Full credit is given to the above companies including the OS that this PDF file was generated!

Red Hat Enterprise Linux Release 9.2 Manual Pages on 'longjmp.3p' command

$ man longjmp.3p

LONGJMP(3P) POSIX Programmer's Manual LONGJMP(3P)

PROLOG

This manual page is part of the POSIX Programmer's Manual. The Linux implementation of this interface may differ (consult the corresponding Linux manual page for details of Linux behavior), or the interface may not be implemented on Linux.

NAME

longjmp ? non-local goto

SYNOPSIS

#include <setjmp.h>

void longjmp(jmp_buf env, int val);

DESCRIPTION

The functionality described on this reference page is aligned with the ISO C standard. Any conflict between the requirements described here and the ISO C standard is unintentional. This volume of POSIX.1?2017 defers to the ISO C standard.

The longjmp() function shall restore the environment saved by the most recent invocation of setjmp() in the same process, with the correspond? ing jmp_buf argument. If the most recent invocation of setjmp() with the corresponding jmp_buf occurred in another thread, or if there is no such invocation, or if the function containing the invocation of setjmp() has terminated execution in the interim, or if the invocation of setjmp() was within the scope of an identifier with variably modi?

fied type and execution has left that scope in the interim, the behav? ior is undefined. It is unspecified whether longjmp() restores the signal mask, leaves the signal mask unchanged, or restores it to its value at the time setjmp() was called.

All accessible objects have values, and all other components of the ab? stract machine have state (for example, floating-point status flags and open files), as of the time longjmp() was called, except that the val? ues of objects of automatic storage duration are unspecified if they meet all the following conditions:

* They are local to the function containing the corresponding setjmp() invocation.
* They do not have volatile-qualified type.
* They are changed between the setjmp() invocation and longjmp() call.

Although longjmp() is an async-signal-safe function, if it is invoked from a signal handler which interrupted a non-async-signal-safe func? tion or equivalent (such as the processing equivalent to exit() per? formed after a return from the initial call to main()), the behavior of any subsequent call to a non-async-signal-safe function or equivalent is undefined.

The effect of a call to longjmp() where initialization of the jmp_buf structure was not performed in the calling thread is undefined.

RETURN VALUE

After longjmp() is completed, program execution continues as if the corresponding invocation of setjmp() had just returned the value speci? fied by val. The longjmp() function shall not cause setjmp() to return 0; if val is 0, setjmp() shall return 1.

ERRORS

No errors are defined.

The following sections are informative.

EXAMPLES

None.

APPLICATION USAGE

Applications whose behavior depends on the value of the signal mask should not use longjmp() and setjmp(), since their effect on the signal mask is unspecified, but should instead use the siglongjmp() and sigsetjmp() functions (which can save and restore the signal mask under application control).

It is recommended that applications do not call longjmp() or sig? longjmp() from signal handlers. To avoid undefined behavior when call? ing these functions from a signal handler, the application needs to en? sure one of the following two things:

1. After the call to longjmp() or siglongjmp() the process only calls async-signal-safe functions and does not return from the initial call to main().
2. Any signal whose handler calls longjmp() or siglongjmp() is blocked during every call to a non-async-signal-safe function, and no such calls are made after returning from the initial call to main().

RATIONALE

None.

FUTURE DIRECTIONS

None.

SEE ALSO

setjmp(), sigaction(), siglongjmp(), sigsetjmp()

The Base Definitions volume of POSIX.1?2017, <setjmp.h>

most likely to have been introduced during the conversion of the source files to man page format. To report such errors, see https://www.kernel.org/doc/man-pages/reporting_bugs.html .