



Программирование на С и С++

Онлайн справочник программиста на С и С++

[Главная](#)[Язык С](#)[Язык С++](#)[Функции](#)[Термины](#)[Блоги](#)[Главная](#)

Случайные статьи

Динамическое выделение и указатели

`tell`

`movedata`

`stime`

Библиотека классов ввода/вывода С++

`intr`

`bar`, `bar3d`

Пробелы и круглые скобки

Инициализация безразмерных массивов

Работа со строками: `fgets()` и `fputs()`

`fread()` и `fwrite()`

`ferror()` и `rewind()`

Выражения

`chmod`

`installuserdriver`

Стиль объявления указателей и ссылок

`fseek()` и произвольный доступ

`strstr`, `_fstrstr`

`acos`, `acosl`

Правила видимости для функций

Операторы выбора

Раннее и позднее связывание

`lfind`, `lsearch`

`biosprint`, `_bios_printer`

`_dos_read`

Файловый ввод/вывод

`_dos_getdate`, `_dos_gettime`

Спецификатор `%n`

`tanh`, `tanh1`

`highvideo`



longjmp



Ураганный онлайн-шутер



Высокочастотные VPS от 1480 ₽/ час
от 370 ₽

void longjmp(jmp_buf, int val)

Прототип:

setjmp.h

Описание:

Функция `longjmp()` вызывает продолжение исполнения программы, начиная с точки последнего вызова функции `setjmp()`. Эти две функции определяют способ переходов между функциями. `longjmp()` функционирует путем переустановки стека в состояние, определяемое параметром `envbuf`, который должен был быть предварительно установлен вызовом `setjmp()`. Это вызывает продолжение исполнения программы, начиная с инструкции, следующей за вызовом `setjmp()`. Тем самым компьютер как бы обманывают, заставляют думать, что он никогда не покидал функцию, вызвавшую `setjmp()`.

Буфер `envbuf` имеет тип `jmp_buf`, определенный в заголовочном файле `setjmp.h`. Буфер должен был быть предварительно установлен посредством вызова `setjmp()`, предшествовавшего вызову функции `longjmp()`.

Величина `val` становится возвращаемым значением функции `setjmp()` и может быть проанализирована с целью определения, откуда был совершен длинный переход. Единственным недопустимым значением является 0.

Важно понимать, что функция `longjmp()` должна быть вызвана перед выходом из функции, которая вызвала `setjmp()`. В противном случае результат технически не определен. (Фактически возникает аварийная ситуация.)

По существу наиболее употребительным использованием `longjmp()` является выход из глубоко вложенных процедур при возникновении фатальной ошибки.

Пример:

Следующая программа выводит «1 2 3»:

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

jmp_buf ebuf;
void f2(void);

int main(void)
{
    char first=1;
```

```

int i;
printf("1 ");
i = setjmp(ebuf);
if (first) {
    first =! first;
    f2();
    printf ("this will not be printed");
}
printf("%d", i);
return 0;
}

void f2(void)
{
    printf("2 ");
    longjmp(ebuf, 3);
}

```

Назначение платежа

На хостинг

Сумма

100 ₽

Пожертвовать

Смешанные функции

abs	assert	atof, atold
atoi	atol	bsearch
div	ecvt	fcvt
gcvt	getenv	getpass
getpid	itoa	labs
ldiv	lfind, lsearch	localeconv
longjmp	ltoa, ultoa	max, min
mblen	mbstowcs	mbtowc
nosound	putenv	qsort

1 2 3

[следующая ›](#)
[последняя »](#)

AMA

СОВКОМБАНК

О «СОВКОМБАНК»
ЛИЦЕНЗИЯ ЦБ РФ №963

**Выдаем кредиты
каждый день!**

Денежный кредит под 19,9%



ЗАПОЛНИТЬ ЗАЯВКУ

[Главная](#)

[Язык С](#)

[Язык С++](#)

[Функции](#)

[Термины](#)

[Блоги](#)

