

Ассоциативность	Оператор	Описание	Пример
1. Нет	::	Глобальная область видимости (унарный)	::name
	::	Область видимости класса (бинарный)	class_name::member_name
2. L -> R	()	Круглые скобки	(expression)
	()	Вызов функции	function_name(parameters)
	()	Инициализация	type name(expression)
	{}	uniform-инициализация (C++11)	type name{expression}
	type()	Конвертация типа	new_type(expression)
	type{}	Конвертация типа (C++11)	new_type{expression}
	[]	Индекс массива	pointer[expression]
	.	Доступ к члену объекта	object.member_name
	->	Доступ к члену объекта через указатель	object_pointer->member_name
	++	Пост-инкремент	lvalue++
	--	Пост-декремент	lvalue--
	typeid	Информация о типе во время выполнения	typeid(type) or typeid(expression)
	const_cast	Cast away const	const_cast(expression)
	dynamic_cast	Type-checked cast во время выполнения	dynamic_cast(expression)
	reinterpret_cast	Конвертация одного типа в другой	reinterpret_cast(expression)
static_cast	Type-checked cast во время компиляции	static_cast(expression)	
3. R -> L	+	Унарный плюс	+expression
	-	Унарный минус	-expression
	++	Пре-инкремент	++lvalue
	--	Пре-декремент	--lvalue
	!	Логическое НЕ (NOT)	!expression
	~	Побитовое НЕ (NOT)	~expression
	(type)	C-style cast	(new_type)expression
	sizeof	Размер в байтах	sizeof(type) or sizeof(expression)
	&	Адрес	&lvalue
	*	Разыменование	*expression
	new	Динамическое выделение памяти	new type
	new[]	Динамическое выделение массива	new type[expression]
	delete	Динамическое удаление памяти	delete pointer
	delete[]	Динамическое удаление массива	delete[] pointer
4. L -> R	->*	Member pointer selector	object_pointer->*pointer_to_member
	.*	Member object selector	object.*pointer_to_member
5. L -> R	*	Умножение	expression * expression
	/	Деление	expression / expression
	%	Деление с остатком	expression % expression
6. L -> R	+	Сложение	expression + expression
	-	Вычитание	expression - expression
7. L -> R	<<	Побитовый сдвиг влево	expression << expression
	>>	Побитовый сдвиг вправо	expression >> expression
8. L -> R	<	Сравнение: меньше чем	expression < expression
	<=	Сравнение: меньше чем или равно	expression <= expression
	>	Сравнение: больше чем	expression > expression
	>=	Сравнение: больше чем или равно	expression >= expression
9. L -> R	Ошибка: 520	Равно	expression == expression
	!=	Не равно	expression != expression
10. L -> R	&	Побитовое И (AND)	expression & expression
11. L -> R	^	Побитовое исключающее ИЛИ (XOR)	expression ^ expression
12. L -> R		Побитовое ИЛИ (OR)	expression expression
13. L -> R	&&	Логическое И (AND)	expression && expression
14. L -> R		Логическое ИЛИ (OR)	expression expression
15. R -> L	?:	Тернарный условный оператор	expression ? expression : expression
	=	Присваивание	lvalue = expression
	*=	Умножение с присваиванием	lvalue *= expression
	/=	Деление с присваиванием	lvalue /= expression
	%=	Деление с остатком и с присваиванием	lvalue %= expression
	+=	Сложение с присваиванием	lvalue += expression
	-=	Вычитание с присваиванием	lvalue -= expression
	<<=	Присваивание с побитовым сдвигом влево	lvalue <<= expression
	>>=	Присваивание с побитовым сдвигом вправо	lvalue >>= expression
	&=	Присваивание с побитовой операцией И (AND)	lvalue &= expression
	=	Присваивание с побитовой операцией ИЛИ (OR)	lvalue = expression
^=	Присваивание с побитовой операцией «Исключающее ИЛИ» (XOR)	lvalue ^= expression	
16. R -> L	throw	Генерация исключения	throw expression
17. L -> R	,	Оператор Запятая	expression, expression