

计算机等级二级JAVA算术运算符 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/138/2021_2022__E8_AE_A1_E7_AE_97_E6_9C_BA_E7_c97_138584.htm

Java 提供了丰富的运算符环境。Java 有4大类运算符：算术运算、位运算、关系运算和逻辑运算。Java 还定义了一些附加的运算符用于处理特殊情况。本章将描述Java所有的运算符，而比较运算符instanceof 将在第12章讨论。注意：如果你对C/C 熟悉，你将会高兴，因为Java 的绝大多数运算符和C/C 中的用法一样。但有一些微妙的差别，所以提醒你要仔细阅读。算术运算符用在数学表达式中，其用法和功能与代数学（或其他计算机语言）中一样，Java 定义了下列算术运算符（见表4-1）：

运算符含义 加法 减法（一元减号） * 乘法/ 除法% 模运算 递增运算 = 加法赋值= 减法赋值 *= 乘法赋值/= 除法赋值%= 模运算赋值--递减运算

算术运算符的运算数必须是数字类型。算术运算符不能用在布尔类型上，但是可以用在char类型上，因为实质上在Java 中，char类型是int 类型的一个子集。

4.1.1 基本算术运算符 基本算术运算符加、减、乘、除可以对所有的数字类型操作。减运算也用作表示单个操作数的负号。记住对整数进行“/”除法运算时，所有的余数都要被舍去。下面这个简单例子示范了算术运算符，也说明了浮点型除法和整型除法之间的差别。

```
// Demonstrate the basic arithmetic operators. class BasicMath { public static void main(String args[]) {
```

```
// arithmetic using integers System.out.println("Integer Arithmetic"). int a = 1 1. int b = a * 3. int c = b / 4. int d = c - a. int e = -d.
```

```
System.out.println("a = " a). System.out.println("b = " b).
```

```
System.out.println("c = " c). System.out.println("d = " d). System.out.println("e = " e). }
```

```
System.out.println("Integer Arithmetic"). int a = 1 1. int b = a * 3. int c = b / 4. int d = c - a. int e = -d. System.out.println("a = " a). System.out.println("b = " b).
```

```
System.out.println("c = " c). System.out.println("d = " d).  
System.out.println("e = " e). // arithmetic using doubles  
System.out.println("\nFloating Point Arithmetic"). double da = 1 1.  
double db = da * 3. double dc = db / 4. double dd = dc - a. double  
de = -dd. System.out.println("da = " da). System.out.println("db = "  
db). System.out.println("dc = " dc). System.out.println("dd = " dd).  
System.out.println("de = " de). } }
```

当你运行这个程序，你会看到输出如下：Integer Arithmetic a = 2 b = 6 c = 1 d = -1 e = 1
Floating Point Arithmetic da = 2.0 db = 6.0 dc = 1.5 dd = -0.5 de = 0.5

4.1.2 模运算符 模运算符%，其运算结果是整数除法的余数。它能像整数类型一样被用于浮点类型（这不同于C/C，在C/C中模运算符%仅仅能用于整数类型）。下面的示例程序说明了模运算符%的用法：

```
class Modulus { public static void  
main(String args[]) { int x = 42. double y = 42.25.
```

```
System.out.println("x mod 10 = " x % 10). System.out.println("y  
mod 10 = " y % 10). } }
```

当你运行这个程序，你会看到输出如下：
x mod 10 = 2 y mod 10 = 2.25

4.1.3 算术赋值运算符 Java 提供特殊的算术赋值运算符，该运算符可用来将算术运算符与赋值结合起来。你可能知道，像下列这样的语句在编程中是很常见的：

```
a = a + 4.
```

在Java中，你可将该语句重写如下：

```
a += 4.
```

该语句使用“+”进行赋值操作。上面两行语句完成的功能是一样的：使变量a的值增加4。下面是另一个例子：

```
a = a % 2.
```

该语句可简写为：

```
a %= 2.
```

在本例中，%=算术运算符的结果是a/2的余数，并把结果重新赋给变量a。这种简写形式对于Java的二元（即需要两个操作数的）运算符都适用。其语句格式为：

```
var = var op expression.
```

可以被重写为：

```
var op =
```

expression. 这种赋值运算符有两个好处。第一，它们比标准的等式要紧凑。第二，它们有助于提高Java的运行效率。由于这些原因，在Java的专业程序中，你经常会看见这些简写的赋值运算符。100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com