Earthly Demon XV: Definition of main (Updates n3480)

Document: n3562

Author: Martin Uecker

Date: 2025-05-25

Changes to n3480: Revise wording and address the possibility of free-standing implementations in the new constraint. Add size expression to second prototype.

Changes to n3419: Rephrase and change to external linkage (SC22WG14.28267), also allow multiple definition to still be UB (this will be addressed later)

Undefined Behavior: Non-supported definition of main (or no definition).

Example:

```
int main(int argc, char *argv[], int *p)
{
// ...
}
```

Analysis:

POSIX defines an extension, but we explicitely allow implementation-defined forms, so this does not fall under this UB in this case. It is UB if there is no corresponding implementation-defined extension.

Recommendation: Make this a constraint violation.

Wording (n3550) (note to editor: addition of size expression to second prototype)

5.2.2.3.2 Program startup

The function called at program startup is named `main`. The implementation declares no prototype for this function. It ~~shall be defined~~ **it is permitted to define it** with a return type of `int` and with no parameters:

```
int main(void) { /* ... */ }
```

or with two parameters (referred to here as `argc` and `argv`, though any names can be used, as they are local to the function in which they are declared):

```
int main(int argc, char *argv[static argc + 1]) { /* ... */ }
```

or ~~equivalent;6) or in some other implementation-defined manner.~~ **using another type that is compatible to one of these two function types, or using some other implementation-defined type.**

6.9.1. General

Constraints

1 In a hosted implementation there shall be a definition for the identifier `main` with external linkage and with a permitted function type as specified in 5.2.2.3.2.