

Homework 5

Due on Thursday, April 23, 2020 before 6:00 pm

Name:	QUID:
--------------	--------------

A barrier is a thread-synchronization mechanism that forces the threads to wait until all have reached a certain point (the barrier). Once all threads have reached this point, they may all continue. An interface for a barrier appears as follows:

```
public interface Barrier{
    /**Each thread calls this method when it reaches the barrier. All threads are released to continue
    processing when the last thread calls this method.*/
    public void waitForOthers();
    /** Release all threads from waiting for the barrier. Any future calls to waitForOthers() will not wait
    until the Barrier is set again with a call to the constructor.*/
    public void freeAll();
}
```

The following code segment of the **Factory** class establishes a barrier and creates 10 **Worker** threads that will synchronize according to the barrier:

```
final int THREADCOUNT = 10;
Barrier barrier = new BarrierImpl(THREAD COUNT);
for (int i = 0; i < THREAD COUNT; i++)
    (new Worker(barrier)).start();
```

Note that the barrier must be initialized to the number of **Worker** threads that are being synchronized and that each **Worker** thread has a reference to the same barrier object **barrier**. Each **Worker** will run as follows:

```
// All threads have access to this barrier
Barrier barrier;
// do some work for a while . . .
// now wait for the others
barrier.waitForOthers();
// now do more work . . .
```

When a thread invokes the method **waitForOthers()**, it will block until all threads have reached this method. Once all threads have reached the method, they may all proceed with the remainder of their code. The **freeAll()** method bypasses the need to wait for threads to reach the barrier; as soon as **freeAll()** is invoked, all threads waiting for the barrier are released.

Submit the java project **HW5** having the full code of the following:

- The interface **Barrier**,
- The classes **BarrierImpl** implementing the **Barrier** interface,
- The class **Worker** extending the class **Thread**
- The class **Factory**