Full credit is given to the above companies including the OS that this PDF file was generated!

Red Hat Enterprise Linux Release 9.2 Manual Pages on 'glob.h.0p' command

$ man glob.h.0p

glob.h(0P) POSIX Programmer's Manual glob.h(0P)

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

glob.h ? pathname pattern-matching types

SYNOPSIS

#include <glob.h>

DESCRIPTION

The <glob.h> header shall define the structures and symbolic constants used by the glob() function.

The <glob.h> header shall define the glob_t structure type, which shall include at least the following members:

size_t gl_pathc Count of paths matched by pattern.

char **gl_pathv Pointer to a list of matched pathnames.

size_t gl_offs Slots to reserve at the beginning of gl_pathv.

The <glob.h> header shall define the size_t type as described in <sys/types.h>.

The <glob.h> header shall define the following symbolic constants as values for the flags argument:

GLOB_APPEND Append generated pathnames to those previously obtained.

GLOB_DOOFFS Specify how many null pointers to add to the beginning of gl_pathv.

GLOB_ERR Cause glob() to return on error.

GLOB_MARK Each pathname that is a directory that matches pattern has a <slash> appended.

GLOB_NOCHECK If pattern does not match any pathname, then return a list consisting of only pattern.

GLOB_NOESCAPE Disable backslash escaping.

GLOB_NOSORT Do not sort the pathnames returned.

The <glob.h> header shall define the following symbolic constants as error return values:

GLOB_ABORTED The scan was stopped because GLOB_ERR was set or (*er? rfunc)() returned non-zero.

GLOB_NOMATCH The pattern does not match any existing pathname, and GLOB_NOCHECK was not set in flags.

GLOB_NOSPACE An attempt to allocate memory failed.

The following shall be declared as functions and may also be defined as macros. Function prototypes shall be provided.

```
int  glob(const char *restrict, int, int(*)(const char *, int),
        glob_t *restrict);
void globfree(glob_t *);
```

The following sections are informative.

APPLICATION USAGE

None.

RATIONALE

None.

FUTURE DIRECTIONS

None.

SEE ALSO

<sys_types.h>

The System Interfaces volume of POSIX.1?2017, glob()

COPYRIGHT