

目录系统函数

pendir(3)

功能：opendir 函数的主要作用是打开指定的目录。在操作系统中，目录也是一种特殊的文件，它记录了该目录下所有文件和子目录的相关信息。通过 opendir 函数，程序可以获取一个指向该目录的“目录流”指针，后续就可以基于这个指针来访问目录中的内容。

原型：DIR *opendir(const char *name);, 其中 name 是要打开的目录的路径名。如果打开成功，函数返回一个指向 DIR 类型的指针，这个指针就代表了打开的目录流；如果打开失败，返回 NULL，并且会设置相应的错误码，可以通过 perror 函数输出错误信息。

readdir(3)

功能：readdir 函数用于从已经打开的目录流中读取下一个目录项。每次调用该函数，它会返回一个指向 struct dirent 结构体的指针，该结构体包含了当前读取到的目录项的相关信息，如文件名等。

原型：struct dirent *readdir(DIR *dirp);, 其中 dirp 是通过 opendir 函数返回的目录流指针。当读取到目录末尾或者发生错误时，函数返回 NULL。

closedir(3)

功能：closedir 函数用于关闭已经打开的目录流。当程序不再需要访问某个目录时，应该调用该函数释放相关的资源，避免资源泄漏。

原型：int closedir(DIR *dirp);, 其中 dirp 是要关闭的目录流指针。如果关闭成功，函数返回 0；如果失败，返回 -1，并设置相应的错误码。

glob(3)

功能：glob 函数主要用于解析匹配的路径名。它可以根据用户提供的通配符模式，在文件系统中查找匹配的文件和目录，并将匹配结果存储在一个 glob_t 结构体中。这在需要批量处理符合特定模式的文件时非常有用，例如查找所有以 .c 结尾的文件。

原型：int glob(const char *pattern, int flags, int (*errfunc) (const char *epath, int errno), glob_t *pglob);

- pattern: 要匹配的路径名模式，可以包含通配符，如 *（匹配任意数量的任意字符）、?（匹配单个任意字符）等。
- flags: 控制 glob 函数行为的标志位，例如 GLOB_NOSORT 表示不排序匹配结果。
- errfunc: 一个错误处理函数指针，当在匹配过程中出现错误时会调用该函数。
- pglob: 指向 glob_t 结构体的指针，用于存储匹配结果。