NAME

`thr_exit` — terminate current thread

LIBRARY

Standard C Library (libc, −lc)

SYNOPSIS

```
#include <sys/thr.h>

void
thr_exit(long *state);
```

DESCRIPTION

This function is intended for implementing threading. Normal applications should call `pthread_exit`(3) instead.

The `thr_exit`() system call terminates the current kernel-scheduled thread.

If the *state* argument is not `NULL`, the location pointed to by the argument is updated with an arbitrary non-zero value, and an `_umtx_op`(2) `UMTX_OP_WAKE` operation is consequently performed on the location.

Attempts to terminate the last thread in the process are silently ignored. Use `_exit`(2) syscall to terminate the process.

RETURN VALUES

The function does not return a value. A return from the function indicates that the calling thread was the last one in the process.

SEE ALSO

`_exit`(2), `thr_kill`(2), `thr_kill2`(2), `thr_new`(2), `thr_self`(2), `thr_set_name`(2), `_umtx_op`(2), `pthread_exit`(3)

STANDARDS

The `thr_exit`() system call is non-standard and is used by 1:1 Threading Library (libthr, −lthr) to implement IEEE Std 1003.1-2001 (“POSIX.1”) `pthread`(3) functionality.