

NAME

fparsefn - return the next logical line from a stream

LIBRARY

System Utilities Library (libutil, -lutil)

SYNOPSIS

```
#include <stdio.h>
```

```
#include <libutil.h>
```

*char **

```
fparsefn(FILE *stream, size_t *len, size_t *lineno, const char delim[3], int flags);
```

DESCRIPTION

The **fparsefn**() function returns a pointer to the next logical line from the stream referenced by *stream*. This string is NUL terminated and it is dynamically allocated on each invocation. It is the responsibility of the caller to free the pointer.

By default, if a character is escaped, both it and the preceding escape character will be present in the returned string. Various *flags* alter this behaviour.

The meaning of the arguments is as follows:

stream

The stream to read from.

len If not NULL, the length of the string is stored in the memory location to which it points.

lineno If not NULL, the value of the memory location to which is pointed to, is incremented by the number of lines actually read from the file.

delim Contains the escape, continuation, and comment characters. If a character is NUL then processing for that character is disabled. If NULL, all characters default to values specified below. The contents of *delim* is as follows:

delim[0] The escape character, which defaults to \, is used to remove any special meaning from the next character.

delim[1] The continuation character, which defaults to \, is used to indicate that the next line should be concatenated with the current one if this character is the last character on the

current line and is not escaped.

delim[2] The comment character, which defaults to #, if not escaped indicates the beginning of a comment that extends until the end of the current line.

flags If non-zero, alter the operation of **fparseln()**. The various flags, which may be *or*-ed together, are:

FPARSELN_UNESCCOMM Remove escape preceding an escaped comment.

FPARSELN_UNESCCONT Remove escape preceding an escaped continuation.

FPARSELN_UNESCESC Remove escape preceding an escaped escape.

FPARSELN_UNESCREST Remove escape preceding any other character.

FPARSELN_UNESCALL All of the above.

RETURN VALUES

Upon successful completion a pointer to the parsed line is returned; otherwise, NULL is returned.

The **fparseln()** function uses internally **fgetln(3)**, so all error conditions that apply to **fgetln(3)**, apply to **fparseln()**. In addition **fparseln()** may set *errno* to ENOMEM and return NULL if it runs out of memory.

SEE ALSO

fgetln(3)

HISTORY

The **fparseln()** function first appeared in NetBSD 1.4 and FreeBSD 4.0.