Operating Systems (coe628) Lab 8

March 27, 2017
Duration 1 week

Description

Implement a basic Semaphore in Java. In order to "block" a Thread in Java, use the "wait()" method (inherited from Object). To wake up something, use the "notify()" method.

For example, the down method may block if the semaphore value is zero. In this case, use "wait()". The "up()" method increments the Semaphore value and never blocks. However, if the value was zero, there may be something waiting and "notify()" should be used.

The Semaphore API

- Click here to download the template for this lab.
- The clas you have to complete is shown below.

```
package coe628.lab8;

/**
 * A <code>Semaphore</code> object implements a semaphore (invented
by Edsger
 * Dijkstra).
 * <p>
 * A semaphore controls access to resources. The number of resources
currently
 * available is the Semaphore <em>value</em>.
 * </p>
 *
 * @see <a
href="http://en.wikipedia.org/wiki/Semaphore_(programming)">wikipedi
a
 * article</a>
 * @author Your Name
 */
public class Semaphore {
```

```
    private int value;

    /**
     * Create a semaphore.
     * @param value The initial value of the Semaphore ( must be
&ge; 0).
     */
    public Semaphore(int value) {
        this.value = value;
    }
    /**
     * Increment the number of available resources.  This method
never blocks.
     * It may wakeup a Thread waiting for the Semaphore. Dijkstra
called this
     * the <em>V</em> operation.  Many also call it <em>signal</em>
or
     * <em>release</em>.
     */
    public void up() {

    }

    /**
     * Request a resource. If no resources are available, the
calling Thread
     * block until a resource controlled by the Semaphore becomes
available.
     *  Dijkstra called this
     * the <em>P</em> operation.  Many also call it <em>wait</em> or
     * <em>acquire</em>.
     */
    public void down() {

    }

}
```

The template also include the ProducerComsumer class. The project will compile and run but the output will be wrong.

Once you implement the Semaphore class, it should work!

And Finally: Submit your lab

To submit your lab do:

1. Zip your Netbeans project into a file called Lab8.zip

2. Submit the zip file with the command: `submit coe628 lab8 Lab8.zip`

That's all folks....