

**NAME**

fseeko, ftello – seek to or report file position

**LIBRARY**

Standard C library (*libc*, *-lc*)

**SYNOPSIS**

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

```
int fseeko(FILE *stream, off_t offset, int whence);
off_t ftello(FILE *stream);
```

Feature Test Macro Requirements for glibc (see **feature\_test\_macros(7)**):

```
fseeko(), ftello():
    _FILE_OFFSET_BITS == 64 || _POSIX_C_SOURCE >= 200112L
```

**DESCRIPTION**

The **fseeko()** and **ftello()** functions are identical to **fseek(3)** and **ftell(3)** (see **fseek(3)**), respectively, except that the *offset* argument of **fseeko()** and the return value of **ftello()** is of type *off\_t* instead of *long*.

On some architectures, both *off\_t* and *long* are 32-bit types, but defining **\_FILE\_OFFSET\_BITS** with the value 64 (before including *any* header files) will turn *off\_t* into a 64-bit type.

**RETURN VALUE**

On successful completion, **fseeko()** returns 0, while **ftello()** returns the current offset. Otherwise, *-1* is returned and *errno* is set to indicate the error.

**ERRORS**

See the ERRORS in **fseek(3)**.

**VERSIONS**

These functions are available since glibc 2.1.

**ATTRIBUTES**

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

Interface	Attribute	Value
<b>fseeko()</b> , <b>ftello()</b>	Thread safety	MT-Safe

**STANDARDS**

POSIX.1-2001, POSIX.1-2008, SUSv2.

**NOTES**

The declarations of these functions can also be obtained by defining the obsolete **\_LARGE-FILE\_SOURCE** feature test macro.

**SEE ALSO**

**fseek(3)**