

Arithmetic & Logic Operators order of precedence

Precedence	Operator	Description	Associativity	example
t = 2, u = 7, v = 9, w = 4, x = 4, y = 2, z = 3				
1	- !	unary negation logical not	right to left right to left	-x ! x (any value other than 4 evaluates to true)
2	* / %	multiplication division modulus (int division remainder)	left to right left to right left to right	x * y (8) x / y (2) x % y (0)
3	+ -	addition subtraction	left to right left to right	x + y (6) x - y (2)
4	< > <= >=	less than? greater than? less than or equal to? greater than or equal to?	left to right left to right left to right left to right	x < y (f) x > y (t) x <= y (f) x >= w (t)
5	== !=	equal to? not equal to?	left to right left to right	x == y (f) x != y (t)
6	&&	AND	left to right	(x > y) && (x == z) both conditions within the compound statement must be true to evaluate to true (f)
7		OR	left to right	(x < y) (w == x) only one condition must be true for the compound statement to evaluate to true (t)
8	^	exclusive or	left to right	(x < y) ^ (w == x) one or the other must be true but not both (f)
	+= -= *= /=	combined assignment combined assignment combined assignment combined assignment	right to left right to left right to left right to left	u += v; (u == 16) v -= w; (v == 5) w *= x; (w == 16) x /= y; (x == 2) z %= y; (z == 1)

Logical Expressions evaluate to:
True (1)
or
False (0)