

**NAME**

`ecvt_r`, `fcvt_r`, `qecvt_r`, `qfcvt_r` – convert a floating-point number to a string

**SYNOPSIS**

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

```
int ecvt_r(double number, int ndigits, int *decpt,
           int *sign, char *buf, size_t len);
```

```
int fcvt_r(double number, int ndigits, int *decpt,
           int *sign, char *buf, size_t len);
```

```
int qecvt_r(long double number, int ndigits, int *decpt,
            int *sign, char *buf, size_t len);
```

```
int qfcvt_r(long double number, int ndigits, int *decpt,
            int *sign, char *buf, size_t len);
```

Feature Test Macro Requirements for glibc (see `feature_test_macros(7)`):

```
ecvt_r(), fcvt_r(), qecvt_r(), qfcvt_r():
```

```
/* Glibc since 2.19: */ _DEFAULT_SOURCE
```

```
|| /* Glibc versions <= 2.19: */ _SVID_SOURCE || _BSD_SOURCE
```

**DESCRIPTION**

The functions `ecvt_r()`, `fcvt_r()`, `qecvt_r()`, and `qfcvt_r()` are identical to `ecvt(3)`, `fcvt(3)`, `qecvt(3)`, and `qfcvt(3)`, respectively, except that they do not return their result in a static buffer, but instead use the supplied `buf` of size `len`. See `ecvt(3)` and `qecvt(3)`.

**RETURN VALUE**

These functions return 0 on success, and `-1` otherwise.

**ATTRIBUTES**

For an explanation of the terms used in this section, see `attributes(7)`.

Interface	Attribute	Value
<code>ecvt_r()</code> , <code>fcvt_r()</code> , <code>qecvt_r()</code> , <code>qfcvt_r()</code>	Thread safety	MT-Safe

**CONFORMING TO**

These functions are GNU extensions.

**NOTES**

These functions are obsolete. Instead, `sprintf(3)` is recommended.

**SEE ALSO**

`ecvt(3)`, `qecvt(3)`, `sprintf(3)`

**COLOPHON**

This page is part of release 5.02 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 <https://www.kernel.org/doc/man-pages/>.