

🏠 (<http://cppstudio.com>)

/ Стандартные заголовочные файлы из Си в C++ (<http://cppstudio.com/cat/309/>)

/ Заголовочный файл cstdio (stdio.h) (<http://cppstudio.com/cat/309/323/>) / Функция fgetc

Функция fgetc



Оценка: **4,50** (голосов: 2)

Чтобы проголосовать, вы должны зарегистрироваться.

Прототип функции fgetc:

```
1 char * fgetc( char * string, int num, FILE * filestream );
```

Заголовочный файл

Название	Язык
stdio.h	C
cstdio	C++

Описание

Функция `fgetc` считывает символы из потока и сохраняет их в виде строки в параметр `string` до тех пор пока не наступит конец строки или пока не будет достигнут конец файла.

Символ новой строки прекращает работу функции `fgetc`, но он считается допустимым символом, и поэтому он копируется в строку `string`.

Нулевой символ автоматически добавляется в строку `str` после прочитанных символов, чтобы сигнализировать о конце строки.

Параметры:

- **string**
Указатель на массив типа `char`, в который сохраняются считанные символы.
- **num**
Максимальное количество символов для чтения, включая нулевой символ.
- **filestream**
Указатель на объект типа `FILE`, который идентифицирует поток, из которого считываются символы.

Для чтения из стандартного ввода, `stdin` может быть использован в качестве этого параметра.

Возвращаемое значение

В случае успеха, функция возвращает указатель на `string`. Если конец файла был достигнут и ни один символ не был прочитан, содержимое `string` остается неизменными и возвращается нулевой указатель. Если происходит ошибка, возвращается нулевой указатель.

Используйте функции `ferror` или `feof` для проверки, произошла ошибка или был достигнут конец-в-файла.

Пример: исходный код программы

```
1  //пример использования функции fgetc
2
3  #include <stdio>
4
5  int main()
6  {
7      FILE * ptrFile = fopen("file.txt" , "r");
8      char mystring [100];
9
10     if (ptrFile == NULL) perror("Ошибка открытия файла");
11     else
12     {
13         if ( fgetc(mystring, 100, ptrFile) != NULL ) // считать символы из файла
14             puts(mystring); // вывод на экран
15         fclose (ptrFile);
16     }
17     return 0;
18 }
```

Пример работы программы

Этот пример читает первую строку `file.txt` или первые 99 символов, в зависимости от того, какое из событий наступит раньше, и выводит их на экран.

💬 Обсудить на форуме (/topics/)

👤 Автор: Marienko L. (/forums/users/liliia/)

📅 Дата: 15.09.2012

🔗 Поделиться:

Похожие статьи:

1. `string`: шаблонный строковый класс STL (<http://cppstudio.com/post/6110/>)
2. Функция `ungetc` (<http://cppstudio.com/post/1644/>)
3. Си-строки (<http://cppstudio.com/post/9567/>)
4. Функция `fread` (<http://cppstudio.com/post/1641/>)
5. Функция `clearerr` (<http://cppstudio.com/post/1571/>)

Оставить комментарий

Вы должны войти (http://cppstudio.com/wp-login.php?redirect_to=http%3A%2F%2Fcppstudio.com%2Fpost%2F1699%2F), чтобы оставить комментарий.

Translation

 (/post/1699/)Русский (/post/1699/)
 (/uk/post/1699/)Українська (/uk/post/1699/)
 (/en/post/1699/)English (/en/post/1699/)
 (/de/post/1699/)Deutsch (/de/post/1699/)
 (/be/post/1699/)Беларуская (/be/post/1699/)
 (/kk/post/1699/)Қазақ тілі (/kk/post/1699/)
 (/uz/post/1699/)O'zbek tili (/uz/post/1699/)
 (/tr/post/1699/)Türkçe (/tr/post/1699/)

Новое

- Особенности Qt: слоты и сигналы, описание QObject и QApplication, виды окон и т.д.
(<http://cppstudio.com/post/11167/>)
- Первая программа на Qt:
(<http://cppstudio.com/post/11127/>)
- Введение – графическая библиотека Qt
(<http://cppstudio.com/post/11097/>)
- Наследование классов
(<http://cppstudio.com/post/10103/>)

Популярное

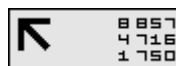
Sorry. No data so far.

➤ Перегрузка операторов в C++
(часть 2)
(<http://cppstudio.com/post/10058/>)

© 2022 CppStudio - Программирование для начинающих на C++



(<https://plus.google.com/u/0/106109650739084338784>)



(<http://www.liveinternet.ru/click>)



(<http://orphus.ru>)