

NAME

stdio — standard buffered input/output package

SYNOPSIS

```
#include <stdio.h>
FILE *stdin, *stdout, *stderr;
```

DESCRIPTION

The functions described in the entries of sub-class 3S of this manual constitute an efficient, user-level I/O buffering scheme. The in-line macros *getc*(3S) and *putc*(3S) handle characters quickly. The macros *getchar*, *putchar*, and the higher-level routines *fgetc*, *fgets*, *fprintf*, *fputc*, *fputs*, *fread*, *fscanf*, *fwrite*, *gets*, *getw*, *printf*, *puts*, *putw*, and *scanf* all use *getc* and *putc*; they can be freely intermixed.

A file with associated buffering is called a *stream* and is declared to be a pointer to a defined type FILE. *Fopen*(3S) creates certain descriptive data for a stream and returns a pointer to designate the stream in all further transactions. Normally, there are 3 open streams with constant pointers declared in the “include” file and associated with the standard open files:

stdin	standard input file
stdout	standard output file
stderr	standard error file.

A constant “pointer” NULL (0) designates the null stream.

An integer constant EOF (−1) is returned upon end-of-file or error by most integer functions that deal with streams (see the individual descriptions for details).

Any program that uses this package must include the header file of pertinent macro definitions, as follows:

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

The functions and constants mentioned in the entries of sub-class 3S of this manual are declared in that “include” file and need no further declaration. The constants and the following “functions” are implemented as macros (redeclaration of these names is perilous): *getc*, *getchar*, *putc*, *putchar*, *feof*, *ferror*, and *fileno*.

SEE ALSO

open(2), *close*(2), *read*(2), *write*(2), *ctermid*(3S), *cuserid*(3S), *fclose*(3S), *ferror*(3S), *fopen*(3S), *fread*(3S), *fseek*(3S), *getc*(3S), *gets*(3S), *popen*(3S), *printf*(3S), *putc*(3S), *puts*(3S), *scanf*(3S), *setbuf*(3S), *system*(3S), *tmpnam*(3S).

DIAGNOSTICS

Invalid *stream* pointers will usually cause grave disorder, possibly including program termination. Individual function descriptions describe the possible error conditions.