Copyright © 1991 by the
Institute of Electrical and Electronics Engineers, Inc.
345 East 47th Street
New York, NY 10017, USA
All rights reserved as an unpublished work.

This is an unapproved and unpublished IEEE Standards Draft, subject to change. The publication, distribution, or copying of this draft, as well as all derivative works based on this draft, is expressly prohibited except as set forth below.

Permission is hereby granted for IEEE Standards Committee participants to reproduce this document for purposes of IEEE standardization activities only, and subject to the restrictions contained herein.

Permission is hereby also granted for member bodies and technical committees of ISO and IEC to reproduce this document for purposes of developing a national position, subject to the restrictions contained herein.

Permission is hereby also granted to the preceding entities to make limited copies of this document in an electronic form only for the stated activities.

The following restrictions apply to reproducing or transmitting the document in any form: 1) all copies or portions thereof must identify the document's IEEE project number and draft number, and must be accompanied by this entire notice in a prominent location; 2) no portion of this document may be redistributed in any modified or abridged form without the prior approval of the IEEE Standards Department.

Other entities seeking permission to reproduce this document, or any portion thereof, for standardization or other activities, must contact the IEEE Standards Department for the appropriate license.

Use of information contained in this unapproved draft is at your own risk.

IEEE Standards Department
Copyright and Permissions
445 Hoes Lane, P.O. Box 1331
Piscataway, NJ 08855-1331, USA
+1 (908) 562-3800
+1 (908) 562-1571 [FAX]

4.33 ln — Link files

⇒ **4.33.1 ln Synopsis.** *Modify the Synopsis to be:*

```
ln [-f] source_file target_file
ln [-f] source_file ... target_dir
ln -s source_file target_file
ln -s source_file ... target_dir
```

⇒ **4.33.2 ln Description.** *Modify the entire Description subclause as follows:*

The *ln* () utility creates either hard or symbolic links, depending on the presence of the `-s` option: hard links shall be created by default, symbolic links with `-s`.

4.33.2.1 Creating Hard Links

<current contents of 4.33.2 ... >

If *source_file* is a symbolic link, `ln` shall follow the link and create a hard link at *destination* to the file referenced by *source_file* . If either *target_file* or *target_dir* is a symbolic link, the third or fourth synopsis forms (respectively) shall be assumed, even though `-s` was not specified; use of `-f` in these cases is undefined.

4.33.2.2 Creating Symbolic Links

In the third synopsis form, the `ln` utility shall create a symbolic link for the file specified by the *source_file* operand, at the *destination* path specified by the *target_file* operand. If *target_file* is a symbolic link to a nondirectory file, `ln` shall overwrite the symbolic link. This third synopsis form shall be assumed when there are two operands and the second is not an existing directory.

In the fourth synopsis form, the `ln` utility shall create a symbolic link for each file specified by a *source_file* operand, at a *destination* path in the existing directory named by *target_dir*. The corresponding destination path for each *source_file* shall be the concatenation of the target directory pathname, a slash character, and the last pathname component of the *source_file*. If *target_dir* is a symbolic link to a directory, `ln` shall follow the link and create a symbolic link to *source_file* in the directory *target_dir*.

For both the third and fourth synopsis forms:

— If the directory in which the destination is to reside does not support {POSIX 2_SYMLINKS }, results are undefined.

— The *source_file* operand can be any pathname and need not represent an existing or accessible file.

Copyright © 1992 IEEE. All rights reserved.
This is an unapproved IEEE Standards Draft, subject to change.

— The *source_file* and *destination* need not reside on the same file system.

— If the last operand specifies an existing file of a type not specified by POSIX .1 {8},the behavior is implementation defined.

For each *source_file* :

(1) If the *destination* path exists, `ln` shall write a diagnostic message to standard error, do nothing more with the current *source_file,* and go on to any remaining *source_file* s.

(2) Actions shall be performed equivalent to the POSIX .1 {8} *symlink* () function using *source_file* as the *pname* argument, and the *destination* path as the *slink* argument.

If *source_file* is itself a symbolic link, `ln` shall not follow the link, but shall create a symbolic link in *destination* to *source_file* (the symbolic link) itself.

⇒ **4.33.3 `ln` Options.** *Add the following option in the proper sorted order:*

`-s` Create symbolic links instead of the default hard links.

⇒ **4.33.4 `ln` Operands.** *Replace all of the Operands subclause with:*

The following operands shall be supported by the implementation:

source_file A pathname of a file to be linked. When `-s` is not specified this can be a regular or special file; whether a directory can be linked is implementation defined. When `-s` is specified, any pathname can be used.

target_file The pathname of the new directory entry or symbolic link to be created.

target_dir A pathname of an existing directory in which the new directory entries or symbolic links are to be created.

HLJ : Is it correct that `-f` *cannot be used with* `-s`*?*

Copyright © 1992 IEEE. All rights reserved.
This is an unapproved IEEE Standards Draft, subject to change.

4.39 `ls` — List directory contents

⇒ **4.39.1 `ls` Synopsis.** *Modify the Synopsis to be:*

```
ls [-CFLRacdilqrtu1][file ...]
```

⇒ **4.39.3 `ls` Options.** *Replace the description of the* `-F` *option with the following:*

`-F` Write a slash (`/`) immediately after each pathname that is a directory, an asterisk (`*`) after each that is executable, a vertical bar (`|`) after each that is a FIFO, and an at-sign (`@`) after each that is a symbolic link.

⇒ **4.39.3 `ls` Options.** *Add the following option in the proper sorted order:*

`-L` When a symbolic link is encountered, write the name of the file or directory the link refers to, rather than the name of the link itself.

⇒ **4.39.6.1 `ls` Standard Output.** *Replace the six-line description of* `-l` *(beginning with* ‘If the -l option is specified, ... ’’*)with:*

If the `-l` option is specified without `-L`, the following information shall be written:

`"%s %u %s %s %u %s %s\n"`, <*file mode*>, <*number of links*>, <*owner name*>, <*group name*>, <*number of bytes in the file*>, <*date and time*>, <*pathname*>

If the file is a symbolic link, this information shall be about the link itself and the <*pathname*> field shall be of the form:

`"%s -> %s"`, <*name of link*>, <*contents of link*>

If both `-l` and `-L` are specified, the following information shall be written:

`"%s %u %s %s %u %s %s\n"`, <*file mode*>, <*number of links*>, <*owner name*>, <*group name*>, <*number of bytes in the file*>, <*date and time*>, <*pathname of link*>

where all fields except <*pathname of link*> shall be for the file resolved from the symbolic link.

In both of the preceding `-l` forms, if <*owner name*> or <*group name*> cannot be determined, they shall be replaced with their associated numeric values using the format `"%u"`.

Copyright © 1992 IEEE. All rights reserved.
This is an unapproved IEEE Standards Draft, subject to change.