NAME

creat − create a new file

SYNOPSIS

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

int creat(const char ***name*****, mode_t** *mode***)**

DESCRIPTION

This interface is made obsolete by open(2), it is equivalent to

open(*name*, O_WRONLY | O_CREAT | O_TRUNC, *mode*)

Creat creates a new file or prepares to rewrite an existing file called *name*, given as the address of a null-terminated string. If the file did not exist, it is given mode *mode*, as modified by the process's mode mask (see **umask**(2)). Also see **chmod**(2) for the construction of the *mode* argument.

If the file did exist, its mode and owner remain unchanged but it is truncated to 0 length.

The file is also opened for writing, and its file descriptor is returned.

NOTES

The *mode* given is arbitrary; it need not allow writing. This feature has been used in the past by programs to construct a simple, exclusive locking mechanism. It is replaced by the O_EXCL open mode, or the advisory locking of the **fcntl**(2) facility.

RETURN VALUE

The value −1 is returned if an error occurs. Otherwise, the call returns a non-negative descriptor that only permits writing.

ERRORS

Creat will fail and the file will not be created or truncated if one of the following occur:

[ENOTDIR] A component of the path prefix is not a directory.

[ENAMETOOLONG]
The path name exceeds PATH_MAX characters.

[ENOENT] The named file does not exist.

[ELOOP] Too many symbolic links were encountered in translating the pathname. (Minix-vmd)

[EACCES] Search permission is denied for a component of the path prefix.

[EACCES] The file does not exist and the directory in which it is to be created is not writable.

[EACCES] The file exists, but it is unwritable.

[EISDIR] The file is a directory.

[EMFILE] There are already too many files open.

[ENFILE] The system file table is full.

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

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

[EROFS] The named file resides on a read-only file system.

[ENXIO] The file is a character special or block special file, and the associated device does not exist.

[EIO] An I/O error occurred while making the directory entry or allocating the inode.

[EFAULT] *Name* points outside the process's allocated address space.

SEE ALSO

open(2), **write**(2), **close**(2), **chmod**(2), **umask**(2).