NAME

grep, egrep, fgrep − search a file for a pattern

SYNOPSIS

grep [option] ... expression [file] ...

egrep [option] ... [expression] [file] ...

fgrep [option] ... [strings] [file]

DESCRIPTION

Commands of the *grep* family search the input *files* (standard input default) for lines matching a pattern. Normally, each line found is copied to the standard output; unless the **−h** flag is used, the file name is shown if there is more than one input file.

Grep patterns are limited regular expressions in the style of *ed*(1); it uses a compact nondeterministic algorithm. *Egrep* patterns are full regular expressions; it uses a fast deterministic algorithm that sometimes needs exponential space. *Fgrep* patterns are fixed strings; it is fast and compact.

The following options are recognized.

−v All lines but those matching are printed.

−c Only a count of matching lines is printed.

−l The names of files with matching lines are listed (once) separated by newlines.

−n Each line is preceded by its line number in the file.

−b Each line is preceded by the block number on which it was found. This is sometimes useful in locating disk block numbers by context.

−s No output is produced, only status.

−h Do not print filename headers with output lines.

−y Lower case letters in the pattern will also match upper case letters in the input (*grep* only).

−e *expression*
Same as a simple *expression* argument, but useful when the *expression* begins with a −.

−f *file* The regular expression (*egrep*) or string list (*fgrep*) is taken from the *file*.

−x (Exact) only lines matched in their entirety are printed (*fgrep* only).

Care should be taken when using the characters $ * [ˆ | ? ´ " ( ) and \ in the *expression* as they are also meaningful to the Shell. It is safest to enclose the entire *expression* argument in single quotes ´ ´.

Fgrep searches for lines that contain one of the (newline-separated) *strings.*

Egrep accepts extended regular expressions. In the following description ‘character’ excludes newline:

> A \ followed by a single character matches that character.
>
> The character ˆ ($) matches the beginning (end) of a line.
>
> A **.** matches any character.
>
> A single character not otherwise endowed with special meaning matches that character.
>
> A string enclosed in brackets [ ] matches any single character from the string. Ranges of ASCII character codes may be abbreviated as in ‘a−z0−9’. A] may occur only as the first character of the string. A literal − must be placed where it can’t be mistaken as a range indicator.
>
> A regular expression followed by * (+, ?) matches a sequence of 0 or more (1 or more, 0 or 1) matches of the regular expression.
>
> Two regular expressions concatenated match a match of the first followed by a match of the second.

Two regular expressions separated by | or newline match either a match for the first or a match for the second.

A regular expression enclosed in parentheses matches a match for the regular expression.

The order of precedence of operators at the same parenthesis level is [ ] then *+? then concatenation then | and newline.

SEE ALSO

ed(1), sed(1), sh(1)

DIAGNOSTICS

Exit status is 0 if any matches are found, 1 if none, 2 for syntax errors or inaccessible files.

BUGS

Ideally there should be only one *grep,* but we don't know a single algorithm that spans a wide enough range of space-time tradeoffs.

Lines are limited to 256 characters; longer lines are truncated.