Vincent Lefevre sent email to CFP pointing out an inconsistency in the definition of the macros `FLT_EXP_MAX`, `DBL_EXP_MAX`, and `LDBL_EXP_MAX` (5.2.4.2.2#18):

- maximum integer such that \( \text{FLT\_RADIX} \) raised to one less than that power is a representable finite floating-point number, \( e_{\text{max}} \)

\[
\text{FLT\_MAX\_EXP}
\]

\[
\ldots
\]

namely, that the number described might not be \( e_{\text{max}} \) for implementations that include finite numbers larger than the maximum normalized floating-point number in the type.

The suggested change below clarifies that the definition refers to normalized floating-point numbers.

We also considered basing `FLT_MAX_EXP`, etc. on all finite numbers in the type and introducing `FLT_MAX_NORM_EXP`, etc. for normalized floating-point numbers. However, this might invalid programs written with the reasonable interpretation (since the definition says \( e_{\text{max}} \) is the value of the macros) that the macros refer to normalized floating-point numbers. And, we didn’t see a real need for the second set of macros.

**Suggested changes:**

In 5.2.4.2.2 #18, change:

- maximum integer such that \( \text{FLT\_RADIX} \) raised to one less than that power is a representable finite normalized floating-point number, \( e_{\text{max}} \)

\[
\text{FLT\_MAX\_EXP}
\]

\[
\ldots
\]