

### Lab 3

Complete the implement of the `mutex.ResourcePool` class found in the attached archive file. This means you will need to implement the bodies of the following methods.

- `public Object getResource()` – two lines of code
- `public void putResource(Object)`
- `public int getAvailable()` – one line of code
- `private Object getNextAvailable()`
- `private int getIndex(Object)` – one line of code

`ResourcePool` uses an internal semaphore to keep track of the number of available resources. The class has a List of Objects representing the resources managed by the pool. There is also a parallel List of Booleans to keep track of which resources are in use and which are available. Keep in mind that the first resource checked out (borrowed) may not be the first resource returned.

The following will be useful in completing this Lab exercise.

- The code in `SemaphoreDemo.main()`
- The JavaDoc for `java.util.concurrent.Semaphore` (find this online)
- The contents of `ResourcePool`

Note that I have added an attribute in the gendocs target of the Ant script to generate JavaDoc comments for private attributes and methods.

Submit a tar or zip archive of your cleaned project to Blackboard.