

ASSERT(3)

ASSERT(3) Linux Programmer's Manual ASSERT(3)

NAME

assert - abort the program if assertion is false

SYNOPSIS

```
#include <assert.h>

void assert(scalar expression);
```

DESCRIPTION

If the macro **NDEBUG** was defined at the moment `<assert.h>` was last included, the macro **assert**() generates no code, and hence does nothing at all. Otherwise, the macro **assert**() prints an error message to standard error and terminates the program by calling **abort**(3) if expression is false (i.e., compares equal to zero).

The purpose of this macro is to help the programmer find bugs in his program. The message “assertion failed in file foo.c, function do_bar(), line 1287” is of no help at all to a user.

RETURN VALUE

No value is returned.

CONFORMING TO

POSIX.1-2001, C89, C99. In C89, expression is required to be of type `int` and undefined behavior results if it is not, but in C99 it may have any scalar type.

BUGS

`assert()` is implemented as a macro; if the expression tested has side-effects, program behavior will be different depending on whether **NDEBUG** is defined. This may create Heisenbugs which go away when debugging is turned on.

SEE ALSO

`abort(3)`, `assert_perror(3)`, `exit(3)`

COLOPHON

This page is part of release 3.54 of the Linux `man-pages` project. A description of the project, and information about reporting bugs, can be found at <http://www.kernel.org/doc/man-pages/>.

2002-08-25 GNU
