**C Operator Precedence**

The following table lists the precedence and associativity of C operators. Operators are listed top to bottom, in descending precedence.

|  |  |  |  |
| --- | --- | --- | --- |
| **Precedence** | **Operator** | **Description** | **Associativity** |
| **1** | ++ --  ()  []  .  ->  (type){list} | Suffix/postfix increment and decrement  Function call  Array subscripting  Structure and union member access  Structure and union member access through pointer  Compound literal (C99) | Left-to-right |
| **2** | ++ --  + -  ! ~  (type)  \*  &  sizeof  \_Alignof | Prefix increment and decrement  Unary plus and minus  Logical NOT and bitwise NOT  Cast  Indirection (dereference)  Address-of  Size-of  Alignment requirement (C11) | Right-to-left |
| **3** | \* / % | Multiplication, division and remainder | Left-to-right |
| **4** | + - | Addition and subtraction |
| **5** | << >> | Bitwise left shift and right shift |
| **6** | < <=  > >= | For relational operators < and ≤ respectively  For relational operators > and ≥ respectively |
| **7** | == != | For relational = and ≠ respectively |
| **8** | & | Bitwise AND |
| **9** | ^ | Bitwise XOR (exclusive OR) |
| **10** | | | Bitwise OR (inclusive OR) |
| **11** | && | Logical AND |
| **12** | || | Logical OR |
| **13** | ?: | Ternary conditional | Right-to-left |
| **14** | =  += -=  \*= /= %=  <<= >>=  &= ^= |= | Simple assignment  Assignment by sum and difference  Assignment by product, quotient and remainder  Assignment by bitwise left shift and right shift  Assignment by bitwise AND, XOR and OR |
| **15** | , | Comma | Left-to-right |