NAME

lseek − reposition read/write file offset

SYNOPSIS

#include <sys/types.h>
#include <unistd.h>

off_t lseek(int *fd***, off_t** *offset***, int** *whence***);**

DESCRIPTION

The **lseek**() function repositions the offset of the open file associated with the file descriptor *fd* to the argument *offset* according to the directive *whence* as follows:

SEEK_SET
The offset is set to *offset* bytes.

SEEK_CUR
The offset is set to its current location plus *offset* bytes.

SEEK_END
The offset is set to the size of the file plus *offset* bytes.

The **lseek**() function allows the file offset to be set beyond the end of the file (but this does not change the size of the file). If data is later written at this point, subsequent reads of the data in the gap (a "hole") return null bytes ('\0') until data is actually written into the gap.

RETURN VALUE

Upon successful completion, **lseek**() returns the resulting offset location as measured in bytes from the beginning of the file. Otherwise, a value of *(off_t) −1* is returned and *errno* is set to indicate the error.

ERRORS

EBADF
fd is not an open file descriptor.

EINVAL
whence is not one of **SEEK_SET**, **SEEK_CUR**, **SEEK_END**; or the resulting file offset would be negative, or beyond the end of a seekable device.

EOVERFLOW
The resulting file offset cannot be represented in an *off_t*.

ESPIPE
fd is associated with a pipe, socket, or FIFO.

CONFORMING TO

SVr4, 4.3BSD, POSIX.1-2001.

NOTES

This document's use of *whence* is incorrect English, but maintained for historical reasons.

Some devices are incapable of seeking and POSIX does not specify which devices must support **lseek**().

On Linux, using **lseek**() on a tty device returns **ESPIPE**.

When converting old code, substitute values for *whence* with the following macros:

old	new
0	SEEK_SET
1	SEEK_CUR
2	SEEK_END
L_SET	SEEK_SET
L_INCR	SEEK_CUR

L_XTND SEEK_END

SVr1-3 returns *long* instead of *off_t*, BSD returns *int*.

Note that file descriptors created by **dup**(2) or **fork**(2) share the current file position pointer, so seeking on such files may be subject to race conditions.

SEE ALSO

dup(2), **fork**(2), **open**(2), **fseek**(3), **lseek64**(3), **posix_fallocate**(3)

COLOPHON

This page is part of release 3.22 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/.