C Operator Precedence Table

C operators are listed in order of *precedence* (highest to lowest). Their *associativity* indicates in what order operators of equal precedence in an expression are applied.

Operator	Description	Associativity
0	Parentheses: grouping or function call	left-to-right
[]	Brackets (array subscript)	
•	Member selection via object name	
-> ++	Member selection via pointer Postfix increment/decrement	
++	Prefix increment/decrement	right-to-left
+ -	Unary plus/minus	0
! ~	Logical negation/bitwise complement	
(<i>type</i>) *	Cast (convert value to temporary value of <i>type</i>)	
*	Dereference Address (of operand)	
sizeof	Determine size in bytes on this implementation	
* / %	Multiplication/division/modulus	left-to-right
+ -	Addition/subtraction	left-to-right
<< >>	Bitwise shift left, Bitwise shift right	left-to-right
< <=	Relational less than/less than or equal to	left-to-right
> >=	Relational greater than/greater than or equal to	
== !=	Relational is equal to/is not equal to	left-to-right
&	Bitwise AND	left-to-right
^	Bitwise exclusive OR	left-to-right
	Bitwise inclusive OR	left-to-right
&&	Logical AND	left-to-right
	Logical OR	left-to-right
?:	Ternary conditional	right-to-left
=	Assignment	right-to-left
+= -=	Addition/subtraction assignment	
*= /= 0/- 8	Multiplication/division assignment	
%= &= ^= =	Modulus/bitwise AND assignment Bitwise exclusive/inclusive OR assignment	
	Bitwise shift left/right assignment	
,	Comma (separate expressions)	left-to-right