COP 4600

Exercise - Race Conditions

Fall 2016

Task:

* write two programs,
* perform experiments,
* report results.

# Part 0:

We will provide an executable that

* opens the file given on the command line, then
* reads the last number in that file,
* increments it, and
* appends the new number to that file.

It will do this the number of times given on the command line in the first argument.

Example:

$ increment 100 foo.txt

* will open and read the file foo.txt,
* search for the last line (verifying that it has an integer N on it),
* then add a line with N+1 on it.

It will then close the file, reopen the file, and read the file again, again find the last line, read it, increment the number that it finds there, and append that to the file. It will do this 100 times.

# Part I:

Write a program *"consecutive"* that

* reads a file of numbers, one per line, then
* outputs two adjacent lines for which the number in the second line is not exactly one greater than the number in the first line.

It shall output the line numbers of each line before that line in a field at least five characters wide with leading spaces, followed by a colon and a space before printing the contents of the line. It shall read from standard in and write to standard out.

Example:

$ cat foo.txt

1

2

3

4

5

5

16

17

$ consecutive < foo.txt

5: 5

6: 5

6: 5

7: 16

Run increment by itself and consecutive on its output to convince yourself they work.

# Part II:

Write a program that forks the number of children specified on the command line.

* Each child executes the program specified on the command line, and
* the parent waits for all the children to terminate, then terminates itself.

The program is named *“spawn”.*

Example:

$ spawn 1 "increment 10 foo.txt"

will create *one* child, that executes the increment program with arguments **10** and **foo.txt**.

$ spawn 2 "increment 10 foo.txt"

will create *two* children, each child will execute the increment program with arguments **10** and **foo.txt**.

# Part III:

Run *spawn* with increment on a file that has as single line with the number 1 on it, with various number of iterations, then check that file using *consecutive*. How long does it take for a race condition to appear?

What is the critical section?

# Submit:

* Your **source code** and **makefile** for *consecutive* and *spawn*, and
* a short report called report.txt with your findings on the experiments you ran.
* Tar the files into a single tarfile for submission.