

**NAME**

`div`, `ldiv`, `lldiv`, `imaxdiv` - compute quotient and remainder of an integer division

**SYNOPSIS**

```
#include <stdlib.h>
```

```
div_t div(int numerator, int denominator);
```

```
ldiv_t ldiv(long numerator, long denominator);
```

```
lldiv_t lldiv(long long numerator, long long denominator);
```

```
#include <inttypes.h>
```

```
imaxdiv_t imaxdiv(intmax_t numerator, intmax_t denominator);
```

Feature Test Macro Requirements for glibc (see [feature\\_test\\_macros\(7\)](#)):

```
lldiv():
```

```
  _XOPEN_SOURCE >= 600 || _ISOC99_SOURCE || _POSIX_C_SOURCE >= 200112L;  
  or cc -std=c99
```

**DESCRIPTION**

The `div()` function computes the value *numerator/denominator* and returns the quotient and remainder in a structure named *div\_t* that contains two integer members (in unspecified order) named *quot* and *rem*. The quotient is rounded toward zero. The result satisfies *quot\*denominator+rem = numerator*.

The `ldiv()`, `lldiv()`, and `imaxdiv()` functions do the same, dividing numbers of the indicated type and returning the result in a structure of the indicated name, in all cases with fields *quot* and *rem* of the same type as the function arguments.

**RETURN VALUE**

The *div\_t* (etc.) structure.

**ATTRIBUTES**

**Multithreading** (see [pthreads\(7\)](#))

The `div()`, `ldiv()`, `lldiv()`, and `imaxdiv()` functions are thread-safe.

**CONFORMING TO**

SVr4, 4.3BSD, C89, C99. The functions `lldiv()` and `imaxdiv()` were added in C99.

**EXAMPLE**

After

```
div_t q = div(-5, 3);
```

the values *q.quot* and *q.rem* are -1 and -2, respectively.

**SEE ALSO**

[abs\(3\)](#), [remainder\(3\)](#)

**COLOPHON**

This page is part of release 3.74 of the Linux *man-pages* project. A description of the project, information about reporting bugs, and the latest version of this page, can be found at <http://www.kernel.org/doc/man-pages/>.