[Назад](#)[Меню](#)[Вперед](#)

## Класс BigInteger

Все примитивные целые типы имеют ограниченный диапазон значений. В целочисленной арифметике Java нет переполнения, целые числа приводятся по модулю, равному диапазону значений.

Для того чтобы было можно производить целочисленные вычисления с любой разрядностью, в состав Java API введен класс `BigInteger`, хранящийся в пакете `java.math`. Этот класс расширяет класс `Number`, следовательно, в нем переопределены методы `doubleValue()`, `floatValue()`, `intValue()`, `longValue()`. Методы `byteValue()` и `shortValue()` не переопределены, а прямо наследуются от класса `Number`.

Действия с объектами класса `BigInteger` не приводят ни к переполнению, ни к приведению по модулю. Если результат операции велик, то число разрядов просто увеличивается. Числа хранятся в двоичной форме с дополнительным кодом.

Перед выполнением операции числа выравниваются по длине распространением знакового разряда.

Шесть конструкторов класса создают объект класса `BigInteger` из строки символов (знака числа и цифр) или из массива байтов.

ШКОЛА БИЗНЕСА  
И ЛИЧНОСТНОГО РАЗВИТИЯ  
— ALEX YANOVSKY —

КАК ПОДНЯТЬ  
САМООЦЕНКУ?

Вы **100%**  
поднимете  
свою  
самооценку!



Получите  
**БЕСПЛАТНЫЕ  
ВИДЕОУРОКИ!**

ПОЛУЧИТЬ  
ВИДЕОУРОКИ

Две константы — ZERO и ONE — моделируют нуль и единицу в операциях с объектами класса BigInteger .

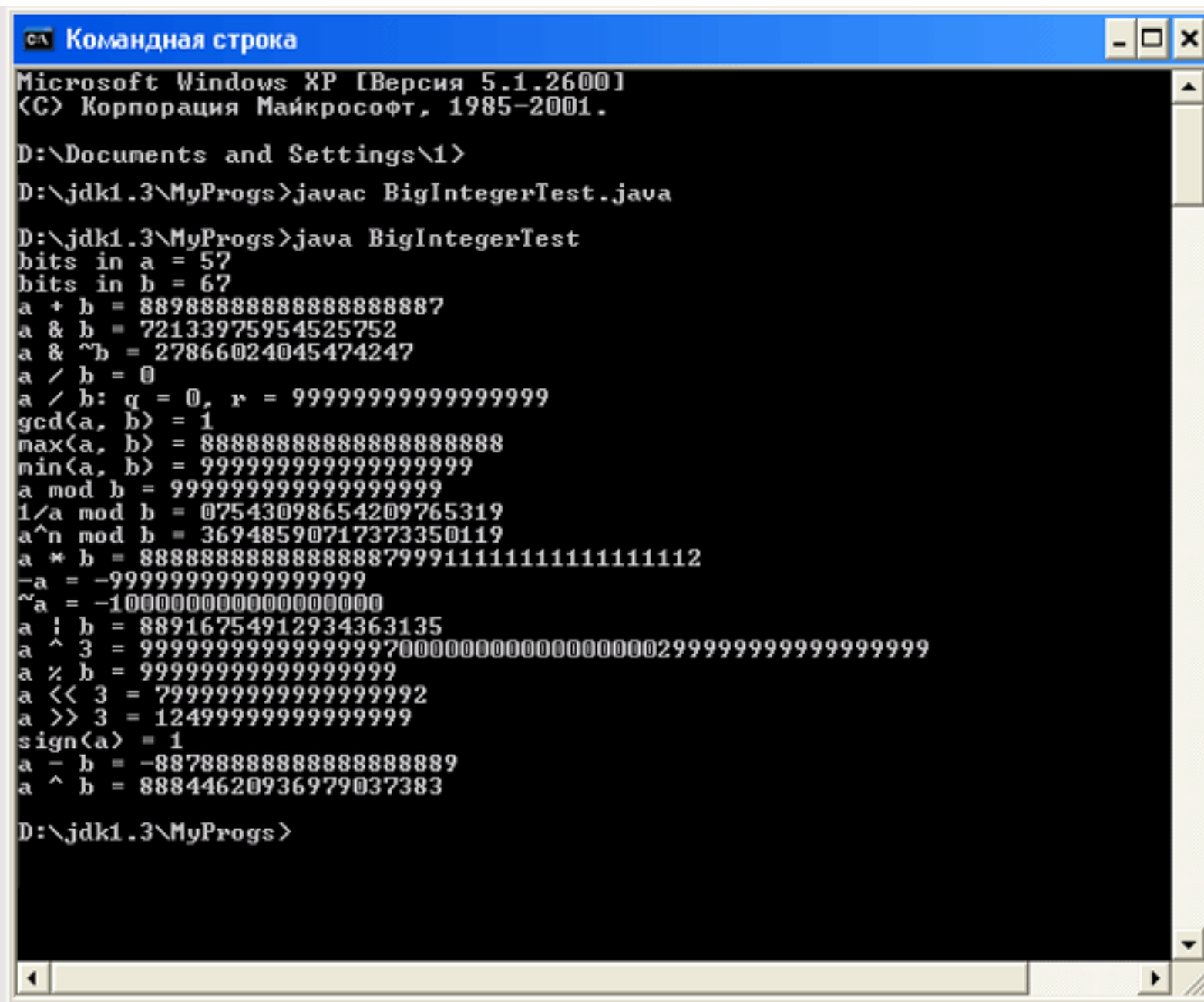
Метод toByteArray() преобразует объект в массив байтов.

Большинство методов класса BigInteger моделируют целочисленные операции и функции, возвращая объект класса BigInteger :

- `abs()` — возвращает объект, содержащий абсолютное значение числа, хранящегося в данном объекте `this` ;
- `add(x)` — операция `this + x` ;
- `and(x)` — операция `this & x` ;
- `andNot(x)` — операция `this & (~x)` ;
- `divide (x)` — операция `this / x` ;
- `divideAndRemainder(x)` — возвращает массив из двух объектов класса BigInteger , содержащих частное и остаток от деления `this` на `x` ;
- `gcd(x)` — наибольший общий делитель, абсолютных, значений объекта `this` и аргумента `x` ;
- `max(x)` — наибольшее из значений объекта `this` и аргумента `x` ; `min(x)` — наименьшее из значений объекта `this` и аргумента `x` ; `mod(x)` — остаток от деления объекта `this` на аргумент метода `x` ;
- `modinverse(x)` — остаток от деления числа, обратного объекту `this` , на аргумент `x` ;
- `modPow(n, m)` — остаток от деления объекта `this` , возведенного в степень `n` , на `m` ;
- `multiply (x)` — операция `this * x` ;
- `negate()` — перемена знака числа, хранящегося в объекте;
- `not()` — операция `~this` ;
- `or(x)` — операция `this | x` ;
- `pow(n)` — операция возведения числа, хранящегося в объекте, в степень `n` ;
- `remainder(x)` — операция `this % x` ;
- `shiftLeft (n)` — операция `this « n` ;
- `shiftRight (n)` — операция `this » n` ;

- `signum()` — функция `sign (x)` ;
- `subtract (x)` — операция `this - x` ;
- `xor(x)` — операция `this ^ x` .

В листинге 4.3 приведены примеры использования данных методов, а рис. 4.4 показывает результаты выполнения этого листинга.



```
Командная строка
Microsoft Windows XP [Версия 5.1.2600]
(C) Корпорация Майкрософт, 1985-2001.

D:\Documents and Settings\1>
D:\jdk1.3\MyProgs>javac BigIntegerTest.java
D:\jdk1.3\MyProgs>java BigIntegerTest
bits in a = 57
bits in b = 67
a + b = 88988888888888888887
a & b = 72133975954525752
a & ~b = 27866024045474247
a / b = 0
a / b: q = 0, r = 999999999999999999
gcd(a, b) = 1
max(a, b) = 88888888888888888888
min(a, b) = 99999999999999999999
a mod b = 99999999999999999999
1/a mod b = 07543098654209765319
a^n mod b = 36948590717373350119
a * b = 888888888888888888879999111111111111112
-a = -99999999999999999999
~a = -100000000000000000000
a ! b = 88916754912934363135
a ^ 3 = 999999999999999999700000000000000000299999999999999999999
a % b = 99999999999999999999
a << 3 = 79999999999999999992
a >> 3 = 12499999999999999999
sign(a) = 1
a - b = -88788888888888888889
a ^ b = 88844620936979037383

D:\jdk1.3\MyProgs>
```

Рис. 4.4. Методы класса BigInteger в программе BigIntegerTest

Листинг 4.3. Методы класса BigInteger в программе BigIntegerTest

```
import Java.math.BigInteger;

class BigIntegerTest{

    public static void main(String[] args){

        BigInteger a = new BigInteger("99999999999999999999") ;

        BigInteger b = new BigInteger("888888888888888888888888");

        System.out.println("bits in a = " + a.bitLength());

        System.out.println("bits in b = " + b.bitLength());

        System.out.println("a + b = " + a.add(b));

        System.out.println("a & b = " + a.and(b));

        System.out.println("a & ~b = " + a.andNot(b));

        System.out.println("a / b = " + a.divide(b));

        BigInteger[] r = a.divideAndRemainder(b);

        System.out.println("a / b: q = " + r[0] + ", r = " + r[1]);

        System.out.println("gcd(a, b) = " + a.gcd(b));

        System.out.println("max(a, b) = " + a.max(b));

        System.out.println("min(a, b) = " + a.min(b));

        System.out.println("a mod b = " + a.mod(b));
```

```
System.out.println("1/a mod b = " + a.modInverse(b));

System.out.println("a^n mod b = " + a.modPow(a, b));

System.out.println("a * b = " + a.multiply(b));

System.out.println("-a = " + a.negate());

System.out.println("~a = " + a.not());

System.out.println("a | b = " + a.or(b));

System.out.println("a л 3 = " + a.pow(3));

System.out.println("a % b = " + a.remainder(b));

System.out.println("a « 3 = " + a.shiftLeft(3));

System.out.println("a » 3 = " + a.shiftRight(3));

System.out.println("sign(a) = " + a.signum());

System.out.println("a - b = " + a.subtract(b));

System.out.println("a л b = " + a.xor(b));

}

}
```

Обратите внимание на то, что в программу листинга 4.3 надо импортировать пакет `Java.math` .



[Назад](#)[Меню](#)[Вперед](#)

Тысячи одиноких русских  
в **США** ищут любовь на  
**RussianHearts.us**



**НАЙДИ  
СВОЮ  
ЛЮБОВЬ**



**Реклама от Google**

► [Скачать](#)

► [Java coding](#)

► [Access 2007](#)

► [C# coding](#)

Copyright © [Realcoding.NET](#) 2003-2007. При перепечатке материалов ссылка на автора материала  
обязательна.

Сообщить об ошибке или написать письмо администрации [через форму контактов](#).

4 113  
2 925  
1 064

