**Unix and C Programming 02**

We reviewed file permissions and checked access to personal web pages on Sciris. The rights to your home directory and to the public\_html directory should look like this:

* drwxr-xr-x 21 wachsmut faculty 4096 Jan 24 16:59 public\_html

i.e. it is a directory (d) with rwxrights to user, rx rights to group, and rx to everyone. Your home directory should have the same rights, incidentally. We then talked about:

* cp, mv, and rm (copy, move (aka rename), and remove)

All of which might do damage by overwriting existing files, so handle with care (or se the –I flag). We also discussed the commands

* cat concatenate files and print on the standard output
* more displays a file one screen at a time; includes search option
* less similar to more but with more options
* tail displays the last few lines of a file

Finally we briefly touched up the command

* grep to search for a pattern in a given input

Then we discussed job control by starting a long Maple computation and interrupting it mid-execution.

* ps lists processes
* jobs lists running jobs
* CTRL-Z stops a running program
* fg, bg moves programs into the foreground (fg) or background (bg)
* & starts a process in the background from the get-go
* kill terminates running processes

Finally we discussed redirection and pipes. Redirection can redirect output to a file, or input from a file. Piping chains the output of one program to the input of a second program.

* > output redirection (variations: >> redirects and append if file exists, 2> redirect only error messages)
* < input redirection (from a file to a program)
* | the pipe command

**Example**: Suppose I want to start a lengthy Maple program that might take hours to complete. The strategy will be:

1. Create the program file as a plain text file. Say our program is to factor a large integers. We use pico to create a text file named cool.maple containing the single line

ifactor(2^450 - 1);

1. Run maple, with input redirected from the previously created program file, output going to some output file, and run everything in the background, as follows:

maple < cool.maple > output.txt &

Our host Sciris will now compute the prime factorization of 2^450 – 1, which will take about a minute. The calculation will continue *even if I logout*, and the answer, when it is finally done, will be saved to the output file output.txt so I can look at it any time (with commands like more or tail).

**Homework**

1. Login in to your home directory. List the stuff *above* your home directory, i.e. list everything in /home
	1. Pipe the output through more to show it one page at a time if necessary
	2. Are there any *files* in /home or only *directories*?
	3. What are the rights of Dr. Morazan’s (morazama) home directory, and when has the directory last changed?
	4. Use ls, a pipe, and grep to display the rights to your own directory in such a way that the directory and its rights are listed in a single line, i.e. nobody else’s directory will show up
2. I have started a process called “secret\_prg” on Sciris. Find its process ID number using the ps command together with appropriate switches (for example –e for *every process* and –l for *long forma*t), a pipe, and the grep command to get the appropriate process ID. Then try to kill that process. Explain.
3. Create a program on Sciris to determine if 2127-1 is a prime number. How about 2521-1? You might want to use redirection and background processing if necessary. Note that these numbers of the form 2^N-1 are called Mersenne numbers and are actively researched in mathematics.