**字符串操作**

**异常处理**

**容器**

**数据溢出的判断（string->int）leetcode8**

Feb.10th 2017

Java源代码中

1.处理符号

2.max=Integer.MAX\_VALUE/10，比较result否比max小，否则，在做一位数的处理，一定溢出

3.在满足2的情况下，处理下一位，next=result\*10+digit(java源代码用的-，最后取负处理，不明白为甚用减处理)，比较next和result的大小（我的方法是

比较result!=（next-digit）/10）(用减号处理的前提下)）,判断是否溢出，

！！！！2147483648作为输入的时候，满足2，处理3的时候，next=-2147483648（已经溢出），但是满足后边的判断，result=214748364，next-digt还是溢出的，得到2147483640，！！！！（Integer.MAX\_VALUE+1,+2,Integer.MIN\_VALUE-1都会出现这种情况）所以无法判断时候溢出，所以在后面要加一条判断是否result为正

if(result>max||result==max&&str.charAt(i)>'7'){

if(flag==1)return Integer.MAX\_VALUE;

return Integer.MIN\_VALUE;

}溢出处理很优秀（把Integer.MIN\_VALUE作为溢出处理，不影响结果）

Blog：http://blog.csdn.net/evan123mg/article/details/45459781

Solution: https://discuss.leetcode.com/topic/2666/my-simple-solution

**If…else…和三元运算符**

expression1?expression2:expression3

if(expression1)

expression2

else

expression3

int x=4;

System.out.println((x>4)?99.0:9);

输出结果为9.0



**定义变量**

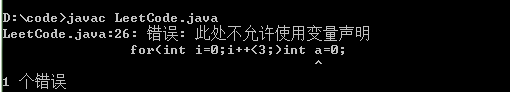
Mar.7th .2017



或者



出现错误

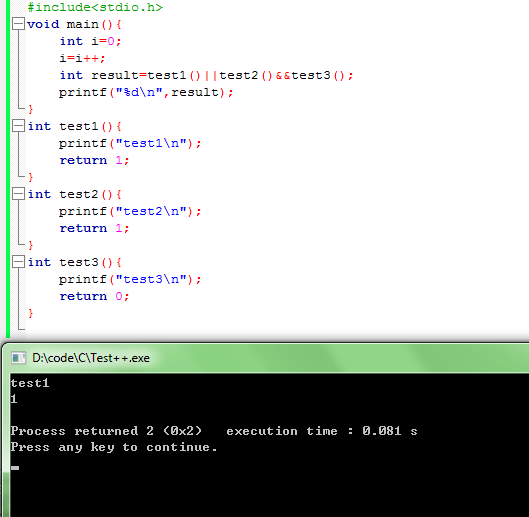


**？？？**

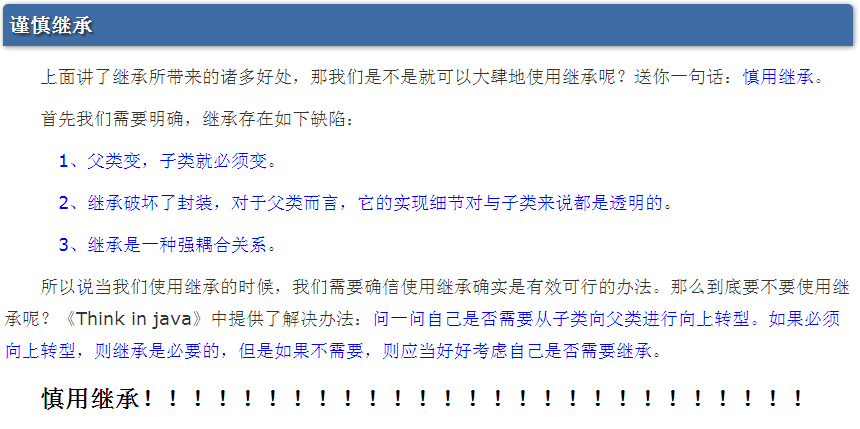
对于继承而已，子类会默认调用父类的构造器，但是如果没有默认的父类构造器，子类必须要显示的指定父类的构造器，而且必须是在子类构造器中做的第一件事(第一行代码)。

**Java运算符优先级高的表达式会被视为整体, 但是执行顺序还是从左到右**

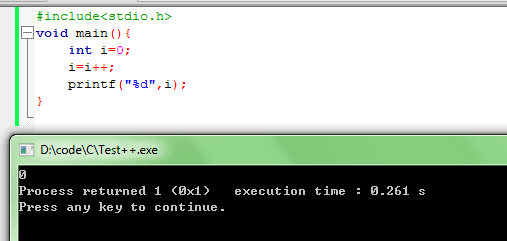
**C语言同理**



 诚然尽管可以使用protected访问修饰符来限制父类属性和方法的访问权限，但是最好的方式还是将属性保持为private(我们应当一致保留更改底层实现)，通过protected方法来控制类的继承者的访问权限。



向上转型存在一些缺憾，那就是它必定会导致一些方法和属性的丢失，而导致我们不能够获取它们。所以父类类型的引用可以调用父类中定义的所有属性和方法，对于只存在与子类中的方法和属性它就望尘莫及了---1

****