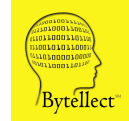


C++ Programming Language Operators



Operator	Description	Operator			Associativity	C++ Only	Precedence	Notes
		Binary Operator	Unary Operator	Ternary Operator				
::	Scope resolution	✓			none	✓	Highest	
:::	Global scope resolution		✓		none	✓		
--	Arithmetic - decrement (postfix)		✓		left-to-right		Operator Precedence	C-style casting
()	Parenthesis, function call	✓			left-to-right			
.	Member of structure/union	✓			left-to-right			
[]	Array subscript	✓			left-to-right			
++	Arithmetic - increment (postfix)		✓		left-to-right			
->	Pointer to structure/union	✓			left-to-right			
typeid	Data type info at runtime		✓		left-to-right	✓		
const_cast	Constant type conversion		✓		left-to-right	✓		
dynamic_cast	Dynamic type conversion		✓		left-to-right	✓		
reinterpret_cast	Reinterpreted type conversion		✓		left-to-right	✓		
static_cast	Static type conversion		✓		left-to-right	✓		
-	Arithmetic - minus sign (negative)		✓		right-to-left			
--	Arithmetic - decrement (prefix)		✓		right-to-left			
!	Logical NOT		✓		right-to-left			
&	Address Reference (address-of)		✓		right-to-left			
*	Address Dereference (indirection)		✓		right-to-left			
~	Bitwise - one's complement		✓		right-to-left			
+	Arithmetic - plus sign (positive)		✓		right-to-left			
++	Arithmetic - increment (prefix)		✓		right-to-left			
delete	Free dynamic object		✓		right-to-left	✓		
new	Allocate dynamic object		✓		right-to-left	✓		
sizeof	Data type size in bytes		✓		right-to-left			
type cast	Explicit data type conversion		✓		right-to-left			
.*	Pointer to member	✓			left-to-right	✓		
->*	Pointer to member	✓			left-to-right	✓		
%	Arithmetic - modulus	✓			left-to-right			
*	Arithmetic - multiplication	✓			left-to-right			
/	Arithmetic - division	✓			left-to-right			
-	Arithmetic - subtraction	✓			left-to-right			
+	Arithmetic - addition	✓			left-to-right			
<<	Bitwise - shift-left	✓			left-to-right			
>>	Bitwise - shift-right	✓			left-to-right			
<	Relational - less-than	✓			left-to-right			
<=	Relational - less-than-or-equal	✓			left-to-right			
>	Relational - greater-than	✓			left-to-right			
>=	Relational - greater-than-or-equal	✓			left-to-right			
!=	Relational - not-equal	✓			left-to-right			
==	Relational - equal	✓			left-to-right			
&	Bitwise AND	✓			left-to-right			
^	Bitwise XOR (exclusive OR)	✓			left-to-right			
	Bitwise OR (inclusive OR)	✓			left-to-right			
&&	Logical AND	✓			left-to-right			
	Logical OR	✓			left-to-right			
? :	Conditional			✓	right-to-left			
%=	Modulus assignment	✓			right-to-left			
&=	Bitwise AND assignment	✓			right-to-left			
*=	Multiplication assignment	✓			right-to-left			
/=	Division assignment	✓			right-to-left			
^=	Bitwise XOR assignment	✓			right-to-left			
=	Bitwise OR assignment	✓			right-to-left			
+=	Addition assignment	✓			right-to-left			
<<=	Bitwise shift-left assignment	✓			right-to-left			
=	Assignment	✓			right-to-left			
-=	Subtraction assignment	✓			right-to-left			
>>=	Bitwise shift-right assignment	✓			right-to-left			
throw	Exception throw		✓		right-to-left	✓		
,	Comma sequence	✓			left-to-right			

Operators in the same group have equal precedence. Operator overloading does not alter precedence.