Proposal for C2Y
WG14 N3406

Title: complex operators, updates N3384
Author, affiliation: C FP group
Date: 2024-11-14
Proposal category: Technical
Reference: N3301

This update to N3384 responds to recent suggestions from Jens Gustedt in WG14 email communications titled "N3384, complex operators".

The suggested changes below move the complex operator definitions that are not particularly related to ISO/IEC 60559 out of Annex G and into the main body of the C standard.

The definitions for multiplicative operators (G.3.2 #1 and #2) and the definitions for additive operators (G.3.3 #1) support ISO/IEC 60559-consistent behavior for signed zeros and infinities, but the same definitions give the natural, most efficient implementation of the operators **+**, **-**, *****, and **/** with operands of mixed type domains (real and complex) for all implementations. Though the C23 usual arithmetic conversions allow implementations to use the formulas in these definitions, doing so is not currently required outside of Annex G. For example, the implementation of "complex op real" could convert the real operand to complex, with unexpected performance implications.

The current definitions in Annex G rely on tables that are not fully explained. The suggested changes are intended to be more explicit.

The suggested changes also introduce definitions for "complex ***** complex", "complex **/** complex", and "real **/** complex", referring to the usual mathematical formulas which the operators are intended to approximate.

In N3301, tables G1, G2, and G3 at no longer correct, presumably a result of editing to remove imaginary types. Details are not provided here because the suggested changes below remove the tables.

Note to editor: In the following changes, mathematical expressions are written entirely in ordinary font (Cambria), and C operators are written in a bold mono font (New Courier).

Suggested changes:

To 6.2.5 paragraph 17, append:

> … In this document complex values are sometimes written in the form $x + i\,y$, where x and y are the values of the real and imaginary parts, and i represents the mathematical imaginary unit, which has the property $i^2 = -1$. The form $x - i\,y$ is equivalent to $x + i\,(-y)$.

In 7.3.1 #3, delete the footnote:

…with the value of the imaginary unit;~~227)~~

~~227)The imaginary unit is a number i such that $i^2 = -1$.~~

In 6.5.6 under Semantics insert:

In the following, the result and any floating-point exceptions are completely specified by the real-valued operations in the definition.

— The ***** operator where one operand is real with value x and the other operand is complex with value $u + i\,v$ is defined by $(x * u) + i\,(x * v)$.

— The **/** operator where the first operand is complex with value $x + i\,y$ and the second operand is real with value u is defined by $(x / u) + i\,(y / u)$.

In the following, the real and imaginary parts of the result approximate the real and imaginary parts, respectively, of the mathematical formula to be computed. The formulas do not indicate how the results are to be evaluated, nor do they necessarily give desirable results when infinities or NaNs are involved (see Annex G).

— The ***** operator where both operands are complex with values $x + i\,y$ and $u + i\,v$ computes $(xu - yv) + i\,(yu + xv)$.

— The **/** operator where both operands are complex with values $x + i\,y$ and $u + i\,v$ computes $(xu + yv)/(u^2 + v^2) + i\,(yu - xv)/(u^2 + v^2)$.

— The **/** operator where the first operand is real with value x and the second operand is complex with value $u + i\,v$ computes $(xu)/(u^2 + v^2) + i\,(-xv)/(u^2 + v^2)$.

Recommended practice

It is recommended that the ***** operator where both operands are complex be commutative (that is, for complex values z and w, $z * w$ and $w * z$ are equivalent). It is also recommended that the ***** operator where both operands are complex, the **/** operator where both operands are complex, and the **/** operator where the first operand is real and the second operand is complex compute their results without undue overflow or underflow (see 7.3.4).

In 6.5.7 under Semantics insert:

In the following, the result and any floating-point exceptions are completely specified by the real-valued operations in the definition.

— The **+** operator where both operands are complex with values $x + i\,y$ and $u + \mathrm{i}\,v$ is defined by $(x + u) + i\,(y + v)$.

— The **+** operator where one operand is real with value *x* and the other operand is complex with value *u* + *i v* is defined by (*x* **+** *u*) + *i v*.

— The **−** operator where the first operand is complex with value *x* + *i y* and the second operand is complex with value *u* + *i v* is defined by (*x* **−** *u*) + *i* (*y* **−** *v*).

— The **−** operator where the first operand is real with value *x* and the second operand is complex with value *u* + *i v*, the difference is defined by (*x* **−** *u*) + *i* (−*v*).

— The **−** operator where the first operand is complex with value *x* + i *y* and the second operand is real with value *u*, the difference is defined by (*x* **−** *u*) + *i y*.

In G.3.2 delete paragraphs 1 and 2.

Delete G.3.3.