CST 334: Operating Systems

Dr. Glenn Bruns

# Awk code to verify a readers-writers lock

**Instructions**. Write an awk script check-rwlock.awk to check the output of your rwlock.c code. The output of rwlock.c is a sequence of lines, each of which is either ‘sr’, ‘dr’, ‘sw’, or ‘dw’. Your awk script should take as input the output of your rwlock code, and produce as output the following:

* the maximum number of concurrently reading threads
* the maximum number of concurrently writing threads
* whether a read operation ever overlapped with a write operation
* whether a write operation ever overlapped with a write operation

Here is some example output from the unedited rwlock-skeleton code:

$ ./rwlock-skeleton | awk -f check-rwlock.awk

maxr = 1, maxw = 2

reader/writer test: FAIL

writer/writer test: FAIL

This rwlock-skeleton code is bad: both the reader/writer and writer/writer tests failed. For example, at some point two writers were writing concurrently. You can see this from the 'writer/writer test' output, and from the 'maxw = 2' output.

Note: you will not always get the same output! Sometimes rwlock-skeleton might pass the tests! You will need to produce lots of output, or run the tests multiple times, to get confidence that the tests really pass.

If your code is correct you will get this output:

$ ./rwlock > temp.txt; awk -f check-rwlock.awk temp.txt

maxr = 3, maxw = 1

reader/writer test: PASS

writer/writer test: PASS

This shows that at most one writer could write at any time. The basic idea on writing the awk script is to keep counts of the number of current readers and the number of current writers. For example, the number of current writers will increase on a ‘sw’ line and decrease on a ‘dw’ line.

You won’t always get output with maxr = 3, even if your code is correct. That will only happen if at some point the three readers were doing overlapping reads. In the example above I didn't pipe the output of rwlock to awk because when I do that I tend to get maxr = 1. You can change constant NUMOPS in rwlock.c to run more reads and writes.

**Testing your code**. In the directory containing rwlock-skeleton.c I also included four test scripts: test1.sh, …, test4.sh. These scripts can be used to test your check-rwlock.awk script. Please look at the scripts to help understand how your awk code should work.

**Submission**: Submit your check-rwlock.awk on iLearn.

**Grading**. I will use 4 tests to grade your code. Some of the tests may be a little different from the ones you were provided. Each test is worth 5 points.