Proposal for C2Y
WG14 N3732

Title:	Floating expressions evaluated in translation
Author, affiliation:	C FP group
Date:	2025-10-07
Proposal category:	Editorial
Reference:	N3447, N3685

Background

6.6.1 #14 says

> If a floating expression is evaluated in the translation environment, the arithmetic range and precision shall be at least as great as if the expression were being evaluated in the execution environment.

In sorting ghost-UB from material-UB, N3447 says

- The case that floating range and precision for the determined value during translation fits the execution environment is not detectable at translation time and gives thus rise to material-UB.

and proposes a new entry in J.2 to cover that case.

Clause 4 says

> In this document, "shall" is to be interpreted as a requirement on an implementation or on a program; conversely, "shall not" is to be interpreted as a prohibition.
>
> If a "shall" or "shall not" requirement that appears outside of a constraint or runtime-constraint is violated, the behavior is undefined. …

The suggested change below eliminates the UB in 6.6.1 #15 by avoiding use of “shall”.

The UB in the current C draft allows a conforming implementation to violate the requirement. We are unaware of any implementation where the translation environment cannot determine bounds on the range and precision for evaluation in the execution environment, though such an implementation is conceivable.

This proposal does not address the issue of whether it is desirable for a violation of a “shall” requirement by the implementation (vs the program) to result in UB.

Suggested Changes

In 6.6.1 #14 change:

> … ~~If a floating expression is evaluated in the translation environment, the arithmetic range and precision shall be~~ at least as great as if the expression were being evaluated in the execution environment.[122]

to:

> … A floating expression evaluated in the translation environment is evaluated with arithmetic range and precision at least as great as if the expression were being evaluated in the execution environment.[122]