NAME

symlink – make a symbolic link to a file

SYNOPSIS

#include <unistd.h>

int symlink(const char **name1*, **const char** **name2***);**

DESCRIPTION

symlink( ) creates a symbolic link *name2* to the file *name1*. Either name may be an arbitrary pathname, the files need not be on the same file system, and *name1* may be nonexistent.

The file to which the symbolic link points is used when an **open**(2) operation is performed on the link. A **stat**(2) on a symbolic link returns the linked-to file, while an **lstat** returns information about the link itself. This can lead to surprising results when a symbolic link is made to a directory. To avoid confusion in programs, the **readlink**(2) call can be used to read the contents of a symbolic link.

RETURN VALUES

Upon successful completion **symlink( )** returns a value of 0; otherwise, it returns –1 and places an error code in **errno**.

ERRORS

The symbolic link is made unless one or more of the following are true:

EACCES Search permission is denied for a component of the path prefix of *name2* .

EEXIST The file referred to by *name2* already exists.

EFAULT *name1* or *name2* points to an illegal address.

EIO An I/O error occurs while reading from or writing to the file system.

ELOOP Too many symbolic links are encountered in translating *name2* .

ENAMETOOLONG The length of the *name2* argument exceeds {**PATH_MAX**}, or the length of a *name2* component exceeds {**NAME_MAX**} while (**_POSIX_NO_TRUNC**) is in effect.

ENOENT A component of the path prefix of *name2* does not exist.

ENOSPC The directory in which the entry for the new symbolic link is being placed cannot be extended because no space is left on the file system containing the directory.

ENOSPC The new symbolic link cannot be created because no space is left on the file system which will contain the link.

ENOSPC There are no free inodes on the file system on which the file is being created.

ENOSYS	The file system does not support symbolic links
ENOTDIR	A component of the path prefix of *name2* is not a directory.
EROFS	The file *name2* would reside on a read-only file system.

SEE ALSO

cp(1), **link**(2), **open**(2), **readlink**(2), **stat**(2), **unlink**(2)