



语法

public class BigInteger extends Number implements Comparable

描述

一个BigInteger(大整型数)代表一个任意精度的整数。一个大整型数值为一种长得不能用 64 位long型数值表示的数值的计算提供了一种方法。这个类中的运算提供了与对 long型数值同样 (甚至更多)种类的运算。除了这种大整型数值从来不溢出，它们在语义上与 java的整型运算相同。如想获得更多的关于java整型运算的信息，请参见《The Java Language Specification , First Edition》。

一个BigInteger类型的对象是不能改变的。也就是说，它包含的值是不能改变的。大多数 BigInteger运算会自动产生一个新的BigInteger对象来保存运算的结果。

如想获得关于这个类的更多信息，请参见《 The Java Class Libraries , Second Edition , Volume 1》。

版本1.2中所作的修改

这个类现在可以实现 Comparable接口。这使得依据收集类排序 BigInteger对象成为可能(参看Collection)。

为了更加方便，加入了两个新的大整形常数 :ZERO和ONE。

成员概述	
构造函数	
BigInteger()	构造一个 BigInteger实例。
常数	
ONE	包含数值 1的一个大整型数值常数。

(续)

成员概述

1.2	ZERO	包含数值0的一个大整型数值常数。
	数学方法	
	abs()	计算这个大整型数值的绝对值。
	add()	计算这个大整型数值与另一个大整型数值的和。
	divide()	计算这个大整型数值除以另一个大整型数值的结果。
	divideAndRemainder()	计算两个大整型数值的商和余数。
	gcd()	计算这个大整型数值与另一个大整型数值的比值的最大分母。
	isProbablePrime()	猜测这个大整型数值是否是素数。
	max()	比较这个大整型数值与另一个大整型数值的大小，返回较大值。
	min()	比较这个大整型数值与另一个大整型数值的大小，返回较小值。
	mod()	计算这个大整型数值与另一个大整型数值的模，返回余数。
	modInverse()	计算这个大整型数的模乘法的相反数。
	modPow()	计算这个大整型数值的平方与另外一个大整型数值的模的余数。
	multiply()	计算这个大整型数值与另一个大整型数值的积。
	negate()	计算这个大整型数值与-1的积。
	pow()	计算这个大整型数的平方。
	remainder()	计算这个大整型数值除以另一个大整型数值的余数。
	signum()	返回这个大整型数值的符号。
	subtract()	计算这个大整型数值与另一个大整型数值的差。
	比较方法	
△	compareTo()	比较这个大整型值与另一个大整型数值。
	equals()	决定是否这个大整型数值与一个对象相等。
	位方法	
	and()	这个大整型数值与另一个大整型数值相与。
	andNot()	这个大整型数值与另一个大整型数值的相反数相与。
	bitCount()	计算这个大整型数值的二进制表示的0或1的个数。
	bitLength()	不包括符号位，这个大整型数值的最小表示方法中的位的个数。
	clearBit()	返回这个大整型数值的二进制表示被清除一位的结果。
	flipBit()	返回这个大整型数值的一个位被求反的结果。
	getLowestSetBit()	找到这个大整型数值的二进制表示的最右边一位。
	not()	求这个大整型数值的二进制表示的各位的补。
	or()	这个大整型数值与另一个大整型数值各位求或。
	setBit()	返回这个大整型数值的二进制表示的一个位被设定的结果
	shiftLeft()	移到这个大整型数值的二进制表示的所有位的最左边一位。
	shiftRight()	移到这个大整型数值的二进制表示的所有位的最右边一位。
	testBit()	决定是否这个大整型数值的二进制表示的某些位是设定的。
	xor()	这个大整型数值与另一个大整型数值做异或。
	转换方法	
	doubleValue()	把这个大整型数值转换为 double 型值。
	floatValue()	把这个大整型数值转换为 float 型值。
	intValue()	把这个大整型数值转换为 int 型值。

(续)

成员概述	
longValue()	把这个大整型数值转换为 long 型值。
toByteArray()	产生一个包含这个大整数值中各位的数组。
valueOf()	返回一个简单整型值的 BigInteger 实例。
对象方法	
hashCode()	计算这个大整型数值的哈希码。
toString()	产生代表这个大整型数值的一个字符串。

参见

java.lang.Comparable, java.util.Collection。

《The Java Class Libraries, Second Edition, Volume 1》中的 BigInteger 部分。

▲ compareTo()

目的	比较这个大整型数值与另一个大整型数值。
语法	public int compareTo(BigInteger val)
从版本 1.2	public int compareTo(Object val)
描述	这个方法比较这个大整型数值与另一个大整型数值 val。如果这个大整型数值小于 val, 返回-1。如果与val相等, 返回0。如果大于val, 返回1。
版本 1.2 中的改动	在版本 1.2 中对 Object 采用的这种方法是新的。
参数	
val	一个非空 BigInteger 对象。
返回	-1、0、或1, 决定于这个大整型数值与另一个大整型数值的比较结果。
参见	equals()。
示例	

```
// Some sample output.
BigInteger bi = BigInteger.valueOf(100);
System.out.println( bi.compareTo(BigInteger.ZERO) );// 1
System.out.println( BigInteger.ZERO.compareTo(bi) );// -1
//System.out.println( bi.compareTo("A String") );    // ClassCastException
//System.out.println( bi.compareTo(null) );           // NullPointerException

// Read numbers from standard in.
BufferedReader is =
    new BufferedReader(new InputStreamReader(System.in));
String line;
List list = new ArrayList();

// Add the big integers to the list.
try {
    while ((line = is.readLine()) != null) {
        list.add(new BigInteger(line));
    }
} catch (Exception e) {
    e.printStackTrace();
}

// Sort the list.
Collections.sort(list);
```

```
// Print sorted list.
for (int i=0; i<list.size(); i++) {
    System.out.println(list.get(i));
}
```

输入

```
2903429847238942
92384729847294729874
03294823984938439483
203489204823904823908420842094820948293
83478278437843784
2387284732
29348723904820984032948
28347238472947294729748923749238
```

输入

```
2387284732
2903429847238942
83478278437843784
3294823984938439483
92384729847294729874
29348723904820984032948
28347238472947294729748923749238
203489204823904823908420842094820948293
```

1.2 ONE

目的	包含数值1的一个大整型数值常数。
语法	public static final BigInteger ONE
描述	对象是一个包含值1的一个方便的常数。
示例	

```
BigInteger bi = BigInteger.valueOf(100);
bi = bi.add(BigInteger.ONE);
System.out.println( bi ); // 101

System.out.println( bi.compareTo(BigInteger.ZERO) ); // 1
System.out.println( BigInteger.ZERO.compareTo(BigInteger.ONE) ); // -1
```

1.2 ZERO

目的	包含数值0的一个大整型数值常数。
语法	public static final BigInteger ZERO
描述	对象是一个包含值0的一个方便的常数。它通常用于大整型数值需要与0比较时。
示例	见ONE。