

Problem 2 Output

Spinlock test:

This test increments a shared counter in 8 processes with and without the spinlock.

Without the lock the final value is lower than expected; with the lock it matches.

```
alexi@AlexsTraptop:/mnt/c/Users/alexi/Documents/Cooper/CompOS/ECE357/PS6/Problem2$ ./spin_test
=== WITHOUT spinlock ===
Expected: 4000000
Observed: 3676063

=== WITH spinlock ===
Expected: 4000000
Observed: 4000000

alexi@AlexsTraptop:/mnt/c/Users/alexi/Documents/Cooper/CompOS/ECE357/PS6/Problem2$
```

FIFO Test (Successful) :

```
alexi@AlexsTraptop:/mnt/c/Users/alexi/Documents/Cooper/CompOS/ECE357/PS6/Problem2$ ./ftest
Beginning acid test with 8 writers, 64000 items each
Writer 5 completed
Writer 1 completed
Reader stream 5 completed
Writer 0 completed
Reader stream 1 completed
Reader stream 0 completed
Writer 2 completed
Reader stream 2 completed
Writer 4 completed
Writer 7 completed
Writer 3 completed
Reader stream 4 completed
Writer 6 completed
Reader stream 7 completed
Reader stream 3 completed
Reader stream 6 completed
Reader: all 512000 items received correctly.
All writer and reader processes completed.
alexi@AlexsTraptop:/mnt/c/Users/alexi/Documents/Cooper/CompOS/ECE357/PS6/Problem2$
```

FIFO Test (Failed):

To demonstrate failure, I temporarily removed the mutex from `fifo_wr` so that writers no longer protect updates to `write_pos` and `buf`. With this change multiple writers can write simultaneously, which corrupts the order of items in the FIFO.

```
alexi@AlexsTraptop: /mnt/c/Users/alexi/Documents/Cooper/CompOS/ECE357/PS6/Problem2$ ./ftest_broken
Beginning acid test with 8 writers, 64000 items each
ERROR: out of sequence from 0
^C
alexi@AlexsTraptop: /mnt/c/Users/alexi/Documents/Cooper/CompOS/ECE357/PS6/Problem2$
```

Code alterations to “break” synchronization:

```
void fifo_wr(struct myfifo *f, unsigned long d)
{
    sem_wait(&f->empty);
    //sem_wait(&f->mutex);

    f->buf[f->write_pos] = d;
    f->write_pos = (f->write_pos + 1) % MYFIFO_BUFSIZ;

    //sem_inc(&f->mutex);
    sem_inc(&f->full);
}
```