EE382C: Multicore Computing
Assignment 2

Instructor: Professor Vijay Garg (email: garg@ece.utexas.edu)
TA: Changyong Hu (email: colinhu9@gmail.com)

Deadline: October 7, 2016

The source code must be uploaded through Canvas before the end of the due date (i.e., 11:59pm on October 7). The assignment should be done in teams of two. You should use the templates downloaded from the course github. You should not change the file names and function signatures. In addition, you should not use package for encapsulation. Note that, you do not need to include a main function for submission. Please zip and name the source code as [EID1_EID2].zip.

1. **(50 points)** Give an implementation of Java class java.util.concurrent.CyclicBarrier using semaphores. A CyclicBarrier is a synchronization aid that allows a set of threads to all wait for each other to reach a common barrier point. CyclicBarriers are useful in programs involving a fixed sized party of threads that must occasionally wait for each other. The barrier is called cyclic because it can be re-used after the waiting threads are released. You need to implement the following methods:

```
public CyclicBarrier(int parties) {
  // Creates a new CyclicBarrier that will trip when
  // the given number of parties (threads) are waiting upon it

  int await() throws InterruptedException {
    // Waits until all parties have invoked await on this barrier.
    // If the current thread is not the last to arrive then it is
    // disabled for thread scheduling purposes and lies dormant until
    // the last thread arrives.
    // Returns: the arrival index of the current thread, where index
    // (parties - 1) indicates the first to arrive and zero indicates
    // the last to arrive.
  }
}
```

2. **(50 points)** In the *shared bathroom problem*, there are two classes of threads, called *male* and *female*. There is a single *bathroom* resource that must be used in the following way:

   (a) Mutual exclusion: persons of opposite sex may not occupy the bathroom simultaneously.

   (b) Starvation-freedom: everyone who needs to use the bathroom eventually enters.

   The protocol is implemented via the following four procedures: `enterMale()` delays the caller until it is ok for male to enter the bathroom, `leaveMale()` is called when a male leaves the bathroom, while `enterFemale()` and `leaveFemale()` do the same for females. For example,

```
enterMale();
teeth.brush(toothpaste);
leaveMale();
```

(a) Implement this class using *locks* and *condition* variables.

(b) Implement this class using *synchronized*, *wait()*, *notify()*, and *notifyAll()*.