NAME

makecontext, **swapcontext** - modify and exchange user thread contexts

LIBRARY

Standard C Library (libc, -lc)

SYNOPSIS

#include <ucontext.h>

void
makecontext(*ucontext_t *ucp, void (*func)(void), int argc, ...*);

int
swapcontext(*ucontext_t *oucp, const ucontext_t *ucp*);

DESCRIPTION

The **makecontext**() function modifies the user thread context pointed to by *ucp*, which must have previously been initialized by a call to getcontext(3) and had a stack allocated for it. The context is modified so that it will continue execution by invoking **func**() with the arguments provided. The *argc* argument must be equal to the number of additional arguments of type *int* provided to **makecontext**() and also equal to the number of arguments of type *int* to **func**(); otherwise , the behavior is undefined.

The *ucp->uc_link* argument must be initialized before calling **makecontext**() and determines the action to take when **func**() returns: if equal to NULL, the process exits; otherwise, **setcontext**(*ucp->uc_link*) is implicitly invoked.

The **swapcontext**() function saves the current thread context in **oucp* and makes **ucp* the currently active context.

RETURN VALUES

If successful, **swapcontext**() returns zero; otherwise -1 is returned and the global variable *errno* is set appropriately.

ERRORS

The **swapcontext**() function will fail if:

[ENOMEM] There is not enough stack space in *ucp* to complete the operation.

SEE ALSO

setcontext(3), ucontext(3)

STANDARDS

The **makecontext**() and **swapcontext**() functions conform to X/Open System Interfaces and Headers Issue 5 ("XSH5") and IEEE Std 1003.1-2001 ("POSIX.1").

The IEEE Std 1003.1-2004 ("POSIX.1") revision marked the functions **makecontext**() and **swapcontext**() as obsolete, citing portability issues and recommending the use of POSIX threads instead. The IEEE Std 1003.1-2008 ("POSIX.1") revision removed the functions from the specification.

The standard does not clearly define the type of integer arguments passed to *func* **via makecontext(); portable applications should not rely on the implementation detail that it may be possible to pass pointer arguments to functions.**

HISTORY

The **makecontext**() and **swapcontext**() functions first appeared in AT&T System V Release 4 UNIX.