# CS 511 – Quiz 8: Model-Checking/Spin

29 November 2023

Names:

## Exercise 1

Cosider the Groovy solution to the cyclic barrier below (barrier of size 2 and 2 threads in the system).

```groovy
import java.util.concurrent.Semaphore

Semaphore mutexE = new Semaphore(1)
Semaphore mutexL = new Semaphore(1)
Semaphore barrier = new Semaphore(0)
Semaphore barrier2 = new Semaphore(0)
final int N = 2
final int B = 2
int enter=0
int leaving=0

N.times {
    int id = it
    Thread.start {
       100.times {
           mutexE.acquire()
           enter++
           if (enter==B) {
               barrier.release(B)
               enter=0
           }
           mutexE.release()

           barrier.acquire()

           mutexL.acquire()
           leaving++
           if (leaving==B) {
               barrier2.release(B)
               leaving=0
           }
           mutexL.release()
           barrier2.acquire()
       }
   }
}
```

You are asked to complete two tasks. The first is to model it in Promela. For that you should use the stub below. You will need to declare extra variables. The second is to prove it is correct using assertions. Correct here means that no one thread gets “ahead” of the others. Hint: use an array (already declared for you) and check that no one thread gets ahead of any other. This may require two nested for-loops.

```
#define N 2  // 2 (resp. 3) - requires setting max_depth to 12000 (resp. 22000)
#define B 2

byte mutexE = 1;
byte mutexL = 1;
byte barrier = 0;
byte barrier2 = 0;

byte c[N];  // array for counting cycles
byte enter=0;
byte leaving=0;

inline acquire(s) {
  skip;
end1:atomic {
    s>0;
    s--
  }
}

inline release(s) {
  s++
}

inline absolute(inp,outp) { // absolute value of inp placed in outp
  if
    :: inp>0 -> outp = inp
    :: else -> outp = -inp
  fi
}

active[N] proctype P() {
  byte i;
  byte j;

  for (i: 1..100 ) {
    // complete


    printf("%d reached at cycle %d\n",_pid, c[_pid]);
    atomic {
      // assertion here

    };

    printf("%d leaves at cycle %d\n",_pid, c[_pid]);

    // complete

  }
}
```

## Submission instructions:

Submit a one file named `cb.pml`.