



第一讲 程序的基本结构

2017.8.20

目录:

1

基础知识

2

分支(选择)结构 if



一.基础知识

➤ 1.程序的基本格式:

例1-1最简单的a+b问题：

```
1. #include<iostream>    //输入输出流文头文件
2. using namespace std;  //标准库名字空间
3. int main(){           //main()是C++的主函数，程序从这个函数开始执行
4.     int a,b,c;         //定义了三个类型为int的整数型变量a,b,c
5.     cin>>a>>b;         //输入整数a和b
6.     c=a+b;             //把整数a与b的和赋值给变量c
7.     cout<<c<<endl;     //输出变量c的值，同时输出换行。
8.     return 0;          //主函数main()结束，返回0
9. }
```

◆ 尝试：.输入长方形的长a和宽b（整数）

◆ 输出：

◆ 长比宽多多少

◆ 面积s

◆ 周长d

◆ a是b的多少倍

```
10 5
5
50
30
2
```

➤ 四种最基本的运算符号：

➤ +

➤ -

➤ *

➤ /

◆例1.2 两个数的交换

- 输入两个int类型的整数，然后交换后输出。
- 如输入：3 4
- 输出：4 3

```
1. #include<iostream>
2. using namespace std;
3. int main() {
4.     int a,b,tem;
5.     cin>>a>>b;
6.     tem=a;a=b;b=tem; //交换a和b的值
7.     cout<<a<<" "<<b<<endl;
8.     return 0;
9. }
```

二. 分支（选择）结构if..else语句

程序的三种结构：

1. 顺序结构
2. 选择结构
3. 循环结构

◆例1-3 . 石头剪刀布

- 游戏的规则是：
- 双方同时各出石头剪刀布中的一个，石头赢剪刀，剪刀赢布，布赢石头。现在甲乙双方进行对弈，根据各自出的进行判断输赢。
- 为了表示方便,现规定：
- 0代表石头；1代表剪刀；2代表布。



石头



剪刀



布

➤ 【输入】

- 一行，两个用一个空格隔开的 x 和 y ， x 代表甲方的选择， y 代表乙方的选择。 x 和 y 一定是0 1 2中的一个。

➤ 【输出】

- 如果甲方赢，输出： A

- 如果乙方赢，输出： B

- 如果平局，输出： O

- 如：

- 输入： 0 1

- 输出： A



石头



剪刀



布

共有多少种情况？



石头

1



剪刀


2



布

3

甲 (A)	乙 (B)	输出
0	0	O
0	1	A
0	2	B
1	0	B
1	1	O
1	2	A
2	0	A
2	1	B
2	2	O



```
1. #include<cstdio>
2. #include<iostream>
3. using namespace std;
4. int main() {
5.     int a, b;
6.     cin>>a>>b;
7.     if (a==b) cout<<"0";
8.     if (a==0 && b==1) cout<<"A";
9.     if (a==0 && b==2) cout<<"B";
10.    if (a==1 && b==0) cout<<"B";
11.    if (a==1 && b==2) cout<<"A";
12.    if (a==2 && b==0) cout<<"A";
13.    if (a==2 && b==1) cout<<"B";
14.    return 0;
15. }
```

◆ 选择结构 if 语句

格式1：单分支

```
if(条件) 语句1;
```

```
if(条件){  
    语句1;  
    语句2;  
    ...  
}
```

- (1) 条件后面无分号“;”，因为不是完整的一条语句
- (2) 如果只有一条语句 {} 可以省略不写；

格式 2：双分支

```
if(条件) 语句1;  
else 语句2;
```

```
if(条件){  
    语句1;  
    语句2;  
}  
Else{  
    语句1;  
    语句2;  
}
```

- 关系运算符: `==` , `!=` , `>` , `<` , `>=` , `<=`
- 逻辑运算符: `&&`: 并且。 `||`: 或者。
- 结果数据类型: `bool`: `true:1` `false:0`

◆例1-4.输入正整数 a , b , c , 判断能否组成三角形。

➤ 如:

➤ 输入

➤ 4 5 6

➤ 输出

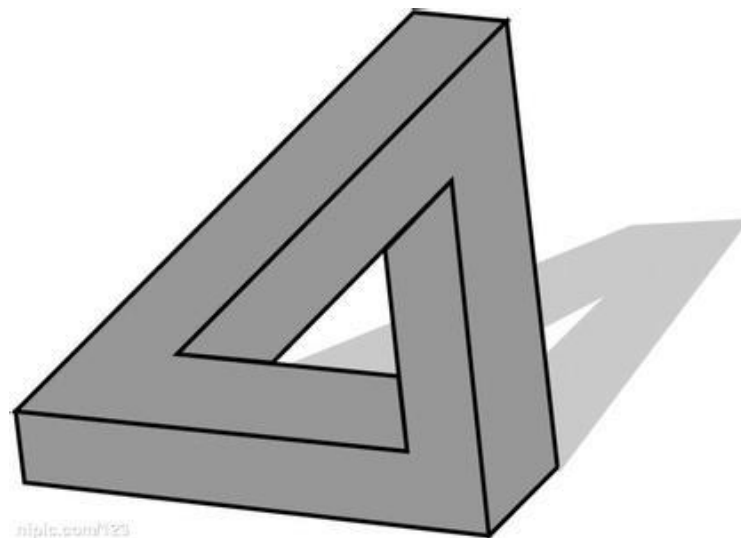
➤ yes

➤ 输入:

➤ 1 2 3

➤ 输出

➤ no



◆例1-5.输入正整数 a , b , c , 判断能否组成直角三角形。

➤ 输入

➤ 3 4 5

➤ 输出

➤ yes

➤ 输入

➤ 4 5 6

➤ 输出

➤ no

◆例1-6. 判断闰年

➤ 【问题描述】

➤ 输入某一年份，判断该年份是否为闰年(是则输出“yes”，否则输出“no”)。

➤ 【输入】 年份。

➤ 【输出】 根据判断结果输出yes或者no。

➤ 如：

➤ 输入：2014

➤ 输出：no

◆判断年y是否为闰年

➤ 四年一闰，百年不闰；四百年闰

➤ $y \% 4 == 0 \ \&\& \ y \% 100 \neq 0 \ || \ y \% 400 == 0$

◆例1-7 . 鸡兔同笼问题。

- 已知鸡和兔子的总数量为 n ，总腿数为 m 。
- 输入 n 和 m ，依次输出鸡的数目和兔子的数目。
- 如：
- 输入：
- **14 32**
- 输出：
- **12 2**

- 输入：
- **10 16**
- 输出：
- **No answer**

◆例1-8 . 输入年y和月m，输出这一年的这个月的天数d。

- 【输入】 年份y和月份m，中间用空格隔开。
- 【输出】 这个月的天数d。
- 【样例输入1】 2015 3
- 【样例输出1】 31
- 【样例输入2】 2015 4
- 【样例输出2】 30
- 【样例输入3】 2000 2
- 【样例输出3】 29
- 【样例输入4】 2001 2
- 【样例输出4】 28



◆年月天数的关系：

- 1, 3, 5, 7, 8, 10, 12月份为31天;
- 4, 6, 9, 11月份为30天;
- 2月: 闰年29天, 其他年份为28天。

◆判断年y是否为闰年

➤ 四年一闰，百年不闰；四百年闰

$(y \% 4 == 0 \ \&\& \ y \% 100 != 0 \ || \ y \% 400 == 0)$

◆ 算法流程：

1. 定义整数 y, m, d ;
2. 输入 y, m ;
3. 如果 $m == 1, 3, 5, 7, 8, 10, 12$ 则 $d = 31$;
4. 如果 $m == 4, 6, 9, 11$ 则 $d = 30$;
5. 如果 $m == 2$
6. 如果 y 是闰年 则 $d = 29$;
7. 否则 $d = 28$;
8. 输出 d ;

例1-9 . 输入三角形的三条边a, b, c。

如果能组成三角形, 则输出三角形的面积, 小数点后面保留2位有效数字;

如果不能组成三角形, 输出 **No answer!**

a, b, c < 100

✓ 3 4 5

✓ 6.00

✓ 2 3 4

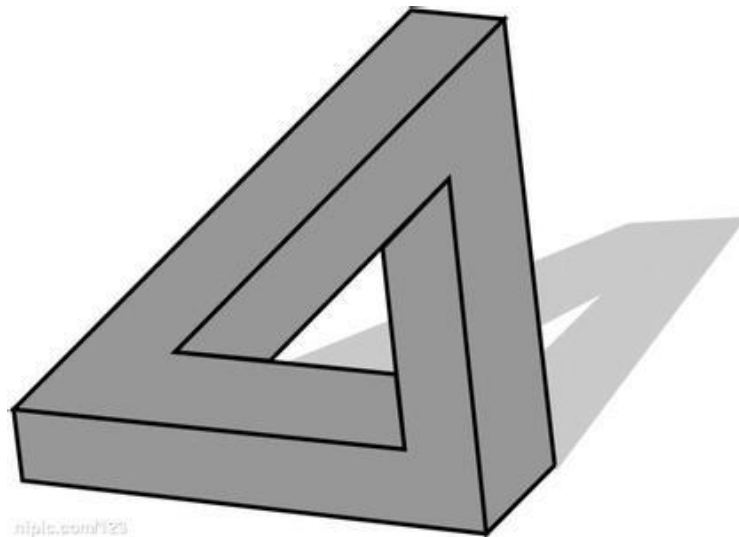
✓ 2.90

✓ 1 1 1

✓ 0.43

✓ 1 2 3

✓ no answer!



海伦公式:

$$p = \frac{a + b + c}{2}$$

$$s = \sqrt{p(p - a)(p - b)(p - c)}$$

秦九韶公式:

$$S = \frac{\sqrt{c^2 a^2 - \left(\frac{c^2 + a^2 - b^2}{2}\right)^2}}{2}$$

算法流程:

1. 定义实数 a, b, c, p, s ; //double
2. 如果能组成三角形
3. 求周长的一半 p ;
4. 求面积 s ;
5. 输出面积 s ;
6. 否则
7. 输出无解;



◆实数类型：double

➤ double a,b,c;

例1-10. 已知方程:

$$ax^2+bx+c=0$$

输入系数a,b,c, $a \neq 0$

输出方程的解(考虑各种情况)。保留两位小数。

(1):两个不同的实数解: 分别输出。

(2):相同的实数解: 输出一个即可。

(3):如果无实数解输出"No answer!";

判别式: $d=b^2-4ac$

$$d>0: \quad x = \frac{-b \pm \sqrt{d}}{2a}$$

$$d=0: \quad x = \frac{-b}{2a}$$

$d<0$: **No answer!**

输入:
1 3 2
输出:
X1=-1.00
X2=-2.00

输入:
2 5 2
输出:
X1=-0.50
X2=-2.00

输入:
1 2 1
输出:
X=-1.00

输入:
1 1 1
输出:
No answer!

◆ 算法流程：

1. 定义 a, b, c, d, x_1, x_2 为实数;
2. 输入实数 a, b, c ;
3. 求判别式 d ;
4. 如果 $d > 0$
 求 x_1
 求 x_2 ;
 输出 x_1 ;
 输出 x_2 ;
5. 若果 $d = 0$
 输出 $-b/(2a)$;
6. 如果 $d < 0$
 输出NO