



Red Hat Enterprise Linux Release 9.2 Manual Pages on 'fwrite.3p' command

\$ man fwrite.3p

FWRITE(3P) POSIX Programmer's Manual FWRITE(3P)

PROLOG

This manual page is part of the POSIX Programmer's Manual. The Linux implementation of this interface may differ (consult the corresponding Linux manual page for details of Linux behavior), or the interface may not be implemented on Linux.

NAME

fwrite ? binary output

SYNOPSIS

```
#include <stdio.h>

size_t fwrite(const void *restrict ptr, size_t size, size_t nitems,
              FILE *restrict stream);
```

DESCRIPTION

The functionality described on this reference page is aligned with the ISO C standard. Any conflict between the requirements described here and the ISO C standard is unintentional. This volume of POSIX.1?2017 defers to the ISO C standard.

The `fwrite()` function shall write, from the array pointed to by `ptr`, up to `nitems` elements whose size is specified by `size`, to the stream pointed to by `stream`. For each object, `size` calls shall be made to the `fputc()` function, taking the values (in order) from an array of `un?` signed `char` exactly overlaying the object. The file-position indicator for the stream (if defined) shall be advanced by the number of bytes

successfully written. If an error occurs, the resulting value of the file-position indicator for the stream is unspecified.

The last data modification and last file status change timestamps of the file shall be marked for update between the successful execution of `fwrite()` and the next successful completion of a call to `fflush()` or `fclose()` on the same stream, or a call to `exit()` or `abort()`.

RETURN VALUE

The `fwrite()` function shall return the number of elements successfully written, which may be less than `nitems` if a write error is encountered.

If `size` or `nitems` is 0, `fwrite()` shall return 0 and the state of the stream remains unchanged. Otherwise, if a write error occurs, the error indicator for the stream shall be set, and `errno` shall be set to indicate the error.

ERRORS

Refer to `fputc()`.

The following sections are informative.

EXAMPLES

None.

APPLICATION USAGE

Because of possible differences in element length and byte ordering, files written using `fwrite()` are application-dependent, and possibly cannot be read using `fread()` by a different application or by the same application on a different processor.

RATIONALE

None.

FUTURE DIRECTIONS

None.

SEE ALSO

Section 2.5, Standard I/O Streams, `ferror()`, `fopen()`, `fprintf()`, `putc()`, `puts()`, `write()`

The Base Definitions volume of POSIX.1?2017, `<stdio.h>`

COPYRIGHT

Portions of this text are reprinted and reproduced in electronic form

from IEEE Std 1003.1-2017, Standard for Information Technology -- Portable Operating System Interface (POSIX), The Open Group Base Specifications Issue 7, 2018 Edition, Copyright (C) 2018 by the Institute of Electrical and Electronics Engineers, Inc and The Open Group. In the event of any discrepancy between this version and the original IEEE and The Open Group Standard, the original IEEE and The Open Group Standard is the referee document. The original Standard can be obtained online at <http://www.opengroup.org/unix/online.html> .

Any typographical or formatting errors that appear in this page are most likely to have been introduced during the conversion of the source files to man page format. To report such errors, see https://www.kernel.org/doc/man-pages/reporting_bugs.html .

IEEE/The Open Group

2017

FWRITE(3P)