The **<fenv.h>** header should define macros for the following sub-exceptions and may define additional macros with the appropriate prefix (**FE_INVALID_** or **FE_DIVBYZERO_**) for other sub-exceptions. The supported exception designations shall include the defined sub-exception macro identifiers (if any). If defined, the macros expand to integer constant expressions. Sub-exceptions corresponding to defined macros occur as specified below, and not in other cases.

— "invalid" floating-point exceptions from add and subtract operators and functions that add or subtract (C23 7.12.14.1, ~~and~~ 7.12.14.2, F.10.11), not caused by signaling NaN input

FE_INVALID_ADD

— "invalid" floating-point exceptions from divide operators and functions that divide (C23 7.12.14.4, F.10.11), not caused by signaling NaN input

FE_INVALID_DIV

— "invalid" floating-point exceptions from functions that compute multiply-add (C23 7.12.13.1, F.10.10.1, ~~and~~ 7.12.14.5, F.10.11) and from contracted multiply and add operators, not caused by signaling NaN input

FE_INVALID_FMA

— "invalid" floating-point exceptions from conversions from floating to integer types (C23 F.4), not caused by signaling NaN input

FE_INVALID_INT

— "invalid" floating-point exceptions from **ilogb** and **llogb** functions C23 F.10.3.8, F.10.3.10), not caused by signaling NaN input

FE_INVALID_ILOGB

— "invalid" floating-point exceptions from multiply operators and functions that multiply (C23 7.12.14.3, F.10.11), not caused by signaling NaN input

FE_INVALID_MUL

— "invalid" floating-point exceptions from the **quantized***N* functions (C23 7.12.15.1), not caused by signaling NaN input

FE_INVALID_QUANTIZE

— "invalid" floating-point exceptions from the **remainder** and **remquo** functions (C23 F.10.7.2, F.10.7.3), not caused by signaling NaN input

FE_INVALID_REM

— "invalid" floating-point exceptions from functions that compute square root or reciprocal of square root (C23 ~~7.12.7.9~~F.10.4.9, ~~7.12.7.10~~F.10.4.10, ~~and 7.12.14.6~~7.12.14.6, F.10.11), not caused by signaling NaN input

FE_INVALID_SQRT

— "invalid" floating-point exceptions caused by signaling NaN input (C23 F.2.1)

FE_INVALID_SNAN

— "invalid" floating-point exceptions from relational operators and comparison macros (C23 6.5.8, 7.12.17, F.10.14.1), not caused by signaling NaN input

FE_INVALID_UNORDERED

— “divide-by-zero” floating-point exceptions from divide operators and functions that divide (C23 7.12.14.4, F.10.11)
`FE_DIVBYZERO_ZERO`

— “divide-by-zero” floating-point exceptions from logarithm and **`logb`** functions (C23 F.10.3.11, F.10.3.12, F.10.3.13, F.10.3.14, F.10.3.15, F.10.3.16, F.10.3.17)
`FE_DIVBYZERO_LOG`