

f is normalized so that either $0.5 \leq f < 1$ or $f = 0$. Stores n in the object pointed to by `exp`.

Returns f , the fractional part of value.

23.3

fscanf *Formatted File Read* `<stdio.h>`

```
int fscanf(FILE * restrict stream,
           const char * restrict format, ...);
```

Reads input items from the stream pointed to by `stream`. The string pointed to by `format` specifies the format of the items to be read. The arguments that follow `format` point to objects in which the items are to be stored.

Returns Number of input items successfully read and stored. Returns EOF if an input failure occurs before any items can be read. 22.3

fseek *File Seek* `<stdio.h>`

```
int fseek(FILE *stream, long int offset, int whence);
```

Changes the file position indicator for the stream pointed to by `stream`. If `whence` is `SEEK_SET`, the new position is the beginning of the file plus `offset` bytes. If `whence` is `SEEK_CUR`, the new position is the current position plus `offset` bytes. If `whence` is `SEEK_END`, the new position is the end of the file plus `offset` bytes. The value of `offset` may be negative. For text streams, either `offset` must be zero or `whence` must be `SEEK_SET` and `offset` a value obtained by a previous call of `ftell`. For binary streams, `fseek` may not support calls in which `whence` is `SEEK_END`.

Returns Zero if the operation is successful, nonzero otherwise.

22.7

fsetpos *Set File Position* `<stdio.h>`

```
int fsetpos(FILE *stream, const fpos_t *pos);
```

Sets the file position indicator for the stream pointed to by `stream` according to the value pointed to by `pos` (obtained from a previous call of `fgetpos`).

Returns Zero if successful. If the call fails, returns a nonzero value and stores an implementation-defined positive value in `errno`. 22.7

ftell *Determine File Position* `<stdio.h>`

```
long int ftell(FILE *stream);
```

Returns Current file position indicator for the stream pointed to by `stream`. If the call fails, returns `-1L` and stores an implementation-defined positive value in `errno`.

22.7

fwide *Get and Set Stream Orientation (C99)* `<wchar.h>`

```
int fwide(FILE *stream, int mode);
```

Determines the current orientation of a stream and, if desired, attempts to set its orientation. If `mode` is greater than zero, `fwide` tries to make the stream wide-oriented if it has no orientation. If `mode` is less than zero, it tries to make the