

ECE 329 HW #2

In this assignment you will write the basics of a simple file system. The file system will be flat (all files in a single directory), and files will not (yet) contain any data. Your entire simulated file system will be stored in a single file on the hard disk. Augment your UNIX shell with the following commands:

- *mkfs rawdevicefile*
creates a file system called *rawdevicefile*. In UNIX, the command formats an existing device, but your command will simply create a file on the hard disk with the name *rawdevicefile*. In UNIX, the raw device would normally be named something like */dev/dsk0*, so you should use a name like *devdsk0*.
- *mount rawdevicefile dir*
mounts the raw device *rawdevicefile* to the directory *dir*. Since you will not be implementing a hierarchical tree structure, the only value allowed for *dir* is *.* (the current directory). When a device is mounted, any other device that happens to already be mounted is hidden. If the option *-p* is given (with no other arguments), then the name of the current mounted device (i.e., the name of *rawdevicefile*) is printed.
- *ls*
lists the files in the directory (single column is okay). If *-l* is given as an argument, then files are listed in long format (permissions, owner, size, last modified date, file name). For this assignment, the size of all files will be zero, and permissions will include two categories of people (“user” and “other”) and three types of access (“read”, “write”, and “execute”).
- *touch file*
creates an empty file. For simplicity, file names can be limited to 14 characters (as in UNIX System V). If a file with the name already exists, this command should do nothing. The default permissions should be *rw----* (i.e., read-write for “user”, nothing for “other”).
- *rm file*
removes a file. Must have permission to do this (i.e., must be *root*, or must be owner of file and *uw* permission is granted, or must be anyone and *ow* permission is granted).
- *chown file user*
changes the owner of a file. Must be *root* to do this.
- *chmod [uo][+-][rwx] file*
changes permissions of a file. Must have permission to do this (same as *rm*).

As indicated above, the commands to modify a file should first check the read and write permissions of the file to ensure that the current user has the appropriate access. If not, an error message should be printed. The file system should always be up to date, so that a crash of the O/S (e.g., pressing Ctrl+C) does not lose any changes made.

Separately, answer the following problems in Chapter 2 of the Tanenbaum textbook: 1, 3, 4, 6, 7, 10, 12, 16.