

NAME

mtx_lock - acquire a mutex

SYNOPSIS

library "threads"

#include <threads.h>

int

mtx_lock(*mtx_t* **mtx*);

DESCRIPTION

The function **mtx_lock**() will acquire a mutex *mtx*. If the mutex is already acquired by an other thread the function will wait until the mutex can be acquired.

If the mutex is already being held by the current thread and locking recursion was not specified to **mtx_init**(3) the thread will be in a deadlock. If recursion is enabled the thread will acquire an additional instance of the mutex. In this case to unlock the mutex **mtx_unlock**(3) must be called an equal number of times as **mtx_lock**(), **mtx_trylock**(3) or **mtx_timedlock**(3) have been called from the same thread on the same mutex, in order to unlock the mutex for other threads.

RETURN VALUES

Upon success **mtx_lock**() will return *thrd_success*. In case of error *thrd_error* will be returned.

SEE ALSO

mtx_init(3) **mtx_destroy**(3) **mtx_unlock**(3) **mtx_trylock**(3) **mtx_timedlock**(3)

HISTORY

The **mtx_lock**() function first appeared in the C11 standard ISO/IEC 9899:2011.

AUTHORS

Jan Adelsbach <jan@jadelsbach.de>