

Proposal for C2x
WG14 N3098

Title: GB-089 on [[fallthrough]]
Author, affiliation: Aaron Ballman, Intel
Date: 2023-01-31
Proposal category: Proposed NB Comment Response
Abstract: Clarifies whether block item meant by “next block item that would be encountered” is in lexical order or in execution order.

GB-089 on `[[fallthrough]]`

Reply-to: Aaron Ballman (aaron@aaronballman.com)

Document No: N3098

Date: 2023-01-31

Summary of Changes

N3098

- Original proposal

Introduction and Rationale

From GB-089:

It is unclear whether “next block item that would be encountered” refers to next lexically, or in execution order (the latter being problematic for a Constraint, since those are translation-time properties). A similar issue was previously raised for C++ - see https://www.open-std.org/jtc1/sc22/wg21/docs/cwg_defects.html#2406 – but it appears the resolution for C++ was not applied for C.

Proposed Straw Poll

Does WG14 want to resolve GB-089 by adopting the proposed wording from N3098?

Proposed Wording

All proposed wording in this document is a diff from WG14 N3054. **Green** text is new text, while **red** text is deleted text.

Modify 6.7.12.5p1:

The attribute token `fallthrough` shall only appear in an attribute declaration (6.7); such a declaration is a `fallthrough` declaration. No attribute argument clause shall be present. A `fallthrough` declaration may only appear within an enclosing switch statement (6.8.4.2). The next block item (6.8.2) that would be encountered after a `fallthrough` declaration shall be a case label or default label associated with the smallest enclosing switch statement **and, if the `fallthrough` declaration is contained in an iteration statement, the next statement shall be part of the same execution of the secondary block of the innermost enclosing iteration statement.**

Modify 6.7.12.5p4: *Editorial note: feel free to change the added comments to have different line wrapping, punctuation, etc.*

EXAMPLE

```
void f(int n) {
    void g(void), h(void), i(void);
    switch (n) {
    case 1: /* diagnostic on fallthrough discouraged */
    case 2:
        g();
        [[fallthrough]];
    case 3: /* diagnostic on fallthrough discouraged */
```

```
do {
    [[fallthrough]]; /* constraint violation: next statement is not
                    part of the same secondary block execution */
} while(false);
case 6:
do {
    [[fallthrough]]; /* constraint violation: next statement is not
                    part of the same secondary block execution */
} while (n--);
case 7:
while (false) {
    [[fallthrough]]; /* constraint violation: next statement is not
                    part of the same secondary block execution */
}
case 5:
h();
case 4: /* fallthrough diagnostic encouraged */
i();
[[fallthrough]]; /* constraint violation */
}
}
```

References

[CWG2406]

[[fallthrough]] attribute and iteration statements. Miller. https://www.open-std.org/jtc1/sc22/wg21/docs/cwg_defects.html#2406