Precedence of Operators

In the following table of operator precedence, the Turbo C++ operators are divided into 16 categories.

The #1 category has the highest precedence; category #2 (Unary operators) takes second precedence, and so on to the Comma operator, which has lowest precedence.

The operators within each category have equal precedence.

The Unary (category #2), Conditional (category #14), and Assignment (category #15) operators associate right-to-left; all other operators associate left-to-right.

#	Category	Operator	What it is (or does)	Pág.
1	Highest	()	Function call	2
		[]	Array subscript	2
		->	C++ indirect component selector	2
		::	C++ scope access/resolution	3
		.	C++ direct component selector	2
2	Unary	!	Logical negation (NOT)	3
		~	Bitwise (1's) complement	3
		+	Unary plus	3
		-	Unary minus	3
		++	Preincrement or postincrement	4
		--	Predecrement or postdecrement	4
		&	Address	5
		*	Indirection	5
		sizeof	(returns size of operand, in bytes)	5
		new	(dynamically allocates C++ storage)	5
		delete	(dynamically deallocates C++ storage)	5
3	Multipli-	*	Multiply	6
	cative	/	Divide	6
		%	Remainder (modulus)	6
4	Member	.*	C++ dereference	3
	access	->*	C++ dereference	3
5	Addittive	+	Binary plus	4
		-	Binary minus	4
6	Shift	<<	Shift left	6
		>>	Shift right	6
7	Relational	<	Less than	7
		<=	Less than or equal to	7
		>	Greater than	7
		>=	Greater than or equal to	7
8	Equality	==	Equal to	7
		!=	Not equal to	7
9		&	Bitwise AND	8
10		^	Bitwise XOR	8
11		|	Bitwise OR	8
12		&&	Logical AND	8
13		||	Logical OR	8
14	Conditional	?:	(a ? x : y means "if a then x, else y")	8
15	Asignement	=	Simple assignement	9
		*=	Assign product	9
		/=	Assign quotient	9
		%=	Assign remainder (modulus)	9
		+=	Assign sum	9
		-=	Assign difference	9
		&=	Assign bitwise AND	9
		^=	Assign bitwise XOR	9
		|=	Assign bitwise OR	9
		<<=	Assign left shift	9
		>>=	Assign right shift	9
16	Comma	,	Evaluate	9

All of the operators in this table can be overloaded except the following:

. C++ direct component selector
.* C++ dereference
:: C++ scope access/resolution
?: Conditional