

[z/OS](#) / [2.3.0](#) / [Change version](#) [Feedback](#) [Product list](#)

readdir() – Read an entry from a directory

Last Updated: 2021-03-03

>

Standards

Standards / Extensions	C or C++	Dependencies
POSIX.1	both	
XPG4		
XPG4.2		
Single UNIX Specification, Version 3		

Format

```
#define_POSIX_SOURCE
#include <dirent.h>

struct dirent *readdir(DIR *dir);
```

General description

Returns a pointer to a `dirent` structure describing the next directory entry in the directory stream associated with *dir*.

A call to `readdir()` overwrites data produced by a previous call to `readdir()` or `__readdir2()` on the same directory stream. Calls for different directory streams do not overwrite each other's data.

>

Each call to `readdir()` updates the `st_atime` (access time) field for the directory.

A `dirent` structure contains the character pointer *d_name*, which points to a string that gives the name of a file in the directory. This string ends in a terminating NULL, and has a maximum of `NAME_MAX` characters.

Save the data from `readdir()`, if required, before calling `closedir()`, because `closedir()` frees the data.

If the contents of a directory have changed since the directory was opened (files added or removed); a call should be made to `rewinddir()` so that subsequent `readdir()` requests can read the new contents.

Special behavior for XPG4: If entries for dot or dot-dot exist, one entry will be returned for dot and one entry will be returned for dot-dot; otherwise they will not be returned.

After a call to `fork()`, either the parent or child (but not both) may continue processing the directory stream using `__readdir2()`, `readdir()`, `rewinddir()`, or `seekdir()`. If both the parent and child processes use these functions, the result is undefined.

Special behavior for XPG4.2: If the entry names a symbolic link, the value of `d_ino` member in *dirent* structure is unspecified.

Returned value

If successful, `readdir()` returns a pointer to a `dirent` structure describing the next directory entry in the directory stream. When `readdir()` reaches the end of the directory stream, it returns a NULL pointer.

If unsuccessful, `readdir()` returns a NULL pointer and sets `errno` to one of the following values:

Error Code

Description

EBADF

dir does not yield an open directory stream.

EINVAL

The buffer was too small to contain any directories.

ENOENT

The current position of the directory stream is invalid.

EOVERFLOW

One of the values in the structure to be returned cannot be represented correctly.

Note: The environment variable `_EDC_SUSV3` can be used to control the behavior of `readdir()` with respect to detecting an `EOVERFLOW` condition. By default, `readdir()` will not detect that values in the structure returned can be represented correctly. When `_EDC_SUSV3` is set to 1, `readdir()` will check for overflow conditions.

Example

CELEBR04

```
/* CELEBR04
```

This example reads the contents of a root directory.

```
*/
#define _POSIX_SOURCE
#include <dirent.h>
#include <errno.h>
#include <sys/types.h>
#undef _POSIX_SOURCE
#include <stdio.h>

main() {
    DIR *dir;
    struct dirent *entry;

    if ((dir = opendir("/")) == NULL)
        perror("opendir() error");
    else {
        puts("contents of root:");
        while ((entry = readdir(dir)) != NULL)
            printf("  %s\n", entry->d_name);
        closedir(dir);
    }
}
```

Output

contents of root:

```
.
..
bin
dev
etc
lib
tmp
u
usr
```

Related information

- [dirent.h](#) – POSIX directory access
- [stdio.h](#) – Standard input and output
- [sys/types.h](#) – typedef symbols and structures
- [closedir\(\)](#) – Close a directory
- [opendir\(\)](#) – Open a directory
- [__opendir2\(\)](#) – Open a directory
- [readdir_r\(\)](#) – Read an entry from a directory
- [__readdir2\(\)](#), [__readdir2_64\(\)](#) – Read directory entry and get file information
- [rewinddir\(\)](#) – Reposition a directory stream to the beginning
- [seekdir\(\)](#) – Set position of directory stream
- [telldir\(\)](#) – Current location of directory stream

Parent topic:

→ [Library functions](#)

Previous

[read\(\)](#) – Read from a file or socket

Next

[__readdir2\(\)](#), [__readdir2_64\(\)](#) – Read directory entry and get file information