## Operaciones Aritmeticos

Como su nombre indica, los operadores aritméticos permiten realizar operaciones aritméticas (suma, resta, producto, etc…).

Podemos distinguir dos tipos de operaciones aritméticas: los que realizan **operaciones básicas** y los que realizan operaciones de **incremento y decremento**.

## Operaciones básicas

Las operaciones aritméticas básicas son las que se indican en la siguiente tabla:

|  |  |  |
| --- | --- | --- |
| **Operador** | **Descripción** | **Sintaxis** |
| + | Suma | a + b |
| – | Resta | a – b |
| \* | Producto | a \* b |
| / | División | a / b |
| % | Módulo | a % b |

### Suma (+)

Permite sumar dos valores (variables, literales, expresiones, etc.) de tipo numérico.

**Ejemplos:**

int a = 20;

int c = 10;

int resultado = a + b;

/\* La variable resultado toma valor 30, que es el resultado de sumar las variables a y b \*/

double a = 5.3;

int b = 2;

double resultado = a + b;

/\* En este caso están implicados en la suma dos datos de diferente tipo, pero numéricos. La variable resultado tomará valor 7.3 \*/

int a = 3;

int b = 9;

int resultado = a + b + 35;

/\* La variable resultado tomará valor 47 (3 + 9 + 35). Se están sumando variables y literales. Es posible puesto que todos son de tipo numérico. Se pueden unir en una misma expresión varias sumas para conseguir una suma total. \*/

### Resta (-)

Permite restar dos valores (variables, literales, expresiones, etc.) de tipo numérico (byte, short, int, double, etc.). Al operando que está en la parte izquierda del operador se le restará el valor del que está a la derecha, pero sin quedar el resultado final almacenado en ninguno de los dos operándos.

**Ejemplos:**

int a = 10;

int c = 2;

int resultado = a + b;

/\* La variable resultado toma valor 8, que es el resultado de restar 10 - 2. Las variables a y b mantendrán sus valores después de la operación \*/

double a = 5.3;

int b = 2;

double resultado = a - b;

/\*La variable resultado tomará valor 3.3 (5.3 - 2). En este caso están implicados en la suma dos datos de diferente tipo, pero numéricos. \*/

int a = 3;

int b = 9;

int resultado = a - b;

/\* La variable resultado tomará valor -6 (3 - 9). El resultado es negativo y así será devuelto por la operación \*/

### Producto (\*)

Permite multiplicar dos valores (variables, literales, expresiones, etc.) de tipo numérico (byte, short, int, double, etc.).

**Ejemplos:**

int a = 10;

int c = 5;

int resultado = a \* b;

/\* La variable resultado toma valor 50, que es el resultado de multiplicar las variables a y b \*/

double a = 2.5;

int b = 2;

double resultado = a \* b;

/\* En este caso están implicados en la multiplicación dos datos de diferente tipo, pero numéricos. La variable resultado tomará valor 5.0 \*/

int a = 4;

int b = 2;

int resultado = a \* b \* 3;

/\* La variable resultado tomará valor 24 (4 \* 2 \* 3). Se están multiplicando variables y literales. Se pueden unir en una misma expresión varios para conseguir un producto total. \*/

### División (/)

Permite dividir dos valores (variables, literales, expresiones, etc.) de tipo numérico (byte, short, int, double, etc.). El operando situado a la izquierda del operador se divide entre el operando situado a la derecha.

Hay que tener en cuenta en tipo de los datos implicados en la operación, puesto que el resultado varía si se trata de datos enteros o de datos en punto flotante.

Cuando se utiliza éste operador con datos de tipo entero se realiza una división entera. Se puede perder precisión en el resultado puesto que se trunca el dato y se pierde la parte decimal.

**Ejemplo:**

int a = 5;

int b = 2;

int resultado = a / b;

/\* 5/2 da como resultado 2.5, pero al tratarse de datos de tipo entero, el resultado también será entero. Se pierde la parte decimal, por lo que la variable resultado tomará valor 2 \*/

Cuando en la operación hay implicado al menos un dato de tipo punto flotante (float o double), el resultado no perderá precisión, ya que también será punto flotante y conservará la parte decimal.

**Ejemplo:**

int a = 5;

float b = 2f;

float resultado = a / b;

/\* El ejemplo es el mismo de antes, con la diferencia de que ahora uno de los operandos es de tipo float. En este caso no se truncará el resultado y no se perderá la parte decimal. La variable resultado almacenará el valor 2.5 \*/

### Módulo (%)

Permite calcular el resto de una división entera. Los operadores implicados tendrán que ser numéricos de tipo entero (variables, literales, expresiones, etc.).

El operando situado en la parte izquierda del operador se divide entre el segundo. El resultado será el resto de dicha división.

**Ejemplo:**

int a = 5;

int b = 2;

int resultado = 5 % 2;

/\* El resto de dividir 5 entre 2 es 2 y el resto de la división es 1. Ese valor se almacena en la variable resultado \*/

## Operaciones de incremento y decremento

Estos operadores permiten incrementar o decrementar en una unidad el valor de una variable de tipo numérico.

|  |  |  |
| --- | --- | --- |
| **Operador** | **Descripción** | **Sintaxis** |
| ++ | Incremento | a++; (notación postfija) ++a; (notación prefija) |
| – – | Decremento | a- -; (notación postfija) – – a; (notación prefija) |

### **Operador de incremento**

Incrementa en una unidad el valor de la variable implicada en la operación. Tanto en notación prefija como postfija.

**Ejemplo:**

int a = 5;

int b = 4;

a++;

++b;

/\* Después de realizar el incremento a tendrá valor 6 y b tendrá valor 5 \*/

Si la operación se utiliza en otra expresión, hay que tener en cuenta la notación usada, puesto que el resultado final se verá afectado.

La **notación postfija** devuelve el valor de la variable antes de realizar el incremento y modificarla.

**Ejemplo:**

int a = 5;

int b = a++;

/\* b toma valor 5 y a toma valor 6. Se asigna a b el valor de a y después se realiza el incremento sobre a \*/

La **notación prefija** incrementa el valor de la variable antes de devolverlo.

**Ejemplo:**

int a = 5;

int b = ++a;

/\* b toma valor 6 y a toma valor 6. Se realiza el incremento sobre a y después se asigna el valor a b \*/

### Operador de decremento

Decrementa en una unidad el valor de la variable implicada en la operación. La notación prefija y postfija se aplican igual que en el operador de incremento, pero decrementándo el valor de la variable.