

**NAME**

**csplit** - split files based on context

**SYNOPSIS**

**csplit** [-ks] [-f *prefix*] [-n *number*] *file args ...*

**DESCRIPTION**

The **csplit** utility splits *file* into pieces using the patterns *args*. If *file* is a dash ('-'), **csplit** reads from standard input.

Files are created with a prefix of "xx" and two decimal digits. The size of each file is written to standard output as it is created. If an error occurs whilst files are being created, or a HUP, INT, or TERM signal is received, all files previously written are removed.

The options are as follows:

**-f** *prefix*

Create file names beginning with *prefix*, instead of "xx".

**-k** Do not remove previously created files if an error occurs or a HUP, INT, or TERM signal is received.

**-n** *number*

Create file names beginning with *number* of decimal digits after the prefix, instead of 2.

**-s** Do not write the size of each output file to standard output as it is created.

The *args* operands may be a combination of the following patterns:

*/regexl*[[+|-]*offset*]

Create a file containing the input from the current line to (but not including) the next line matching the given basic regular expression. An optional *offset* from the line that matched may be specified.

*%regex%*[[+|-]*offset*]

Same as above but a file is not created for the output.

*line\_no*

Create containing the input from the current line to (but not including) the specified line number.

*{num}*

Repeat the previous pattern the specified number of times. If it follows a line number pattern, a new file will be created for each *line\_no* lines, *num* times. The first line of the file is line number 1 for historic reasons.

After all the patterns have been processed, the remaining input data (if there is any) will be written to a new file.

Requesting to split at a line before the current line number or past the end of the file will result in an error.

## ENVIRONMENT

The LANG, LC\_ALL, LC\_COLLATE and LC\_CTYPE environment variables affect the execution of **csplit** as described in environ(7).

## EXIT STATUS

The **csplit** utility exits 0 on success, and >0 if an error occurs.

## EXAMPLES

Split the mdoc(7) file *foo.1* into one file for each section (up to 21 plus one for the rest, if any):

```
csplit -k foo.1 '%^\.Sh%' '/^\.Sh/' '{20}'
```

Split standard input after the first 99 lines and every 100 lines thereafter:

```
csplit -k - 100 '{19}'
```

## SEE ALSO

sed(1), split(1), re\_format(7)

## STANDARDS

The **csplit** utility conforms to IEEE Std 1003.1-2001 ("POSIX.1").

## HISTORY

A **csplit** command appeared in PWB UNIX.

## BUGS

Input lines are limited to LINE\_MAX (2048) bytes in length.