

Project 4: Part 2

Friday, April 15, 2016

12:04 PM

- Mknod
 - Used to make a new file
 - Testing:
 - Need to use: `echo " " > newfile.ext`
 - Step 1:
 - `Scanff-->dir, file, ext`
 - Step 2: check for errors
 - `ENAMETOOLONG`: when `strlen(filename) > 8 || strlen(ext) > 3`
 - `EPERM`: if `strlen(filename) = 0`
 - `EEXIST`: root --> subdir --> if the file exists
 - Step 3:
 - Create new file
 - Update fields of subdirectory
 - Find a free block for the inode
 - Find a free block for the first data block; store it in list of pointers
- Write
 - Testing: Need to use: `echo "string" > f1.ext`
 - `Echo "string" >> f1.ext (>> will append)`
 - Step 1:
 - `Scanff-->dir, file, ext`
 - Step 2: check for errors
 - `-EFBIG`: `offset > file size`
 - Step 3: write: `block to start = offset/BLOCK SIZE + starting block of file`
 - `Position in block = offset % BLOCK SIZE`
 - `Fwrite and shit, transfer data into data field of cs150_disk_block`
 - Increase `fsize` in `file_directory`
- Read
 - Testing: `cat filename.ext`
 - Step 1:
 - `Scanff-->dir, file, ext`
 - Step 2: check for errors

- Step 2: check for errors
 - -EISDIR: strlen(filename) == 0 (trying to read for directory, which isn't possible)
- Step 3:
 - Need to make sure that offset < filesize
- Step 4:
 - Read from offset (which is generally 0) to end of file
- Unlink
 - Test: rm filename.ext
 - Step 1:
 - Scanff --> dir, file, ext
 - Step 2: check for errors
 - -EISDIR: strlen(filename) == 0
 - -ENOENT: !fileExists()
 - Step 3:
 - Removing:
 - Go through list of files inside the subdirectory. If filename matches:
 - ◆ Bit map:
 - ◇ inode block:
 - ▶ List of pointers: remove pointer
 - ◆ Files[pos] = files[pos + 1] (shift to the left); or copy last element to file[pos] and delete last element