NAME
>   cp − copy files and directories

SYNOPSIS
>   **cp** [*OPTION*]... [*-T*] *SOURCE DEST*
>   **cp** [*OPTION*]... *SOURCE*... *DIRECTORY*
>   **cp** [*OPTION*]... *-t DIRECTORY SOURCE*...

DESCRIPTION
>   Copy SOURCE to DEST, or multiple SOURCE(s) to DIRECTORY.
>
>   Mandatory arguments to long options are mandatory for short options too.

> **−a**, **−−archive**
>>   same as **−dR −−preserve=***all*

> **−−attributes−only**
>>   don't copy the file data, just the attributes

> **−−backup**[=*CONTROL*]
>>   make a backup of each existing destination file

> **−b**   like **−−backup** but does not accept an argument

> **−−copy−contents**
>>   copy contents of special files when recursive

> **−d**   same as **−−no−dereference −−preserve=***links*

> **−f**, **−−force**
>>   if an existing destination file cannot be opened, remove it and try again (this option is ignored
>>   when the **−n** option is also used)

> **−i**, **−−interactive**
>>   prompt before overwrite (overrides a previous **−n** option)

> **−H**   follow command−line symbolic links in SOURCE

> **−l**, **−−link**
>>   hard link files instead of copying

> **−L**, **−−dereference**
>>   always follow symbolic links in SOURCE

> **−n**, **−−no−clobber**
>>   do not overwrite an existing file (overrides a previous **−i** option)

> **−P**, **−−no−dereference**
>>   never follow symbolic links in SOURCE

> **−p**   same as **−−preserve=***mode*,ownership,timestamps

> **−−preserve**[=*ATTR_LIST*]
>>   preserve the specified attributes (default: mode,ownership,timestamps), if possible additional at-
>>   tributes: context, links, xattr, all

> **−−no−preserve=***ATTR_LIST*
>>   don't preserve the specified attributes

> **−−parents**
>>   use full source file name under DIRECTORY

> **−R**, **−r**, **−−recursive**
>>   copy directories recursively

> **−−reflink**[=*WHEN*]
>>   control clone/CoW copies. See below

−−remove−destination
    remove each existing destination file before attempting to open it (contrast with **−−force**)

−−sparse=*WHEN*
    control creation of sparse files. See below

−−strip−trailing−slashes
    remove any trailing slashes from each SOURCE argument

−s, **−−symbolic−link**
    make symbolic links instead of copying

−S, **−−suffix**=*SUFFIX*
    override the usual backup suffix

−t, **−−target−directory**=*DIRECTORY*
    copy all SOURCE arguments into DIRECTORY

−T, **−−no−target−directory**
    treat DEST as a normal file

−u, **−−update**
    copy only when the SOURCE file is newer than the destination file or when the destination file is
    missing

−v, **−−verbose**
    explain what is being done

−x, **−−one−file−system**
    stay on this file system

−Z   set SELinux security context of destination file to default type

−−context[=*CTX*]
    like **−Z**, or if CTX is specified then set the SELinux or SMACK security context to CTX

−−help   display this help and exit

−−version
    output version information and exit

By default, sparse SOURCE files are detected by a crude heuristic and the corresponding DEST file is
made sparse as well. That is the behavior selected by **−−sparse**=*auto*. Specify **−−sparse**=*always* to create
a sparse DEST file whenever the SOURCE file contains a long enough sequence of zero bytes. Use
−−sparse=*never* to inhibit creation of sparse files.

When **−−reflink**[=*always*] is specified, perform a lightweight copy, where the data blocks are copied only
when modified. If this is not possible the copy fails, or if **−−reflink**=*auto* is specified, fall back to a stan-
dard copy. Use **−−reflink**=*never* to ensure a standard copy is performed.

The backup suffix is '˜', unless set with **−−suffix** or SIMPLE_BACKUP_SUFFIX. The version control
method may be selected via the **−−backup** option or through the VERSION_CONTROL environment vari-
able. Here are the values:

none, off
    never make backups (even if **−−backup** is given)

numbered, t
    make numbered backups

existing, nil
    numbered if numbered backups exist, simple otherwise

simple, never
    always make simple backups

As a special case, cp makes a backup of SOURCE when the force and backup options are given and

SOURCE and DEST are the same name for an existing, regular file.

AUTHOR

Written by Torbjorn Granlund, David MacKenzie, and Jim Meyering.

REPORTING BUGS

GNU coreutils online help: <https://www.gnu.org/software/coreutils/>
Report any translation bugs to <https://translationproject.org/team/>

COPYRIGHT

Copyright © 2020 Free Software Foundation, Inc. License GPLv3+: GNU GPL version 3 or later <https://gnu.org/licenses/gpl.html>.
This is free software: you are free to change and redistribute it. There is NO WARRANTY, to the extent permitted by law.

SEE ALSO

Full documentation <https://www.gnu.org/software/coreutils/cp>
or available locally via: info '(coreutils) cp invocation'