Módulo 1: Elementos Básicos del Lenguaje

3. Operaciones Básicas, Operadores y Jerarquía de Operadores

Operadores Aritméticos

Operador en C++	Significado
+	suma
-	resta
*	multiplicación
/	división
%	residuo

Todos los operadores que se muestran en esta tabla son binarios; es decir, trabajan con dos operandos.

Los operadores **+**, **-** y ***** funcionan de la manera conocida.

El operador **/** funciona de diferente manera si trabaja con datos de tipo entero o de tipo flotante. Con datos de tipo flotante

funciona de la manera tradicional; pero al realizarse una división entre dos números enteros, el operador **/** regresa el cociente de

la división entera; es decir, regresa la parte entera del resultado (si hay fracción la elimina).

Por ejemplo:

2/3 da como resultado 0

pero

2.0/3.0 da como resultado 0.66666

Si hay operandos de diferentes tipos de datos, se convierten al tipo de datos más amplio y el tipo del valor resultante es del tipo

más amplio. Por ejemplo, si hay enteros y flotantes, todos los números se convierten a flotantes y el resultado se calcula como

flotante.

Por ejemplo:

4/3.0 da como resultado 1.3333

El operador **%** calcula el residuo de la división entera y sólo existe para datos de tipo entero

Por ejemplo:
10%3 da como resultado 1

Otros operadores de Asignación

En C++ es posible abreviar algunas expresiones de asignación como se muestra en la siguiente tabla:

Operador	Expresión equivalente
v + = e	v = v + e
v - = e	v = v - e
v * = e	v = v * e
v / = e	v = v / e
v % = e	v = v % e

Otros Operadores aritméticos

En C++ existen también los siguientes operadores aritméticos:

++ incremento

-- decremento

Es decir:
x++ ó ++x es equivalente a x = x+1

x-- ó --x es equivalente a x = x-1

Estos operadores son unitarios, es decir, trabajan con un solo operando y solamente se pueden utilizar con variables de tipo

entero.

Los operadores se pueden utilizar antes o después del nombre de la variable y funcionan de diferente manera:

- Si se ponen antes, primero se realiza la operación (incremento o decremento) y luego se utiliza el valor de la variable en la

  expresión en la que se encuentre.

- Si se pone después, primero se utiliza el valor de la variable en la expresión y luego se lleva a cabo la operación (incremento

  o decremento).

Por ejemplo:

Supón que a = 10 y c = 4

La operación v = a * c++; v toma el valor de 40 y c queda con el valor de 5

La operación v = a * ++c; v toma el valor de 50 y c queda con el valor de 5

Jerarquía de los operadores aritméticos

Prioridad	Operadores	Asociatividad
1	( )	Empezando por los paréntesis más internos
2	++, --, +(positivo), -(negativo)	De derecha a izquierda, ++ y -- dependiendo de la posición
3	*, /, %	De izquierda a derecha
4	+, -	De izquierda a derecha
5	=,+=,-=,*=, /=,%=	De derecha a izquierda

Algunas Funciones Matemáticas Predefinidas

C++ contiene una serie de funciones matemáticas que puedes utilizar en tus programas, algunas de estas funciones son:

abs(x) obtiene el valor absoluto de x, x debe ser entero

sqrt(x) obtiene la raíz cuadrada de x, x debe ser positivo

pow(x,y) calcula x elevado a la potencia y, pueden ser enteros o flotantes

el uso de estas funciones requiere de la librería <math.h>

Operadores de Relacionales

Los operadores relacionales que tiene C++ son :

Operador en C++	Significado
==	Igual
!=	Diferente
<	Menor que
>	Mayor que
<=	Menor o igual que

>=	Mayor o igual que

Operadores Lógicos

Los operadores lógicos que maneja C++ son:

Operador en C++	Significado
||	or
&&	and
!	not

El resultado de las expresiones que incluyen operadores relacionales o lógicos generan resultados verdaderos o falsos. Para C++ el

cero representa falso y cualquier otro número verdadero.

Ejemplos de construcción de expresiones

Expresión para saber si un número es par:

(num % 2 == 0)

Expresión para saber si un número A está en el rango 5 a 300 incluyendo los extremos

(num > = 5) && (num <=300)

Ejercicios

I. Realiza las siguientes expresiones en C++ para obtener el valor con el que quedará la variable x.

1. x = 7 + 3 * 6 / 2 - 1; ver solución

2. x = (3 * 9 * (3 + (9 * 3 / (3)))) ; ver solución

Dados los valores iniciales de a = 15, b = 3, cual será el valor final de las variables después de ejecutar las siguientes expresiones (individualmente).

3. a = a * 6 / a ++ ; ver solución

4. b = --b * a++ / b; ver solución

Construye las siguientes expresiones.

5. Expresión para saber si 3 número son iguales.

6. Expresión para determinar que la variable edad está fuera del rango de 15 a 40 sin incluir los extremos

ver solución de ambos ejercicios

Material de Apoyo

Presentacion de Expresiones

Ligas sugeridas

http://www.cplusplus.com/doc/tutorial/

http://www.cs.wustl.edu/~schmidt/C++/

Regresar

Siguiente módulo