

fgets

Defined in header <stdio.h>

```
char *fgets( char *str, int count, FILE *stream );    (until C99)
char *fgets( char *restrict str, int count, FILE *restrict stream );    (since C99)
```

Reads at most `count - 1` characters from the given file stream and stores them in the character array pointed to by `str`. Parsing stops if a newline character is found, in which case `str` will contain that newline character, or if end-of-file occurs. If bytes are read and no errors occur, writes a null character at the position immediately after the last character written to `str`.

Parameters

str - pointer to an element of a char array
count - maximum number of characters to write (typically the length of `str`)
stream - file stream to read the data from

Return value

`str` on success, null pointer on failure.

If the end-of-file condition is encountered, sets the *eof* indicator on stream (see `feof()`). This is only a failure if it causes no bytes to be read, in which case a null pointer is returned and the contents of the array pointed to by `str` are not altered (i.e. the first byte is not overwritten with a null character).

If the failure has been caused by some other error, sets the *error* indicator (see `ferror()`) on stream. The contents of the array pointed to by `str` are indeterminate (it may not even be null-terminated).

Notes

POSIX additionally requires (<http://pubs.opengroup.org/onlinepubs/9699919799/functions/fgets.html>) that `fgets` sets `errno` if it encounters an failure other than the end-of-file condition.

Although the standard specification is unclear (<https://stackoverflow.com/questions/23388620>) in the cases where `count <= 1`, common implementations do

- if `count < 1`, do nothing, report error
- if `count == 1`,
 - some implementations do nothing, report error,
 - others read nothing, store zero in `str[0]`, report success

Example

Run this code

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

int main(void)
{
    FILE* tmpf = tmpfile();
    fputs("Alan Turing\n", tmpf);
    fputs("John von Neumann\n", tmpf);
    fputs("Alonzo Church\n", tmpf);

    rewind(tmpf);

    char buf[8];
    while (fgets(buf, sizeof buf, tmpf) != NULL)
        printf("%s\n", buf);

    if (feof(tmpf))
        puts("End of file reached");
}
```

Output:

```
"Alan Tu"
"ring
"
"John vo"
"n Neuma"
"nn
"
"Alonzo "
"Church
"
End of file reached
```

References

- C11 standard (ISO/IEC 9899:2011):
 - 7.21.7.2 The fgets function (p: 331)
- C99 standard (ISO/IEC 9899:1999):
 - 7.19.7.2 The fgets function (p: 296)
- C89/C90 standard (ISO/IEC 9899:1990):
 - 4.9.7.2 The fgets function

See also

scanf	
fscanf	
sscanf	reads formatted input from stdin, a file stream or a buffer
scanf_s (C11)	(function)
fscanf_s (C11)	
sscanf_s (C11)	
gets (removed in C11)	reads a character string from stdin
gets_s (C11)	(function)
fputs	writes a character string to a file stream
getline	
getwline (dynamic memory TR)	read from a stream into a automatically resized buffer until delimiter/end of line
getdelim	(function)
getwdelim	

C++ documentation for **fgets**

Retrieved from "https://en.cppreference.com/mwiki/index.php?title=c/io/fgets&oldid=123647"