Operator precedence and association

C++/C#		Java		JS		PHP		PERL		Python		Ruby		VB	
OP	A	OP	A	OP	A	OP	A	OP	A	OP	A	OP	A	OP	A
()	L	()	L	()	N	()	N	()	N	()	L	[][]=	N	()	N
[]	L	[]	L	++	R	[]	N	++	N	**	R	**	R	^	R
.	L	.	L	--	R	!	R	--	N	*	L	!	R	*	L
->	L	++	L	-- N	R	~	R	-	R	@	L	~	R	/	L
++	L	--	L	!	R	++	R	~	R	/	L	+	R	\	L
--	L	++	R	*	L	--	R	!	R	%	L	-	R	Mod	L
++	R	--	R	/	L	*	L	**	R	//	L	*	L	+	L
--	R	+	R	%	L	/	L	=~	L	+	L	/	L	-	L
+	R	-	R	+	L	%	L	!~	L	-	L	%	L	&	L
-	R	!	R	-	L	+	L	*	L	<<	L	+	L	=	R
!	R	~	R	+ C	R	-	L	/	L	>>	L	-	L	=	L
~	R	*	L	<	L	.	L	%	L	<	L	<<	L	<	L
*	R	/	L	<=	L	<<	L	x	L	<=	L	>>	L	>	L
&	R	%	L	>	L	>>	L	+	L	>		&	L	<=	L
*	L	+	L	>=	L	<	N	-	L	>=		^	L	>=	L
/	L	-	L	==	L	<=	N	.	L	==	L	|	L	<>	L
%	L	<<	L	!=	L	>	N	<<	L	!=	L	<	L	And	L
+	L	>>	L	===	L	>=	N	>>	L	is	L	<=	L	Eqv	L
-	L	>>>	L	!==	L	==	N	-e	N	is not	L	>	L	Imp	L
<<	L	<	L	&&	L	!=	N	-r	N	in		>=	L	Or	L
>>	L	<=	L	||	L	===	N	<	L	not in		<=>	L	Xor	L
<	L	>		?:	R	!===	N	<=	L	&	L	==	L	Not	L
<=	L	>=		=	R	&	L	>	L	^	L	===	L		
>	L	==	L	+=	R	^	L	>=	L	|	L	!=	L		
>=	L	!=	L	-=	R	|	L	lt	L	not	R	=~	L		
==	L	&	L	*=	R	&&	L	le	L	and	L	!~	L		
!=	L	^	L	/=	R	||	L	gt	L	or	L	&&	L		
&	L	|	L	%=	R	?:	R	ge	L	=	R	||	L		
^	L	&&	L	**=	R	=	R	==	L	+=	R	.. …	L		
|	L	||	L			+=	R	!=	L	-=	R	?:	R		
&&	L	?:	R			-=	R	<=>	L	*=	R	=	R		
||	L	=	R			*=	R	eq	L	/=	R	+=	R		
?:	R	+=	R			**=	R	ne	L	%=	R	-=	R		
=	R	-=	R			/=	R	cmp	L	//=	R	*=	R		
+=	R	*=	R			.=	R	&	L	**=	R	/=	R		
-=	R	/=	R			%=	R	|	L	&=	R	%=	R		
*=	R	%=	R			&=	R	^	L	^=	R	**=	R		
/=	R	&=	R			|=	R	&&	L	|=	R	&&=	R		
%=	R	|=	R			^=	R	||	L	<<=	R	||=	R		
&=	R	^=	R			<<=	R	..	L	>>=	R	&=	R		
^=	R	<<=	R			>>=	R	? and :	R			|=	R		
|=	R	>>=	R			??=	R	+=	R			^=	R		
<<=	R					and	L	-=	R			<<=	R		
>>=	R					xor	L	*=	R			>>=	R		
,	L					or	L	/=	R			not	R		
						,	L	%=	R			or	L		
								**=	R			and	L		
								,	L						
								not	L						
								and	L						
								xor	L						
								or	L						

Operator precedence and associativity symbols by computer language. In this table, operators enclosed in the same border have equal precedence and their associativity is shown on the column next to the symbols. The pink color of a cell indicates a group of operators and the light yellowish color indicates single operators per level. Note that the abbreviation OP means Order of Precedence; A = Associativity; N = Order of direction is not applicable here - non-associative; L = left-to-right; R = right-to-left. Some lesser known and used operator symbols are not shown here. The plus and minus signs belonging to addition and subtraction can be seen immediately below multiplication and division. Other plus or minus symbols present either above or below that position have dual roles, such as the plus sign in JavaScript which uses the symbol for both concatenation and addition. Other interesting observations are: In VB the "\" means integer division; in Ruby "=~" means matching operator; also in Ruby "!~" means NOT match. In C# the "^" means bitwise XOR, whereas in VB it means exponentiation.

Paul A. Gagniuc. *An Introduction to Programming Languages: Simultaneous Learning in Multiple Coding Environments*. Synthesis Lectures on Computer Science. Springer International Publishing, 2023, pp. 1-280.