

Análisis de Expresiones en C

Expresiones Aritméticas

Son aquellas que involucran operadores matemáticos y devuelven valores numéricos.

Importante:

- Si ambos operandos son enteros (`int`), la división es entera.
- Si al menos uno es `float` o `double`, se obtiene un resultado decimal.

Operadores Aritméticos en C

Operador	Descripción	Ejemplo	Resultado
+	Suma	$5 + 3$	8
-	Resta	$5 - 3$	2
*	Multiplicación	$5 * 3$	15
/	División	$5 / 3$	1 (división entera)
%	Módulo	$5 \% 3$	2 (residuo de la división)

Expresiones Relacionales

Son aquellas que comparan valores y devuelven un resultado **booleano** (0 para falso y 1 para verdadero).

Operador	Descripción	Ejemplo (a = 5, b = 3)	Resultado
==	Igual a	a == b	0 (Falso)
!=	Diferente de	a != b	1 (Verdadero)
>	Mayor que	a > b	1
<	Menor que	a < b	0
>=	Mayor o igual	a >= b	1
<=	Menor o igual	a <= b	0

Expresiones Lógicas

Se usan para combinar condiciones y devuelven un resultado booleano (0 o 1).

Operador	Descripción	Ejemplo (a = 5, b = 3)	Resultado
&&	AND lógico	(a > 2) && (b < 5)	1
	OR logico	`(a > 2) (b < 5)	1
!	NOT lógico	!(a == 5)	0

Precedencia y Asociatividad de Operadores

Cuando se combinan varios operadores, se evalúan siguiendo un orden de precedencia.

Prioridad	Operadores	Asociatividad
1 (Mayor)	()	Izquierda a derecha
2	!, ++, --, - (unario)	Derecha a izquierda
3	*, /, %	Izquierda a derecha
4	+, -	Izquierda a derecha
5	<, <=, >, >=	Izquierda a derecha
6	==, !=	Izquierda a derecha
7	&&	Izquierda a derecha
8		Izquierda a derecha
9 (Menor)	=	Derecha a izquierda

REFERENCIAS

ISO/IEC. (2018). *Programming languages — C* (ISO/IEC 9899:2018). International Organization for Standardization.

<https://www.iso.org>

Kernighan, B. W., & Ritchie, D. M. (1988). *The C programming language* (2nd ed.). Prentice Hall.

King, K. N. (2008). *C programming: A modern approach* (2nd ed.). W. W. Norton & Company.

GeeksforGeeks. (2023). *Operators in C and C++*. <https://www.geeksforgeeks.org/operators-in-c-and-cpp/>

Gaddis, T. (2021). *Starting out with C++ from control structures through objects* (10th ed.). Pearson.