Updating “Restrictions on exception handling”

GB 41 says of [res.on.exception.handling] that “the "any other function" sentence in p4 contradicts the restriction placed in p2.” Discussion in LWG revealed inconsistent interpretations of the text.

Proposed resolution: rewrite [res.on.exception.handling] to avoid ambiguity in which “other functions” are covered. Avoid saying “Throws clause”, as “clause” has another meaning in the Standard; say “Throws: paragraph” consistently. While here, address GB 42 by turning a “should” footnote into normative encouragement.

Drafting note: Please apply these edits after those for P0003.

1 Any of the functions defined in the C++ standard library can report a failure by throwing an exception of a type described in its Thrown: paragraph, or of a type derived from a type named in the Thrown: paragraph that would be caught by an exception handler for the base type. An implementation may strengthen the exception specification for a non-virtual function by adding a non-throwing noexcept-specification:

2 Functions from the C standard library shall not throw exceptions except when such a function calls a program-supplied function that throws an exception.

3 Destructor operations defined in the C++ standard library shall not throw exceptions. Every destructor in the C++ standard library shall behave as if it had a non-throwing exception specification.

4 Functions defined in the C++ standard library that do not have a Thrown: paragraph but do have a potentially-throwing exception specification may throw implementation-defined exceptions. Implementations should report errors by throwing exceptions of or derived from the standard exception classes (18.6.3.1, 18.8, 19.2). Any other functions defined in the C++ standard library that do not have an exception specification may throw implementation-defined exceptions unless otherwise specified. An implementation may strengthen this implicit exception specification by adding an explicit one.
5. An implementation may strengthen the exception specification of a non-virtual function by adding a non-throwing exception specification.

GB 42:

“The word 'should' makes footnote 188 sound like normative encouragement, if not an actual mandate.”

Proposed resolution: make it normative encouragement, per the re-write of [res.on.exception.handling] for GB41 above. Remove it from the footnote.

188) In particular, they can report a failure to allocate storage by throwing an exception of type bad_alloc, or a class derived from bad_alloc (18.6.3.1). Library implementations should report errors by throwing exceptions of or derived from the standard exception classes (18.6.3.1, 18.8.19.2).

Acknowledgements

My thanks to Alisdair Meredith for assisting in understanding the effects of P0003 on this text.