be calculated with respect to the beginning of the file, the current position, or the end of the file. <stdio.h> defines three macros for this purpose:

SEEK_SET	Beginning of file
SEEK_CUR	Current file position
SEEK_END	End of file

The second argument is a (possibly negative) byte count. To move to the beginning of a file, for example, the seek direction would be SEEK_SET and the byte count would be zero:

```
fseek(fp, 0L, SEEK_SET);      /* moves to beginning of file */
```

To move to the end of a file, the seek direction would be SEEK_END:

```
fseek(fp, 0L, SEEK_END);      /* moves to end of file */
```

To move back 10 bytes, the seek direction would be SEEK_CUR and the byte count would be −10:

```
fseek(fp, -10L, SEEK_CUR);    /* moves back 10 bytes */
```

Note that the byte count has type long int, so I've used 0L and -10L as arguments. (0 and -10 would also work, of course, since arguments are converted to the proper type automatically.)

Normally, fseek returns zero. If an error occurs (the requested position doesn't exist, for example), fseek returns a nonzero value.

The file-positioning functions are best used with binary streams, by the way. C doesn't prohibit programs from using them with text streams, but care is required because of operating system differences. fseek in particular is sensitive to whether a stream is text or binary. For text streams, either (1) offset (fseek's second argument) must be zero or (2) whence (its third argument) must be SEEK_SET and offset a value obtained by a previous call of ftell. (In other words, we can only use fseek to move to the beginning or end of a text stream or to return to a place that was visited previously.) For binary streams, fseek isn't required to support calls in which whence is SEEK_END.

ftell

errno variable ➤24.2

The ftell function returns the current file position as a long integer. (If an error occurs, ftell returns -1L and stores an error code in errno.) The value returned by ftell may be saved and later supplied to a call of fseek, making it possible to return to a previous file position:

```
long file_pos;
...
file_pos = ftell(fp);            /* saves current position */
...
fseek(fp, file_pos, SEEK_SET);  /* returns to old position */
```

If fp is a binary stream, the call ftell(fp) returns the current file position as a byte count, where zero represents the beginning of the file. If fp is a text stream, however, ftell(fp) isn't necessarily a byte count. As a result, it's best not to perform arithmetic on values returned by ftell. For example, it's not a good