

N3160: Add `min`, `max` for integers to C

Christoph Grüninger*

2023-08-19

Document: N3160

Audience: WG14

Proposal category: New feature

Document hosted at <https://github.com/gruenich/n3160-add-min-max-to-c>

1 Summary of proposed changes

Add the functions `min` and `max` for integers to the C programming language.

2 Rationale

For floats and doubles, `fmin` and `fmax` exists, but not for integer types. Other programming languages provide them including C++ by `std::min` and `std::max`. Several C libraries provide them as macros, too, e.g., GNU C library [1], Linux [2], SuperLU [3], Cairo [4]. Stack Overflow has a question where to find C's `min/max` function with more than 400 up-votes [5], indicating that many C programmers expect this function to be part of the C language.

Some may argue these two functions are trivial to implement. C provides other functions that would be trivial to implement like `abs` and `fabs`.

3 Proposal

Add the `min` and `max` functions for integers to `stdlib.h`.

The formal definitions of the two functions are

$$\min(a, b) := \begin{cases} b & a > b \\ a & \text{else} \end{cases}$$

*Unaffiliated, foss@grueninger.de

and

$$\max(a, b) := \begin{cases} b & b > a \\ a & \text{else} \end{cases} .$$

The following functions should be added to `stdlib.h` implementing above definition:

```
int min (int a, int b);
int max (int a, int b);
long int lmin (long int a, long int b);
long int lmax (long int a, long int b);
long long int llmin (long long int a, long long int b);
long long int llmax (long long int a, long long int b);
```

As a consequence, the function-like macros `min` and `max` can be implemented.

4 Impact

The introduction of the functions will conflict with existing code already using the names. In most cases functions with the same name probably do exactly the same as described in this proposal.

Other uses of the names are imaginable like returning the smallest or biggest element from a set or vector. This would lead to name clashes and compile errors.

Using a library version-test macro can help with backwards compatible code. We do not see the need of a dedicated test macro `__has_c_min_max`.

5 Proposed wording

We propose to add a new sub-section 7.24.6.2:

7.24.6.2 The `min`, `lmin`, `llmin`, `max`, and `lmax`, `llmax` functions

Synopsis

```
#include <stdlib.h>;
int min (int a, int b);
int max (int a, int b);
long int lmin (long int a, long int b);
long int lmax (long int a, long int b);
long long int llmin (long long int a, long long int b);
long long int llmax (long long int a, long long int b);
```

Description

The `min`, `lmin`, and `llmin` functions compute the smaller value of the two integers `a` and `b`. The `max`, `lmax`, and `llmax` functions compute the bigger value of the two integers `a` and `b`.

Returns

The `min`, `lmin`, and `llmin` functions return the smaller value. The `max`, `lmax`, and `llmax` functions return the bigger value.

6 Looking for help

If someone is interested to chip in with experience, WG14 seniority, or as a champion to present the proposal in a committee meeting, we welcome additional authors to this proposal.

7 Acknowledgments

We would like to thank Martin Uecker for encouragement and feedback on the first draft.

References

- [1] GNU C Library source code containing macros for MIN and MAX, <https://sourceware.org/git/?p=glibc.git;a=blob;f=misc/sys/param.h;hb=HEAD>.
- [2] Linux source code containing macros for MIN and MAX, <https://github.com/torvalds/linux/blob/master/include/linux/minmax.h>.
- [3] SuperLU source code containing macros for MIN and MAX, https://github.com/xiaoyeli/superlu/blob/master/SRC/slu_util.h.
- [4] Cairo source code containing macros for MIN and MAX, <https://cgit.freedesktop.org/cairo/tree/perf/cairo-perf.h>.
- [5] Stackoverflow question *MIN and MAX in C*, <https://stackoverflow.com/questions/3437404/min-and-max-in-c>.