

Lab 3

9 Apr 2009

In this lab, you'll implement `ls`.

1 The basics

To implement a bare-bones `ls`, you only need to know about two of the directory functions: `opendir` and `readdir`.

There are two new types you'll see here. One is `struct dirent`, which contains info for one "directory entry". Details of its format are on the `dirent` manpage. The actual file that represents a directory just contains a bunch of these structs, and you could in fact read them using the same `open` and `read` you've used before; but then you'd have to keep track of the size of the struct, and various things could go wrong, so someone's written `opendir` and `readdir` to manage this for you. These functions use the second new type, `DIR`, a "directory stream", which is just a wrapper for an open file descriptor. It is pointers to this `DIR` type that you'll get from `opendir` and pass to `readdir`.

Implement a version of `ls` that just lists the names of all the files in a given directory. If no directory name is given, use the current working directory (which you can get by opening `."`).

2 Adding an option

The real `ls` can accept command-line options. Yours will accept some, too, although a restricted list and in a limited format. To start with, implement a single option: `-a`. When this option is *not* present as the first argument to the program, filenames that begin with a period are omitted from the listing. When it is present, the full listing is given as before.

3 Accessing file metadata

To find out much more than the file's name, you generally have to look beyond the `dirent` info. (On some filesystems, the `dirent` does at least say if the file is a regular file or a directory, but not all of them.) The usual mechanism for this is the `stat` system call. To call `stat`, provide it with a filename *including absolute or relative path* and a pointer to a `struct stat` that you've already allocated, and it'll put the info about that file in that struct.

Add a second option: if the user selects the `p` option, then all filenames that correspond to subdirectories will have a slash appended to them. (Type `'ls -p'` or `'ls -ap'` to see how this works in the real `ls`.)

The format for accepting multiple options will be limited: when present, the options must come all together, as the first argument, preceded by a single hyphen. (The real `ls` will let you say `"ls -a -p"` but yours would require `"ls -ap"` or `"ls -pa"`.)

(Helpful note for processing multiple options: the `strchr` function finds the first occurrence of a given letter in a given string, and returns `NULL` if that letter is not found.)

4 Hand in

At the end of the period, or once you finish the previous section, hand in what you've done as `lab3`.