**Proposal for C2Y**
**WG14 n3405**

**Title:** **i**mproved wording for treatment of error conditions in <math.h>
**Author, affiliation:** CFP group
**Date:** 2024-11-04
**Proposal category:** Editorial
**Reference:** N3301

This document proposes a clarification of the language around how errors arise in the mathematical domain and may or may not be reported in the C computational domain.

**Note to the editor:**

If accepted, this change should be implemented after change N3383, which would add a sentence after the first sentence of 7.12.2#4 but is otherwise orthogonal to this change.

**Suggested change:**

Change to add a new paragraph after 7.12.2#1

> An error is said to occur when the underlying exceptional computational condition arises and the error is reported using the mechanisms described in this subclause. Not all exceptional conditions are required to be reported as errors.

Change to the first sentence of 7.12.2#2

> For all functions, a domain error ~~occurs~~ can occur if and only if . . .

Change to the first sentence of 7.12.2#3

> Similarly, a pole error ~~occurs~~ can occur in cases where . . .

Change to the first sentence of 7.12.2#4

> Likewise, a range error ~~occurs~~ can occur if and only if . . .

**Rationale:**

The problem addressed here is that the current wording in section 7.12.2 specifies that "error X occurs if and only if condition Y is met", while later on each function specification has language about which errors (must) "occur" or "may occur".

That is, the unconditional “if and only if” specification in 7.12.2 is at odds with the piecemeal function-by-function specification. The relaxed "can occur" wording of this proposal bridges the gap between the computational context (what exceptional cases arise) and the C execution context (which cases must or may be reported).

Exceptional circumstances in the various functions can arise in a variety of contexts. The issue may be mathematical, as in log(0.0), or it may be tied to the underlying arithmetic, as in overflow of a value too large to represent. The C standard defines whether such circumstances can or must be reported to users.

First, the new paragraph emphasizes that errors "occur" (that is, are reported) in the C domain.

Next, the three definitions connect domain, pole, and range errors to the underlying computational circumstances.

Finally, the long list of function specifications defines which errors must occur, and mentions other circumstances where such errors "may occur".

This change requires no implementation changes.