

Semaphores y Thread Executors

P4

En la siguiente practica se intenta modelar la clase *Mysemaphore*. Esta intenta duplicar el funcionamiento de *java.util.concurrent.Semaphore*.

Para ello la clase *MySemaphore* proporcionada consta de dos métodos, *up()* y *down()* que emulan a *Semaphore.acquire()* y *Semaphore.release()* respectivamente.

Para ello se reutiliza el código de la semana pasada modificando su estructura de bloqueo basado en *Lock* y utilizando *MySemaphore*.

El resultado obtenido es la ejecución alternada de Productor y Consumidor:

```
Consumidor is reading, Product size is: 0  
Productor is writting, Product size is: 1  
Consumidor is reading, Product size is: 0  
Productor is writting, Product size is: 1
```

En el segundo apartado se modifica el programa dejando de usar la clase implementada *MySemaphore* y usando *java.util.concurrent.Semaphore*. El resultado obtenido es el mismo, lo que implica que la implementación hecha en esta entrega parece correcta.