

PRECEDENZA DEGLI OPERATORI IN C

La seguente tavola delle precedenze degli operatori in C dovrebbe essere sempre a portata di mano. Soprattutto quando si sta imparando il linguaggio di programmazione. Ma, anche i più esperti potranno trarre beneficio dall'aver riassunti in una unica tabella tutti gli operatori con le loro precedenze e associatività (= ordine di precedenza che viene seguito quando si trovano più operatori di uguale precedenza nella stessa espressione).

Operatore	Descrizione	Associatività
()	Parentesi (chiamata a funzione) (vedi Nota 1)	da sx a dx
[]	Parentesi quadre (array)	
.	selezione di membri di oggetti dal nome	
->	come sopra, ma dal puntatore	
++ --	incrementi/decrementi postfissi (vedi Nota 2)	
++ --	incrementi/decrementi prefissi	da dx a sx
+ -	più/meno unari	
! ~	negazione logica/complemento bit a bit	
(type)	cast (conversione di tipi)	
*	deferenza	
& sizeof	indirizzo (di operandi) dimensioni in byte (dipendente dalla piattaforma)	
* / %	moltiplicazione/divisione/modulo	da sx a dx
+ -	Addizione/sottrazione	da sx a dx
<< >>	Shift a sinistra/destra bit a bit	da sx a dx
< <=	operatori relazionali minore di/minore o uguale a	da sx a dx
> >=	operatori relazionali maggiore di/maggiore o uguale a	
== !=	operatore relazionale uguale a/ non uguale a	da sx a dx
&	AND bit a bit	da sx a dx
^	OR esclusivo bit a bit	da sx a dx
	OR bit a bit	da sx a dx
&&	AND logico	da sx a dx
	OR logico	da sx a dx
? :	condizionale ternario	da dx a sx
=	assegnazione	da dx a sx
+= -=	assegnazione con addizione/sottrazione	
*= /=	assegnazione con moltiplicazione/divisione	
%= &=	assegnazione con modulo/AND bit a bit	
^= =	assegnazione con OR esclusivo bit a bit/Or bit a bit	
<<= >>=	assegnazione con shift a sinistra/shift a destra bit a bit	da sx a dx
,	virgola (separatore di espressioni)	da sx a dx

Nota 1: Le parentesi sono usate anche per raggruppare sottoespressioni per forzare le precedenze. Nel caso di parentesi nidificate, l'ordine di precedenza è dalla più interna alla più esterna.

Note 2: Gli operatori di incremento postfissi hanno alta precedenza, ma l'incremento/decremento reale è rimandato fino alla fine della valutazione dell'espressione. Ad esempio nell'esecuzione di $a=3*i++$, se i valeva 1 prima di questa istruzione, a varrà 3 e i varrà 2. Cioè i è incrementato al termine dell'esecuzione dell'istruzione.