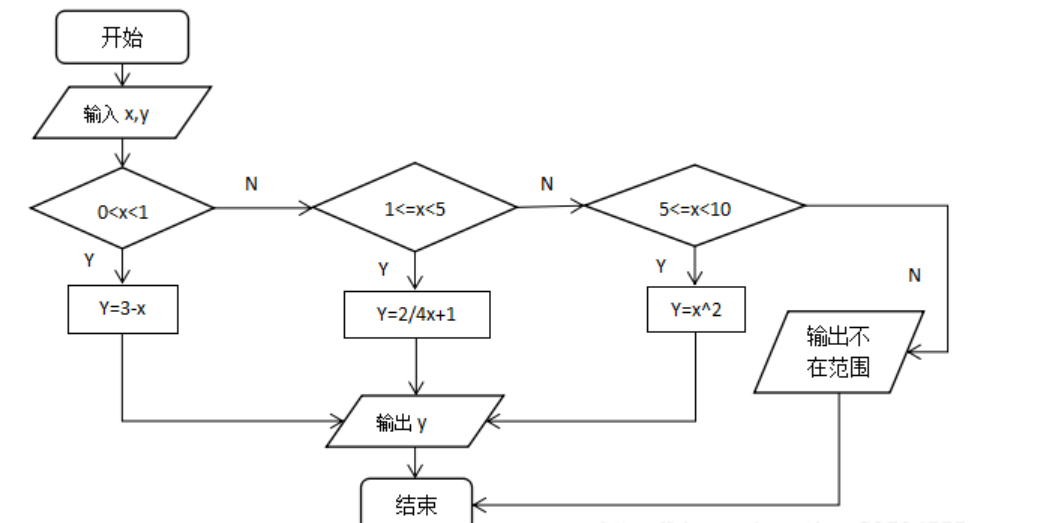
}

return 0;

}

2.

算法：



程序：

#include<iostream>

#include<string>

using namespace std;

int main()

{

double x,y;

cin >> x;

if (x < 1 && x>0)

{

y = 3 - 2 \* x;

cout << y << endl;

}

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

{

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

cout << y << endl;

}

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

{

y = x \* x;

cout << y<< endl;

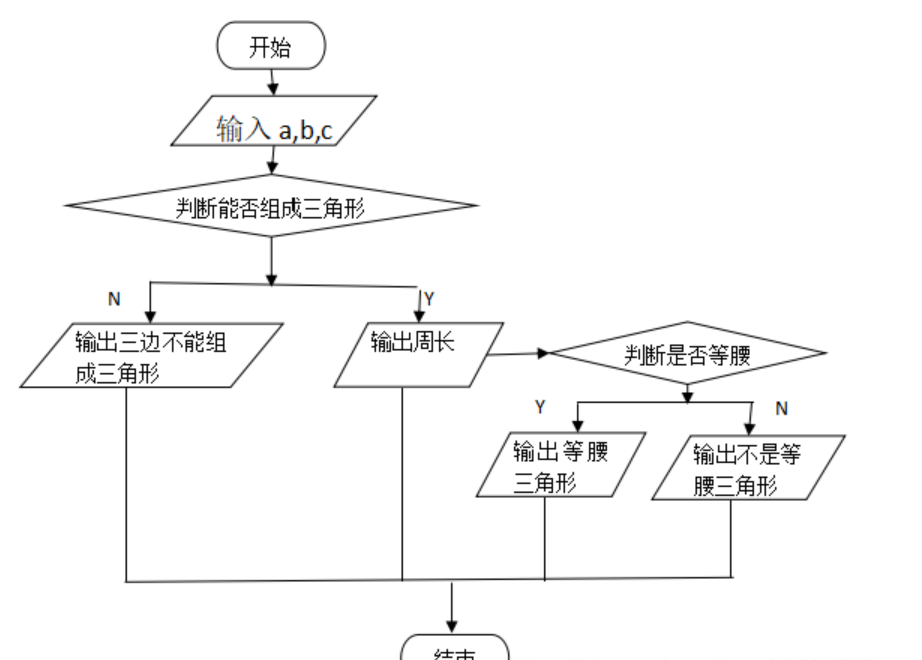
}

return 0;

}

3.

算法：



程序：

#include <iostream>

using namespace std;

int main()

{

double a , b , c , L;

cout<<"Please enter the three sides a , b , c : "<<endl;

cin>>a>>b>>c;

L=a+b+c;

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

cout<<L<<endl;

if(a!=b&&a!=c&&b!=c)

cout<<"This is not an isosceles triangle"<<endl;

else

cout<<"This is an isosceles triangle"<<endl;

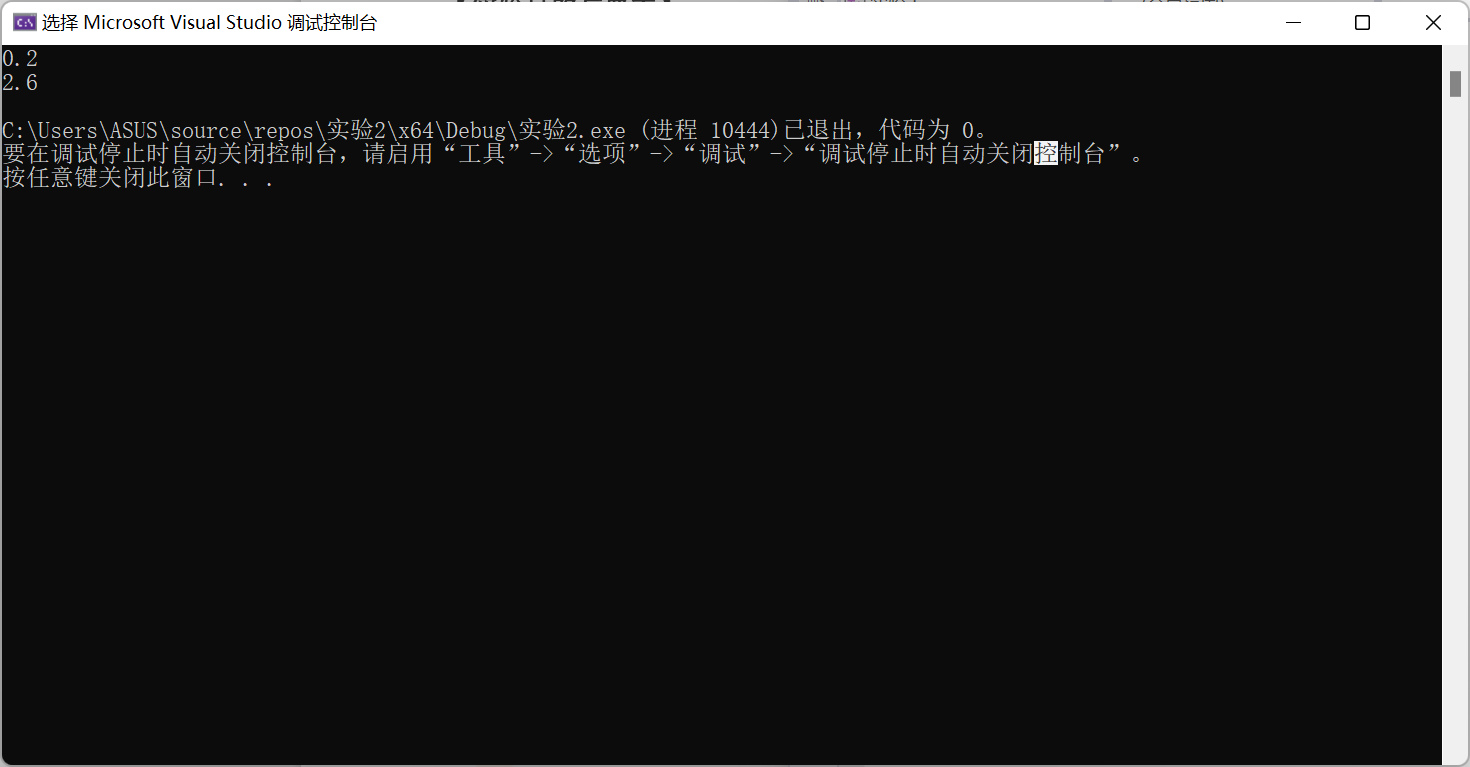
}

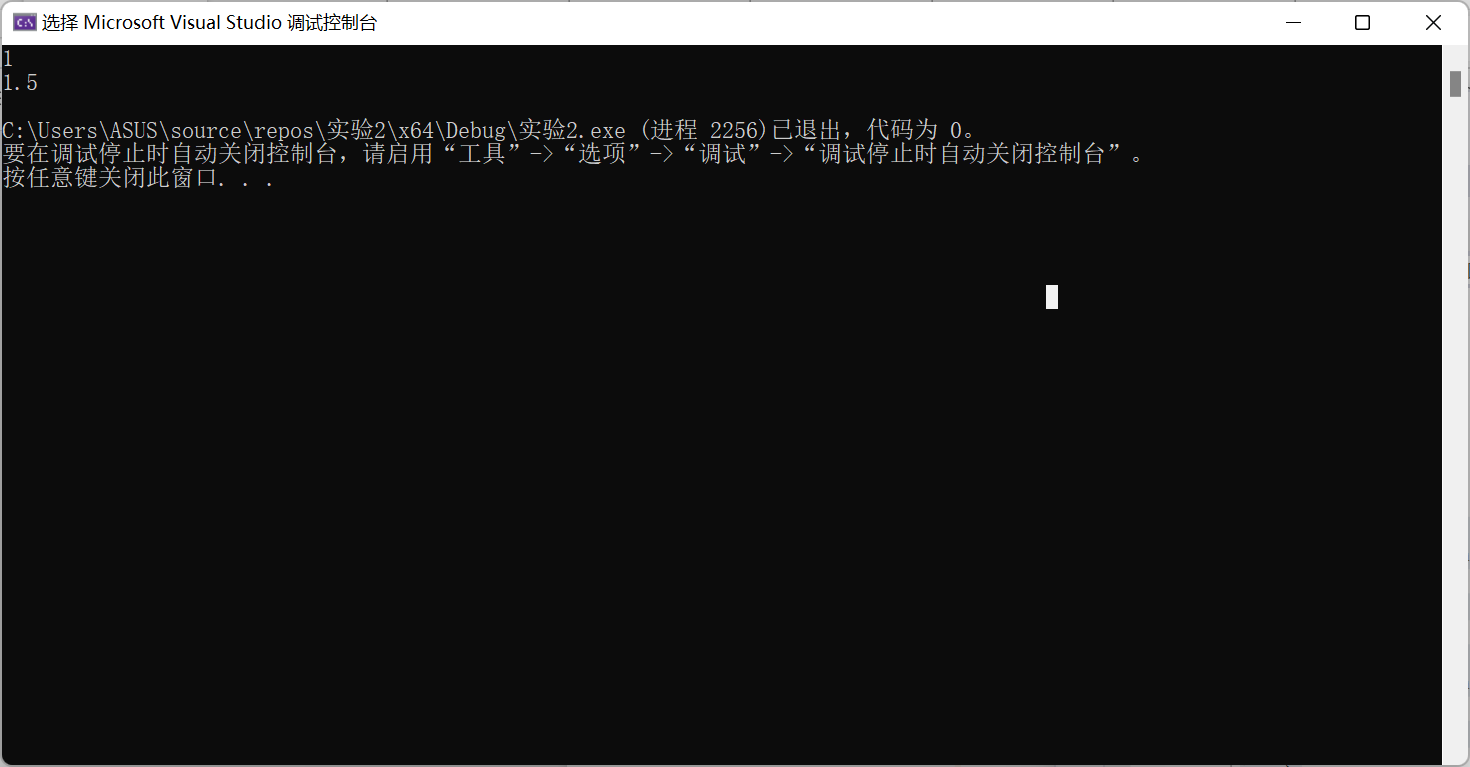
else

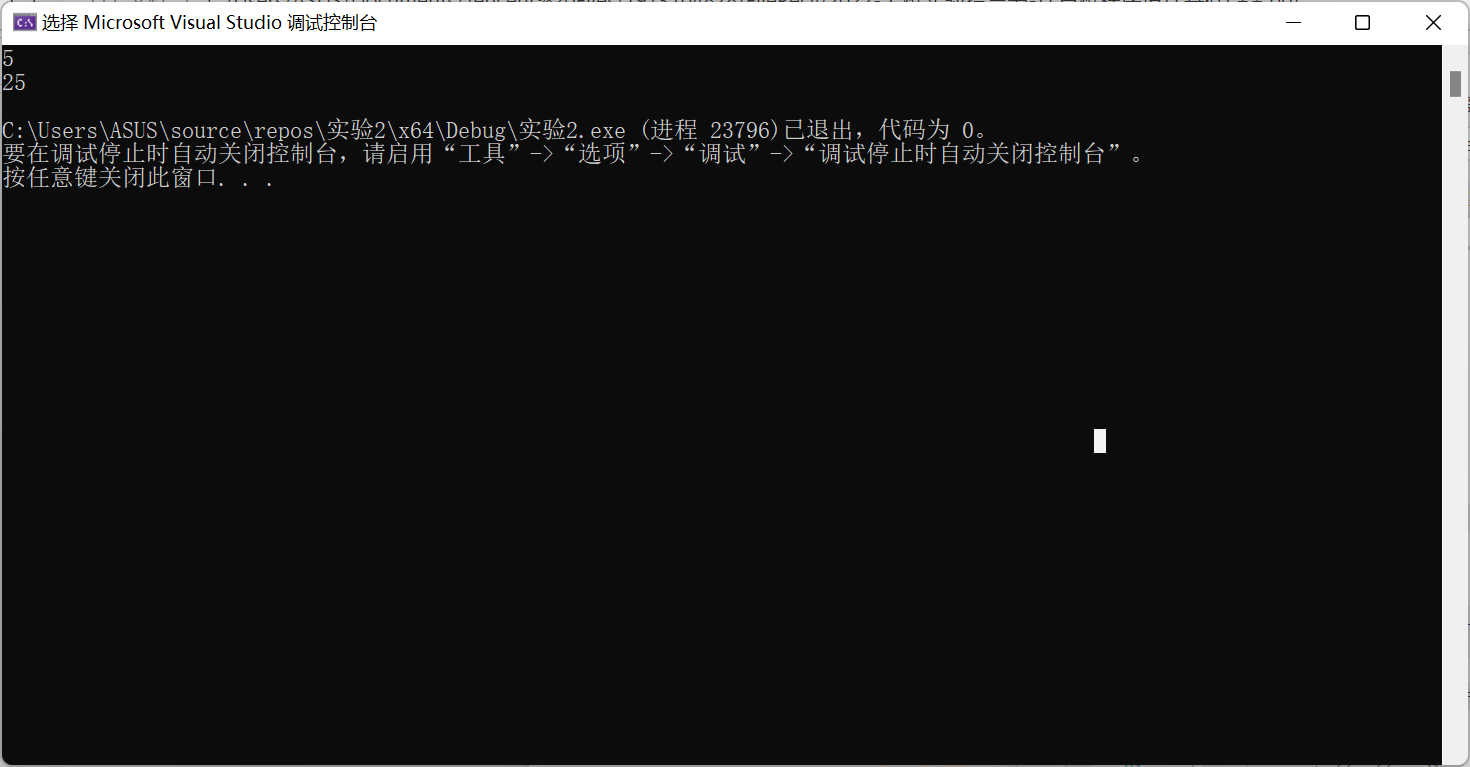
cout<<"Three sides do not meet the definition of triangle. Please enter the three sides again"<<endl;

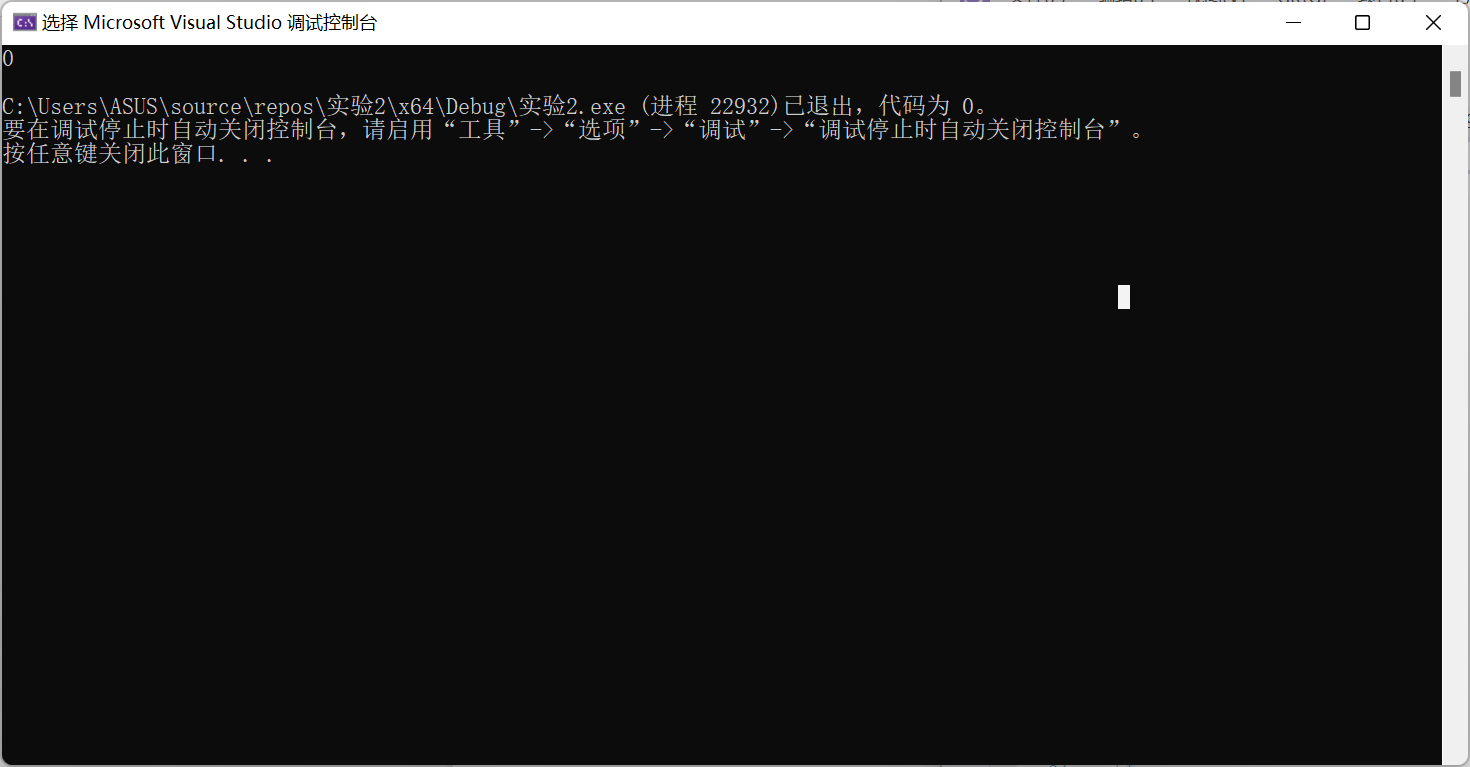
return 0;

}



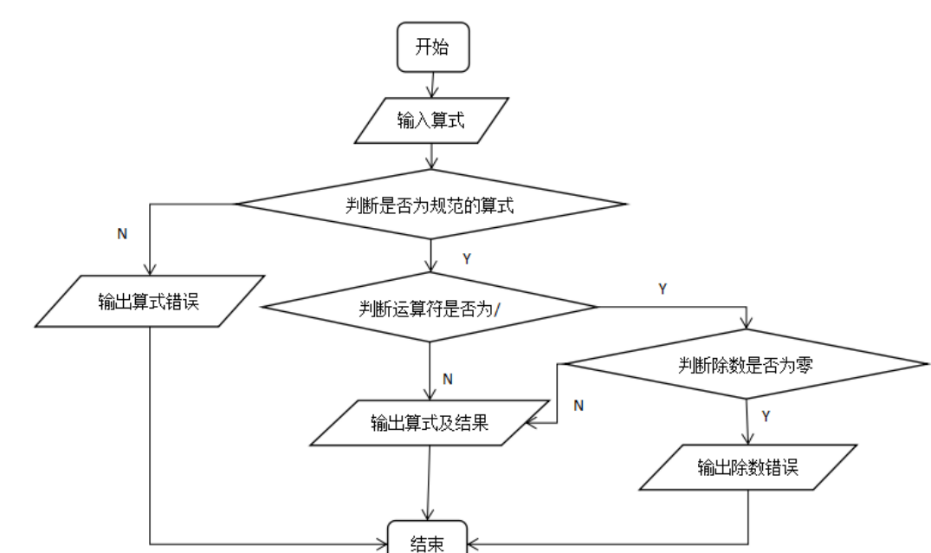






4.、

算法：



程序：

#include<iostream>

#include<string>

using namespace std;

int main()

{

double x, y;

char op;

cout << "输入两个数: ";

cin >> x >> y;

cout << "输入运算符：+、-、\*、/ 、%: ";

cin >> op;

switch (op)

{

case'+':

cout << x + y;

break;

case '-':

cout << x - y;

break;

case '\*':

cout << x \* y;

break;

case'/':

if (y == 0)

{

cout << "表达式不合法" << endl;

break;

}

else

cout << x / y;

break;

case'%':

cout << (int)x % (int)y;

break;

default:

cout << "Error! 请输入正确运算符。";

break;

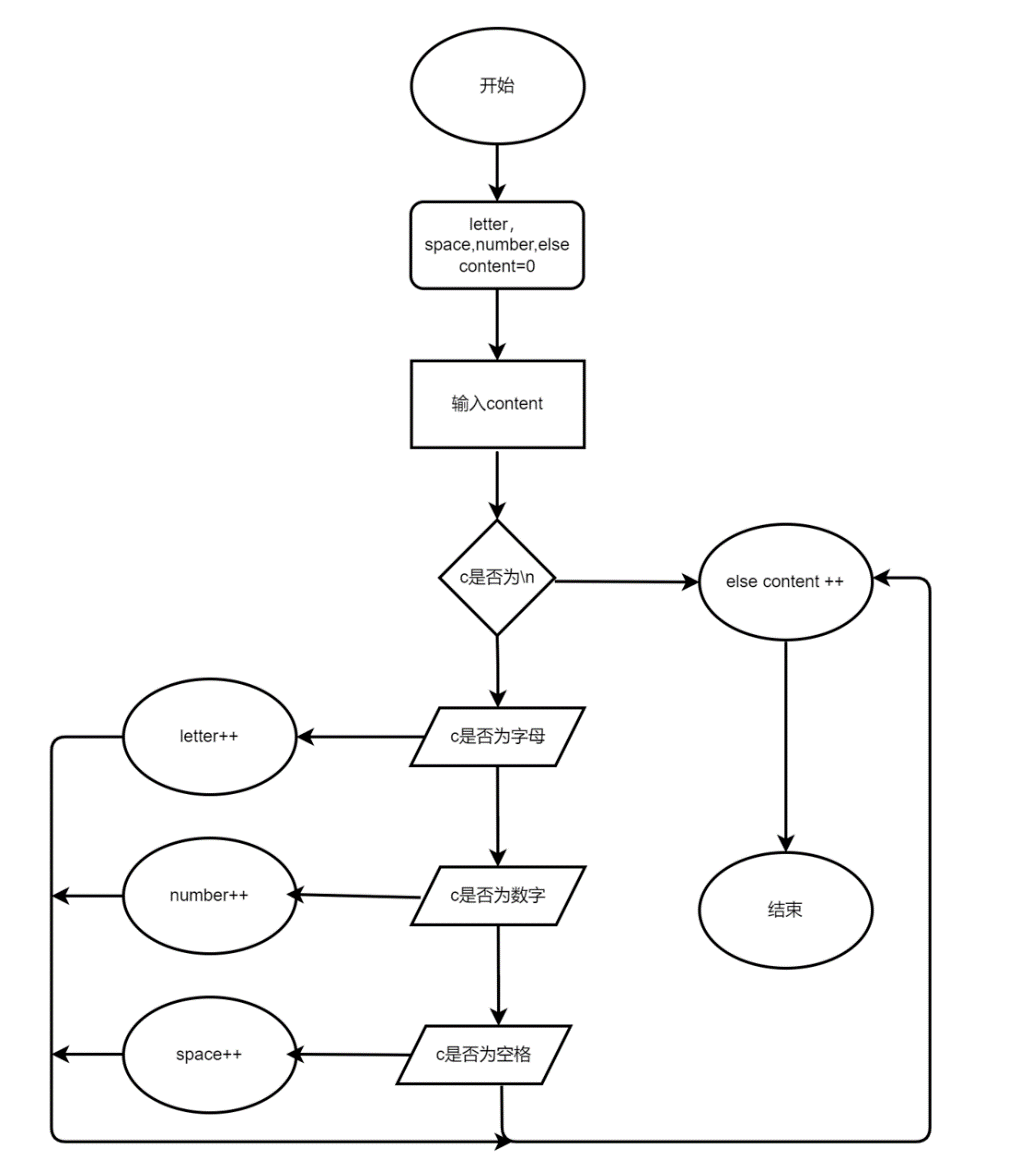
}

return 0;

}

5.

算法：



程序：

#include<iostream>

#include<string>

using namespace std;

int main()

{

//当程序调用getchar时，程序就等着用户按键。用户输入的字符被存放在键盘缓冲区输入流中。

//直到用户按回车为止。当用户键入回车之后，getchar才开始从输入流中每次读入一个字符

/\*注意！

①使用getchar()时，用户在键盘输入一行的数字，当遇到回车符时，这一行会被存进输入流。

②如果没有用循环的话，getchar（）每次只能从输入流中读取一个字符

③getchar()只能读取输入流回车符前面的字符，不可以读取回车符！

④如用户在按回车之前输入了不止一个字符，其他字符会保留在键盘缓存区中，等待后续getchar调用读取。

也就是说，后续的getchar调用不会等待用户按键，

而直接读取缓冲区中的字符，直到缓冲区中的字符读完后，才等待用户按键。\*/

//在程序中第一次调用getchar()函数从输入缓冲区中读取一个字节的数据。

// 需要注意的是，如果此时在程序中第二次调用getchar()函数，

// 因为此时输入缓冲区中还有回车键的数据没有被读出，第二个getchar()函数读出的是回车符。

char content;

int letter = 0, space = 0, number = 0, else\_content = 0;

while ((content = getchar())&& (content!= '\n'))

{

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

letter = letter + 1;

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

number = number + 1;

else if (content == ' ')

space = space + 1;

else

else\_content = else\_content + 1;

}

cout << "英文字母个数为：" << letter << endl;

cout << "空格个数为：" << space << endl;

cout << "数字个数为：" << number << endl;

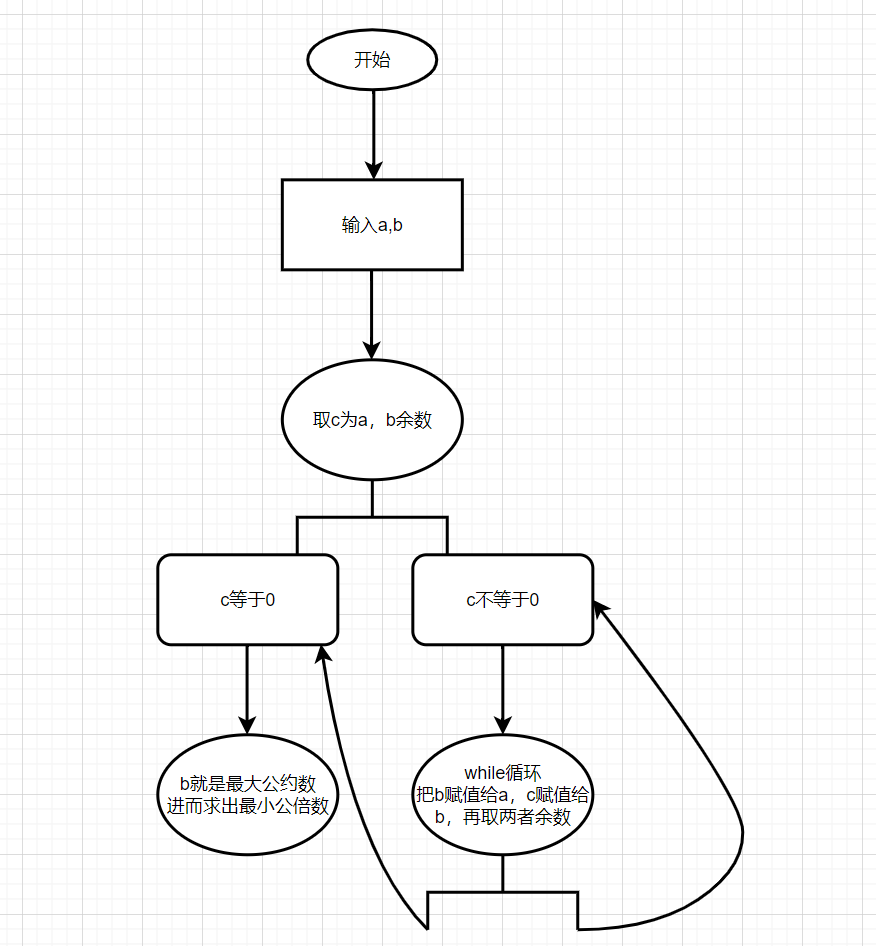
cout << "其他字符个数为：" << else\_content << endl;

return 0;

}

6.

算法：



程序：

#include<iostream>

#include<string>

using namespace std;

int main()

{

int a, b,c,m,n;

cin >> a >> b;

m = a;

n = b;

c = a % b;

if (c == 0)

{

cout << "两数的最大公约数为" << b << endl;

cout << "两数的最小公倍数为" << a \* b / b << endl;

}

while (c != 0)

{

a = b;

b = c;

c = a % b;

if (c == 0)

{

cout << "两数的最大公约数为" << b << endl;

cout << "两数的最小公倍数为" << m\* n/ b << endl;

}

}

return 0;

}

7.

程序：  
#include<iostream>

#include<string>

using namespace std;

int main()

{

int i, t, k;

for (int i = 0; i <= 4; i++)

{

for (int t = 1; t <=4-i; t++)

{

cout << " " ;

}

for (k = 0; k < i + 1; k++)

{

cout << "\*" ;

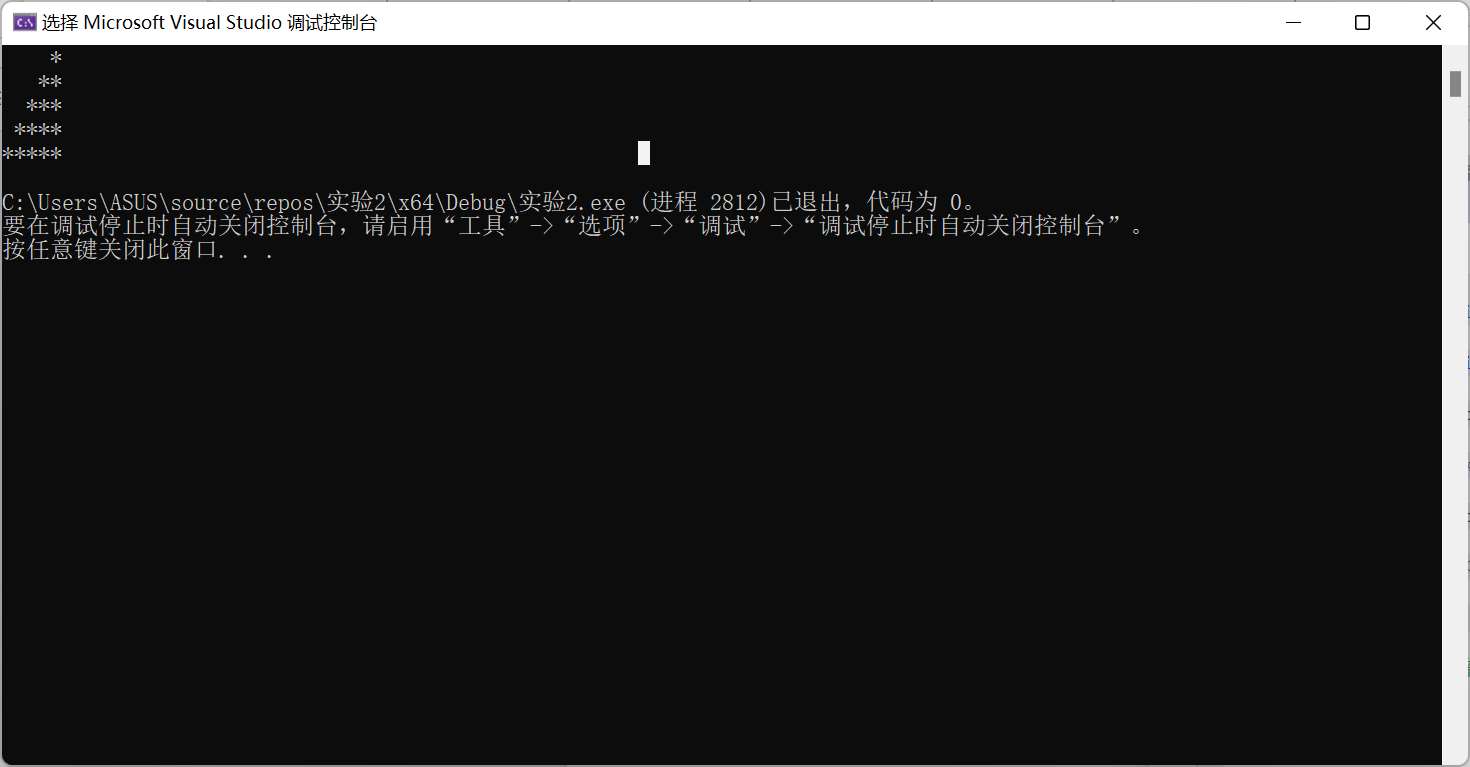
}

cout << endl;

}

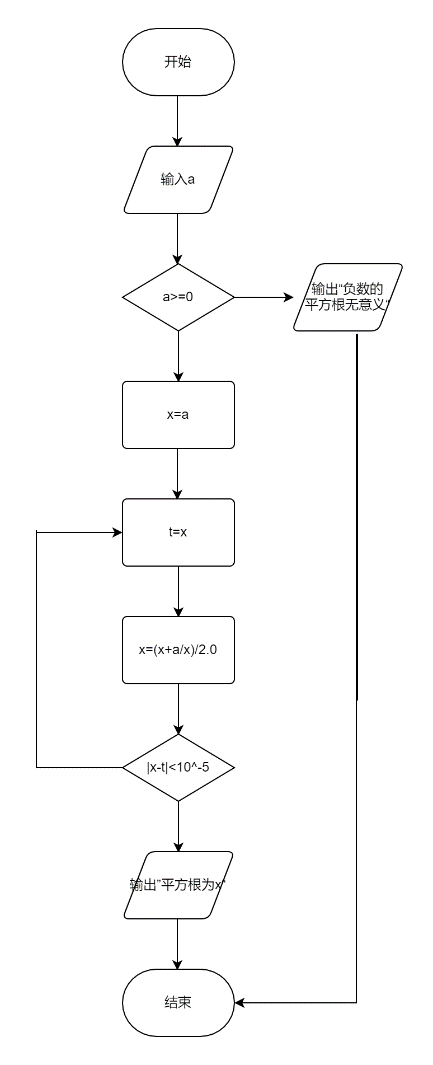
return 0;

}



8.

算法：



程序：

#include<iostream>

#include<cmath>

using namespace std;

int main()

{

double a,b,x0,x1;

cin >> a;

x0 = a;

x1 = (x0 + a / x0) / 2;

while (fabs(x1 - x0) >= 1e-5)

{

x0 = x1;

x1 = (x0 + a / x0) / 2;

}

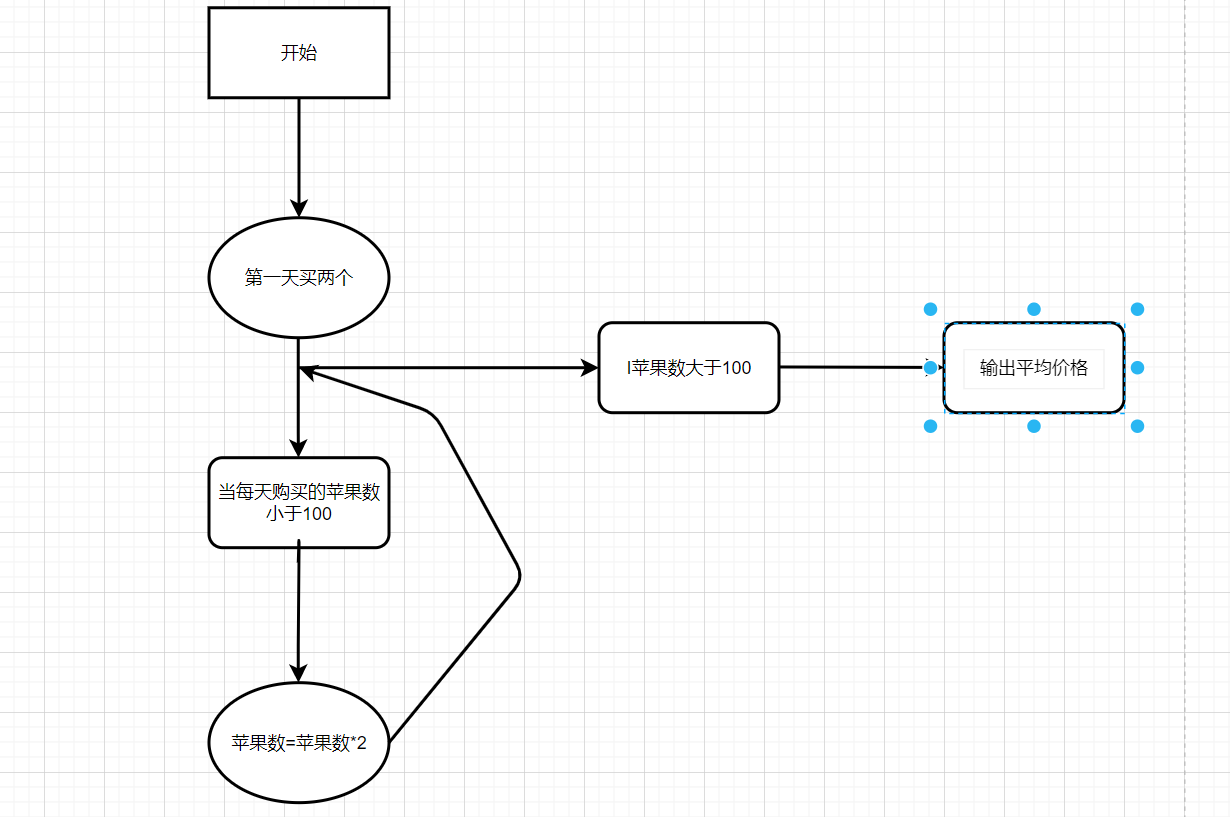
cout << x0 << endl;

return 0;

}

9.

算法：



程序：

#include<iostream>

#include<cmath>

using namespace std;

int main()

{

double a = 0.8,b,c=0,d=0;

int t = 2;

while (t <= 100)

{

b = t \* a;

t = t \* 2;

cout << b << endl;

c = c + b;

d++;

}

cout << "每天平均花："<<c / d << endl;

return 0;

}

四、遇到的问题与解决办法

遇到了大量自己不了解的代码，这十分考验自己的自学能力。我通过大量查阅资料，了解了getchar（）函数的用法

当程序调用getchar时，程序就等着用户按键。用户输入的字符被存放在键盘缓冲区输入流中。

直到用户按回车为止。当用户键入回车之后，getchar才开始从输入流中每次读入一个字符

而且在编写程序时遗忘了部分基础知识，不得不再去查询课本来解决

五、体会

C++是一个需要长期学习的编程语言，我还需要不断学习，在写程序时出现了编写5分钟调试半小时的情况，自己还是需要继续学习基础知识，巩固自己的基础