Operator precedence and Associativity in C programming language

Operator Precedence in C
Operator precedence determines which operator is evaluated first when an expression has more than one operators. For example 100-2*30 would yield 40, because it is evaluated as 100 – (2*30) and not (100-2)*30. The reason is that multiplication * has higher precedence than subtraction(-).

Associativity in C
Associativity is used when there are two or more operators of same precedence is present in an expression. For example multiplication and division arithmetic operators have same precedence, lets say we have an expression 5*2/10, this expression would be evaluated as (5*2)/10 because the associativity is left to right for these operators. Similarly 20/2*5 would be calculated as (20*2)/5.

Operator precedence and Associativity Table in C Programming

Description	Operator	Associativity
Function expression	( )	Left to Right
Array Expression	[]	Left to Right
Structure operators	->	Left to Right

Unary minus	–	Right to Left
Increment & Decrement	— ++	Right to Left
One’s compliment	~	Right to Left
Pointer Operators	& *	Right to Left
Type cast	(data type)	Right to Left
size of operator	sizeof	Right to Left
Left and Right Shift	>> <<	
Arithmetic Operators		
Multiplication operator, Divide by, Modulus	*, /, %	Left to Right
Add, Substract	+, –	Left to Right

Relational Operators		
Less Than	<	Left to Right
Greater than	>	Left to Right
Less than equal to	<=	Left to Right
Greater than equal to	>=	Left to Right
Equal to	==	Left to Right
Not equal	!=	Left to Right
Logical Operators		
AND	&&	Left to Right
OR	||	Left to Right

NOT	!	Right to Left
Bitwise Operators		
AND	&	Left to Right
Exclusive OR	^	Left to Right
Inclusive OR	|	Left to Right
Assignment Operators		
	=	Right to Left
	*=	Right to Left
	/=	Right to Left
	%=	Right to Left

	+=	Right to Left
	-=	Right to Left
	&=	Right to Left
	^=	Right to Left
	|=	Right to Left
	<<=	Right to Left
	>>=	Right to Left
Other Operators		
Comma	,	Right to Left
Conditional Operator	?:	Right to Left