

🏠 (<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 fopen

Функция fopen



Оценка: **4,00** (голосов: 2)

Чтобы проголосовать, вы должны зарегистрироваться.

Прототип функции fopen:

```
1 FILE * fopen( const char * fname, const char * modeopen );
```

Заголовочный файл

Название	Язык
stdio.h	C
cstdio	C++

Описание

Функция `fopen` открывает файл, имя которого указано в параметре `fname` и связывает его с потоком, который может быть идентифицирован для выполнения различных операций с файлом. Операции с потоком, выполнение которых разрешено определяются параметром `modeopen`. Среда программирования поддерживает, по крайней мере, `FOPEN_MAX` – количество открытых файлов одновременно, где `FOPEN_MAX` является макро-константа, определенная в `<cstdio>`.

Параметры:

- **fname**

Си-строка, содержащая имя файла, который необходимо открыть. Этот параметр должен соответствовать правилам именования файлов в используемой системе, и может включать в себя путь, если система поддерживает его.

- **modeopen**

Строка, содержащая режим доступа к файлу. Далее приведён список режимов и их описание:

«r»	Режим открытия файла для чтения. Файл должен существовать.
-----	--

«w»	Режим создания пустого файла для записи. Если файл с таким именем уже существует его содержимое стирается, и файл рассматривается как новый пустой файл.
«a»	Дописать в файл. Операция добавления данных в конец файла. Файл создается, если он не существует.
«r+»	Режим открытия файла для обновления чтения и записи. Этот файл должен существовать.
«w+»	Создаёт пустой файл для чтения и записи. Если файл с таким именем уже существует его содержимое стирается, и файл рассматривается как новый пустой файл.
«a+»	Открыть файл для чтения и добавления данных. Все операции записи выполняются в конец файла, защищая предыдущее содержания файла от случайного изменения. Вы можете изменить позицию (<code>fseek</code> , перемотка назад) внутреннего указателя на любое место файла только для чтения, операции записи будет перемещать указатель в конец файла, и только после этого дописывать новую информацию. Файл создается, если он не существует.

Выше указанные спецификаторы режима доступа к файлу используются только в текстовых файлах. Для того, чтобы открыть двоичный файл, символ `b` должен быть включен в режим доступа. Этот дополнительный символ `b` может быть добавлен в конец строки, что даёт следующие режимы доступа к бинарным файлам: `rb`, `wb`, `ab`, `r+b`, `w+b`, `a+b` или может быть вставлен между буквой и знаком `+`, в случае со смешанными режимами: `rb+`, `wb+`, `ab+`.

В последовательности спецификаторов доступа могут находиться дополнительные символы, хотя они никак не влияют на работу программы. Например, символ `'t'` иногда прилагается, чтобы показать, что файл является текстовым.

В случае текстовых файлов, в зависимости от среды, в которой работает приложение, некоторые специальные символы преобразования могут в операциях ввода/вывода могут быть адаптированы к системе, из-за специфического формата текстового файла. Во многих системах, например, UNIX-системах, это не имеет значения ни для открытия файла в текстовом режиме ни для открытия файла в двоичном режиме. Оба попадают точно так же, но адаптация служебных символов рекомендуется для лучшей переносимости.

Для смешанных режимов (те, которые включают в себя знак «+»), когда разрешены операции чтения и записи (или добавления), следует обнулить поток функцией `fflush` или изменить позицию внутреннего указателя (`fseek` `fsetpos`, `rewind`) после операции чтения, за которой следует операция записи и, наоборот, после операции записи, за которой следует операция чтения.

Возвращаемое значение

Если файл был успешно открыт, функция возвращает указатель на объект файла, который используется для идентификации потока и выполнения операций с файлом. В противном случае, возвращается нулевой указатель.

Пример: исходный код программы

```
1 //пример использования функции fopen
2
3 #include <iostream>
4 #include <cstdio>
5
6 int main ()
7 {
8     FILE * ptrFile = fopen("file.txt", "w");
9
10    if (ptrFile != NULL)
11    {
12        fputs("Пример использования функции fopen ", ptrFile); // записать строку в
13        fclose (ptrFile);
14    }
15    return 0;
16 }
```

Пример работы программы

Записать в только что открытый файл file.txt строку "Пример использования функции fopen" .

Обсудить на форуме (/topics/)

Автор: admin (/forums/users/admin/)

Дата: 10.09.2012

Поделиться:

Похожие статьи:







1. Работа с файлами в C++ (<http://cppstudio.com/post/446/>)
2. Функция freopen (<http://cppstudio.com/post/1257/>)
3. Классы в C++ (<http://cppstudio.com/post/439/>)
4. Функция setvbuf (<http://cppstudio.com/post/1265/>)
5. Функция ungetc (<http://cppstudio.com/post/1644/>)



Комментарии

Оставить комментарий

Вы должны войти (http://cppstudio.com/wp-login.php?redirect_to=http%3A%2F%2Fcppstudio.com%2Fpost%2F1253%2F), чтобы оставить комментарий.

Translation

 (/post/1253/)Русский (/post/1253/)
 (/uk/post/1253/)Українська (/uk/post/1253/)
 (/en/post/1253/)English (/en/post/1253/)
 (/de/post/1253/)Deutsch (/de/post/1253/)
 (/be/post/1253/)Беларуская (/be/post/1253/)
 (/kk/post/1253/)Қазақ тілі (/kk/post/1253/)

 (/uz/post/1253/) O'zbek tili (/uz/post/1253/)
 (/tr/post/1253/) Türkçe (/tr/post/1253/)

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/>)

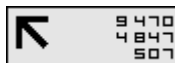
Популярное

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>)