

Search:

Go

Logged in as: legioner9

Account logout

Reference <cstdio> freopen

C++
Information
Tutorials
Reference
Articles
Forum

Reference
C library: <assert> (assert.h) <cctype> (ctype.h) <errno> (errno.h) <fcntl> (fcntl.h) <float> (float.h) <inttypes> (inttypes.h) <iso646> (iso646.h) <limits> (limits.h) <locale> (locale.h) <math> (math.h) <setjmp> (setjmp.h) <signal> (signal.h) <stdarg> (stdarg.h) <stdbool> (stdbool.h) <stddef> (stddef.h) <stdint> (stdint.h) <stdio> (stdio.h) <stdlib> (stdlib.h) <string> (string.h) <tgmath> (tgmath.h) <time> (time.h) <uchar> (uchar.h) <wchar> (wchar.h) <wctype> (wctype.h) Контейнеры: Ввод/вывод: Многопоточность: Другое:

<cstdio> (stdio.h)
функции: clearerr fclose feof ferror fflush fgetc fgetpos fgets fopen fprintf fputc fputs fread freopen fscanf fseek fsetpos ftell fwrite getc getchar получает perror printf putc putchar ставит удалить переименовать перемотка назад scanf setbuf setvbuf snprintf sprintf scanf tmpfile tmpnam ungetc vfprintf vscanf vprintf vscanf vsprintf vsscanf Объекты: stderr stdin stdout типы: ФАЙЛ fpos_t

function

freopen

<cstdio>

FILE * freopen (const char * filename, const char * mode, FILE * stream);

Reopen stream with different file or mode

Повторно *использует stream*, чтобы либо открыть файл, указанный *именем* файла, либо изменить его *режим доступа*.

Если указано новое *имя* файла, функция сначала пытается закрыть любой файл, уже связанный с *stream* (третий параметр), и разъединяет его. Затем, независимо от того, был ли этот поток успешно закрыт или нет, *freopen* открывает файл, указанный *именем* файла, и связывает его с *поток*ом так же, как это сделал бы *fopen*, используя указанный *режим*.

Если *имя* файла является нулевым указателем, функция пытается изменить *режим* потока. Хотя конкретной реализации библиотеки разрешено ограничивать разрешенные изменения и при каких обстоятельствах.

Индикатор *ошибки* и *индикатор eof* автоматически очищаются (как если бы был вызван *clearerr*).

Эта функция особенно полезна для перенаправления предопределенных потоков, таких как *stdin*, *stdout* и *stderr*, в определенные файлы (см. Пример ниже).

Параметры

имя файла

Строка C, содержащая имя файла, который нужно открыть.

Его значение должно соответствовать спецификациям имени файла рабочей среды и может включать путь (если поддерживается системой).

Если этот параметр является нулевым указателем, функция пытается изменить режим *потока*, как если бы использовалось имя файла, связанное в данный момент с этим потоком.

режим

Строка C, содержащая режим доступа к файлу. Это может быть:

"r"	читать: Открыть файл для операций ввода. Файл должен существовать.
"w"	запись: Создание пустого файла для операций вывода. Если файл с таким же именем уже существует, его содержимое отбрасывается, и файл рассматривается как новый пустой файл.
"a"	добавить: Открыть файл для вывода в конце файла. Операции вывода всегда записывают данные в конце файла, расширяя его. Операции перемещения (<i>fseek</i> , <i>fsetpos</i> , <i>rewind</i>) игнорируются. Файл создается, если он не существует.
"r+"	чтение/обновление: Откройте файл для обновления (как для ввода, так и для вывода). Файл должен существовать.
"w+"	запись/обновление: Создайте пустой файл и откройте его для обновления (как для ввода, так и для вывода). Если файл с таким же именем уже существует, его содержимое отбрасывается, и файл рассматривается как новый пустой файл.
"a+"	добавление/обновление: Откройте файл для обновления (как для ввода, так и для вывода) со всеми операциями вывода, записывающими данные в конце файла. Операции перемещения (<i>fseek</i> , <i>fsetpos</i> , <i>rewind</i>) влияют на следующие операции ввода, но операции вывода перемещают позицию обратно в конец файла. Файл создается, если он не существует.

С помощью спецификаторов режима выше файл открывается как *текстовый файл*. Чтобы открыть файл в виде *двоичного файла*, в строку *mode* должен быть включен символ "b". Этот дополнительный символ "b" может быть либо добавлен в конце строки (таким образом, делая следующие составные режимы: "rb", "wb", "ab", "r+b", "w+b", "a+b"), либо вставлен между буквой и знаком "+" для смешанных режимов ("rb+", "wb+", "ab+").

Новый стандарт C (C2011, который не является частью C++) добавляет новый стандартный подспецификатор ("x"), который может быть добавлен к любому спецификатору "w" (для формирования "wx", "wbx", "w+x" или "w+bx"/"wb+x"). Этот подспецификатор заставляет функцию терпеть неудачу, если файл существует, вместо того, чтобы перезаписывать его.

Если за последовательностью следуют дополнительные символы, поведение зависит от реализации библиотеки: некоторые реализации могут игнорировать дополнительные символы, так что, например, дополнительный "t" (иногда используемый для явного указания *текстового файла*) принят.

В некоторых реализациях библиотек открытие или создание текстового файла в режиме обновления может обрабатывать поток как двоичный файл.

поток

указатель на *файловый* объект, идентифицирующий поток для повторного открытия.

Возвращаемое значение

Если файл успешно открыт заново, функция возвращает указатель, переданный как *поток параметров*, который можно использовать для идентификации вновь открытого потока.
В противном случае возвращается нулевой указатель.
В большинстве реализаций библиотек переменная *errno* также устанавливается в системный код ошибки при сбое.

Пример

```
1 /* freopen example: redirecting stdout */
2 #include <stdio.h>
3
4 int main ()
5 {
6     freopen ("myfile.txt","w",stdout);
7     printf ("This sentence is redirected to a file.");
8     fclose (stdout);
9     return 0;
10 }
```

Редактирование
и
запуск

```
макроконстанты:  
BUFSIZ  
EOF  
FILENAME_MAX  
FOPEN_MAX  
L_tmpnam  
NULL  
TMP_MAX
```

Этот пример кода перенаправляет вывод, который обычно идет на стандартный вывод, в файл с именем myfile.txt, что после выполнения этой программы содержит:

This sentence is redirected to a file.

См. Также

<code>fopen</code>	Открыть файл (функция)
<code>fclose</code>	Закрыть файл (функцию)