

C9X Revision Proposal

=====

Title: Signed Integer Division
Author: Tom MacDonald
Author Affiliation: Cray Research, Inc.
Postal Address: 655F Lone Oak Drive
Eagan, MN 55121, USA
E-mail Address: tam@cray.com
Telephone Number: +1 612 6835818
Fax Number: +1 612 6835307
Sponsor: X3J11
Date: 1995 February 23

Proposal Category:

☒ Other: Eliminating some current implementation-defined behavior.

Area of Standard Affected:

☒ Language

Prior Art: Fortran

Target Audience: Everyone

Related Documents (if any): ANSI X3.9-1978, ISO 1539-1980(E)
programming language FORTRAN, ANSI X3.198:1992,
ISO/IEC 1539:1992 Fortran 90

Proposal Attached: Yes

Abstract: Currently signed integer division has implementation-defined semantics if either operand is negative. This proposal proposes to remove the implementation defined semantics and replace them with the Fortran rules.

Proposal: Change the following words in the current C Standard

Clause 6.3.5 Multiplicative Operators

From:

When integers are divided and the division is inexact, if both operands are positive the result of the "/" operator is the largest integer less than the algebraic quotient and the result of the "%" operator is positive. If either operand is negative, whether the result of the "/" operator is the largest integer less than or equal to the algebraic quotient or the smallest integer greater than or equal to the quotient is implementation-defined, as is the sign of the result of the "%" operator. If the quotient "a/b" is representable, the expression "(a/b)*b + a%b" shall equal "a".

To:

When integers are divided, the result of the "/" operator is the integer value closest to the mathematical quotient, and between zero and the mathematical quotient inclusively.

If the quotient "a/b" is representable, the expression
"a%b" shall equal "a-(a/b)*b".

Examples: $(-8) / 3 == (-2)$
 $(-8) \% 5 == -3$
 $8 \% (-5) == 3$
 $(-8) \% (-5) == -3$

Comments: The above wording is easier to understand, removes
implementation-defined behavior from the standard,
and is consistent with Fortran 90.