

In an effort to bring the current manpage for the C operator table up to the current official Standard, I have gone back and compared the current manpage operator table against K&R, 1st edition, plus the C89/90, C99, C11, and C18 Draft Standards documents. I do not have access to the Official ISO C Standards documents, with the exception of the official ANSI published version of the C89/90 Standard in PDF form.

The following documents, and only these documents, are used as official references:

K&R	The C Programming Language, Kernighan & Ritchie,	1978
C89/90	ANSI/ISO 9899-1990 (Published by ANSI)	8/3/1992
C99	ISO/IEC 9899:TC3 WG14/N1256 Committee Draft	9/7/2007
C11	N1570 ISO/IEC9899:201x Committee Draft	4/12/2011
C18	ISO/IEC 9899:2017 N2176 C17 Ballot	6/2017

I have examined the C2x Working Group Draft simply to look forward for any potential changes in the future Standard. None found, at this time.

Current manpage, "operator":

-----

The current manpage for the C operators matches K&R, 1st edition.

I was quite surprised to see the manpage was never updated in the past 30 years, for any of the changes in any of the ANSI/ISO Standards.

I feel that the manpage for the operator table should be a true representation of the current ISO Standard. Why create a ANY Standard if we simply ignore them?

Other web sites, books, etc... use this operator table as the assumed Standard, instead of referring to the current Standard. Even the current version of the GNU C Reference Manual uses the K&R version of the table! See: <https://www.gnu.org/software/gnu-c-manual/gnu-c-manual.html#Operator-Precedence>

The current precedence table has 15 levels. C89/90 adds one new level. The new proposed Precedence Table should have 16 levels.

Other current C Language manpages may, and probably do need to be updated to comply with the current Standard. I am only concerned with the operator table manpage in this document.

Procedures used:

-----

In K&R, 1st edition, Appendix A, Section 7.x, each of the levels are listed in a different section, in the order of precedence, top to bottom, with references to associativity. I list the various sections, from K&R, in the current manpage operator table, in the PDF attached.

Each section shows operators on separate lines with appropriate terms such as "expression", "lvalue", "type-name", etc... to indicate the use of the operator.

The ANSI/ISO Standards follow this convention of separating the precedence levels in different sections. It is this convention of the separation of levels, and the operators, in bold font, in each section, that I use to determine the levels of the various operators in each of the updated precedence tables.

I am presenting the additions & changes for each of the ANSI & ISO Standards Documents, in separate precedence tables, in the PDF file attached, with the additions, or changes highlighted.

ISO refers to the C17/C18 Standard as C18, and I follow this naming convention in this document.

See: [http://www.iso-9899.info/wiki/The\\_Standard](http://www.iso-9899.info/wiki/The_Standard)

C18 makes no changes to the operator table, as far as I can see. The precedence table for C11 is presented in the PDF representing both C11 & C18.

The current draft, dated, March 13, 2019, of the C2x working committee (WG14) also shows no additional changes to the proposed new Standard. See: <http://www.open-std.org/jtc1/sc22/wg14/www/docs/n2346.pdf>

Perhaps the newly presented operator manpage, should also refer to the appropriate section of the current Standard, similar to how I have presented them in the attached PDF. A link to the current C11 and/or C18 Draft Standard would be needed if the Sections are listed.

C89/90 Standard:  
-----

The ANSI C89, and the ISO C90 Standards are for all practical purposes, equivalent.

"C89 and C90 are identical except for the frontmatter and section numbering."

[http://www.iso-9899.info/wiki/The\\_Standard](http://www.iso-9899.info/wiki/The_Standard)

As the date of publication in the copy I am using, is 1992, I believe the document represents and all changes made by the ISO in the C90 Standard.

"ANSI/ISO 9899-1990  
(revision and redesignation of  
ANSI X3.159-1989)"

and

"Approved August 3, 1992"

From Section 6.3, Expressions, on page 38 of my copy:

"Except as indicated by the syntax[Footnote 35] or otherwise specified later (for the function-call operator (), &&, ||, ?:, and the comma operator), the order of evaluation of subexpressions and the order in which side effects take place as both unspecified"

[Footnote 35]

"...

Within each major subclause, the operators have the same precedence. Left or right associativity is indicated in each subclause by the syntax for the expressions discussed therein."

Based on the above quote, and in later Standards, it is my interpretation that except for specific exceptions, as indicated in the quote above, the ISO has basically withdrawn from assigning specific associativity to the various levels of precedence. IMHO, compilers, books, GNU/GCC, etc..., have for the most part, accepted the current operator manpage, as is. IMHO we should leave all the current associativity for each level as is, unless specified below.

Postfix increment and decrement operators:  
-----

Postfix operators "++", and "--", (Section 6.3.2) have been promoted to level 1, with prefix versions of the operators left on level 2.

Cast operator:  
-----

The cast operator "(type)", (Section 6.5.4) has been removed from the Unary operators, (Section 7.2 of K&R, and the current table) and a new 3rd level inserted between the 2nd level, "6.5.3 Unary Operators" and now 4th level "6.5.5 Multiplicative operators", expanding the table to 16 levels. IMHO, the associativity should remain, right-to-left as in the 2nd level.

C99 Standard:  
-----

C99, Section 6.5, in paragraph #3, reaffirms that associativity is "unspecified", as do all later Standards.

"... Except as specified later (for the function-call (), &&, ||, ?:, and comma operators), the order of evaluation of subexpressions and the order in which side effects take place are both unspecified."

C99 offers no additions or changes to the operator table as defined by the C89/90 Standard.

I have included the table in the attached PDF to be thorough.

C11 Standard:  
-----

A new operator `_Alignof`, has been added to Level 2 of the operator table.

A new header file has been added to the C11 Standard, `stdalign.h` (7.15). This `#defines` `alignof` to the `_Alignof` operator/keyword. C11 also `#defines` `alignas` to a related keyword, `_Alignas`.

In conclusion:  
-----

I believe this is now accurate, and I ask you to confirm my interpretation of the Standards.

I would be happy to update the manpage upon approval of the changes to be made.

Thank you!

Rick Stanley  
rstanley@rsiny.com  
(917) 822-7771