

Apêndice F

Precedência e associatividade no C++

A tabela seguinte sumariza as regras de precedência e associatividade dos operadores do C++. Operadores colocados na mesma secção têm a mesma precedência. As secções são separadas por § e apresentadas por ordem decrescente de precedência. Quanto à associatividade, apenas os operadores unários (com um único operando) e os operadores de atribuição se associam à direita: todos os outros associam-se à esquerda, como é habitual.

Descrição	Sintaxe (itálico: partes variáveis da sintaxe)
resolução de âmbito	<i>nome_de_classe :: membro</i>
resolução de âmbito	<i>nome_de_espaco_nominativo :: membro</i>
global	<i>:: nome</i>
global	<i>:: nome_qualificado</i>
§	
selecção de membro	<i>objecto . membro</i>
selecção de membro	<i>ponteiro -> membro</i>
indexação	<i>ponteiro [expressão_inteira]</i>
invocação de função	<i>expressão (lista_expressões)</i>
construção de valor	<i>tipo (lista_expressões)</i>
incrementação sufixa	<i>lvalor ++</i>
decrementação sufixa	<i>lvalor --</i>
identificação de tipo	<i>typeid (tipo)</i>
identificação de tipo durante a execução	<i>typeid (expressão)</i>
conversão verificada durante a execução	<i>dynamic_cast < tipo > (expressão)</i>
conversão verificada durante a compilação	<i>static_cast < tipo > (expressão)</i>
conversão não verificada (evitar)	<i>reinterpret_cast < tipo > (expressão)</i>
conversão constante (evitar)	<i>const_cast < tipo > (expressão)</i>
§	
tamanho de objecto	<i>sizeof expressão</i>
tamanho de tipo	<i>sizeof (tipo)</i>
incrementação prefixa	<i>++ lvalue</i>

Descrição	Sintaxe (itálico: partes variáveis da sintaxe)
decrementação prefixa	-- <i>lvalue</i>
negação <i>bit-a-bit</i> ou complemento para um	compl <i>expressão</i> (ou ~)
negação	not <i>expressão</i>
simétrico	- <i>expressão</i>
identidade	+ <i>expressão</i>
endereço de	& <i>lvalue</i>
conteúdo de	* <i>expressão</i>
construção de variável dinâmica	new <i>tipo</i>
construção de variável dinâmica com inicializadores	new <i>tipo</i> (<i>lista_expressões</i>)
construção localizada de variável dinâmica	new (<i>lista_expressões</i>) <i>tipo</i>
construção localizada de variável dinâmica com inicializadores	new (<i>lista_expressões</i>) <i>tipo</i> (<i>lista_expressões</i>)
destruição de variável dinâmica	delete <i>ponteiro</i>
destruição de matriz dinâmica	delete[] <i>ponteiro</i>
conversão de tipo de baixo nível (crime!)	(<i>tipo</i>) <i>expressão</i>
§ _____	
selecção de membro	<i>objecto</i> . * <i>ponteiro_para_membro</i>
selecção de membro	<i>ponteiro</i> -> * <i>ponteiro_para_membro</i>
§ _____	
multiplicação	<i>expressão</i> * <i>expressão</i>
divisão	<i>expressão</i> / <i>expressão</i>
resto da divisão inteira	<i>expressão</i> % <i>expressão</i>
§ _____	
adição	<i>expressão</i> + <i>expressão</i>
subtracção	<i>expressão</i> - <i>expressão</i>
§ _____	
deslocamento para a esquerda	<i>expressão</i> << <i>expressão</i>
deslocamento para a direita	<i>expressão</i> >> <i>expressão</i>
§ _____	
menor	<i>expressão</i> < <i>expressão</i>
menor ou igual	<i>expressão</i> <= <i>expressão</i>
maior	<i>expressão</i> > <i>expressão</i>
maior ou igual	<i>expressão</i> >= <i>expressão</i>
§ _____	
igual	<i>expressão</i> == <i>expressão</i>
diferente	<i>expressão</i> != <i>expressão</i> (ou not_eq)
§ _____	
conjunção <i>bit-a-bit</i>	<i>expressão</i> bitand <i>expressão</i> (ou &)
§ _____	
disjunção exclusiva <i>bit-a-bit</i>	<i>expressão</i> xor <i>expressão</i> (ou ^)
§ _____	
disjunção <i>bit-a-bit</i>	<i>expressão</i> bitor <i>expressão</i> (ou)

Descrição	Sintaxe (itálico: partes variáveis da sintaxe)
conjunção	§ <i>expressão and expressão</i> (ou <i>&&</i>)
disjunção	§ <i>expressão or expressão</i> (ou <i> </i>)
operador condicional	§ <i>expressão ? expressão : expressão</i>
atribuição simples	§ <i>lvalue = expressão</i>
multiplicação e atribuição	<i>lvalue *= expressão</i>
divisão e atribuição	<i>lvalue /= expressão</i>
resto e atribuição	<i>lvalue %= expressão</i>
adição e atribuição	<i>lvalue += expressão</i>
subtracção e atribuição	<i>lvalue -= expressão</i>
deslocamento para a esquerda e atribuição	<i>lvalue <=< expressão</i>
deslocamento para a direita e atribuição	<i>lvalue >>= expressão</i>
conjunção <i>bit-a-bit</i> e atribuição	<i>lvalue &= expressão</i> (ou <i>and_eq</i>)
disjunção <i>bit-a-bit</i> e atribuição	<i>lvalue = expressão</i> (ou <i>or_eq</i>)
disjunção exclusiva <i>bit-a-bit</i> e atribuição	<i>lvalue ^= expressão</i> (ou <i>xor_eq</i>)
lançamento de excepção	§ <i>throw expressão</i>
sequenciamento	<i>expressão , expressão</i>

