Proposal for C23
WG14 N2805

Title: overflow and underflow definitions (N2746 update)
Author, affiliation: C FP group
Date: 2021-08-22
Proposal category: Editorial
Reference: N2596, N2746

This document updates N2746. The suggested changes below improve on those in N2746 in the following ways:

1. The revised overflow definition is based on the result being too large for representation in the floating-point model, where error expectations (based on the model) can be met, rather than on the result being too larger for representation in the type. This change is needed for double-double formats.

2. The revised underflow definition avoids the blanket inclusion of results with magnitude equal to the minimum normalized number, but allows such underflows, which occur in IEC 60559 implementations in certain cases.

3. A footnote is added to explain the term “ordinary accuracy”.

Otherwise, the following is as in N2746.

7.12.1 in the current C23 draft (N2596) defines overflow and underflow:

> [5] A floating result overflows if the magnitude (absolute value) of the mathematical result is finite but so large that the mathematical result cannot be represented without extraordinary roundoff error in an object of the specified type. …
>
> [6] The result underflows if the magnitude (absolute value) of the mathematical result is nonzero and less than the minimum normal number in the type.249) …
>
> 249)The term underflow here is intended to encompass both "gradual underflow" as in IEC 60559 and also "flush-to-zero" underflow.

Problem 1: The use of “mathematical result” is not appropriate here. It might well be taken to mean the infinitely precise value of the mathematical function corresponding to the C function. But C doesn’t require correct rounding. The implementation might compute an estimate that overflows where the mathematical result would not. The following suggested changes eliminate these uses of “mathematical result”.

The C23 draft contains one other use of “mathematical result”, which the changes in another proposal eliminate.

Problem 2: The definition of underflow excludes the IEC 60559 underflows that are outside the normal range before but not after rounding. This is contrary to footnote 249. The following suggested changes broaden the definition of underflow to include all IEC 60559 underflows. Broadening the definition does not break implementations because reporting of underflow range errors is optional in C.

The changes also add a sentence to footnote 249 explaining why the definition is broader than might be expected.

Suggested changes:

Changes in 7.12.1:

> [5] A floating result overflows if ~~the~~ a finite result value with ordinary accuracy*) would have magnitude (absolute value) ~~of the mathematical result is finite but so large that the mathematical result cannot be represented without extraordinary roundoff error~~ larger than the maximum normalized number in the specified type. (A result that is an exact infinity does not overflow.)
>
> . ...
>
> [6] The result underflows if ~~the~~ a nonzero result value with ordinary accuracy would have magnitude (absolute value) ~~of the mathematical result is nonzero and~~ less than the minimum normalized number in the type; however, a zero result that is specified to be an exact zero does not underflow. Also, a result with ordinary accuracy and the magnitude of the minimum normalized number may underflow.249) ...
>
> *)Ordinary accuracy is determined by the implementation. It refers to the accuracy of the function where results are not compromised by extreme magnitude.
>
> 249)The term underflow here is intended to encompass both "gradual underflow" as in IEC 60559 and also "flush-to-zero" underflow. IEC 60559 underflow can occur in cases where the magnitude of the rounded result (accurate to the full precision of the type) equals the minimum normalized number in the format.