Chapter 2 Lecture (first half, page 1-10)

Purpose: This assignment covers the file operations.

A) Programming:

Get a copy of the file `/net/326/copy.c`.

This program in this file takes one argument. That arguement should be the name of a (text) file. The program copies that file to the screen.

Modify the program so it takes two arguements. The first will be the name of the source file, the second will be the name of the destination file. Further modify the program so it opens (creates) the destination file and copies the source file into it. Don't forget to close the destination.

Finally, add some protection to the program. If the source file doesn't exist print an error message to the screen and exit. If the destination file already exists (open with the correct flags) print an error message to the screen and exit.

Hint: You can tell there was an error on open if it returns an integer less than 0.

Running the resulting command should look something like `"copy.out file1 file2"` with the resulting file2 being identical to file1.

Optional: The `"diff"` command shows differences between two files. If there is no output from the command, the files are identical.

Run something like `"diff file1 file2"` to verify that your copied file is identical to the original.

B) Exploring the system:

On the Linux server `cd` to `/etc`. Then list the contents of that directory with an `ls -al`. `ls` shows hidden files with the `-a` argument and gives details with `-l`. Notice the modification dates on the files.

Question: when was `rc.local` last modified?

Just in front of the date is the number of bytes in the file. The number of bytes in `rc.local` is accurate (pretty small). Directories are padded out to the nearest multiple of 4096, so the sizes given for directories is not terribly helpful.

Question: how many bytes are there in the file `rc.local`?

The third column of the output shows the owner of the file. The fourth shows the group. This is in front of the number of bytes.

Question: who owns `rc.local`?

Demo: Your copy program and show the Instructor your code. Leave the code on the server.

Submit: Your answers to the questions in part B. The answers for part B may be handwritten or printed. You may wish to keep notes on the server so you have study material.