NAME

readdir – read a directory entry

SYNOPSIS

/usr/ucb/cc [*flag* ...] *file* ...

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

struct direct *readdir(*dirp***);**
DIR ***dirp*;**

DESCRIPTION

readdir( ) returns a pointer to a structure representing the directory entry at the current position in the directory stream to which *dirp* refers, and positions the directory stream at the next entry, except on read-only filesystems. It returns a **NULL** pointer upon reaching the end of the directory stream, or upon detecting an invalid location in the directory. **readdir( )** shall not return directory entries containing empty names. It is unspecified whether entries are returned for dot or dot-dot. The pointer returned by **readdir( )** points to data that may be overwritten by another call to **readdir( )** on the same directory stream. This data shall not be overwritten by another call to **readdir( )** on a different directory stream. **readdir( )** may buffer several directory entries per actual read operation. **readdir( )** marks for update the *st_atime* field of the directory each time the directory is actually read.

RETURN VALUES

readdir( ) returns **NULL** on failure and sets **errno** to indicate the error.

ERRORS

readdir( ) will fail if one or more of the following are true:

EAGAIN Mandatory file/record locking was set, **O_NDELAY** or **O_NONBLOCK** was set, and there was a blocking record lock.

EAGAIN Total amount of system memory available when reading using raw I/O is temporarily insufficient.

EAGAIN No data is waiting to be read on a file associated with a tty device and **O_NONBLOCK** was set.

EAGAIN No message is waiting to be read on a stream and **O_NDELAY** or **O_NONBLOCK** was set.

EBADF The file descriptor determined by the **DIR** stream is no longer valid. This results if the **DIR** stream has been closed.

EBADMSG Message waiting to be read on a stream is not a data message.

EDEADLK The **read( )** was going to go to sleep and cause a deadlock to occur.

EFAULT *buf* points to an illegal address.

EINTR A signal was caught during the **read( )** or **readv( )** function.

EINVAL Attempted to read from a stream linked to a multiplexor.

EIO A physical I/O error has occurred, or the process is in a background process group and is attempting to read from its controlling terminal, and either the process is ignoring or blocking the **SIGTTIN** signal or the process group of the process is orphaned.

ENOENT The current file pointer for the directory is not located at a valid entry.

ENOLCK The system record lock table was full, so the **read( )** or **readv( )** could not go to sleep until the blocking record lock was removed.

ENOLINK *fildes* is on a remote machine and the link to that machine is no longer active.

ENXIO The device associated with *fildes* is a block special or character special file and the value of the file pointer is out of range.

SEE ALSO

getdents(2), **scandir**(3B), **directory**(3C)

NOTES

Use of these interfaces should be restricted to only applications written on BSD platforms. Use of these interfaces with any of the system libraries or in multi-thread applications is unsupported.