

🏠 (<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 ungetc

Функция ungetc



Оценка: **4,00** (голосов: 1)

Чтобы проголосовать, вы должны зарегистрироваться.

Прототип функции ungetc:

```
1 int ungetc( int character, FILE * filestream );
```

Заголовочный файл

Название	Язык
stdio.h	C
cstdio	C++

Описание

Функция `ungetc` возвращает только что прочитанный символ обратно в поток ввода `filestream`, через параметр `character`. Внутренний индикатор позиции файла уменьшается обратно, на предыдущее положение, так что этот символ возвращается при следующем вызове операции чтения для этого потока.

Параметр `character` может содержать любой символ, например, последний символ прочитанный из потока в предыдущей операции или любой другой. В обоих случаях, значение, полученное по следующей операции чтения является значением функции `ungetc`, независимо от символа `character`.

Обратите внимание, что данная функция влияет только на следующую операцию чтения для данного потока, а не на содержание файла, связанного с потоком, который не изменяется при любом вызове этой функции.

Если внутренний показатель конца файла EOF был установлен, то после вызова этой функции он очищается.

Вызовы функций `fseek`, `fsetpos` или `rewind` для потока `stream` совместно с функцией `ungetc`, будут отбрасывать любые символы назад.

Если аргумент параметра `character` – EOF, функция завершается, не изменяя входной поток.

Параметры:

- **character**

Символ, возвращаемый обратно в поток. Символ передается, как значение типа `int`.

- **filestream**

Указатель на объект типа `FILE`, который идентифицирует входной поток.

Возвращаемое значение

В случае успеха, возвращается целочисленное значение символа, который был перенесен обратно в поток. В противном случае, возвращается значение `EOF`, и поток остается неизменным.

Пример: исходный код программы

```
1 //пример использования функции ungetc
2
3 #include <stdio>
4 #include <iostream>
5
6 int main()
7 {
8     FILE * ptrFile = fopen("file.txt", "rt");
9
10    char buffer [256];
11
12    if (ptrFile == NULL) perror("Ошибка открытия файла");
13    else
14    {
15        while ( !feof(ptrFile) )           // пока не конец файла
16        {
17            int c = getc(ptrFile);           // считать символ из файла
18
19            if (c == '%')                   // если считанный символ - %
20                ungetc('~', ptrFile);       // вернуть символ '~' в поток ввода
21            else
22                ungetc(c, ptrFile);          // вернуть считанный символ в поток в
23
24            fgets(buffer, 255, ptrFile);     // считать символы из файла и сохранит
25            fputs(buffer, stdout);           // вывод на экран содержимого буфера
26        }
27
28        fclose(ptrFile);
29    }
30
31    return 0;
32 }
```

Пример работы программы

Эта программа открывает существующий файл с именем `file.txt` для чтения и выводит его строки, но сначала получает первый символ каждой строки и кладет его обратно в поток, за исключением случаев, когда строка начинается с символа `%`, и в этом случае такой символ не будет заменен на `~`.

Обсудить на форуме (/topics/)

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

Дата: 15.09.2012

Поделиться:

Похожие статьи:

1. Функция fread (<http://cppstudio.com/post/1641/>)
2. Функция fflush (<http://cppstudio.com/post/1249/>)
3. Функция rewind (<http://cppstudio.com/post/1576/>)
4. Функция fopen (<http://cppstudio.com/post/1253/>)
5. Тип данных FILE (<http://cppstudio.com/post/1544/>)









Комментарии

Оставить комментарий

Вы должны войти (<http://cppstudio.com/wp-login.php?>

[redirect_to=http%3A%2F%2Fcppstudio.com%2Fpost%2F1644%2F](http://cppstudio.com/wp-login.php?redirect_to=http%3A%2F%2Fcppstudio.com%2Fpost%2F1644%2F)), чтобы оставить комментарий.

Translation

 (/post/1644/)Русский (/post/1644/)
 (/uk/post/1644/)Українська (/uk/post/1644/)
 (/en/post/1644/)English (/en/post/1644/)
 (/de/post/1644/)Deutsch (/de/post/1644/)
 (/be/post/1644/)Беларуская (/be/post/1644/)
 (/kk/post/1644/)Қазақ тілі (/kk/post/1644/)
 (/uz/post/1644/)O'zbek tili (/uz/post/1644/)
 (/tr/post/1644/)Türkçe (/tr/post/1644/)

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/>)

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

Популярное

Sorry. No data so far.

© 2022 CppStudio – Программирование для начинающих на C++



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

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



(<http://orphus.ru>)