1concurrency-and-threads-lab

CST 334 (Operating Systems)
Dr. Glenn Bruns

Lab: Intro to pthreads

The purpose of this lab is to give you exposure to the low-level interleaving of programs.

1. Login to mlc104 and copy this file to a directory of your own:

   `/home/CLASSES/brunsglenn/cst334/labs/concurrency/t0.c`

2. Read the file and understand what it does. Write down your answers to these questions:
   - Is a pthreads library include file used? What is it's name?
   - Which pthreads library functions are called in this program?
   - How many threads are created, and what are their names?
3. Compile the program like this:

   ```
   $ gcc -o t0 t0.c -pthread
   ```

4. Run the program. What do you get as output? Run the program a few more times. Do you get the same output?

5. Create a Makefile that will compile t0. Your Makefile should include a target `t0`, and that target should depend on `t0.c`.

6. Here's an example bash 'for loop', to remind you of what they look like:

   ```
   $ for i in {1..10}; do echo $i; done
   ```

   Run this code yourself at the command line.

7. Write a loop to run program `t0` 100 times and write the output to file temp.txt. Do you see examples where A runs before B, and B runs before A?

8. Write some awk code to count how many times A was printed before B. You can modify t0.c to make your life easier.

9. If you have time, add a third process, and print 'C' when it executes.

10. If you still have time, try repeating the lab but on your own machine, or on cloud9.

Hints:

5. This Makefile will work:

```
t0:   t0.c
    gcc -o t0 t0.c -pthread
```